International Journal of Interactive Multimedia and Artificial Intelligence

March 2022, Vol. VII, Number 3
ISSN: 1989-1660

unir LA UNIVERSIDAD EN INTERNET

> *“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.”*
>
> *Tom M. Mitchel*

Special Issue on Artificial Intelligence in Economics, Finance and Business

INTERNATIONAL JOURNAL OF INTERACTIVE MULTIMEDIA AND ARTIFICIAL INTELLIGENCE

ISSN: 1989-1660 –VOL. 7, NUMBER 3

Foreword

Newman
Escuela de Posgrado

This time, in the Special Issue on Artificial Intelligence in Economics, Finance and Business, we present a series of publications focused on artificial intelligence and finance. This compilation of research will bring new information to researchers in different disciplines, and at the same time, it will be an ideal space to present studies that have an international scope.

UNIR, dedicated to the training of professionals in different academic programs, through its journal is consolidating a culture of research and expanding the knowledge that contributes to an excellent education. For this reason, we consider the dissemination of scientific articles essential, since this guarantees the transfer of results, in addition to the conclusions of high-impact research.

Currently the world is going through a complicated scenario, a fluctuating economy and problems in health services that require immediate attention; in this sense, science and knowledge management open space to opportunities in search of medium and long-term solutions.

It is a great honor to present this issue of the International Journal of Interactive Multimedia and Artificial Intelligence, whose contribution to the knowledge society is invaluable.

Dr. Jehovanni Fabricio Velarde Molina
Director of Research at Escuela de Posgrado Newman - Peru

DOI: 10.9781/ijimai.2022.02.010

Editor's Note

Machine learning (ML) is generating new opportunities for innovative research in areas apparently unrelated such as, economics, business or/and finance [1]. Specifically, it has also been widely used in applications related to the economic and financial analysis, such as economic recessions prediction, labor market trends, risk management, prices analysis among others [2].

However, it is important to note the differences between classical statistics/econometrics and machine learning. On the one hand, econometrics set out to build models designed to describe economic problems, while machine learning uses algorithms, generally for prediction, classification, and also, can manage a large amount of structured and unstructured data and make fast decisions or forecasts. As S. Athey [3] points out, perhaps "a key advantage of ML is that it frames empirical analysis in terms of algorithms that estimate and compare many alternative models. This approach contrasts with econometrics, where (in principle, though rarely in reality) the researcher picks a model based on principles and estimates it once".

This Special Issue presents nine contributions that illustrate both approaches in the domain of economics, finance and business. We have classified them in three broad categories: economic modeling, language processing and business applications.

Among the first set, Cadahia et al., applied a decision-tree ensemble method to examine the variable importance of Treasury term spreads to predict US economic recessions with a balance of generating rules for US economic recession detection, demonstrating that machine learning methods are useful for interpretation comparing many alternative algorithms. This contribution is followed by the contribution of Rodríguez-Santiago, who manages a new dataset of 117 countries in the 2005-2019, using a Bayesian Model Averaging (BMA) allowing fixed effects and investigating the existence of heterogeneity, allowing interactions of the focus variable with other regressors it evaluates the robustness of determinants of the variation of self-employment rates across countries by variations in the unemployment, the stage of economic development and the variations in labor market frictions.

Sanchez Fuentes closes this initial set introducing a solution of the Parameterized Expectations Algorithm, a widely applied method for solving nonlinear stochastic dynamic models with rational expectations, based on asymptotic properties.

Natural language processing is among the most interesting areas of artificial intelligence. Recent developments in this field have enabled very significant advances in financial applications ranging from market sentiment analysis to fraud detection. This special issue features two studies. One focused on news extraction and the other on stock trend prediction.

The former, authored by Dogra et al., discusses multiclass financial text news classification. The article describes the difficulties posed by imbalanced datasets and elaborates on the solutions that have been proposed in the literature, such as over-sampling, down-sampling, and ensemble approach. It then reports the results of a benchmarking exercise of different classifiers on banking news extraction. The latter, by Chen et al., proposes an approach to build an ensemble classifier using sentiment in Chinese news at sentence level and technical indicators to predict stock trends. The system combines three different classifiers, a positive and a negative stock trend prediction model based on sentiment values, and another one that relies on Bollinger bands.

The third set of articles illustrates the potential of machine learning to tackle real-world business problems. The body literature on this matter is extremely abundant, as the range of possibilities is almost endless. We present four contributions in this regard.

Alejandro Baldominos et al. focus their attention on the spot instance price prediction in AWS cloud. These authors benchmark nine classic machine learning algorithms and observe that performance varies very significantly among instance types. They subsequently describe how they use these models to develop a prediction-as-a-service system in the cloud.

Ejiyi et al. present a similar exercise in the domain of building insurance prediction. They test the potential of six algorithms to predict whether a customer will submit claims on his/her property or not based on the attributes of the building and some characteristics of the policy. The authors also dive deep in the analysis of feature relevance using Shapley Additive Explanations.

The third piece of research on business applications of machine learning discusses a very interesting case study on food and beverage sourcing for the hospitality industry. Sánchez Torres et al. explore the potential of hierarchical agglomerative clustering to identify similar products in the catalogs made available by suppliers to hotels. The decision support system described in the paper has the potential to have a positive impact both in customer satisfaction and cost effectiveness.

Finally, the extraction of information derived from a large amount of structured data, is the main topic in the work by Asensio et al. They collect data from customers who enquire about university programs, showing that data-driven business management offers itself as a solution to improve the design of promotional strategies.

The issue ends with a tenth article, not specific to the core topic of this special issue, but on a general topic of interest for the community of scientific authors. Razzaq et al. propose the use of dynamic co-authorship and citation networks to study the influence of research collaboration on different aspects such as area, quality or performance of the research.

We would like to thank both the authors and the reviewers that contributed to the special issue. We are also grateful to the editorial team at IJIMAI, Dr. Elena Verdú, Dr. Javier Martínez Torres and Dr. Rubén González Crespo for their support and kind assistance during the whole process.

Antonio A. Golpe[1]
Pedro Isasi[2]
Juan-Manuel Martín-Álvarez[3]
David Quintana[2]

[1] University of Huelva
[2] Universidad Carlos III de Madrid
[3] Universidad Internacional de La Rioja

References

[1] H. Ghoddusi, G. C. Creamer, and N. Rafizadeh, "Machine learning in energy economics and finance: A review," Energy Economics, vol. 81, pp. 709-727, 2019.

[2] P. Gogas and T. Papadimitriou, "Machine learning in economics and finance," Computational Economics, vol. 57, no. 1, pp. 1-4, 2021.

[3] S. Athey, "The impact of machine learning on economics," The economics of artificial intelligence: An agenda, pp. 507-547, 2018.

DOI: 10.9781/ijimai.2022.02.009

TABLE OF CONTENTS

OPEN ACCESS JOURNAL

ISSN: 1989-1660

The International Journal of Interactive Multimedia and Artificial Intelligence is covered in Clarivate Analytics services and products. Specifically, this publication is indexed and abstracted in: *Science Citation Index Expanded, Journal Citation Reports/ Science Edition, Current Contents®/Engineering Computing and Technology.*

The Yield Curve as a Recession Leading Indicator. An Application for Gradient Boosting and Random Forest

Pedro Cadahia Delgado[1], Emilio Congregado[1], Antonio A. Golpe[1], José Carlos Vides[2]*

[1] Department of Economics, University of Huelva, Huelva (Spain)
[2] Instituto Complutense de Estudios Internacionales (ICEI – UCM) & Department of Applied and Structural Economics and History, Faculty of Economics and Business, Complutense University of Madrid, Madrid (Spain)

Received 15 April 2021 | Accepted 24 January 2022 | Published 21 February 2022

Abstract

Most representative decision-tree ensemble methods have been used to examine the variable importance of Treasury term spreads to predict US economic recessions with a balance of generating rules for US economic recession detection. A strategy is proposed for training the classifiers with Treasury term spreads data and the results are compared in order to select the best model for interpretability. We also discuss the use of SHapley Additive exPlanations (SHAP) framework to understand US recession forecasts by analyzing feature importance. Consistently with the existing literature we find the most relevant Treasury term spreads for predicting US economic recession and a methodology for detecting relevant rules for economic recession detection. In this case, the most relevant term spread found is 3-month–6-month, which is proposed to be monitored by economic authorities. Finally, the methodology detected rules with high lift on predicting economic recession that can be used by these entities for this propose. This latter result stands in contrast to a growing body of literature demonstrating that machine learning methods are useful for interpretation comparing many alternative algorithms and we discuss the interpretation for our result and propose further research lines aligned with this work.

Keywords

Gradient Boosting Machine, Random Forest, Rules Detection, Shapely, Treasury Yield Curve.

DOI: 10.9781/ijimai.2022.02.006

I. Introduction

Since the decade of the '80s, economic crises have been more recurrent and deeper. In this respect, researchers and practitioners have tried to understand, model, and even predict a recession differently. One popular forecasting tool suggested in the literature and followed by economists is the analysis of the slope of the yield curve or the term spread, i.e., the difference between long-term and short-term interest rates [1].

According to this idea, in a competitive financial environment, the term structure should respond to international market forces, considered as key for assessing the impact of monetary policy and more importantly, to express the economy's behavior. Indeed, if a monetary policy is effective, changes in short-term policy interest rates should impact long-term ones [2]. In this sense, the need to forecast and prevent economic recessions has become of great importance to policymakers, practitioners and researchers. In this respect, the use of economic and financial variables as predictive information containers joint to the application of several econometric methods and machine learning models have focused on detecting a better accuracy in predicting the possible turning points of the business cycle and, more deeply, economic recessions [3]. This literature review has tried to shed some light on the more important and highlighted topic works.

As previously mentioned, the term structure holds implications in macroeconomics or finance and the shape of the yield curve (see [4] for a survey). According to this, an upward sloping yield curve suggests that future short-term rates are expected to rise. Contrariwise, a descending sloping yield curve may mean that future short-term rates are expected to drop. Like [5] states, the yield curve's slope –the difference between the longer maturity of interest rates and the shorter maturity– gives an important source of information of the real economy evolution. Accordingly, they found that a positive curve slope is associated with future increases in real economic activity when using macroeconomic variables, possessing a significant predictive power or its economic implications in the monetary policy [6], [7]. To understand the background of the term structure, we briefly treat the Expectations Hypothesis of Term Structure (EHTS). This hypothesis illustrates the relationship between short and long-term interest rates and represents the most influential theory explaining the term structure relations. This hypothesis establishes that long-term interest rates are defined by an average of the contemporary and expected short-term interest rate [8]. Therefore, this relationship between both types of interest rates indicates that their spread holds meaningful information on future changes in short-term rates and is an important function in the potential effectiveness of monetary policy [9], [10] or reflecting economic agents' anticipations of future events such as recessions, for instance (see [11] for a survey). According to [12], the inversion of the yield curve is viewed as a consistent predictor of recessions and future economic activity, providing an important reason to explain the flattening or inversion of the yield curve: a monetary lightening. A

* Corresponding author.

E-mail address: jvides@ucm.es

tightening monetary policy would be considered a rise in short-term interest rates, focusing on reducing inflation. The consequence of the monetary tightening is that the economy may slow down.

Consequently, shorter-term interest rates are considered indicators of demand for credit and future inflation. Therefore, longer-term interest rates would tend to decrease and flatten the yield curve, an example of the relation between the yield curve behavior and recessions. Definitely, the yield curve's steepness would help us predict and determine a future recession [13].

The literature on this topic has tried to demonstrate the role of the term structure or the yield curve as a good forecasting tool for recessions [14]. The influential papers of [5] and [15] should be noted. These works evidenced that the yield curve might be employed to predict real growth in consumption, investment, or aggregate GNP, and more importantly, they demonstrated the relation with NBER-dated recessions. For its part, [16] suggests that among different variables used in his work, the term spread is the significant predictor of recessions at horizons beyond three months. In this respect, many previous papers have treated the topic by relating the GDP growth with the yield curve slope (see [17]-[25], among others or [26] for a deep survey of the topic.). Another important work by [27] argues the convenience of applying models which use the yield curve to predict recessions. In other influential papers in the literature, the term spread is also useful in predicting recession even for professional forecasters, as [28] suggested and [29] combined the term spread with stock returns to measure the accuracy of the term spread the latter to predict recessions. His results were positive, and the term spread was found as a valuable predictor of recessions for German and US economies. In a similar work by [28], [30] compared the strength of the yield curve in forecasting recessions with the data used in [28], evidencing the power of the former and suggesting the suitability of using this indicator. For its part, [31] also treated the capability of predicting recessions of the term structure and highlighted the power of this indicator over other leading indicators. Its strength decreased as a predictor after the financial crisis due to the volatility of macroeconomic variables, but unfortunately, its predictive power over the last decade has fallen.

Furthermore, [3] in line with the previous literature, find that the ability of the term structure to predict recessions is stronger over the twelve-month horizon when using a similar probit model than [5] or [13] used. Additionally, [32] further evidenced the potential of the yield curve in forecasting future situations of the US economy over horizons ranging from one quarter to two years. Besides, [33] recognized that the yield curve contains information on future GDP growth and that its predictability varies with time, forecast horizons, and quantiles of the distribution of future growth; nonetheless, a significant empirical contribution of their work is that it seems more efficient to predict future expansionary phases, which are more common than recessions, for which the latter appears to perform better. Finally, although [34] find that developments in the stock market diminish the efficacy of the yield curve in forecasting future economic activity, they show the fitness of this indicator for predicting economic activity in many most important world economies, such as the US, Canada and Europe and, more importantly, when periods of financial stress are analyzed.

From another empirical perspective, it emerges in the literature the use of techniques based on machine learning algorithms. In this sense, [35] claims the suitability of machine learning techniques on central banking or monetary policy issues as applied in other real-life topics. In this sense, [36] demonstrated the yield curve as a robust and consistent predictor of economic activity when US business cycle turning points are checked by using four different methods, i.e., equally-weighted forecasts, Bayesian Model Averaging (BMA), and linear and non-linear machine learning boosting algorithms. An important paper in the literature by [37] compares different Support Vector Machine (SVM hereafter) and logit models when using the yield curve as a leading indicator, being "the first empirical investigation on the relation between the yield curve and an economy's real output, using an SVM classifier". The model created is helpful for policymakers in order to forecast future recessions. In order to reaffirm this latter study, [38] the yield curve is a useful tool for assessing future economic activity, achieving a 100% forecasting accuracy for recessions. For its part, [39] demonstrated that the predictive power of boosted regression trees is considerably better than standard probit models. Their findings show that short rates and the yield curve are crucial leading indicators for recession forecasts during the 1974-2014 period. Finally, [40] employs several machine learning methods such as Least Absolute Shrinkage and Selection Operator (LASSO), and Elastic Net, Discriminant Analysis classifiers, Bayesian classifiers, and classification and regression trees (CART), in line with the existing literature and reveal the ability of the yield curve to act as an early warning system to predict recessions in the United States is reconfirmed. Specifically, the yield curve keeps on a consistent and reliable predictor of recession over the 12-month forecast horizon and [41] also applies a battery of machine learning methods: decision trees, random forests, extremely randomized trees, support vector machines (SVM), and artificial neural networks, finding that almost all the machine learning models appropriately predict the global financial crisis of 2007-2008 and, additionally, they indicate that the flatter or more inverted the yield curve is, the higher the chance of a crisis, exposing the tendency of chasing performance or increased risk-taking that can often be seen before financial crises.

To the best of our knowledge, our approach, i.e., Gradient Boosting and Random Forest Machine Learning methods, allows us to reach a better accuracy than in those previous papers on the topic. These Machine Learning algorithms let us identify the more relevant variables associated with the main variable, which has not been done before in the literature. Additionally, we extend the time horizon, i.e., we update data compared to previous studies. Indeed, our results indicate that our algorithm let us signal and choose the most influential variables for predicting economic recessions amongst the term spreads analyzed. This case highlights some of the most important term spreads as 3-month–6-month, 2-year–5-year and 5-year–10-year. Furthermore, concerning these variables, the lift metric is computed to detect intervals with a higher probability of accounting for a recession, applied to the rules description methods. Results suggest that the most important term spread is 3-month–6-month compared with the term spreads mentioned in the literature. Results give some considerations for monetary authorities, policymakers and practitioners, such as the monitorization of this term spread above mentioned as a tool for evidencing economic recessions.

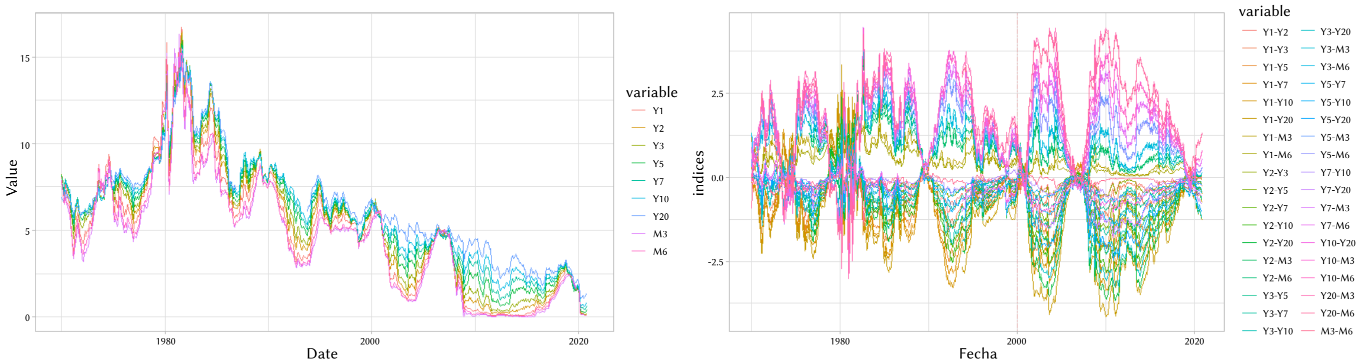
The rest of the paper is as follows. Section II presents the data and methodology used in the paper. Later, section III show and discuss the results; the concluding remarks are in section

II. Data and Methodology

A. Introduction

A supervised method is proposed to predict economic crisis cycles and can also identify the key factors that lever this phenomenon. Assessing variable importance is an important task; this is reflected in many studies fields; besides, several approaches address this question [42]-[45].

A decision-tree ensemble classification method is proposed for interpretability rather than only predicting economic recessions from the different term spread as independent variables. In this way, the variable importance is computed to measure which variables are the most relevant to predict economic crisis cycles. More interpretation of



(a)

(b)

Fig. 1. Original data interest rates (a) & Computed Term spreads (b).

the model is performed by analyzing the dependencies with the most correlated variables and the feature value dependency regarding the target variable to understand this phenomenon better. Finally, a rule extraction process is proposed that could be useful for interpreting and detecting economic recession.

B. Data Description

For our empirical analysis, we employ a monthly sample of Treasury Constant interest rates at nine different maturities from January 1969 to November 2020 (amounting to 601 observations for each interest rate series). The data corresponds to the constant maturity rates of 3-month, 6-month, 1-year, 2-year, 3-year, 5-year, 7-year, 10-year and 20-year.

The data is collected from the Federal Reserve Economic Data (FRED) collected by the Economic Research Division of the Federal Reserve Bank of St. Louis. Since the 1-month Treasury Constant maturity rate is only accessible since January 2001, we have picked these maturities considering the availability of consistent interest rate data with the period studied. We reveal 3-month, 6-month and 1-year as short-run, including the latter variable 1-year as short-term because it offers more robustness in our assessment. Conversely, we contemplate the rest of the maturity rates as long term. Table I shows descriptive statistics related to each interest rate in different maturities. These variables show similar behavior in terms of volatility, and Fig. 1.A and Fig. 1.B presents a plot analysis of the time series traced for all maturities.

TABLE I. Descriptive Statistics for the Data

	M3	M6	Y1	Y2	Y3	Y5	Y7	Y10	Y20
Mean	4.57	4.69	5.08	5.18	5.54	5.84	6.07	6.23	6.31
Median	4.86	4.95	5.27	5.03	5.77	5.97	6.17	6.20	6.01
Min	0.01	0.04	0.10	0.13	0.16	0.27	0.56	0.62	1.06
Max	16.30	15.52	16.72	16.46	16.22	15.93	15.65	15.32	15.13
SD.	3.41	3.40	3.64	3.78	3.51	3.53	3.23	3.11	3.05

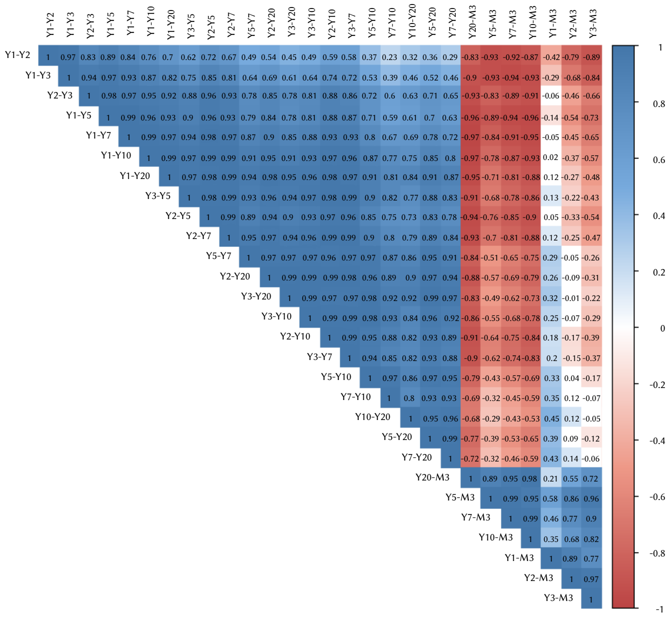
[a] Data from January of 1969 to November of 2020.

[b] M and Y refers to month and year respectively.

From 9 interest rates, 36 spread variables are obtained, the calculation being a subtraction of two elements; this follows a combination without repetition C(n,r), being n and r the set and subset size, respectively. As shown in Table I, the interest rates show similar statistical properties. Nevertheless, the short term interest rates 3-month and 6-month presents lower mean and median and higher standard deviation. On the contrary, long term interest rates show the opposite higher mean and median and lower standard deviation. Henceforth for representing term spread at figures and tables, due to saving space, an abbreviation is used, being M and Y for month and year interest rates respectively, i.e. M3-Y10 for 3-month–10-year term spread.

At Fig. 1.A, the interest rates are plotted where the general trend is decreasing, Fig. 1.B shows the computed Term spread for all combinations of interest rates, it is stated that there are some expansion stages with the behavior of divergence and flattening stage where the term spreads are inverted with the behavior of convergence which could be an early indicator of economic recession.

As a combinatory result, the term spread variables show several strong correlations. The correlation coefficient is used to verify collinearity, and it is argued that collinearity is certain at the 0.9 level of a correlation coefficient or higher [46]. A correlation analysis is shown between variables at Fig. 2, where the correlation plot shows the coefficients.



	Y1-Y2	Y1-Y3	Y2-Y3	Y1-Y5	Y1-Y7	Y1-Y10	Y1-Y20	Y3-Y5	Y2-Y5	Y2-Y7	Y5-Y7	Y2-Y20	Y3-Y20	Y3-Y10	Y2-Y10	Y3-Y7	Y5-Y10	Y7-Y10	Y10-Y20	Y5-Y20	Y7-Y20	Y20-M3	Y5-M3	Y7-M3	Y10-M3	Y1-M3	Y2-M3	Y3-M3
Y1-Y2	1	0.97	0.83	0.89	0.84	0.76	0.7	0.62	0.72	0.67	0.49	0.54	0.45	0.49	0.59	0.58	0.37	0.23	0.32	0.36	0.29	-0.83	-0.93	-0.92	-0.87	-0.42	-0.79	-0.89
Y1-Y3		1	0.94	0.97	0.93	0.87	0.82	0.75	0.85	0.81	0.64	0.69	0.61	0.64	0.74	0.72	0.53	0.39	0.46	0.52	0.46	-0.9	-0.93	-0.94	-0.93	-0.29	-0.68	-0.84
Y2-Y3			1	0.98	0.97	0.95	0.92	0.88	0.96	0.93	0.78	0.85	0.78	0.81	0.88	0.86	0.72	0.6	0.63	0.71	0.65	-0.93	-0.83	-0.89	-0.91	-0.06	-0.46	-0.66
Y1-Y5				1	0.99	0.96	0.93	0.9	0.96	0.93	0.79	0.84	0.78	0.81	0.88	0.87	0.71	0.59	0.61	0.7	0.63	-0.96	-0.89	-0.94	-0.96	-0.14	-0.54	-0.73
Y1-Y7					1	0.99	0.97	0.94	0.98	0.97	0.87	0.9	0.85	0.88	0.93	0.93	0.8	0.67	0.69	0.78	0.72	-0.97	-0.84	-0.91	-0.95	-0.05	-0.45	-0.65
Y1-Y10						1	0.99	0.97	0.99	0.99	0.91	0.95	0.91	0.93	0.97	0.96	0.87	0.77	0.75	0.85	0.8	-0.97	-0.78	-0.87	-0.93	0.02	-0.37	-0.57
Y1-Y20							1	0.97	0.98	0.99	0.94	0.98	0.95	0.96	0.98	0.97	0.91	0.81	0.84	0.91	0.87	-0.95	-0.71	-0.81	-0.88	0.12	-0.27	-0.48
Y3-Y5								1	0.98	0.99	0.93	0.96	0.94	0.97	0.98	0.99	0.9	0.82	0.77	0.88	0.83	-0.91	-0.68	-0.78	-0.86	0.13	-0.22	-0.43
Y2-Y5									1	0.99	0.89	0.94	0.9	0.93	0.97	0.96	0.85	0.75	0.73	0.83	0.78	-0.94	-0.76	-0.85	-0.9	0.05	-0.33	-0.54
Y2-Y7										1	0.95	0.97	0.94	0.96	0.99	0.99	0.9	0.8	0.79	0.89	0.84	-0.93	-0.7	-0.81	-0.88	0.12	-0.25	-0.47
Y5-Y7											1	0.97	0.97	0.97	0.96	0.97	0.97	0.87	0.86	0.95	0.91	-0.84	-0.51	-0.65	-0.75	0.29	-0.05	-0.26
Y2-Y20												1	0.99	0.99	0.99	0.98	0.96	0.89	0.9	0.97	0.94	-0.88	-0.57	-0.69	-0.79	0.26	-0.09	-0.31
Y3-Y20													1	0.99	0.97	0.97	0.98	0.92	0.92	0.99	0.97	-0.83	-0.49	-0.62	-0.73	0.32	-0.01	-0.22
Y3-Y10														1	0.99	0.99	0.98	0.93	0.84	0.96	0.92	-0.86	-0.55	-0.68	-0.78	0.25	-0.07	-0.29
Y2-Y10															1	0.99	0.95	0.88	0.82	0.93	0.89	-0.91	-0.64	-0.75	-0.84	0.18	-0.17	-0.39
Y3-Y7																1	0.94	0.85	0.82	0.93	0.88	-0.9	-0.62	-0.74	-0.83	0.2	-0.15	-0.37
Y5-Y10																	1	0.97	0.86	0.97	0.95	-0.79	-0.43	-0.57	-0.69	0.33	0.04	-0.17
Y7-Y10																		1	0.8	0.93	0.93	-0.69	-0.32	-0.45	-0.59	0.35	0.12	-0.07
Y10-Y20																			1	0.95	0.96	-0.68	-0.29	-0.43	-0.53	0.45	0.12	-0.05
Y5-Y20																				1	0.99	-0.77	-0.39	-0.53	-0.65	0.39	0.09	-0.12
Y7-Y20																					1	-0.72	-0.32	-0.46	-0.59	0.43	0.14	-0.06
Y20-M3																						1	0.89	0.95	0.98	0.21	0.55	0.72
Y5-M3																							1	0.99	0.95	0.58	0.86	0.96
Y7-M3																								1	0.99	0.46	0.77	0.9
Y10-M3																									1	0.35	0.68	0.82
Y1-M3																										1	0.89	0.77
Y2-M3																											1	0.97
Y3-M3																												1

Fig. 2. Pearson correlation between term spread variables.

Pearson's correlation results in Fig. 2 shows high correlated features. In line with the literature, results show a consistent negative relationship in the difference between long-term and short-term interest rates and consequently in the term spreads [1]. This is taken into account to interpret the importance of the features exposed in the results.

Literature mainly focused on continuous variables whose values, for instance, growth rates in GNP, GDP, industrial production, consumption, investment, among others [1]. In this work, only interest rates are used as predictors as the main purpose of this work is not to offer the better predictive model results of literature but to understand the relationships, importance and rules regarding interest rates with an economic recession.

1. Variable Target Lift

In machine learning, Lift is a metric used to assess the performance of a targeting model at predicting or classifying cases as having an enhanced response concerning the population as a whole.

This metric is pretty straightforward to understand, and a targeting model is performing well if the response within the target is much better than the average for the population. In other words, Lift is simply the ratio of these values: target response divided by average response [47]. It is defined as:

$$Lift = \frac{P(A \cap B)}{P(A)P(B)} \tag{1}$$

These indicators, shown in Table II, are useful in the exploratory data analysis stage to understand at each variable's decile which range of values of the response variable has more impact on positive target. This can be used as an early exploratory rule for detecting economic recession, and this is complementary information as the decile split does not guarantee the optimal value range for a variable for maximizing the lift; on the contrary, the computed lift for tree base rules ranges may give a better separation as it is a supervised method, for this reason, it helps initially to understand this economic processes.

TABLE II. Lift for Crisis per Deciles for the Most Relevant Features

Decile	M3-M6	Y3-M3	Y5-Y10	Y2-Y5	Y2-M6	Y3-Y7
1	**1.46**	**1.09**	0.46	0.16	**1.52**	0.33
2	0.74	**1.60**	**1.14**	0.91	0.62	0.87
3	**1.20**	0.75	0.90	**1.40**	0.00	**1.11**
4	0.51	0.53	0.64	**1.67**	0.91	0.96
5	0.85	**1.42**	**1.63**	**1.55**	**1.71**	**1.26**
6	0.77	**1.29**	0.56	0.78	**1.71**	0.62
7	0.34	**1.42**	0.31	0.62	0.62	**1.09**
8	0.41	0.66	0.71	**1.26**	0.30	0.33
9	**1.88**	0.54	0.62	0.30	0.78	**1.42**
10	**1.79**	0.75	**2.95**	**1.34**	**1.83**	**2.04**

[a] Term spread abbreviations contains M and Y for monthly term and yearly term interest rates respectively.

For the sake of simplicity, in Table II the target lift is computed only for the most important variables, as shown in section III. From this table, some initial patterns there can be found. Generally, almost every term spread at high deciles has a high lift in economic recession except for 3-year–3-month. On the contrary, the 3-year-3–month and 2-year–6-month term spreads show high lift for low and mid deciles. This is an initial indicator due to the higher probability of recession in those deciles; for specific range values, the decile's interval table can be found in the appendix.

C. Methodology

The main purpose of this work is not only to offer a model for predicting economic recessions but also to offer a methodology of a good enough model that is able to explain variable importance, dependencies and economic recession detection rules.

Decision-tree ensemble methods are supervised learning methods for modeling the relationship between the dependent variable y with the characteristic vector x. Besides, these techniques are a common choice on the actual machine learning research scenario, it has a wide range of applications for regression, classification and other tasks [48], [49].

The two main decision-tree ensemble methods in bagging and boosting for classification scenario are applied in this work for estimating the economic crisis cycles. The advantage of this methods is that often provides predictive accuracy that cannot be beat, it can optimize on different loss functions and provides several hyperparameter tuning options that make the function fit flexible, generally no data pre-processing required and often works great with categorical and numerical values.

To train the models, a training and test data split is performed, where the training set consists on all available variables for all observations from January of 1969 to December of 1999 and the test set comprises from January of 2000 to January of 2020, with the correspondent binary supervised target of economic crisis cycle. In other words, the models should learn which features are relevant in order to predict from a time interval selected for another more recent time interval which should be relevant not only for predicting the economic crisis cycles but also for Interpretability of the actual situation.

1. Random Forest Classifier

Random Forest (RF) was proposed by [50] as an ensemble method for regression based on individual decision trees, the original classification approach based on Stochastic Discrimination was proposed by [51], [52].

In this way, Ranger is a fast implementation of RF [53] or recursive partitioning, particularly suited for high dimensional data. The R implementation Ranger was used to adjust a RF model respectively the considered optimal settings [54].

Which makes Random forest powerful is that builds several weak decision trees in parallel, resulting computationally cheap process, by combining the trees to form a single, strong learner by averaging or taking the majority vote results often to be accurate learning algorithms.

The pseudocode is illustrated at Algorithm scheme I. The algorithm works as follows: for each tree in the forest, a bootstrap sample is selected from S where S(i) is the ith bootstrap. Then it is trained a decision-tree as follows: at each node of the tree, instead of examining all possible feature-splits, a random features subset selection is made f ⊆ F. where F is the set of features. The node then splits on the best feature in f rather than F. In practice f is much smaller than F. By narrowing the set of features, it drastically speeds up the learning of a tree.

Algorithm I. Random Forest algorithm

Precondition: A training set $S:=(x_1, y_1), \ldots,(x_n, y_n)$, *being* $\boldsymbol{F}$ *the features and* $\boldsymbol{B}$ *number of trees in forest.*

```
function: RandomForest($S$, $F$)
    $H \leftarrow \emptyset$
      for $i \in 1, \ldots, B$ do:
        $S^{(i)} \leftarrow$ A boostrap sample from $S$
        $h_i \leftarrow$ RandomizedTreeLearn ($S^{(i)}$, $F$)
        $H \leftarrow H \cup \{h_i\}$
      end for
    return $H$
end function
function RandomizedTreeLearn ($S$, $F$)
    At each node:
      $f \leftarrow$ small subset of $F$
      Split on best feature in $f$
    return learned tree
end function
```

RF algorithm is a bagging technique for building an ensemble of decision trees, and this technique is known to reduce the variance of the algorithm. Traditionally bagging with decision trees, the constituent decision trees may be highly correlated because the same features will tend to be used repeatedly to split the bootstrap samples. At the same time, restricting each split-test to a small, random sample of features decreases the correlation between trees in the ensemble and improves the performance of the algorithm.

2. Gradient Boosting Machine

The gradient boosting machines (GBM) proposed by [55] is a robust machine learning algorithm due to its flexibility and efficiency in performing regression tasks [55].

The main difference between boosting and traditional machine learning techniques is that optimization is held out in the function space. In other words, the function estimate $\hat{f}$ is parametrized in the additive functional form:

$$\hat{f}(x) = \hat{f}^M(x) = \sum_{i=0}^{M} \hat{f}_i(x) \tag{2}$$

In this notation, M is the number of iterations, $\hat{f}_0$ is the initial guess and $\{\hat{f}_i\}_{i=1}^{M}$ are the function increments, also known as "boosts".

To ensure that the functional approach is achievable in practical terms, a comparable approach to parameterization of the family of functions can be implemented. It is introduced to the reader the parameterized "base-learner" functions $h(x, \theta)$ to differentiate it the overall ensemble functions estimates $\hat{f}(x)$. Different families of basic learners can be chosen, such as decision trees and loss functions.

The "greedy stagewise" approach of function incrementing with the base-learners can be formulated.

For the function estimate at the t-th iteration, the optimization function is:

$$\hat{f}_t \leftarrow \hat{f}_{t-1} + \rho_t h(x, \theta_t) \tag{3}$$

$$(\rho_t, \theta_t) = \arg\min_{\rho,\theta} \sum_{i=1}^{N} \psi(y_i, \hat{f}_{t-1}) + \rho h(x, \theta_t) \tag{4}$$

The optimal step-size ρ, should specified at each iteration.

The gradient boosting algorithm proposed by Friedman [55], can be summed up with the following pseudocode at algorithm II.

Algorithm II. Friedman's GBM algorithm

Precondition:

- Input data $(x, y)_{i=1}^{N}$
- Number of iterations M
- Choice of loss-function $\Psi(y, f)$
- Choice of the base-learner model $h(x, \theta)$

```
Initialize $\hat{f}_0$ with a constant
for t = 1 to M do:
    compute the negative gradient $g_t(x)$
    fit a new base-learner function $h(x, \theta_t)$
    find the best gradient descent step size $\rho_t$:
        $\rho_t = \arg\min_{\rho,\theta} \sum_{i=1}^{N} \psi(y_i, \hat{f}_{t-1}) + \rho h(x, \theta_t)$
    update the function estimate: $\hat{f}_t \leftarrow \hat{f}_{t-1} + \rho_t h(x, \theta_t)$
end for
```

The theory and formulation of GBM are available in reference [55], which interested readers in a more profound explanation for a better understanding of this method.

In this work, the so-called Extreme Gradient Boosting Training (XGB), proposed by [56], a version of GBM, was applied as a boosting method for classification with the R library xgboost.

3. Classifier Evaluation

For training the model, a data partition was performed; as explained in the previous sections, the predictive accuracy of the models was measured by splitting the data into training and test sets.

The training set comprehends from 1970 to 1999 with 360 instances and a binary target variable with 16% positives (5 crisis cycles). The test set comprehends from 2000 to 2020, which are 251 instances with 14% of positives in the binary target (3 crisis cycles).

As a classification task, the error assessment was performed using the predicted class for the selected models and computing some accuracy metrics from the confusion matrix, the computed metrics are shown in Table III.

Let {P, N} the positive a negative instance class and let $\{\tilde{P}, \tilde{N}\}$ be the predictions produced by a classifier. Let P(P|I) be the posterior probability that an instance I is positive.

TABLE III. Classification Metrics for Classification Model Assessment

Metric	Formula
Recall(TPR)	$P(\tilde{P}\,\lvert P) \approx \frac{\textit{positives correctly classified}}{\textit{total positives}}$
Specificity(TNR)	$P(\tilde{N}\lvert N) \approx \frac{\textit{negatives correctly classified}}{\textit{total negatives}}$
Precision(PPV)	$\frac{\textit{positives correctly classified}}{\textit{positives correctly classified} + \textit{Negatives correctly classified}}$

There is no unique metric for assessing a classification task, depending on the characteristics to be evaluated, we consider precision as the most suitable metric for this purpose as considers the positives correctly classified within the observations correctly classified.

4. Model Interpretation

The interpretability of a statistic model helps to understand why certain decisions or predictions have been made; for this reason, measuring variable importance is an important task in many applications. In this sense, this is the era of making machine learning explainable; several authors have conducted an extensive review of methods [57], [58].

The most common variable importance based has been tested by several researchers using both simulated and real data; this metric tends to be biased in many scenarios [58]-[60]. As studied in subsection II.B., there is the presence of mutually correlated and collinearity; Gini variable importance is expected to be biased [59], [60].

Nevertheless, there is also another classification for interpretability, and it could be either local or global; in other words, it is explaining an individual prediction or the entire model behavior [61].

a) SHAP Variable Importance

SHapley Additive exPlanations (SHAP) is a model additive explanation approach in which each prediction is explained by the contribution of the features of the dataset to the model's output [62], [63]. SHAP comes from the game theory field, that is, the solution for the problem of computing the contribution to a model's prediction of every subset of features given a dataset with m features.

A model retraining is required on all feature subsets $S \subseteq F$, where F are all the available features. A value of importance it is assigned to every variable that accounts for the impact on the model's prediction of incorporating that feature. A model $f_{S \cup \{i\}}$ is trained with that feature present and another model f_S is trained with the feature withheld in order to compute this effect. Then, both models predictions are compared on the current input $f_{S\cup\{i\}}(x_{S\cup\{i\}}) - f_S(x_S)$, where x_S are the values of the input variables in the set S. Since the effect of withholding a feature depends on other features in the model, the preceding differences are computed for all possible subsets $S \subseteq F \setminus \{i\}$. The feature attributions are the computed Shapley values.

They are a weighted average of all possible subsets of S in F:

$$\phi_i = \sum_{S\in F\setminus\{i\}} \frac{|S|!(M-|S|-1)!}{M!}\left[f_{S\cup\{i\}}(x_{S\cup\{i\}}) - f_S(x_S)\right] \qquad (5)$$

SHAP value is the only possible locally accurate and consistent feature contribution values [62], [63], they can provide high quality explanation both local and global.

Calculating the importance of the features based on SHAP contributions, the mean of each feature is retrieved for each SHAP matrix. Then, the resulting vectors are summed.

b) SHAP Dependence Plots

For every feature and data instance, a point is plotted with the feature value on the x-axis and the corresponding Shapley value on the y-axis, this is the SHAP feature dependence plot.

Mathematically, the plot contains the following points:

$$\left\{\left(x_j^{(i)}, \phi_j^i\right)\right\}_{i=1}^{n} \qquad (6)$$

SHAP dependence plots are an alternative to partial dependence plots and accumulated local effects. While other methods show average effects, SHAP dependence also shows the variance on the y-axis.

c) Rules Extraction

Tree ensembles such as random forests and boosted trees are accurate but difficult to understand. In this work, the framework of the interpretable tree (inTrees) is used to extract, measure, prune, select, and summarize rules from a tree ensemble and calculate frequent variable interactions [64].

Tree ensemble methods consist of multiple decision trees [53], [55]. A rule can be extracted by means of a decision tree's root node to a leaf node.

This rule summarization process explained at algorithm 3, is relevant in order to understand and filter the rules for phenomenon interpretability.

```
Algorithm III. ruleExtract algorithm
Precondition:
 • Input: : ruleSet ← null, node ← rootNode, C ← null
 • Output: ruleSet
function: ruleExtract(ruleSet, node, C)
    if leafNode = true then
        currentRule ⟵ {C → $pred_{node}$}
        ruleSet ← {ruleSet → currentRule}
        return ruleset
    end if
    for $child_i$ = every child of node do:
        C ← C ∧ $C_{node}$
        ruleSet ← ruleExtract(ruleSet, child, C)
    end for
    return ruleSet
end function
```

Given a rule {C ⇒ T}, where C is the condition's rule, being a conjunction of variable-value pairs aggregated from the path from the root node to the current node, C_{node} denote the variable-value pair used to split the current node, $leaf_{Node}$ denote the flag whether the current node is a lead node, $pred_{node}$ denote the prediction at a leaf node, and T for rule's output.

The method ruleExtract explained at pseudocode Algorithm 3 shows the method used to extract rules from a decision tree. As tree ensembles are multiple decision trees, the final rules are a combination of rules extracted from each decision tree in the tree ensemble.

In the following work, it is applied the inTrees framework to the data set. For the winning classifier, the ruleExtract method is applied. As a result, several rules are extracted, and a post-processing rules step is performed. This post-processing comprises de-duping rules and rules metrics computation for rules quality. The rule's metrics are length which is the number of conditions within a rule, support which is the percentual frequency of observations that fulfil the rule, the rule's error for classification tasks which is the number of correctly classified instances within a rule condition and the target lift (epigraph II.B.1) for every rule as the number proportion of positive targets in the rule condition compared with the variable range.

III. Results & Discussion

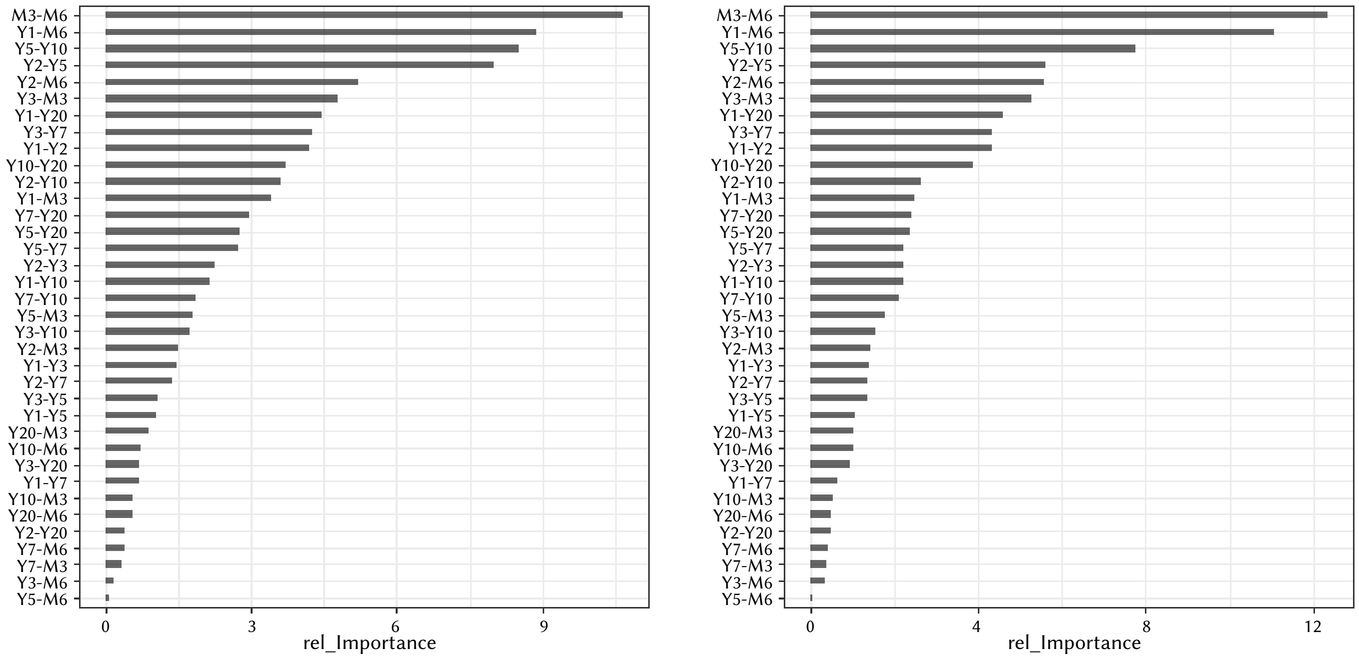
In this work, a methodology is proposed for understanding the economic recession phenomenon and extracting rules as an early economic recession detection method with a balance of getting a model with a suitable accuracy for prediction, which is the main scope of interpretable models in machine learning. This methodology begins with benchmarking proposed models to get the feature importance for the winning model (see epigraph II.C.4.a). From this step, the main variables that lever the economic recession are detected by understanding the dependencies with the most correlated variables and the feature value interaction regarding the target variable to understand this phenomenon better (see epigraph II.C.4.b). To conclude, a rule extraction process is performed for proposing rules useful for early detection of economic recession (see epigraph II.C.4.c).

As the first step, two tree-based classification models are fitted to the data; as a result, Table IV shows the results for the proposed accuracy metrics for the fitted models. When assessing the predictive accuracy, the yield curve performs quite well. Additional information can improve its predictive performance [65]. Thus, the main purpose of this work is through term spreads as unique independent variables to build a model for interpretability with a balance on predictive accuracy.

TABLE IV. Classification Metrics Results

Model	Class	Precision	Recall	Specificity
RF	0	0.88	0.96	0.25
	1	0.52	0.25	0.96
XGB	0	0.96	1.00	0.80
	1	1.00	0.80	1.00

Despite adding only variables about interest rate nature, suitable classification metrics are obtained employing term spread variables for predicting an economic recession. XGB model has better classification metrics results; for the positive target class, the precision shows us how no false positives are obtained; for this reason, specificity also has the maximum value. However, recall has a high value but not the maximum, showing that despite a balanced classification of negative and positive labels, false negatives are present. After fitting and selecting the winning model, the model interpretation for understanding the phenomenon as the most important part of this work comes with the feature importance as the first relevant output to interpret which variables are the main predictors for economic



(a)

(b)

Fig. 3. Training (a) & Test (b) SHAP values for the variables.

recessions. The variable importance is obtained by computing the mean of absolute SHAP value for all instances for every feature at the training and test set. As a result, Table V, which is in the appendix, is plotted in Fig. 3 for better understanding. In Fig. 3, the features are sorted by variable importance in descending order from top to bottom for the most relevant and less relevant, respectively. Besides, by only considering the presence of variables Fig. 3.A and Fig. 3.B shows similar results at the most important variables; however, as the test set has the more recent data, it is expected to be more representative for future values and may be more accurate in order to extrapolate this information for a near future, due to this, the main analysis is focused in the test set analysis.

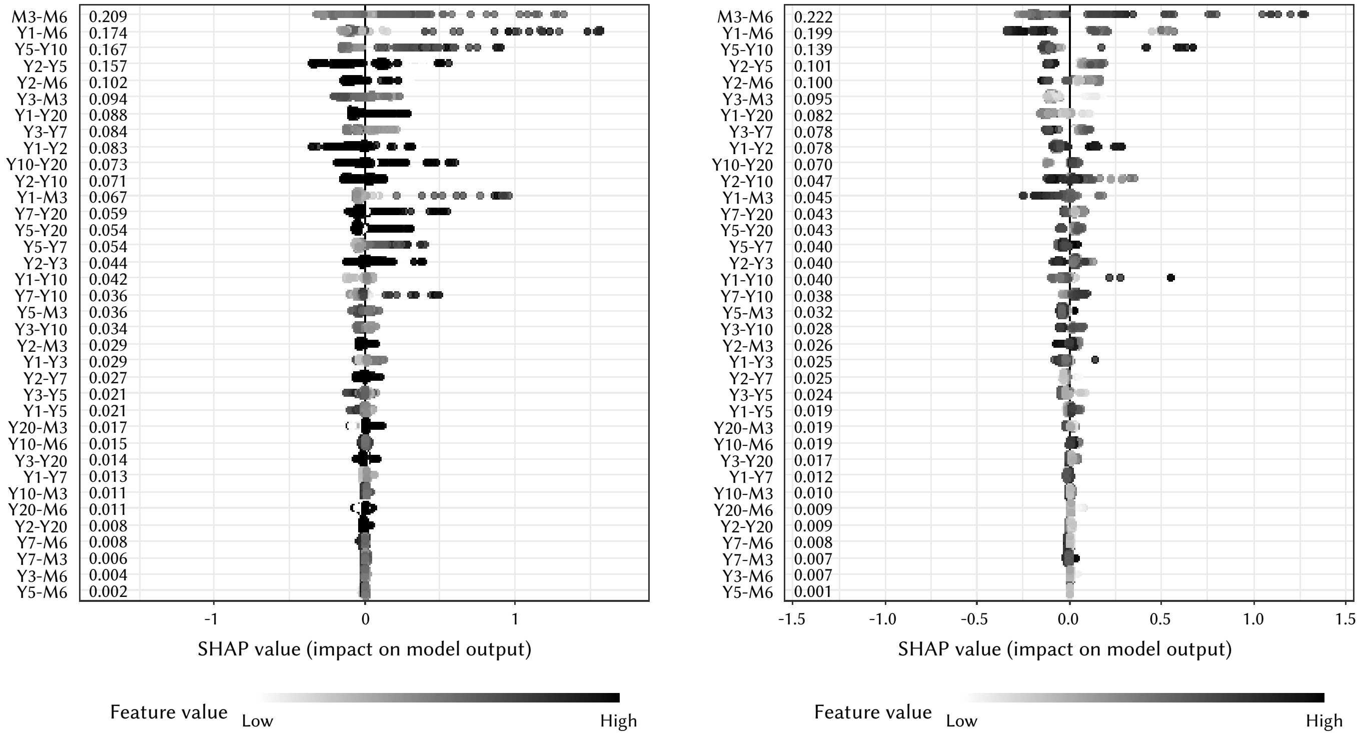
In previous studies, the best results are obtained when forecasting an economic recession by taking the difference between two interest rates whose maturities are far apart. [65] suggested that the 3-month–10-year term spread provides a suitable combination of accuracy and validity in the long term to predict economic recessions. However, most term spreads are highly correlated and provide similar information about the economy, so the particular choices regarding the maturity amount mainly to fine-tuning process.

Results suggest that the most important term spreads are 3-month–6-month, 2-year–5-year, 5-year–10-year, 3-year–7-year, 3-year–3-month and 2-year–6-month. Although this work has more recent data than previous studies, the literature suggests as a rule of thumb that the difference between 10-year and 3-month Treasury rates becomes negative in early recessions providing a reasonable accuracy and time prevalence [65]. Despite not having this term spread as the more relevant, most term spreads are highly correlated and provide similar information about the economy's behavior, so the particular choices concerning maturity amount mainly to fine-tuning and not to reversal of results [65]. The cautionary is that a reference point that works for one spread may not work for others. For example, the 2-year to 10-year term spread may reverse in advance of the 3-month to 10-year term spread, which tends to be higher [1]. In this line, some of the most critical variables like 5-year - 10-year term spread align with the literature statements as could invert earlier than 10-year–3-month term spread.

SHAP contribution values are plotted for training and test sets in Fig. 4.A and Fig. 4.B. This method estimates an individual sample because they are local explainers. Nonetheless, this can lead to different results as training and test set have different instances; in this case, there are slight differences between both results. Besides, this plot retrieves additional information about the feature value analysis and the position of the instances on the plot. The horizontal location shows whether the effect of that value is associated with a higher or lower prediction from right to left; respectively, the vertical location shows the variable importance. The color gradient shows whether that variable is high (dark) or low (light) for that observation.

As argued before, the analysis is focused on test set results, SHAP contribution values analysis could be complementary to decile target lift results at Table II as it is a preliminary analysis that has not the best splitting method for finding a range with the maximum split. SHAP contribution analysis shows that 3-year–6-month and 5-year–10-year term spreads have a higher lift for higher values, the 9-10 deciles. The term mentioned above spreads shows this relationship information at the SHAP contribution plot at Fig. 4, the dark gradient color for instances are at the right side of the plot and the light ones at the left, which indicates that high values are associated with positive predictions of economic recession. On the contrary, an opposite behavior is shown on 2-year–5-year, 3-year–7-year, 3-year–3-month and 2-year–6-year spreads, which is somehow aligned with the decile target lift values of Table II, the lower values, the higher lift, in other words, higher probability of economic recession. As the SHAP contribution plot shows local interpretability and the decile target lift is not an optimized method for splitting ranges for maximizing lift, these complementary results also may present different nuances at both results due to are different perspective analyses.

Once the main features that impact economic recession prediction are detected, the dependent variables with more important variables on the target variable are studied. Dependence plots have been explained at epigraph II.C.4.b; more information can be found at [62], [63]. In essence, this plot shows feature values of the most important variables on the x-axis and SHAP values of the most correlated variable on the y-axis; additionally, a gradient color to the points by the feature value of the designated variable is added.



(a)

(b)

Fig. 4. Training (a) & Test (b) SHAP contribution values results.

For selecting the most correlated variable, the pairwise Pearson's correlation is performed at subsection II.B. By sorting the correlation coefficient, the most important variable is selected as the most correlated feature; as a result, Table VI at the appendix. Results suggest that the most correlated variables for the most important ones are in the same time term; for long term time spreads, most correlated are long term ones. The relevance of this information is to complement the previous findings with the dependencies of other variables to know the dependence and relationship between the most important variables and the most correlated to them; this helps complete the overview of the processes that affect the economic recessions.

The dependence plot for the most important variables is shown in Fig. 5. At the x-axis, the horizontal location is the actual value from the most correlated variable, and at the y-axis, the vertical location shows what having that value did to the prediction. Additionally, the relationship between both information is shown with a loess regression line. For positive slopes, this trend says that the more variable value, the higher the model's prediction is for the most correlated variable; it is the opposite with negative slopes.

As a result, two kinds of relationships are found: one with a positive trend at Fig. 5 plots A, B, D and F with a positive slope, having the highest correlation with 20-year–6month and 1-year–3-months term spread respectively. Besides, the positive trend with an asymptotic behavior at Fig. 5 plots C and E is found to correlate a 1-year–2-year term spread. In addition, the color gradient shows the y-axis feature value from light to dark when variables value is low to high, respectively. Generally speaking, the more considerable value of the most correlated variables, the smaller the SHAP value of this variable is. At this point, decile target lift, feature contribution, feature importance and feature dependence are presented; this information let understanding as early indicators which initial range variable values have more probability of having economic recessions and which variable are the most relevant for the economic recession process respectively.

To finalize, at epigraph II.C.4.c is proposed a methodology for identifying rules for economic recession detection. As a result of rules extraction and initial postprocessing, 359 rules are extracted followed by rules metrics; due to saving space, the table is not presented in the appendix; but this can be requested to the authors.

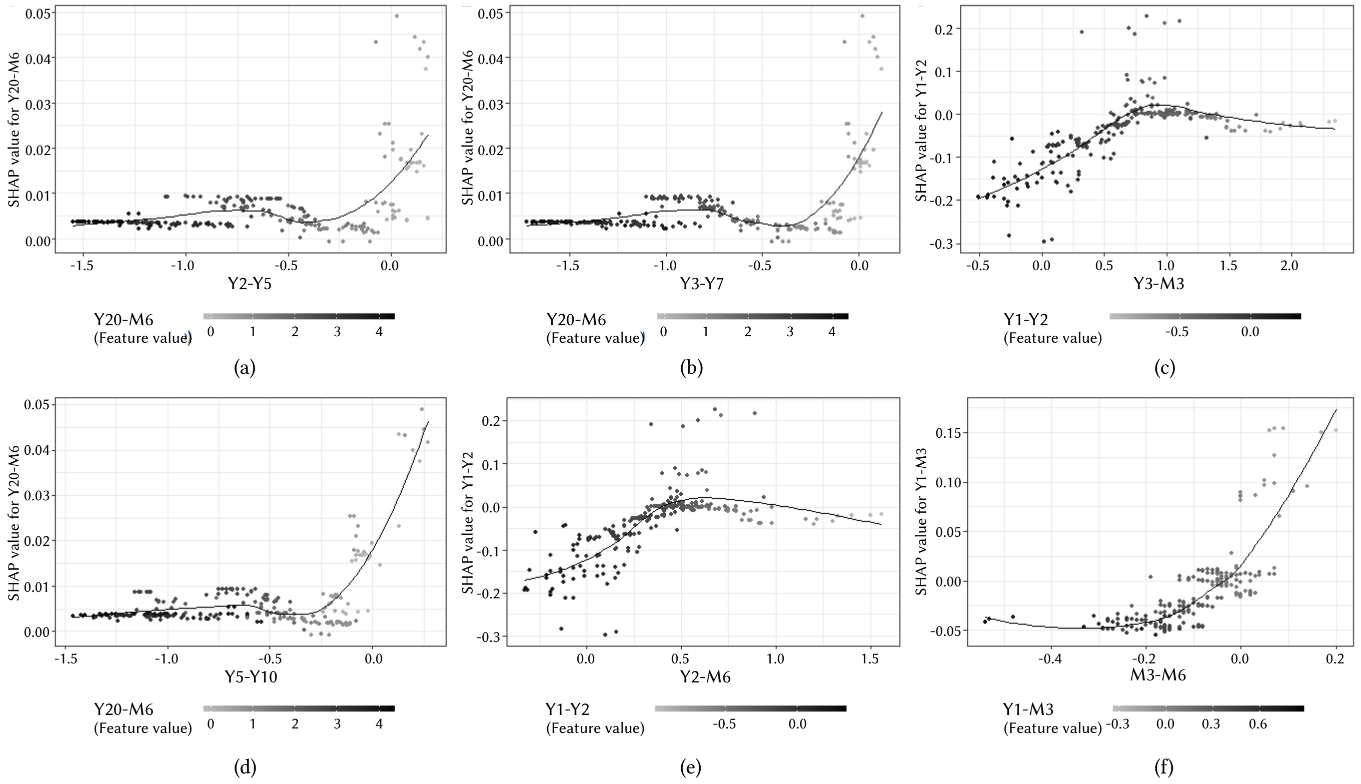
The extracted rules from the winning model can be filtered in several ways; as an initial exploratory study, this work proposes a frequency maximization and Lift Maximization criterion for discovering interesting rules. Frequency maximization criterion is when rules are sorted by support in descending order, and the first rules are the most frequent. The frequency maximization criterion does not sort results by lift, error or length metric for the rules.

TABLE VII. Top 5 XGB Max Support Rules

Rule	Error	Length	Support	Lift
M3-M6 ≤ 0.19	0.14	1	0.95	1.02
Y5-Y10 ≤ 0.35	0.12	1	0.95	0.84
Y2-Y5 ≤ 0.15	0.13	1	0.90	0.95
Y3-M3 > 0.26	0.12	1	0.85	0.90
Y3-Y7 ≤ -0.1	0.09	1	0.74	0.59

[a] Source is in an enclosed document, rules are obtained by sorting by support and selecting by the presence of top variables from SHAP results.
[b] M and Y are referred for monthly term and yearly respectively.

Table VII shows some rules for the Frequency maximization criterion, and results show a maximum Support for a rule of 0.95% of observations that satisfy the condition. By analyzing lift criterion, these rules show values nearly to 1, which is equivalent to saying that these rules could guarantee that there is no special probability of finding an economic recession compared with other data range; however, a rule with values near to 0 could show a high probability of not finding an economic recession. As previously explained, XGB is a tree-ensemble model through assembling simple trees, making a complex non-linear model. In this way, the rules extraction may provide rules with a low



(a) (b) (c) (d) (e) (f)

Fig. 5. SHAP dependence plot for most important variables and their most correlated features.

level of complexity. Due to this sorting method, the most important rules present low Length, low Lift and error rate, qualifying these as simplistic and inaccurate rules.

By sorting rules by lift in descending order, the first rules impact the economic recession detection more. Nevertheless, these rules could affect little observations, but as a recession is a rare event, support for recession identification should be a small percentage.

TABLE VIII. Top 5 XGB Max Lift and Support Rules

Rule	Error	Length	Support	Lift
Y2-M6≤-0.145 & Y20-M3>0.79	0	2	0.01	7.17
Y2-Y3 ≤ -0.12 & Y5-Y10 ≤ 0.04 & M3-M6 > 0.01	0	3	0.03	6.32
Y1-Y2 > -0.585 & Y2-M6 > 1.02	0	2	0.04	6.32
Y5-Y10 > 0.12 & Y5-Y20 ≤ 0.43 & Y20-M6 > -0.66	0	3	0.04	6.32
Y3-M3 > 0.45 & Y5-Y20 > 0.22 & Y5-M3 ≤ 1.32	0	3	0.04	6.32

[a] Source is in an enclosed document, rules are obtained by lift and selecting by the presence of top variables from SHAP results.
[b] M and Y are referred for monthly term and yearly respectively.

Table VIII shows some rules for lift maximization criterion; results show a maximum Lift for a rule of 7.17 times more probability of economic recession for the observations that satisfy the condition comparing the overall observations. Nonetheless, as an economic recession is a rare event, these rules usually have low support due to the nature of the economic recession, which is a rare event. More complex rules are found by this sorting criterion, with a low error rate and high probability of economic recession; therefore, the more interesting rules may be found. The interpretation of these rules is pretty straightforward, and a condition value is presented for every term spread involved in the rule; when this condition is satisfied, support, the percentage of observation that satisfies this rule is computed with the respective lift.

For the first rule, 2-year–6-month and 20-year–3-month are involved; this also indicates an interaction in the rule between these variables regarding the economic recession detection. Besides, the 20-year–the 3-month term spread is also an important term spread indicator as it may invert earlier than the 3-month–the 10-year term spread stated as relevant in previous studies [65].

Regarding the threshold values interpretation, the values are compared with the min, mean and max values for all the historical data for every term spread (see Table IX at appendix) in order to interpret the threshold value as a small, average or big value as those thresholds are closer to any of this feature descriptive statistics, in the case a value is close to two statistics the priority for the average is given. As a result, the first threshold number is labelled as a small value and the second as an average value. In this way, the qualitative interpretation of this rule will be formulated as follows: “When the 2-year–6-month term spread is lower or equal a small value and 20-year–3-month term spread is greater than the average value there is over seven times more probability of economic recession than the probability of economic recession for the complementary conditions”. Besides, historically this rule fulfilled the economic recessions accounted for 2008.

For the second rule, 2-year–3-year, 5-year–10-year and 3-month–6-month are involved, mainly describing an interaction between these variables regarding the economic recession detection. “When the Y2–Y3 and Y5–Y10 term spread is lower or equal of the average value of this term spread and greater than the average value of M3–M6 term spread, there is over six times more probability of economic recession than the probability of economic recession for the complementary conditions”. Besides, these conditions were fulfilled in the economic recessions accounted at 1990, 1991, 2001 and 2008.

The other rules from Table VIII can be described similarly to the previously explained rules, and these rules fulfil the conditions of the economic recession accounted at 1980, 1981, 1982, 1974 & 1970 years. This technique allows us to have a set of rules for detecting economic recession; with proper data updating & model retraining, these rules can be used in real life and act consequently with economic policies, among other uses.

To summarize the findings, Table X shows the main results except for dependencies analysis results.

TABLE X. Summary TABLE of Empirical Results

Variables	Most Correlated	Decile lift	SHAP(+)	Rules Support	Rules Lift
M3-M6	Y1-M3	Low-High	High	✓	✓
Y2-Y5	Y20-M6	Mid-High	Low-Mid	✓	×
Y5-Y10	Y20-M6	Mid-High	High	✓	✓
Y3-Y7	Y20-M6	Mid-High	Low-Mid	✓	×
Y3-M3	Y1-Y2	Low-Mid	Low-Mid	✓	✓
Y2-M6	Y1-Y2	Low-High	Low	×	✓

As a result, main variables on predicting economic recession are detected, and the variable dependence concerning the most correlated is studied; the SHAP value for positive economic recession is taken into account with the preliminary information of Decile Target Lift. Besides, some of the top rules contain the most important variables and fulfil the ideas mentioned in this work.

IV. Conclusion

Regarding the term structure, long-term rates could explain changes in future short-term rates. Understanding the term structure and yield curve, our goal is to create an interpretable forecasting model that can accurately inform us about future recessions, which could be a valuable tool for practitioners, researchers, governments and central banks. For three main groups, the public sector and the private sector are households, banks and investors, and the Federal Reserve. From an investors point of view, this information could be useful to make the right decisions for investing considering different strategies regarding this information, as the expanding economic activity is correlated with the stock market expansion [66]. By using the term spread to know in advance a possible economic recession, Federal Reserve could modify the interest rates to try to reduce the effect of this phenomenon.

Relevant term spreads are found, 3-month - 6-month, 2-year–5-year, 5-year–10-year, 3-year–7-year, 3-year–3-month and 2-year–6-month. Furthermore, for these variables, the lift metric is computed in order to detect initial intervals with a higher probability of accounting for a recession which is complementary to the SHAP contribution values analysis, applied into the rules description methods implementing the necessary policy mix they can dampen the effects of the recession, minimize its duration, or steer the economy away from it altogether. As the model provides some false negative alarms, we expect that implementing fiscal and monetary policy may put some inflationary pressure on the economy.

Finally, the methodology proposes a novelty application in this topic by extracting rules for economic recession understanding and detection. With this technique, several descriptive conditions allow the user to understand this phenomenon and have indicators with the goal of detecting to minimize the magnitude of the effect of the recession.

It is important to note that the yield curve's predictive power is statistical evidence and that, despite its accuracy, it is impossible to assure future results.

Thus, we encourage validating and updating these rules with reasonable frequency as the market evolves.

The literature suggests that the USA's best predictor of economic recessions is the 3-month-10-year term spread. Nevertheless, we found that the 3-month-6-month spread is the most relevant for detecting recessions, including the main recession detection rules. Therefore, monitoring this spread can be a useful tool for recession identification and a valid indicator for market expectations. In this context, it is found that the best rule associates this short-term 3-month-6-month predictor with the long-term term spreads, such as 5-year-10-year and 2-year-3-year, illustrating the rule as "When the Y2-Y3 and Y5-Y10 term spread is lower or equal of the average value of this term spread and greater than the average value of M3-M6 term spread there is over six times more probability of economic recession than the probability of economic recession for the complementary conditions".

As a future work suggestion, several paths can be followed. On one accuracy side, the improvement of the model predictive accuracy is relevant to have tools with high quality and impact on predicting this phenomenon. On the interpretability side, as different exogenous variables can be added, more study on the variable interactions can be performed to understand the yield curve inversion with other variables relevant for generating policies to prevent and control. On the rules generation side, as rules are potentially changing over time as variable importance may variate, a predictive maintenance system could be proposed to keep rules updated and valid over time.

Appendix

TABLE V. SHAP Values for Train and Test Set

Feature	Training	Test
M3-M6	0.2095	0.2217
Y2-Y5	0.1571	0.1986
Y5-Y10	0.1674	0.1394
Y3-Y7	0.0837	0.1012
Y3-M3	0.0939	0.1002
Y2-M6	0.1022	0.0949
Y1-M6	0.1745	0.0824
Y1-Y20	0.0877	0.0778
Y2-Y10	0.071	0.0778
Y1-Y10	0.0422	0.0699
Y2-Y3	0.0442	0.0474
Y1-Y2	0.0827	0.0445
Y5-M3	0.0355	0.0432
Y3-Y10	0.0337	0.0431
Y10-Y20	0.0731	0.0403
Y5-Y7	0.0536	0.0403
Y7-Y10	0.0363	0.0403
Y1-Y3	0.0292	0.0381
Y5-Y20	0.0545	0.0321
Y2-Y7	0.0272	0.0278
Y3-Y5	0.0213	0.0262
Y7-Y20	0.0585	0.025
Y2-M3	0.0294	0.0248
Y1-M3	0.0675	0.0244
Y1-Y7	0.0134	0.0194
Y10-M6	0.0145	0.0188
Y1-Y5	0.0209	0.0186
Y10-M3	0.011	0.017
Y2-Y20	0.0081	0.0116
Y7-M3	0.0065	0.0096
Y20-M3	0.0173	0.0093
Y3-M6	0.0036	0.0087
Y7-M6	0.0076	0.0078
Y3-Y20	0.0137	0.0067
Y20-M6	0.0108	0.0066
Y5-M6	0.0015	0.0007

[a] Term spread are sorted by SHAP values in percent scale of test set.

TABLE VI. Pearson Correlation Coefficient for the Most Correlated Variable

Variable	Correlated	Correlation
Y1-Y10	Y20-M6	-0,98
Y20-M6	Y1-Y10	-0,98
Y1-Y7	Y20-M6	-0,97
Y1-Y5	Y10-M6	-0,97
Y10-M6	Y1-Y5	-0,97
Y1-Y20	Y20-M6	-0,97
Y7-M6	Y1-Y5	-0,97
Y2-Y5	Y20-M6	-0,96
Y20-M3	Y1-Y7	-0,96
Y2-Y7	Y20-M6	-0,96
Y10-M3	Y1-Y5	-0,96
Y1-Y3	Y7-M6	-0,96
Y5-M6	Y1-Y3	-0,96
Y7-M3	Y1-Y3	-0,95
Y1-Y2	Y5-M6	-0,94
Y2-Y10	Y20-M6	-0,94
Y2-Y3	Y20-M6	-0,94
Y5-M3	Y1-Y3	-0,94
Y3-Y5	Y20-M6	-0,93
Y3-Y7	Y20-M6	-0,93
Y3-M6	Y1-Y2	-0,92
Y2-Y20	Y20-M6	-0,91
Y3-Y10	Y20-M6	-0,90
Y3-M3	Y1-Y2	-0,90
Y5-Y7	Y20-M6	-0,87
Y3-Y20	Y20-M6	-0,87
Y5-Y10	Y20-M6	-0,83
Y2-M6	Y1-Y2	-0,81
Y5-Y20	Y20-M6	-0,80
Y2-M3	Y1-Y2	-0,79
Y1-M3	M3-M6	-0,75
M3-M6	Y1-M3	-0,75
Y7-Y20	Y20-M6	-0,73
Y7-Y10	Y20-M6	-0,73
Y10-Y20	Y20-M6	-0,65
Y1-M6	M3-M6	-0,44

TABLE IX. Term Spread Descriptive Statistics

Feature	mean	median	min	max	sd
Y1-Y2	-0.29	-0.31	-1.06	0.95	0.34
Y1-Y3	-0.46	-0.51	-1.63	1.77	0.55
Y1-Y5	-0.75	-0.77	-2.50	2.35	0.81
Y1-Y7	-0.98	-1.02	-2.87	2.82	0.99
Y1-Y10	-1.14	-1.18	-3.40	3.07	1.15
Y1-Y20	-1.42	-1.33	-4.15	3.33	1.38
Y1-M3	0.52	0.43	-0.94	2.93	0.44
Y1-M6	0.39	0.31	-0.39	1.60	0.32
Y2-Y3	-0.17	-0.17	-0.59	0.83	0.22
Y2-Y5	-0.49	-0.46	-1.55	1.41	0.53
Y2-Y7	-0.74	-0.71	-2.28	1.88	0.72
Y2-Y10	-0.93	-0.85	-2.83	2.13	0.91
Y2-Y20	-1.30	-1.14	-3.67	2.39	1.19
Y2-M3	0.79	0.72	-1.76	3.86	0.66
Y2-M6	0.69	0.61	-0.82	2.44	0.54
Y3-Y5	-0.30	-0.27	-0.99	0.58	0.31
Y3-Y7	-0.53	-0.50	-1.72	1.05	0.52
Y3-Y10	-0.69	-0.60	-2.36	1.30	0.71
Y3-Y20	-1.01	-0.82	-3.27	1.56	1.00
Y3-M3	0.97	0.98	-2.01	4.11	0.80
Y3-M6	0.85	0.86	-1.20	2.74	0.69
Y5-Y7	-0.23	-0.21	-0.76	0.47	0.22
Y5-Y10	-0.39	-0.31	-1.46	0.72	0.42
Y5-Y20	-0.73	-0.60	-2.47	1.25	0.72
Y5-M3	1.27	1.33	-2.25	4.33	0.99
Y5-M6	1.15	1.20	-1.56	3.12	0.89
Y7-Y10	-0.16	-0.11	-0.74	0.38	0.22
Y7-Y20	-0.50	-0.42	-1.80	0.84	0.52
Y7-M3	1.50	1.56	-2.49	4.46	1.12
Y7-M6	1.37	1.43	-2.03	3.31	1.03
Y10-Y20	-0.34	-0.34	-1.06	0.87	0.34
Y10-M3	1.66	1.74	-2.65	4.42	1.24
Y10-M6	1.53	1.59	-2.28	3.64	1.16
Y20-M3	1.93	2.01	-3.00	4.44	1.41
Y20-M6	1.80	1.79	-2.54	4.36	1.35
M3-M6	-0.12	-0.10	-1.45	1.01	0.19

References

[1] A. Estrella, “The yield curve as a leading indicator: frequently asked questions,” *Federal Reserve Bank of New York*, 2005.

[2] M. J. Holmes, J. Otero, and T. Panagiotidis, “The expectations hypothesis and decoupling of short- and long-term US interest rates: A pairwise approach,” *The North American Journal of Economics and Finance*, vol. 34, pp. 301-313., 2015, doi: 10.1016/j.najef.2015.09.014.

[3] W. Liu and E. Moench, “What predicts US recessions?,” *International Journal of Forecasting*, vol. 34, no. 4, 2016, doi: 10.1016/j.ijforecast.2016.02.007.

[4] R. J. Shiller and J. Huston McCulloch, “Chapter 13 The term structure of interest rates,” *Handbook of Monetary Economics*. vol. 1, pp. 627-722, 1990, doi: 10.1016/S1573-4498(05)80016-5.

[5] A. Estrella and G. A. Hardouvelis, “The Term Structure as a Predictor of Real Economic Activity,” The *Journal of Finance*, vol. 46, pp. 555-576, 1991, doi: 10.1111/j.1540-6261.1991.tb02674.x.

[6] E. Weber and J. Wolters, “The US term structure and central bank policy,” *Applied Economics Letters*, vol. 41, pp. 41-45, 2012, doi: 10.1080/13504851.2011.566171.

[7] E. Weber and J. Wolters, “Risk and Policy Shocks on the US Term Structure,” *Scottish Journal of Political Economy*, vol. 19, pp. 41-45, 2013, doi: 10.1111/sjpe.12004.

[8] J. Y. Campbell, “Some Lessons from the Yield Curve,” *Journal of economic perspectives*, vol. 9, no. 3, pp. 129-152, 1995, doi: 10.1257/jep.9.3.129.

[9] B. S. Bernanke and A. S. Blinder, “The federal funds rate and the channels of monetary transmission,” *The American Economic Review., pp. 901-921*, 1992, doi: 10.2307/2117350.

[10] J. C. Vides, J. Iglesias, and A. A. Golpe, “The term structure under non-linearity assumptions: New methods in time series,” in *Contributions to Management Science*, pp. 117-136, 2018.

[11] K. R. Vetzal, “A survey of stochastic continuous time models of the term structure of interest rates,” *Insurance: Mathematics and Economics*, vol. 14, no. 2, pp. 139-161, 1994, doi: 10.1016/0167-6687(94)00009-3.

[12] A. Estrella and M. R. Trubin, “The Yield Curve as a Leading Indicator: Some Practical Issues,” *Current issues in Economics and Finance*, vol. 12, no. 5, 2006.

[13] A. Estrella and F. S. Trubin, “The yield curve as a predictor of US recessions,” Current issues in economics and finance, vol. 2, no. 7, 1996.

[14] W. Poole, R. H. Rasche, and D. L. Thornton, “Market Anticipations of Monetary Policy Actions,” *Federal Reserve Bank of St. Louis Review*, vol. 84, no. 4, pp. 65-94, 2002, doi: 10.20955/r.84.65-94.

[15] A. Estrella and G. A. Hardouvelis, “Possible roles of the yield curve in monetary policy.” Federal Reserve Bank of New York (Ed.), Intermediate targets and indicators for monetary policy, pp. 339-362, 1990.

[16] M. J. Dueker, “Strengthening the Case for the Yield Curve as a Predictor of US Recessions,” *Federal Reserve Bank of St. Louis Review*, pp. 41-51, 1997.

[17] R. D. Laurent, “An interest rate-based indicator of monetary policy,” Economic Perspectives, pp. 3-14, 1988.

[18] C. R. Harvey, “Forecasts of Economic Growth from the Bond and Stock Markets,” *Financial Analysts Journal*, vol 45. no. 5, pp. 38-45, 1989, doi: 10.2469/faj.v45.n5.38.

[19] J. H. Stock & M. W. Watson, “New indexes of coincident and leading economic indicators,” NBER macroeconomics annual, vol. 4, pp. 351-394, 1989.

[20] N. F. Chen, “Financial investment opportunities and the macroeconomy,” The Journal of Finance, vol. 46, no. 2, pp. 529-554, 1991, doi: 10.1111/j.1540-6261.1991.tb02673.x.

[21] C. R. Harvey, “Term structure forecasts economic growth. Financial Analysts Journal”, vol. 49, no. 3, pp. 6-8, 1993, doi: 10.2469/faj.v49.n3.6.2.

[22] M. Dotsey, “The predictive content of the interest rate term spread for future economic growth,” FRB Richmond Economic Quarterly, vol. 84, no. 3, pp. 31-51, 1998.

[23] J.D. Hamilton & D.H. Kim, "A re-examination of the predictability of the yield spread for real economic activity," Journal of Money, Credit, and Banking, vol. 34, no. 2, pp. 340-360, 2002.

[24] A. Estrella, “Why does the yield curve predict output and inflation?,” *The Economic Journal*, vol. 115, no. 505, pp. 722-744, 2005, doi: 10.1111/j.1468-0297.2005.01017.x.

[25] A. Ang, M. Piazzesi, and M. Wei, “What does the yield curve tell us about GDP growth?,” *Journal of econometrics*, vol. 131, no. 1-2, pp. 359-403, 2006, doi: 10.1016/j.jeconom.2005.01.032.

[26] D. C. Wheelock and M. E. Wohar, "Can the term spread predict output growth and recessions? A survey of the literature," *Federal Reserve Bank of St. Louis Review*, vol. 91, no. 5, pp. 419-440, 2009, doi: 10.20955/r.91.419-440.

[27] A. Estrella, A. P. Rodrigues, and S. Schich, "How stable is the predictive power of the yield curve? Evidence from Germany and the United States," *Review of Economics and Statistics*, vol. 85, no. 3, pp. 629-644, 2003, doi: 10.1162/003465303322369777.

[28] G. D. Rudebusch and J. C. Williams, "Forecasting recessions: The puzzle of the enduring power of the yield curve," *J ournal of Business & Economic Statistics*, pp. 492-503, 2009, doi: 10.1198/jbes.2009.07213.

[29] H. Nyberg, "Dynamic probit models and financial variables in recession forecasting," *Journal of Forecasting*, 2010, doi: 10.1002/for.1161.

[30] Lahiri, K., Monokroussos, G., & Zhao, Y. "The yield spread puzzle and the information content of SPF forecasts," Economics Letters, vol. 118, no. 1, pp. 219-221, 2013.

[31] M. Chinn and K. Kucko, "The Predictive Power of the Yield Curve Across Countries and Time," *International Finance*, vol. 18, no. 2, pp. 129-156, 2015, doi: 10.1111/infi.12064.

[32] A. Evgenidis, A. Tsagkanos, and C. Siriopoulos, "Towards an asymmetric long run equilibrium between stock market uncertainty and the yield spread. A threshold vector error correction approach," *Research in International Business and Finance*, 2017, doi: 10.1016/j.ribaf.2016.08.002.

[33] B. Gebka and M. E. Wohar, "The predictive power of the yield spread for future economic expansions: Evidence from a new approach," *Economic Modelling*, 2018, doi: 10.1016/j.econmod.2018.06.018.

[34] A., Evgenidis, S. Papadamou, and C. Siriopoulos, "The yield spread's ability to forecast economic activity: What have we learned after 30 years of studies?," Journal of Business Research, vol. 106, pp. 221-232.

[35] S. Ng, "Opportunities and Challenges: Lessons from Analyzing Terabytes of Scanner Data," in *Advances in Economics and Econometrics*, 2017.

[36] T. J. Berge, "Predicting Recessions with Leading Indicators: Model Averaging and Selection over the Business Cycle," *Journal of Forecasting*, 2015, doi: 10.1002/for.2345.

[37] P. Gogas, T. Papadimitriou, M. Matthaiou, and E. Chrysanthidou, "Yield Curve and Recession Forecasting in a Machine Learning Framework," *Computational Economics*, 2015, doi: 10.1007/s10614-014-9432-0.

[38] P. Gogas, T. Papadimitriou and E. Chrysanthidou, "Yield curve point triplets in recession forecasting," International Finance, 2015, doi: 10.1007/s10614-014-9432-0.

[39] J. Döpke, U. Fritsche, and C. Pierdzioch, "Predicting recessions with boosted regression trees," *International Journal of Forecasting*, 2017, doi: 10.1016/j.ijforecast.2017.02.003.

[40] S. D. Vrontos, J. Galakis and I. D. Vrontos, "Modeling and predicting US recessions using machine learning techniques," *International Journal of Forecasting*, 2020, doi: 1016/j.ijforecast.2020.08.005

[41] K. Bluwstein, M. Buckmann, A. Joseph, M. Kang, S. Kapadia, and Ö. Simsek, "Credit Growth, the Yield Curve and Financial Crisis Prediction: Evidence from a Machine Learning Approach," *SSRN Electron. J.*, 2020, doi: 10.2139/ssrn.3520659.

[42] P. Wei, Z. Lu, and J. Song, "Variable importance analysis: A comprehensive review," *Reliability Engineering and System Safety.* 2015, doi: 10.1016/j.ress.2015.05.018.

[43] Y. H. Yun, B. C. Deng, D. S. Cao, W. T. Wang, and Y. Z. Liang, "Variable importance analysis based on rank aggregation with applications in metabolomics for biomarker discovery," *Analytica chimica acta*, 2016, doi: 10.1016/j.aca.2015.12.043.

[44] S. Yang, W. Tian, Y. Heo, Q. Meng, and L. Wei, "Variable Importance Analysis for Urban Building Energy Assessment in the Presence of Correlated Factors," *Procedia Engineering*, 2015, doi: 10.1016/j.proeng.2015.08.1069.

[45] E. Vladislavleva, T. Friedrich, F. Neumann, and M. Wagner, "Predicting the energy output of wind farms based on weather data: Important variables and their correlation," *Renewable Energy*, 2013, doi: 10.1016/j.renene.2012.06.036.

[46] I. R. Dohoo, C. Ducrot, C. Fourichon, A. Donald, and D. Hurnik, "An overview of techniques for dealing with large numbers of independent variables in epidemiologic studies," *Preventive veterinary medicine*, 1997, doi: 10.1016/S0167-5877(96)01074-4.

[47] S. Tufféry, "*Data Mining and Statistics for Decision Making,*" *John Wiley & Sons*, 2011.

[48] A. J. Ferreira and M. A. T. Figueiredo, "Boosting algorithms: A review of methods, theory, and applications," in *Ensemble Machine Learning: Methods and Applications*, 2012.

[49] Q. Sun and B. Pfahringer, "Bagging ensemble selection," *In Australasian Joint Conference on Artificial Intelligence*, 2011, doi: 10.1007/978-3-642-25832-9_26.

[50] T. K. Ho, "Random decision forests," In Proceedings of 3rd international conference on document analysis and recognition, vol. 1, pp. 278-282, 1995. doi: 10.1109/ICDAR.1995.598994.

[51] E. M. Kleinberg, "Stochastic discrimination," *Annals of Mathematics and Artificial intelligence,* vol. 1, no. 1-4, pp. 278-282, 1990, doi: 10.1007/BF01531079.

[52] E. M. Kleinberg, "On the algorithmic implementation of stochastic discrimination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000, doi: 10.1109/34.857004.

[53] L. Breiman, "Random forests," Machine learning, vol. 45, no. 1, pp. 5-32, 2001.

[54] M. N. Wright and A. Ziegler, "Ranger: A fast implementation of random forests for high dimensional data in C++ and R," *Journal of Statistical Software*, 2017, doi: 10.18637/jss.v077.i01.

[55] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, 2001, doi: 10.1214/aos/1013203451.

[56] Tianqi Chen and Carlos Guestrin, "XGBoost: A Scalable Tree Boosting System," 22nd SIGKDD Conference on Knowledge Discovery and Data Mining, 2016, https://arxiv.org/abs/1603.02754

[57] C. Otte, "Safe and interpretable machine learning: A methodological review," *Studies in Computational Intelligence*, 2013, doi: 10.1007/978-3-642-32378-2_8.

[58] P. Wei, Z. Lu, and J. Song, "Variable importance analysis: A comprehensive review," *Reliability Engineering and System Safety.* 2015, doi: 10.1016/j.ress.2015.05.018.

[59] C. Strobl, A. L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures: Illustrations, sources and a solution," *BMC Bioinformatics*, 2007, doi: 10.1186/1471-2105-8-25.

[60] C. Strobl, A. L. Boulesteix, T. Kneib, T. Augustin, and A. Zeileis, "Conditional variable importance for random forests," *BMC Bioinformatics*, 2008, doi: 10.1186/1471-2105-9-307.

[61] S. M. Lundberg *et al.*, "From local explanations to global understanding with explainable AI for trees," *Nature machine intelligence*, 2020, doi: 10.1038/s42256-019-0138-9.

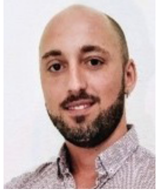
[62] S. Lipovetsky and M. Conklin, "Analysis of regression in game theory approach," *Applied Stochastic Models in Business and Industry*, 2001, doi: 10.1002/asmb.446.

[63] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," In Advances in neural information processing systems, pp. 4765-4774, 2017.

[64] H. Deng, "Interpreting tree ensembles with intrees," International Journal of Data Science and Analytics, vol. 7, no. 4, pp. 277-287, 2019. doi: 10.1007/s41060-018-0144-8.

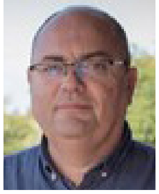
[65] A. Estrella, and F. S. Mishkin, "Predicting US recessions: Financial variables as leading indicators," Review of Economics and Statistics, vol. 80, no. 1, pp. 45-61, 1998.

[66] D. R. Cameron, "The expansion of the public economy: A comparative analysis," The American Political Science Review, pp. 1243-1261, 1978.



Pedro Cadahía Delgado

Principal Data Scientist PepsiCo. PhD Computer Science student in Economics. He has worked as Data Scientist in different fields and firms such as: DB Schenker, Minsait by Indra, Schibsted (Infojobs), Cofidis and PepsiCo.



Emilio Congregado

Emilio Congregado is Full Professor in the Economics Department at the University of Huelva. His main research interests are in the area of Entrepreneurship and labor economics. His work has been published in several scientific journals including Small Business Economics, Journal of Business Venturing, Empirical Economics, Journal of Policy Modeling, Annals of Regional Science or Economic Modeling.

Antonio Golpe Moya

Antonio A. Golpe is Associate Professor in the Economics Department at University of Huelva. His main research interests are in the area of Applied Economics. His work has been published in several scientific journals including Energy Economics, Annals of Regional Science, Renewable and Sustainable Energy Review, International Small Business Journal, Empirical Economics, Journal of Policy Modeling, Economic Modeling or Tourism Management.

Jose Carlos Vides

Jose Carlos Vides is Assistant Professor in the Department of Applied and Structural Economics and History at Complutense University of Madrid. His main research interests are in the area of Applied Economics. His work has been published in several scientific journals including Energy Economics, International Review of Economics and Finance, Journal of Policy Modeling, Empirica, Tourism Economics or FinanzArchiv: Public Finance Analysis.

Re-Evaluating the Relationship Between Economic Development and Self-Employment, at the Macro-Level: A Bayesian Model Averaging Approach

Ana Rodriguez-Santiago*

Vienna University of Economics and Business (Austria), University of Huelva (Spain) & CCTH - Centro Científico Tecnológico Huelva (Spain)

Received 21 June 2021 | Accepted 24 January 2022 | Published 26 January 2022

unir
LA UNIVERSIDAD
EN INTERNET

Abstract

We re-evaluate the relationship between stages of economic development and entrepreneurship, at the macro level. We first conduct a literature review of previous empirical research on cross-country determinants of entrepreneurship in order to put our contribution in perspective. To circumvent problems related to model uncertainty we use Bayesian Model Averaging (BMA) to evaluate the robustness of determinants of economic growth in a new dataset of 117 countries in the 2005-2019 period, allowing fixed effects and investigating the existence of heterogeneity allowing interactions of our focus variable with other regressors. Our empirical analysis then shows that the variation of self-employment rates across countries are mainly determined by variations in the unemployment, the stage of economic development and the variations in labor market frictions. When interactions are taken into account, results confirm that there is a differential effect of labor market frictions in countries with different levels of income. Frictions in labor market may encourage becoming self-employed in richer countries.

Keywords

Bayesian Model Averaging, Entrepreneurship, Macro-level Determinants, Model Uncertainty, Panel, Self-employment.

DOI: 10.9781/ijimai.2022.01.007

I. Introduction

The empirical literature on the macro-level determinants of entrepreneurship/self-employment[1] has analyzed a wider set of predictors as potential entrepreneurship drivers. These potential determinants relate to human capital[2], the level of development[3] and institutions[4]. There is a great number of studies in which a large set of regressors are included in so-called ʻad-hocˊ regressions, based on previous hypotheses and theoretical propositions.[5]

Whatever the type of specification is -structural or not[6] – and independently of the inclusion of a focus variable, we have a set of theories and propositions not mutually exclusive and, as in other fields of economics research, most of the empirical results in previous literature on the determinants of entrepreneurship at the macro-level have potential problems of model uncertainty, that is, regarding the choice of predictors.

To the best of our knowledge, we only can find two previous attempts to circumvent these potential problems. On the one hand, [16] adopted an algorithmic approach based on resampling and bootstrap techniques in a cross section of 69 countries for the year 2014, using data drawn from the Global Entrepreneurship Monitoring Database (henceforth, GEM). In short, the method is a step-by-step approach for finding the subset of explanatory variables leading the best possible prediction accuracy. With this strategy they select the more relevant regressors for explaining the national total entrepreneurship activity (TEA). The strength of Innovation and research and the level of entrepreneurial education are the best predictors in their analysis. [17] adopted an alternative solution. They applied a Bayesian model averaging (henceforth, BMA) to address the issue of model uncertainty in the framework of the literature on the determinants of self-employment, following the seminal contribution of [19], who combined the Bayesian Information criteria model weights and

[1] Throughout this paper, we use the terms entrepreneurship and self-employment synonymously and interchangeably. This operationalization of entrepreneurship as self-employment is dictated by data availability considerations.

[2] Educational attainment and sociodemographic characteristics.

[3] Economic development, macroeconomic stability -unemployment, inflation, government size–, financial development and access to finance and technological progress.

[4] Labor market institutions, Globalization, Administrative complexity and the rule of law, Taxes and Government.

[5] These works may be classified into two groups: with or without focus variable. For example, among the former are the works of [1]-[14] and among the latter the works of [15]-[17].

[6] The adjective structural describes how the specification is derived from a theoretical model. As [18] states, this approach allows to understand how the model is identified. The works of [3], [4], [13] and [14] are examples of this approach in the empirical literature on the determinants of entrepreneurship.

* Corresponding author.

E-mail address: ana.rodriguez@dege.uhu.es

maximum likelihood estimates for model selection, later revisited in the works of [20] and [21]. By using 32 predictors, aggregated into three groups – human capital, level of development and Institutions–, they use the BMA approach for correcting model uncertainty. With a short panel of 80 observations drawn from the GEM, the gross domestic product per capita, the unemployment and tax rates and the volatility of inflation are identified as the best predictors of the entrepreneurship rate, when model uncertainty is corrected for.

Despite the advantages of this last approach, the poor quality of the database and short period of observation and the non-consideration of interactions awake serious concerns about the robustness of the last two previous contributions. The problem may be particularly worrying if the relationship between self-employment and the potential regressors was dependent on the state of economic development, as suggested several previous contributions [22]-[25].

The present study aims to re-evaluate the robustness of the statistical significance of 21 macrolevel variables as predictors of the cross-country differences in the level of self-employment taking into account the potential parameter heterogeneity according to country development level. To this end, we use an extension of the BMA, suggested by [26], to re-evaluate the robustness of 21 determinants of self-employment in a new larger dataset of 117 countries during the period from 2005 to 2019, and investigate the existence of parameter heterogeneity allowing interactions between potential regressors and the stage of economic development based in panel data with fixed effects.

This article contributes to the previous empirical literature on self-employment determinants on the following grounds.

First, we provide new (and updated) empirical evidence on the drivers of self-employment in a much larger dataset than in the available empirical literature, including both developed and developing countries. As usual in prior related literature joint to our focus variable –the economic development proxied by GDP per capita–, a set of control variables are also included –e.g., proxies of different type of institutions, human capital, openness and technological progress, among others–.

Second, although previous empirical literature devoted to the identification the drivers of entrepreneurship across countries is considerable, there is a lack of consensus. Empirical evidence has not provided unambiguous results and as a result some controversies, about what are the drivers (and barriers) of entrepreneurship, have emerged, with deep policy implications. These inconsistencies may be due to the poor quality of data, to problems related with measurement issues of some variables and to the discretionary choice of predictors, the so-called model uncertainty [27].[7] To circumvent this problem of specification we use an extended version of the BMA for panel data allowing interactions and parameter heterogeneity [28] and [26] in which inference is based on a weighted average of all possible model specifications, not in a particular one. To the best of our knowledge this contribution is the first attempt to use the BMA approach with interactions in the context of the empirical literature on the determinants of entrepreneurship/self-employment.

Third, the data collected in the new database have been drawn from different sources –International Labor Organization Statistics, OECD Statistics, Penn World Tables (10.0), World Bank and World Intellectual Property Organization–. For measuring some explanatory variables, alternative indicators were taken into consideration to enlarge the sample.

Empirical support is found for the view that national self-employment rate is affected by unemployment, labor market frictions and the level of economic development –a nonlinear relationship consistent with the observed U-shape relationship between GDP and self-employment–. When interactions are considered, the key finding is that labor market frictions for the most advanced countries economic are found to be associated with higher self-employment rates.

The paper proceeds as follows. In Section II we conduct a brief description of the methodology that we employ and data. Section III describes the empirical results and, finally, Section IV concludes.

II. Methods and Data

A. Data

Our sample consists of a balanced panel data set formed by 117 economies over the period 2005–2019. Entrepreneurship is operationalized in terms of self-employment, reflecting data availability at the time-series level[8]. Entrepreneurship is defined as the self-employment rate, which is the number of business owners –employers and solo self-employed workers– divided by the total labor force.[9] The self-employment rate is drawn from the International Labor Organization Statistics (ILO-Statistics).

To explain the cross-national variations on self-employment rate, we include the 21 following variables (see Table A.II in the appendix for sources and descriptive statistics):

GDP per capita on purchasing power parity (PPP): gross domestic product converted to international dollars using purchasing power parity rates. Data are in constant 2017 international dollars.

Agriculture, Services and Industry correspond to the ISIC divisions 1-5, 50-99 and 10-45, respectively.

Exports/Imports of goods and services represent the value of all goods and other market services provided/received to/from the rest of the world.

Patent applications are worldwide patent applications filed through the Patent Cooperation Treaty procedure or with a national patent office for exclusive rights for an invention.

Internet users. This indicator captures the proportion of individuals using the Internet based on results from national household surveys.

Human capital index. Index provided by the Penn World Tables based on the average years of schooling and an assumed rate of return to education, based on Mincer equation estimates around the world.

Female Labor force participation rate. Proportion of females aged 15 and older who are economically active.

Frictions in Labor Markets. Following [13] we use the unemployment-wage employment ratio as an indicator of labor market frictions. He argues that labor market frictions make it more difficult to find a job and cause high levels of unemployment relative to wage employment, reducing the opportunity cost of self-employment.

Unemployment (Youth unemployment). Share the labor force that is without work but available for and seeking employment (in the age interval 15-24, for the younger age group).

Rural population. It refers to people living in rural areas as defined by national statistical offices. It is calculated as the difference between total population and urban population.

Total population. It is "de facto" definition of population, which counts all residents regardless of legal status or citizenship.

[7] The potential bias of ignoring this uncertainty is discussed in the works of [22], [23], [26] and [27]. See [29] for a detailed and recent survey.

[8] Table A.I and Fig. A.1 in the appendix show a list of the countries included on the sample and a map with the average self-employment rate over the sample period, respectively.

[9] This is a common practice, for convenience although it is aware that entrepreneurship is a multifaceted concept and look for better indicators is a major challenge for empirical research.

Inflation. Proxied by the annual growth rate of the GDP implicit deflator.

Taxes. The total tax and contribution rate measures the amount of taxes and mandatory contributions borne by the business in the second year of operation, expressed as a share of commercial profit. The labor tax and contributions measures all government mandated labor contributions that are borne by the business in the second year of operation, expressed as a share of commercial profit.

Doing Business. The score for starting a business is the simple average of the scores for each of the component indicators: the procedures, time and cost for an entrepreneur to start and formally operate a business, as well as the paid-in minimum capital requirement.

Control of Corruption. This index captures perceptions of the extent to which public power is exercised for private gain, including both petty and grand forms of corruption, as well as "capture" of the state by elites and private interests.

Government Effectiveness. It captures perceptions of the quality of public services, the quality of the civil service and the degree of its independence from political pressures, the quality of policy formulation and implementation, and the credibility of the government's commitment to such policies.

B. Methodology

As we mentioned, our objective is to select the appropriate specification or statistical model for the determinants of self-employment avoiding the personal discretion of the researcher. Consider the general model,

$$y = \alpha + X_k\beta_k + \varepsilon, \qquad \varepsilon \sim N(0,\sigma^2 I) \tag{1}$$

Where y is the self-employment rate and k is the number of regressors included, from all the possible regressors K. We are interested in the effect β of every particular variable and interaction included in X. With 21 possible variables, the cardinality of the model space including interactions would be 241, number of combinations of the 41 variables/interactions in models of size from 1 to 41. It is not possible to estimate around 2.199 billion models. If we could estimate all the models and get the probabilities of every model, the posterior distribution of the parameter β would be a weighting of the estimate of β from every particular model M_i times the probability that this model is true given the data.

$$p(\beta|y) = \sum_{i=1}^{2^K} p(\beta|y, M_i)\, p(M_i|y) \tag{2}$$

We use a BMA approach, first introduced by [19], to assess the implicit uncertainty across models. With BMA we assign a prior probability to a set of models and update it according to the data. Then, the posterior model probabilities (PMP) of the top models are averaged to calculate the posterior inclusion probabilities (PIP) for the potential determinants.

The PMP of every model is approximated by the marginal likelihood times the prior probability of the model, not conditional on the data.

$$p(M_i|y) \propto p(y|M_i)\, p(M_i) \tag{3}$$

The researcher is in charge of including the prior beliefs on the model prior. Non-informative prior will assume $p(M_i)=1/2^K$, assessing the same probability to all the possible models. Under this prior, the posterior model probability will be proportional to the marginal likelihood. It is the likelihood function after integrating away all the parameters of the model (α, β, σ):

$$p(y|M_i) = \iiint p(y|\, M_i, \alpha, \beta_k, \sigma)\, p(\alpha, \beta_k, \sigma)\; d\alpha\; d\beta_k\, d\sigma \tag{4}$$

Priors for model-specific parameters. Setting uninformative prior, we let the data speak. We establish non-informative priors on intercept $p(\alpha) \propto 1$ and on the deviation $p(\sigma) \propto 1/\sigma$. But, in order to find an analytical solution of the marginal likelihood, we need barely informative prior for coefficients β. We assume informative prior on β given σ by the g-prior by [30]

$$p(\beta_k|\sigma) \sim \mathcal{N}(0, \sigma^2 (gX'X)^{-1}) \tag{5}$$

This prior requires only elicitation of g. The variance-covariance matrix of β has the same structure of the variance-covariance matrix of OLS estimator, scaled with g, that determines the shrinkage in the regression parameters

$$E(\beta|y, M_i) = \frac{1}{1+g}(X'X)^{-1}X'y = \frac{1}{1+g}\hat{\beta}_{OLS} \tag{6}$$

The marginal likelihood for model M_i is given by

$$p(y|M_i) \propto \left(\frac{g}{1+g}\right)^{\frac{k_i}{2}} \left[\frac{1}{1+g} y' M_X\, y + \frac{g}{1+g}(y - \bar{y}_n)'(y - \bar{y}_n)\right]^{-\frac{n-1}{2}} \tag{7}$$

with the residual matrix $M_X = (I - X(X'X)^{-1} X')$.

The Bayes factor comparing M_i to the null model is given by

$$BF[M_i : M_0] = \frac{p(y|M_i)}{p(y|M_0)} = \left(1 + \frac{1}{g}\right)^{\frac{n-k_i-1}{2}} \left[1 + \frac{1}{g}(1 - R_i^2)\right]^{-\frac{n-1}{2}} \tag{8}$$

Fixing g, the marginal likelihood depends on how well the model fits the data and the size of the model. The use of the g-prior leads to a marginal likelihood which incorporates Occam's razor properties: For a given value of k_i, $p(y\,|\,M_i)$ and $BF[M_i : M_0]$ increase as goodness of fits increases, and for a given goodness of fit, $p(y\,|\,M_i)$ and $BF[M_i : M_0]$ increase as k_i decreases.

Literature has provided different options when choosing g. Unit Information Prior (UIP), proposed by [31], establishes $g = n$, which implies that the Bayes Factor mimics BIC [32]. Risk Inflation Criterion (RIC), proposed by [33], sets $g = K^2$, that minimizes the maximum increase in risk due to selecting rather than knowing the correct predictors. According to [34], we use the Benchmark prior (BRIC), $g = \max(n, K^2)$, that will decide between UIP or BIC depending on the number of potential regressors K and the sample size n.

Priors over the model space. We follow [21] for the specification of the prior model probabilities. We establish a fully random prior for the model and a binomial-beta hyperprior over prior inclusion probability with prior expected model size $\tilde{k} = K/2$. This hyper-prior leads to flat prior inclusion probability[10].

Related predictors. In order to know the different determinants of self-employment rate depending on the income level, we include in our model interactions of all the variables with the GDP per capita. Since we want to analyze the determinants of the self-employment comparing different level of development, we need to control by the effect of individual variables to compare the effect of the interactions. Following [26], we include the specification of strong heredity principle based on [35], which is a special case of George's dilution priors [36]. This way, we define prior probabilities across models where interactions are not present or are present with parent variables, and assign zero prior probability to models with interactions where some parent variable is not present.

The rationale behind this specification is that using a uniform prior over the model space we are interpreting an interaction term as an exclusive effect of that particular product of covariates and ignoring the independent effects of the interacted variables. Since we want to assess the differential effect of the covariates depending on GDP, we need to evaluate the significance of these interactions in a model which contains linear terms in both variables in addition to the interaction variable.

[10] In order to check robustness, we tried an informative specification for expected model size ($\tilde{k} = 5$). Results do not show significant change.

Computational Issues. Sampling from the model space. Following [20] we use Markov Chain Monte Carlo Model Composition (MC3) to approximate the posterior model probability. Starting with a random model with a random number of variables, we compute the posterior model probability and then propose a candidate model in the neighborhood of the first model, with one variable more or less, randomly chosen. Then, we can compare the posterior model probability with the previous one and keep the model with a higher value, that will be compared with a new candidate from the neighborhood. This procedure will visit models with higher non-negligible posterior model probability. Convergence of the MC3 sampler can be checked by computing the correlation between analytical and frequency-based posterior model probabilities for a region of the model space. We estimate 5.000.000 draws, discard the first 1.000.000 draws as the burn-in sample, and compute the results based on the top 100 models visited by the Markov chain.

Using the extension of the BMA methodology [20] to a panel data framework [28], we estimate a country fixed effects panel, including interactions terms with GDP per capita under the strong heredity prior over the model space. We present posterior inclusion probabilities (PIP)[11], the mean of the posterior distribution for each parameter (and interaction) and the corresponding posterior standard deviation (SD).

III. Results

Table I presents the results of the BMA exercise. We use the benchmark BRIC prior and establishes a binomial-beta prior on a prior expected model size of $K/2 = 20.5$. Using the strong heredity priors, we only evaluate models which contain the parent variables when interaction terms are included.

Fig. 1 shows the variables inclusion of variables with highest PIP on the top visited models and the sign when included. Our analysis, based on 21 covariates and the interaction of GDPpc with all the variables, presents a posterior mean model size of 11 variables but identifies only five variables/terms as significatively determinants of the self-employment.

First, GDP per capita presents a negative and statistically significant relationship with self-employment rate, in line with previous literature [13], [15], [17]. Cross-country analysis show self-employment rates are lower in richer countries [13] while some propositions and theories have attempted to provide a rationale for this negative relationship [37]-[38]. [23] distinguishes three major stages of development in self-employment rates. The first is characterized by high rates of non-agricultural self-employment. The second is characterized for a growing number of transitions to the wage-employment sector. As the economy becomes more developed fewer people become self-employed. In the third, the business sector expanded relative to manufacturing and the improvement in information technologies increase the returns of entrepreneurship. From this perspective, a U-shape relationship between self-employment and economic development emerges. Both arguments suggest a non-linear relationship as the significance of the coefficient associated to the quadratic GDPpc seems to indicate [39].

The next variables appearing as dominant determining the self-employment rates are related to the labor market. Unemployment rate emerges as negative and statistically significant, providing support to the entrepreneurial-pull hypothesis. As [40] states it has been a traditional source of controversy among economists, caused by the two competing hypotheses provided by the theory. The recession-push hypothesis which states that in times of crisis the lack of job opportunities pushes unemployed into self-employment. By contrast, the prosperity-pull mechanism suggests a positive comovement between self-employment and economic opportunities. If this relationship prevails in times of crisis, entrepreneurs are "pulled" out of self-employment, suggesting the existence of negative relationship between unemployment and self-employment. Previous empirical literature provides a large array of different results. As a result, the exact nature of the relation is still not clear, since we can only aspire to capture the net effect [15], [17]. Our results support the prosperity pull hypothesis.

TABLE I. BMA Results

Variable	PIP	M	SD
GDPpc	**1,00**	-40,71***	12,23
AGR	0,05	0,03	0,17
EXP	0,48	-0,01	0,01
IMP	**0,53**	-0,01	0,02
SER	0,11	0,00	0,02
IND	0,02	0,00	0,00
PAT	0,01	0,00	0,00
INT	**1,00**	-0,03	0,03
HUC	0,08	-0,40	2,15
LFF	**0,56**	0,04	0,05
UWE	**1,00**	-171,20***	12,28
UNE	**1,00**	-1,28***	0,10
UNY	0,08	0,00	0,03
RUR	0,21	0,14	0,31
POP	0,01	0,00	0,00
INF	0,02	0,00	0,00
TTX	0,05	0,00	0,01
LTX	**1,00**	0,03	0,17
BUS	**0,70**	-0,01	0,01
COR	**0,73**	-5,24	4,60
GOV	0,05	-0,03	0,25
GDPpc2	**0,95**	1,70***	0,59
GDPpc x AGR	0,04	0,00	0,02
GDPpc x EXP	0,01	0,00	0,00
GDPpc x IMP	0,02	0,00	0,00
GDPpc x SER	0,00	0,00	0,00
GDPpc x IND	0,00	0,00	0,00
GDPpc x PAT	0,00	0,00	0,00
GDPpc x INT	0,06	0,00	0,00
GDPpc x HUC	0,03	0,03	0,20
GDPpc x LFF	0,01	0,00	0,00
GDPpc x UWE	**1,00**	23,94***	1,60
GDPpc x UNE	0,01	0,00	0,01
GDPpc x UNY	0,00	0,00	0,00
GDPpc x RUR	0,19	-0,02	0,03
GDPpc x POP	0,00	0,00	0,00
GDPpc x INF	0,00	0,00	0,00
GDPpc x TTX	0,01	0,00	0,00
GDPpc x LTX	0,17	0,01	0,02
GDPpc x BUS	0,01	0,00	0,00
GDPpc x COR	**0,57**	0,49	0,47
GDPpc x GOV	0,00	0,00	0,02
PMS	**11,77**		
Corr. PMP	0,9998		

PIP, Posterior inclusion probability; M, mean of the posterior distribution parameter; SD, posterior standard deviation of the parameter; PMS, posterior mean model size; PMP, posterior model probability. Statistics based on the 100 most visited models by the Markov chain. Bold entries refer to variables who PIP>0.5. *, p<0.10; **, p<0.05; ***, p<0.01.

Finally, the frictions on the labor market are found to be a determinant of the variation of self-employment across countries. The relationship between the ratio unemployed by wage employees and self-employment is significant and negative. When checking the importance of interaction terms of GDPpc, only the one with the ratio U/WE appears to be significant. It means that economies with more frictions on the labor market tend to present lower self-employment rate, unless they have higher level of development, in which case the relationship between frictions and self-employment turns positive. This outcome is in line with the results provided by [1], [4], [13].

[11] PIP is considered robust when higher than the prior inclusion probability (π), which is expected model size by the number of variables. For the flat prior over the model space $\tilde{k} = K/2$, π = $\tilde{k}/K = 0.5$.

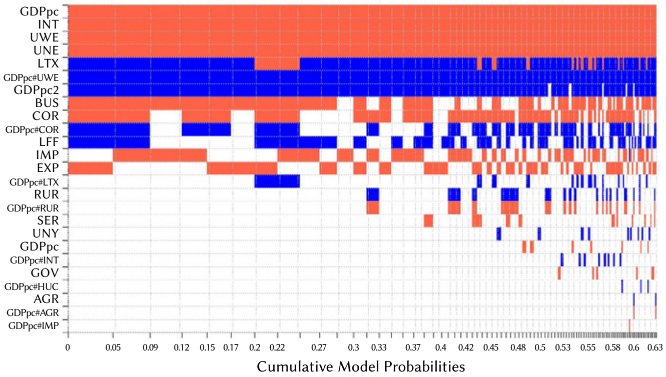


Fig. 1. Selected models probabilities. Inclusion and sign of variables. In blue, positive sign; in red, negative sign; and in white, non-inclusion.

IV. Conclusion

The contribution of this paper was to provide empirical evidence on the drivers of self-employment in a new and much larger –and harmonized– dataset than in the available empirical literature including 117 countries observed 20 periods and a set of 21 potential entrepreneurship determinants. As usual in prior related literature, joint to our focus variable –the economic development proxied by GDP per capita– a large battery of control variables is also included –e.g., GDP per capita square, institutions, human capital, openness and technological progress, among others– and data and we include a new proxy for capturing frictions in the labor market suggested by [13]. To circumvent problems associated to model uncertainty we adopted a BMA approach for panel. Our results provide a new explanation of the cross-country differences in the level of self-employment. We show that the unemployment rate, the frictions in the labor market and the stage of economic development are strong determinants of self-employment across the 117 countries included in our sample. Other potential drivers are not significantly correlated with self-employment.

Appendix

TABLE A.I. Countries in the Sample

Albania
Algeria
Angola
Argentina
Australia
Austria
Bangladesh
Belgium
Belize
Benin
Bolivia
Botswana
Brazil
Bulgaria
Burkina Faso
Burundi
Cambodia
Cameroon
Canada
Chile
China
Colombia
Rep. of Congo
Costa Rica
Cote d'Ivoire
Croatia
Czech Rep.
Denmark
Dominican Rep.
Ecuador
Egypt
El Salvador
Estonia
Eswatini
Fiji
Finland
France
Gabon
Gambia
Germany
Ghana
Greece
Guatemala
Haiti
Honduras
Hungary
Iceland
India
Indonesia
Iran
Iraq
Ireland
Israel
Italy
Jamaica
Japan
Jordan
Kazakhstan
Kenya
South Korea
Kyrgyz Rep.
Latvia
Lithuania
Luxembourg
Madagascar
Malawi
Malaysia
Mali
Mauritania
Mauritius
Mexico
Moldova
Mongolia
Morocco
Mozambique
Namibia
Nepal
Netherlands
New Zealand
Nicaragua
Nigeria
Norway
Pakistan
Panama
Paraguay
Peru
Philippines
Poland
Portugal
Romania
Russia
Rwanda
Saudi Arabia
Senegal
Serbia
Sierra Leone
Singapore
Slovak Rep.
Slovenia
South Africa
Spain
Sri Lanka
Sudan
Sweden
Switzerland
Thailand
Tunisia
Turkey
Uganda
Ukraine
UA Emirates
United Kingdom
United States
Uruguay
Vietnam
Zambia
Zimbabwe

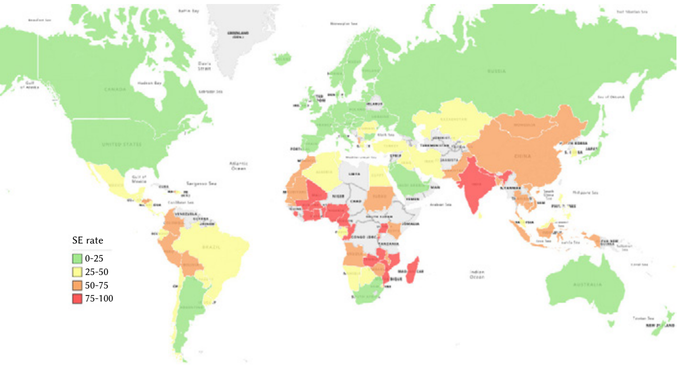


Fig. A.1. Average Self-Employment rate, 2005-2019.

TABLE A.II. Variable Description, Source and Statistics

Covariate	Code	Source	Mean	Min	Max
Dependent					
Self-employed (% of total employment)	SE	ILOSTAT	39,67	2,94	94,79
GDP and components					
Log GDP per capita, PPP (constant 2017 $)	GDPpc	World Bank	9,42	6,62	11,66
Agriculture, forestry, and fishing (% of GDP)	AGR	World Bank	10,03	0,03	60,28
Exports of goods and services (% of GDP)	EXP	World Bank	41,32	5,32	228,99
Imports of goods and services (% of GDP)	IMP	World Bank	44,29	9,00	208,33
Services (% of GDP)	SER	World Bank	54,53	12,81	79,33
Industry, including construction (% of GDP)	IND	World Bank	27,18	0,96	66,76
Technological Progress					
Patent applications, per million people	PAT	WIPO	217,75	0,00	4212,02
Individuals using the internet (% of population)	INT	World Bank	42,18	0,22	99,15
Human Capital					
Human Capital Index	HUC	PWT	2,62	1,12	4,35
Labor Market					
Labor force participation rate, female	LFF	ILOSTAT	51,73	11,28	87,12
Ratio Unemployed by Wage employees	UWE	ILOSTAT	0,17	0,00	1,13
Unemployment (% of total labor force)	UNE	ILOSTAT	7,50	0,39	29,25
Unemployment, youth (% of labor force 15-24yo)	UNY	ILOSTAT	16,58	0,60	58,00
Population					
Rural population (% of total population)	RUR	World Bank	39,65	0,00	90,63
Population, total in millions	POP	World Bank	55,00	0,28	1397,72
Institutions					
Inflation, GDP deflator	INF	World Bank	5,66	-26,10	95,41
Profit tax (% of profit)	TTX	Doing Business	44,96	14,10	285,90
Labor tax and contributions (% of profit)	LTX	Doing Business	19,17	0,00	68,00
Score-Starting a business	BUS	Doing Business	76,36	13,09	99,98
Control of Corruption	COR	Worldwide Governance Indicator	0,06	-1,53	2,47
Government effectiveness	GOV	Worldwide Governance Indicator	0,15	-2,08	2,44

Acknowledgment

The author wishes to thank Emilio Congregado, Jesús Crespo-Cuaresma and Concepción Román for many useful discussions. The author would like to thank the guest editor, Antonio Golpe, and two anonymous reviewers for their insightful comments that contributed substantially to the development of this paper. Financial assistance from the Spanish Ministry of Economy and Competitiveness through projects ECO2017-86402-C2-2-R, from Junta de Andalucía through Research Group SEJ-487 (Spanish Entrepreneurship Research Group – SERG) and from Research and Transfer Policy Strategy (Estrategia de Política de Investigación y Transferencia) 2020 of the University of Huelva is gratefully acknowledged.

References

[1] M.T. Robson, “Does stricter employment protection legislation promote self-employment?,” *Small Business Economics*, vol. 21, no. 3, pp. 309-319, 2003.

[2] J.W. Spencer, and C. Gomez, “The relationship among national institutional structures, economic factors, and domestic entrepreneurial activity: a multicountry study,” *Journal of business research*, vol. 57, no. 10, pp. 1098-1107, 2004.

[3] R. Torrini, “Cross-country differences in self-employment rates: The role of institutions,” *Labour Economics*, vol. 12, no. 5, pp. 661-683, 2005.

[4] V. Kanniainen, and T. Vesala, “Entrepreneurship and labor market institutions,” *Economic Modelling*, vol. 22, no. 5, pp. 828-847, 2005.

[5] R.S. Sobel, J.R. Clark, and D.R. Lee, “Freedom, barriers to entry, entrepreneurship, and economic progress,” *The Review of Austrian Economics*, vol. 20, no. 4, pp. 221-236, 2007.

[6] K. Nyström, “The institutions of economic freedom and entrepreneurship: evidence from panel data,” *Public Choice*, vol. 136, no. 3, pp. 269-282, 2008.

[7] C. Bjørnskov, and N.J. Foss, “Economic freedom and entrepreneurial activity: Some cross-country evidence,” *Public Choice*, vol. 134, no. 3, pp. 307-328, 2008.

[8] Y. Kim, W. Kim, and T. Yang, “The effect of the triple helix system and habitat on regional entrepreneurship: Empirical evidence from the US,” *Research Policy*, vol. 41, no. 1, pp. 154-166, 2012.

[9] P. Stenholm, Z.J. Acs, and R. Wuebker, “Exploring country-level institutional arrangements on the rate and type of entrepreneurial activity,” *Journal of Business Venturing*, vol. 28, no. 1, pp. 176-193, 2013.

[10] M.T.T. Thai, and E. Turkina, “Macro-level determinants of formal entrepreneurship versus informal entrepreneurship,” *Journal of Business Venturing*, vol. 29, no. 4, pp. 490-510, 2013.

[11] E. Autio, and K. Fu, “Economic and political institutions and entry into formal and informal entrepreneurship,” *Asia Pacific Journal of Management*, vol 32, no. 1, pp. 67-94, 2015.

[12] S. Aparicio, D. Urbano, and D. Audretsch, “Institutional factors, opportunity entrepreneurship and economic growth: Panel data evidence,” *Technological forecasting and social change*, vol. 102, pp. 45-61, 2016.

[13] M. Poschke, “Wage Employment, Unemployment and Self-Employment across countries,” *IZA DP No. 12367*, 2019.

[14] A.F. Shapiro, and F.S. Mandelman, “Digital adoption, automation, and labor markets in developing countries,” *Journal of Development Economics*, vol. 151, pp. 102656, 2021.

[15] C. Pietrobelli, R. Rabellotti, and M. Aquilina, “An empirical study of the determinants of self-employment in developing countries,” *Journal of International Development*, vol. 16, no. 6, pp. 803-820, 2004.

[16] J.I. Gimenez-Nadal, M. Lafuente, J.A. Molina, and J. Velilla, “Resampling and bootstrap algorithms to assess the relevance of variables: applications to cross section entrepreneurship data,” *Empirical Economics*, vol. 56, no. 1, pp. 233-267, 2019.

[17] K.P. Arin, V.Z. Huang, M. Minniti, A.M. Nandialath, and O.F. Reich, “Revisiting the determinants of entrepreneurship: A Bayesian approach,” *Journal of Management*, vol. 41, no. 2, pp. 607-631, 2015.

[18] H. Low, and C. Meghir “The use of structural models in econometrics,” *Journal of Economic Perspectives*, vol. 31, no. 2, pp. 33-58, 2017.

[19] A.E. Raftery, “Bayesian model selection in social research,” *Sociological methodology*, pp. 111-163, 1995.

[20] C. Fernandez, E. Ley, and M.F. Steel, “Model uncertainty in cross-country growth regressions,” *Journal of applied Econometrics*, vol. 16, no. 5, pp. 563-576, 2001.

[21] E. Ley, and M.F. Steel, “On the Effect of Prior Assumptions in Bayesian Model Averaging with Applications to Growth Regression,” *Journal of Applied Econometrics*, vol. 24, pp. 651-674, 2009.

[22] Z.J. Acs, D.B. Audretsch, and D.S. Evans, “Why does the self-employment rate vary across countries and over time?,” Discussion Paper No. 871, *Center for Economic Policy Research*, 1994.

[23] Z.J. Acs, S. Desai, and J. Hessels, “Entrepreneurship, economic development and institutions,” *Small Business Economics*, vol. 31, no. 3, pp. 219-234, 2008.

[24] M. Carree, A. van Stel, R. Thurik, and S. Wennekers, “Economic development and business ownership: An analysis using data of 23 OECD countries in the period 1976–1996,” *Small Business Economics*, vol. 19, no. 3, pp. 271-290, 2002.

[25] S. Wennekers, A. van Stel, R. Thurik, and P. Reynolds, “Nascent entrepreneurship and the level of economic development,” *Small Business Economics*, vol. 24, no. 3, pp. 293-309, 2005.

[26] J. Crespo-Cuaresma, “How different is Africa? A Comment on Masanjala and Papageorgiou,” *Journal of Applied Econometrics*, vol. 26, pp. 1041-1047, 2011.

[27] C. Young, “Model uncertainty in sociological research: an application to religion and economic growth,” *American Sociological Review*, vol. 74, no. 3, pp. 380-397, 2009.

[28] E. Moral-Benito, “Determinants of economic growth: a Bayesian panel data approach,” *Review of Economics and Statistics*, vol. 94, no. 2, pp. 566-579, 2012.

[29] E. Moral-Benito, “Model averaging in economics: An overview,” *Journal of Economic Surveys*, vol. 29, no. 1, pp. 46-75, 2015.

[30] A. Zellner, “Bayesian estimation and prediction using asymmetric loss functions,” *Journal of the American Statistical Association*, vol. 81, no. 394, pp. 446-451, 1986.

[31] R.E. Kass, and L. Wasserman, “A reference Bayesian test for nested hypotheses and its relationship to the Schwarz criterion,” *Journal of the American Statistical Association*, vol. 90, no. 431, pp. 928-934, 1995.

[32] F. Liang, R. Paulo, G. Molina, M.A. Clyde, and J.O. Berger, “Mixtures of g priors for Bayesian variable selection,” *Journal of the American Statistical Association*, vol. 103, no. 481, pp. 410-423, 2008.

[33] D.P. Foster, and E.I. George, “The risk inflation criterion for multiple regression,” *The Annals of Statistics*, pp. 1947-1975, 1994.

[34] C. Fernandez, E. Ley, and M.F. Steel, “Benchmark priors for Bayesian model averaging,” *Journal of Econometrics*, vol. 100, no. 2, pp. 381-427, 2001.

[35] H. Chipman, “Bayesian variable selection with related predictors,” *Canadian Journal of Statistics*, vol. 24, no.1, pp. 17-36, 1996.

[36] E.I. George, “[Bayesian Model Averaging: A Tutorial]: Comment,” *Statistical Science*, vol. 14, no. 4, pp. 409-412, 1999.

[37] G. Yamada, “Urban informal employment and self-employment in developing countries: theory and evidence,” *Economic development and cultural change*, vol. 44, no. 2, pp. 289-314, 1996.

[38] S. Kuznets, “Modern economic growth,” *Yale University Press*, 1966.

[39] S. Wennekers, A. van Stel, M. Carree, and R. Thurik, “The relationship between entrepreneurship and economic development: Is it U-shaped?,” *Now Publishers Inc.*, 2010.

[40] E. Congregado, A. Golpe, and A. van Stel, “The ‘recession-push’ hypothesis reconsidered,” *International Entrepreneurship and Management Journal*, vol. 8, no. 3, pp. 325-342, 2012.

Ana Rodríguez-Santiago

She is a visiting researcher in the Department of Economics, Vienna University of Economics and Business. She is a PhD student in the program in Economics, Business, Finance and Computer Science (University of Huelva). Ana was granted with a FPU fellowship by the Spanish Ministry of Education. Her main areas of research are Macroeconometrics and Labor Economics and her research focuses on Business Cycle analysis, Bayesian econometrics methods, time series analysis and finite mixture models. Her thesis focuses on the application of Bayesian methodologies on self-employment to fight model uncertainty and cluster economies based on individual characteristics.

Finite Sample Properties of Parameterized Expectations Algorithm Solutions; Is the Length So Determinant?

A. Jesús Sánchez-Fuentes

Complutense Institute of International Studies (ICEI-UCM) & GEN-UVigo (Spain)

Received 22 December 2020 | Accepted 10 February 2022 | Published 25 February 2022

Abstract

The solution of the Parameterized Expectations Algorithm (PEA) is well defined based on asymptotic properties. In practice, it depends on the specific replication of the exogenous shock(s) used for the resolution process. Typically, this problem is reduced when a sufficiently long replication is considered. In this paper, we suggest an alternative approach which consists of using several, shorter replications. A centrality measure (the median) is used then to discriminate among the different solutions using two different criteria, which differ in the information used. On the one hand, the distance to the vector composed by median values of PEA coefficients is minimized. On the other hand, distances to the median impulse response is minimized. Finally, we explore the impact of considering alternative approaches in an empirical illustration.

Keywords

Nonlinear Models, Numerical Solution Methods, Optimal Growth, Parameterized Expectations Algorithm.

DOI: 10.9781/ijimai.2022.02.007

I. Introduction

The Parameterized Expectations Algorithm (PEA) is a widely applied method for solving nonlinear stochastic dynamic models with rational expectations (see [1]-[8]) The PEA scheme involves approximating the conditional expectation functions in the Euler equations with certain parametric functions, and the use of a numerical optimization method to estimate parameter values.

A common problem with the PEA is that the solutions are obtained from a specific replication of exogenous random process(es). Due to the asymptotic properties of the algorithm and increasing computational costs, authors typically have considered only one replication of the exogenous shocks but with a sufficiently long simulation length. However, as [9] pointed out: *"... it may be necessary to use an extremely long simulation in order to obtain the same fitted coefficients of the approximating function across replications of the simulation"*. By contrast, [10] claim that only quantitative differences are observed (qualitative properties remain). In any event, they all agree that basing one's conclusions on a "non-unique" solution may cause results to be less robust than otherwise.

Contrary to [9] proposal which involves a parallel implementation of the PEA algorithm that enables a sufficiently long replication, we adopt a different approach consisting of sampling a sufficient number of shorter replications, in the framework of a Montecarlo experiment. A centrality measure (the median) is used to discriminate among the different solutions. The criteria differ in the information used. On the one hand, the distance to the vector composed by median values of PEA coefficients is minimized. On the other hand, distances to the median impulse response is minimized.

* Corresponding author.

E-mail address: ajsanchezfuentes@ucm.es

We consider two models to frame the discussion: the simple neoclassical growth model, and the Cooley and Hansen (1989) model, that adds to the previous model a non-convexity, indivisible labour and introduces money via a cash-in-advance constraint in consumption.

Our main conclusion is that the criterion choosing the replication closest to the median impulse-response function appears to be the most suitable criterion for several reasons: (i) there is no bias in using shorter simulations which allows one to face much lower computational costs, (ii) it shows how different solutions behave when a transitory/permanent shock is applied (qualitative robustness is indirectly checked), and (iii) it provides a band of confidence around the final choice (the level of uncertainty is measured). Additionally, summary statistics may be obtained from the distribution of estimated coefficients.

II. The PEA

Consider an economy characterized by a vector of **n** endogenous variables, $\mathbf{z}_t$, and by a vector of **s** exogenously given shocks, $\mathbf{u}_t$. Let the process $\{\mathbf{z}_t , \mathbf{u}_t\}$ be represented by a system

$$g(E_t[\phi(z_{t+1}, z_t)], z_t, z_{t-1}, u_t) = 0, for\ all\ t \tag{1}$$

where g: $\mathbb{R}^m\ x\ \mathbb{R}^n\ x\ \mathbb{R}^n\ x\ \mathbb{R}^s \rightarrow\ \mathbb{R}^q$ and ϕ: $\mathbb{R}^{2n} \rightarrow \mathbb{R}^m$; the vector $\mathbf{z}_t$ includes all endogenous and exogenous variables that are inside the expectation, and $\mathbf{u}_t$ follows a first-order Markov process. It is assumed that $\mathbf{z}_t$ is uniquely determined by (1) if the rest of arguments are given. The functions $\mathbf{g}(\cdot)$ and $\boldsymbol{\phi}(\cdot)$ are known functions once the structural parameters of the economy are fixed. Alternatively, let the solution be expressed as a law of motion **h** such that the vector $\mathbf{z}_t$ generated by $z_t = h(z_{t-1}, u_t)$ fulfills (1), given that all past information relevant to forecast $\phi(z_{t+1}, z_t)$ can be summarized in a finite-dimension function of $\{\mathbf{z}_{t-1} , \mathbf{u}_t\}$.

Obtaining a solution to (1) using PEA consists of finding a parametric function $\boldsymbol{\varphi}(\boldsymbol{\beta}; \mathbf{z}_{t-1} , \mathbf{u}_t)$, such that for a positive integer $\mathbf{v}$, $\beta \in \mathbb{D}^\nu$, where $\mathbb{D}^\nu \subset \{\beta \in \mathbb{R}^\infty$: *i-th element of β is zero if $i > \nu$*$\}$, the process

$\{\mathbf{z}_t(\boldsymbol{\beta})\}$ satisfies for all t the set of equations

$$g(E_t[\psi(\beta;z_{t-1},u_t)],z_t(\beta),z_{t-1}(\beta),u_t)=0 \tag{2}$$

and the order of **v** is such that when solving $G(\beta)=\arg\min_{\beta\in\mathbb{D}^{\nu}} E_t\left[\phi\left(z_{t+1}(\beta),z_t(\beta)\right)-\psi(\beta;z_{t-1}(\beta),u_t)\right]^2$, then $\beta = G(\beta)$. This problem is solved by use of the following Gauss-Newton updating rule: $\beta^i=\beta^{i-1}+\lambda G(\beta^{i-1})$ at each iteration *i*. Given these conditions, the stochastic process $\{\mathbf{z}_t(\boldsymbol{\beta})\}$ is the PEA approximated solution. Under certain regularity conditions over the functions defining the equilibrium in (1), the function $\mathbf{g}(\cdot)$ is invertible in its second argument, and equation (2) can be written as (see [3])

$$z_t(\beta)=h_\beta(z_{t-1}(\beta),u_t) \tag{3}$$

for stationary and ergodic processes. [11] shows that under those regularity conditions, fulfilled by standard business cycle models, it is always possible to find an approximated function $\mathbf{h}_\beta(\cdot)$ arbitrarily close to the true law of motion of the system $\mathbf{h}(\cdot)$. Under the true law of motion $h(z_{t-1},u_t)$, the true process $\{\mathbf{z}_t,\mathbf{u}_t\}_{-\infty}^{+\infty}$ verifying (1) is stationary. For the approximation to be acceptable, given initial conditions $\{\mathbf{z}_0,\mathbf{u}_0\}$ and an initial vector $\boldsymbol{\beta}$, the resulting process $\{\mathbf{z}_t(\boldsymbol{\beta})\}_{t=1}^{T}$ verifying (2) has to be stationary.

The PEA as presented in [2] can be written as follows:

- Step 1. Fix initial conditions $\mathbf{u}_0$ and $\mathbf{z}_0$; draw and fix a random series $\{\mathbf{u}_t\}_{t=1}^{T}$ from a given definition. Replace the conditional expectation in (1) with a function $\boldsymbol{\varphi}(\boldsymbol{\beta};\mathbf{z}_{t-1},\mathbf{u}_t)$ and compute $\boldsymbol{z}_t(\boldsymbol{\beta})$ from (3).
- Step 2. For a given $\beta\in\mathbb{D}^{\nu}$ recursively calculate $\{\mathbf{z}_t(\boldsymbol{\beta})\}_{t=1}^{T}$ according to $z_t(\beta)=h_\beta(z_{t-1}(\beta),u_t), if\ \underline{z}\le z_t(\beta)\le\overline{z}$
- Step 3. Find a $\mathbf{G}(\boldsymbol{\beta})$ that satisfies $G(\beta)=\arg\min_{\beta\in\mathbb{D}^{\nu}} E_t\left[\phi\left(z_{t+1}(\beta),z_t(\beta)\right)-\psi(\beta;z_{t-1}(\beta),u_t)\right]^2$. In order to perform this step, one can run a nonlinear least squares regression with the sample $\{z_t(\beta),u_t\}$, taking $\phi(z_{t+1}(\beta),z_t(\beta))$ as a dependent variable, $\boldsymbol{\varphi}(\cdot)$ as an explanatory function, and $\boldsymbol{\xi}$ as a parameter vector to be estimated.
- Step 4. Compute the vector $\boldsymbol{\beta}^{i+1}$ for the next iteration, $\beta^{i+1}=\beta^i+\lambda G(\beta^i), \lambda\in(0,1)$

Iterate on Steps 2-4 until $\|\beta^{i+1}-\beta^i\|$ is below a certain tolerance value, for all *t*.

III. The Models

For simplicity, and without loss of generality, we have selected two quite standard models to address the discussion: the one-sector stochastic growth model (SN henceforth) and the Cooley and Hansen model (CH henceforth) presented in [12]. Consider firstly the SN model,

$$\max_{\{c_t,k_t\}_{t=0}^{\infty}} E_0\sum_{t=0}^{\infty}\delta^t\frac{c_t^{1-\gamma}-1}{1-\gamma},\ s.t.\ c_t+k_t=(1-d)k_{t-1}+\theta_t k_{t-1}^{\alpha}$$

where $\log\theta_t=\rho\log\theta_{t-1})+\epsilon_t$ with $\epsilon_t\sim\mathcal{N}(0,\sigma^2)$, the initial condition (k_{-1},θ_0) is given. $\mathbf{c}_t$ is consumption at time *t*, k_{t-1} the beginning of period *t* capital stock, $\mathbf{0<\boldsymbol{\delta}<1}$ is the subjective discount factor, $\mathbf{0<\boldsymbol{\alpha}<1}$ the capital share in production, $\mathbf{0<d<1}$ the depreciation rate, and $\mathbf{0<\boldsymbol{\rho}<1}$. But for the case with logarithmic utility, $\gamma=1$, and full depreciation of capital, $\mathbf{d=1}$, a closed-form solution to this model is not known. Following [4], we approximate the conditional expectation by

$$E_t\left[c_{t+1}^{-\gamma}(1-d+\alpha\theta_{t-1}k_{t-1}^{\alpha})\right]\cong exp(\beta_0+\beta_1\log\theta_t+\beta_2\log k_{t-1})$$

where $\beta=(\beta_0,\beta_1,\beta_2)$ is a vector of coefficients to be found. To simulate the model, parameter values are fixed as:

$$\alpha=0.33,\delta=0.95,d=0.02,\rho=0.95,\sigma=0.01,k_{-1}=k_{ss}$$

(the subscript *ss* refers to the steady-state values) and $\theta_0=1$.

The CH model is slightly more complex in that it includes a non-convexity, indivisible labour. Money is introduced via a cash-in-advance constraint in consumption. The competitive equilibrium is non-Pareto-optimal and the second welfare theorem does not apply. The representative firm solves a standard profit maximization problem, while households seek to maximize their time preferences subject to their holdings of money balances and a set of standard budget constraints. There are two sources of uncertainty in this economy: an autoregressive shock to technology, θ_t, and an autoregressive logged money growth rate, $\log g_t$. Reference [3] preferred specification for the approximating function $\boldsymbol{\varphi}(\cdot)$ to the expectation term is a third-order polynomial such that,

$$\begin{aligned}E_t\left[c_{t+1}^{-\gamma}(+\alpha\theta_{t-1}k_{t-1}^{\alpha}N_{t-1}^{1-\alpha}+1-d)\right]&\\ \cong exp(\beta_0+\beta_1\log k_{t-1}+\beta_2\log\theta_t+\beta_3\log g_t&\\ +\beta_4(\log k_{t-1})^2&\\ +\beta_5\log k_{t-1}\log\theta_t&\\ +\beta_6(\log\theta_t)^2+\beta_7(\log\theta_t)^3)&\end{aligned}$$

where $\boldsymbol{\mu}_t$ is the Lagrange multiplier attached to the household's budget constraint, and N denote hours worked. Following [3], we will adopt as baseline parameterization:

$$\alpha=0.36,\delta=0.99,d=0.025,\rho=0.95,\rho_g=0.48,$$
$$\sigma_{\epsilon_g}=0.009,\sigma=0.00721,g_{ss}=1.15,A_N=2.86$$

IV. Selection Criteria Between PEA Solutions

Potential criteria should rely on the statistical properties of distributions obtained from the simulation of the models: estimated coefficients, simulated variables or impulse-response functions. Prior intuitions are next discussed. Firstly, how each solution behaves cannot be determined using the simulated series as they depend on the replication used.

Secondly, it looks reasonable, in line with studies in the field of the functional analysis facing similar problems (see [13]), to choose methods that reduce the probability of obtaining an "extreme" (less representative) solution: that with highest or lowest values of the reference distribution. Thus, commonly used concepts of centrality within a distribution may be considered. In this respect, we use the concept of median but our conclusions are robust to changes in the chosen measure.[1]

Thirdly, general or model-specific criteria based on economic theory can be applied as well. A chosen replication should verify some meaningful constraint(s) dictated by economic theory. Impulse-response functions are typically used to determine solutions behaviour. As an instance, in the context of the SN model, [10] discusses the convenience of a monotonic response of consumption to a technology shock (pp. 12-13).

Fourthly, note that values coming from different replications do not represent a PEA solution. Therefore, choosing independently *ideal* values is not an option and a compromised choice should be done.

In the light of previous arguments, we suggest the following selection criteria:

- *Minimum distance to median coefficients*

  First, we compute a vector composed by the median value for each coefficient $(\tilde{\beta})$. Next, we compute the distance of each vector

[1] Results remain if distances are computed with respect to either average values or the concept of *statistical depth* (see [16]).

of coefficients to the reference vector. Finally, the replication minimizing this distance, $\tilde{\beta}_{\tilde{r}}$, is chosen. Analytically,

$$r_{\tilde{\beta}}^{min} = \min_{r=1,\dots,R} \left\|\tilde{\beta} - \beta^{r}\right\|$$

where β^r is the coefficient vector of replication *r*.

- *Minima distance to median impulse response*

First, we compute the *median impulse response*, $\{\widetilde{IR}\}$, as a vector composed by the median value of different responses at period *h*, $\widetilde{ir}_h$, (*h* = 1, ..., *H*). Next, we compute the distance of each response to the reference response and, again, we choose those minimizing it. Analytically,

$$r_{\widetilde{IR}}^{min} = \min_{r=1,\dots,R} \left\|\widetilde{IR} - IR^{r}\right\|$$

where $\widetilde{IR} = \{\widetilde{ir}_h\}_{h=1}^{H}$ and $IR^r = \{ir_h\}_{h=1}^{H}$ is the vector composed by the response at period *h*, (*h*=1, ..., *H*) of replication *r*, ir_h^r.

Within this category, we distinguish between transitory ($r_{\widetilde{IR}_t}^{min}$) and permanent ($r_{\widetilde{IR}_p}^{min}$).

We consider previous criteria to discriminate among solutions. Consequently, we may define $r_{\tilde{\beta}}^{max}, r_{\widetilde{IR}_t}^{max}$ *and* $r_{\widetilde{IR}_p}^{max}$ as the replications which respectively maximize the corresponding distances. If no significant differences are observed between both choices (*min* and *max*), we conclude that the criterion becomes non-informative.

These criteria are systematically checked in our Monte Carlo experiment. We consider 250 independent draws of exogenous shocks of varying size **T** for each model. We choose **T** ∈ {1000, 10000, 20000, 30000, 40000, 50000} with the aim of checking progressively the gains obtained from increases in the simulation length.[2]

As a convergence criterion, we use the L²-distance between the **β**-vectors obtained in two subsequent iterations be less than 10^{-5}. With respect to the initialization of the algorithm, to the light of results shown in [14], we consider the approach suggested by [15].

V. Results and Discussion

Table I presents the number of the simulation (among the 250 independent draws) chosen according to each criteria. It can be observed that the different criteria rarely agree. Indeed, the only coincidence is referred to the replication maximizing L² distance to reference values obtained in the case of CH model. Therefore, in order to formulate a final proposal, we are forced to discriminate in this section between the different criteria suggested before.

Table II includes summary statistics which allow us to explore, firstly, the relevance of the central issue of this paper (to determine whether there are significant differences between different PEA solutions) and, secondly, how the selection criteria exposed above perform in the cases of the SN and CH models.

The main conclusion is that distributions are much more similar for simulation lengths greater than 1,000. A decreasing trend in σ_K is also observed when **T** grows, in line with the asymptotic properties of this algorithm, but this parameter is mostly constant within each distribution.

Furthermore, there are significant differences regarding the cross correlation between the responses of central variables of the CH model. By contrast, the SN model shows no big differences among the solutions. This latter result indicates the relevance of considering selection criteria when the complexity of model raises. Additionally, selection criteria based on impulse-responses achieve reasonable (and stable) values for all the simulation lengths considered -they are in the range achieved for the longest distribution of simulations- whereas criterion based on median coefficients satisfy this condition only from *T=10,000.*

TABLE I. Chosen Replications. r_C^{min}/r_C^{max} Refer to Replications Minimizing/Maximizing L² Distance to Reference Values ($C = \tilde{\beta}, \widetilde{IR}_t$ *and* $\widetilde{IR}_p$)

Choices minimizing L² distance to reference values

Replication length	SN model			CH model		
	$\tilde{\beta}$	$\widetilde{IR}_t$	$\widetilde{IR}_p$	$\tilde{\beta}$	$\widetilde{IR}_t$	$\widetilde{IR}_p$
1,000	129	236	13	27	240	219
10,000	68	80	51	54	247	40
20,000	221	226	161	92	111	95
30,000	158	151	164	38	93	136
40,000	125	71	32	27	111	154
50,000	99	200	201	71	149	48

Choices maximizing L² distance to reference values

Replication length	SN model			CH model		
	$\tilde{\beta}$	$\widetilde{IR}_t$	$\widetilde{IR}_p$	$\tilde{\beta}$	$\widetilde{IR}_t$	$\widetilde{IR}_p$
1,000	41	41	41	25	199	199
10,000	115	21	217	6	51	51
20,000	195	41	184	154	98	98
30,000	18	57	7	32	143	143
40,000	57	198	30	44	117	117
50,000	195	212	91	195	94	44

Notes: (1) Monte Carlo experiment: (a) 250 independent replications are computed for each length and model, (b) L2 distance between subsequent vectors obtained is required to be less than 10^{-5}, (c) Reference [15] approaches is used to initialise the algorithm. (2) Figures indicate the number of the simulation (among the 250 independent draws) chosen according to each criteria.

We further look in detail how the values of the coefficients change according to each replication length and selection criteria. Fig. 1 presents the distribution of coefficients for the SN model, and Fig. 2 those of the CH model. On each box, the central mark is the median, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually. Moreover, solutions minimizing and maximizing distance to reference values of different criteria selected above are remarked.

Some conclusions can be drawn: (i) we confirm again, from a different perspective, that the variance is reduced when the replication length is increased. The gain loses importance for lengths higher than 20000. (ii) Median values of coefficients are almost equal for lengths higher than 10000.[3] (iii) The criteria based on coefficients and those based on median responses are not coincident (as we commented before) in the sense that those based on median responses do not clearly discriminate between choices minimizing and maximizing distance within the distribution of the coefficients. (iv) The criterion based on median coefficients is more precise on those with a high variability. Otherwise, choices may be further away from the corresponding median value. As an instance, see coefficients $\boldsymbol{\beta}_8$ and $\boldsymbol{\beta}_4$ of the CH model.

Fig. 3, Fig. 4, Fig. 5 and Fig. 6 show the responses of *k* (capital in both models) to a transitory/permanent technology shock for each simulation length and model. Additionally, replications minimizing/maximizing L² distance to reference values of different criteria are remarked.[4]

[2] Indeed, a wider grid of lengths has been explored. For the sake of brevity, only results relative to those mentioned here are shown.

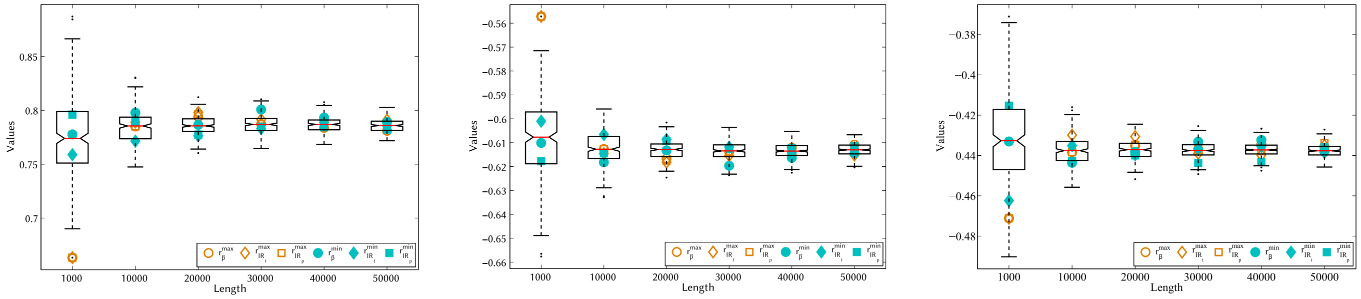
[3] Tests statistics confirm this finding. The results are available upon request.

[4] For the sake of simplicity, criteria based on different types of shocks are not compared within the same graph.

TABLE II. Summary Statistics. r_C^{min}/r_C^{max} Refer to Replications Minimizing/Maximizing L² Distance to Reference Values ($C = \tilde{\beta}, \widetilde{IR}_t$ and $\widetilde{IR}_p$)

		Standard Neoclassical model											
		Transitory shock						*Permanent shock*					
		T=1000	**T=10000**	**T=20000**	**T=30000**	**T=40000**	**T=50000**	**T=1000**	**T=10000**	**T=20000**	**T=30000**	**T=40000**	**T=50000**
σ_k	max	0,0276	0,0089	0,0053	0,0044	0,0040	0,0033	0,2635	0,0855	0,0550	0,0482	0,0407	0,0367
	min	0,0273	0,0088	0,0052	0,0043	0,0039	0,0032	0,0273	0,0088	0,0052	0,0043	0,0039	0,0032
$\rho(IR(c)_t, IR(k)_t)$	**max**	**0,9858**	**0,9776**	**0,9766**	**0,9768**	**0,9760**	**0,9757**	**0,9990**	**0,9986**	**0,9985**	**0,9985**	**0,9985**	**0,9985**
	r_β^{min}	0,9678	0,9735	0,9735	0,9734	0,9735	0,9744	0,9980	0,9983	0,9983	0,9983	0,9983	0,9984
	r_{IRt}^{min}	0,9781	0,9737	0,9737	0,9739	0,9738	0,9741	0,9986	0,9983	0,9983	0,9983	0,9983	0,9984
	r_{IRp}^{min}	0,9745	0,9736	0,9740	0,9739	0,9739	0,9738	0,9983	0,9983	0,9983	0,9983	0,9983	0,9983
	min	**0,9603**	**0,9690**	**0,9712**	**0,9710**	**0,9717**	**0,9718**	**0,9972**	**0,9980**	**0,9982**	**0,9982**	**0,9982**	**0,9982**
$\rho(IR(c)_t, IR(c)_{t+1})$	**max**	**0,9971**	**0,9968**	**0,9968**	**0,9967**	**0,9967**	**0,9967**	**0,9999**	**0,9999**	**0,9999**	**0,9998**	**0,9998**	**0,9998**
	r_β^{min}	0,9968	0,9967	0,9967	0,9967	0,9967	0,9967	0,9998	0,9998	0,9998	0,9998	0,9998	0,9998
	r_{IRt}^{min}	0,9966	0,9967	0,9967	0,9967	0,9967	0,9967	0,9998	0,9998	0,9998	0,9998	0,9998	0,9998
	r_{IRp}^{min}	0,9967	0,9967	0,9967	0,9967	0,9967	0,9967	0,9998	0,9998	0,9998	0,9998	0,9998	0,9998
	min	**0,9964**	**0,9965**	**0,9966**	**0,9966**	**0,9966**	**0,9966**	**0,9998**	**0,9998**	**0,9998**	**0,9998**	**0,9998**	**0,9998**
$\rho(IR(k)_t, IR(k)_{t+1})$	**max**	**0,9915**	**0,9897**	**0,9893**	**0,9892**	**0,9892**	**0,9891**	**0,9998**	**0,9998**	**0,9998**	**0,9998**	**0,9998**	**0,9998**
	r_β^{min}	0,9874	0,9886	0,9886	0,9886	0,9886	0,9888	0,9998	0,9998	0,9998	0,9998	0,9998	0,9998
	r_{IRt}^{min}	0,9897	0,9886	0,9886	0,9887	0,9886	0,9887	0,9998	0,9998	0,9998	0,9998	0,9998	0,9998
	r_{IRp}^{min}	0,9889	0,9886	0,9887	0,9887	0,9887	0,9886	0,9998	0,9998	0,9998	0,9998	0,9998	0,9998
	min	**0,9859**	**0,9877**	**0,9879**	**0,9881**	**0,9882**	**0,9882**	**0,9998**	**0,9998**	**0,9998**	**0,9998**	**0,9998**	**0,9998**

		Cooley and Hansen (1989) model											
		Transitory shock						*Permanent shock*					
		T=1000	**T=10000**	**T=20000**	**T=30000**	**T=40000**	**T=50000**	**T=1000**	**T=10000**	**T=20000**	**T=30000**	**T=40000**	**T=50000**
σ_k	max	0,0650	0,0190	0,0150	0,0125	0,0097	0,0088	0,0669	0,0194	0,0153	0,0130	0,0102	0,0091
	min	0,0639	0,0188	0,0149	0,0123	0,0096	0,0087	0,0639	0,0188	0,0149	0,0123	0,0096	0,0087
$\rho(IR(y)_t, IR(N)_t)$	**max**	**0,9584**	**0,8982**	**0,9129**	**0,8936**	**0,8750**	**0,8786**	**0,9991**	**0,9083**	**0,9193**	**0,9089**	**0,8701**	**0,8842**
	r_β^{min}	0,3592	0,7210	0,7641	0,7680	0,7201	0,7347	-0,8260	0,6743	0,6084	0,6617	0,5262	0,5802
	r_{IRt}^{min}	0,5512	0,7985	0,7790	0,7632	0,7940	0,7826	0,2206	0,7272	0,7073	0,6689	0,7143	0,7260
	r_{IRp}^{min}	0,6709	0,7685	0,8604	0,7456	0,7420	0,8112	0,7593	0,4874	0,7982	0,6063	0,6458	0,7919
	min	**-0,9993**	**0,4826**	**0,5613**	**0,3613**	**0,6190**	**0,6132**	**-1,0000**	**-0,1905**	**-0,1305**	**-0,6397**	**0,1462**	**0,0887**
$\rho(IR(y)_t, IR(\pi)_t)$	**max**	**0,9101**	**-0,9413**	**-0,9426**	**-0,9251**	**-0,9468**	**-0,9455**	**0,9570**	**-0,2571**	**-0,3187**	**0,2101**	**-0,5609**	**-0,5123**
	r_β^{min}	-0,8879	-0,9527	-0,9527	-0,9530	-0,9515	-0,9523	0,4732	-0,9191	-0,8756	-0,9065	-0,8303	-0,8628
	r_{IRt}^{min}	-0,9455	-0,9538	-0,9536	-0,9530	-0,9538	-0,9537	-0,6302	-0,9375	-0,9299	-0,9112	-0,9315	-0,9389
	r_{IRp}^{min}	-0,9523	-0,9515	-0,9549	-0,9522	-0,9529	-0,9542	-0,9622	-0,7959	-0,9624	-0,8764	-0,9012	-0,9664
	min	**-0,9580**	**-0,9556**	**-0,9554**	**-0,9552**	**-0,9549**	**-0,9549**	**-0,9987**	**-0,9979**	**-0,9973**	**-0,9963**	**-0,9887**	**-0,9918**
$\rho(IR(y)_t, IR(y)_{t+1})$	**max**	**0,9809**	**0,4634**	**0,4630**	**0,4614**	**0,4569**	**0,4581**	**0,9997**	**0,9902**	**0,9898**	**0,9893**	**0,9872**	**0,9878**
	r_β^{min}	0,4980	0,4480	0,4486	0,4489	0,4473	0,4477	0,9928	0,9790	0,9768	0,9784	0,9745	0,9761
	r_{IRt}^{min}	0,4451	0,4505	0,4500	0,4488	0,4503	0,4502	0,9683	0,9808	0,9803	0,9786	0,9804	0,9809
	r_{IRp}^{min}	0,4521	0,4484	0,4534	0,4489	0,4480	0,4522	0,9831	0,9737	0,9833	0,9767	0,9781	0,9837
	min	**0,4349**	**0,4442**	**0,4438**	**0,4452**	**0,4451**	**0,4462**	**0,9683**	**0,9681**	**0,9690**	**0,9696**	**0,9695**	**0,9694**

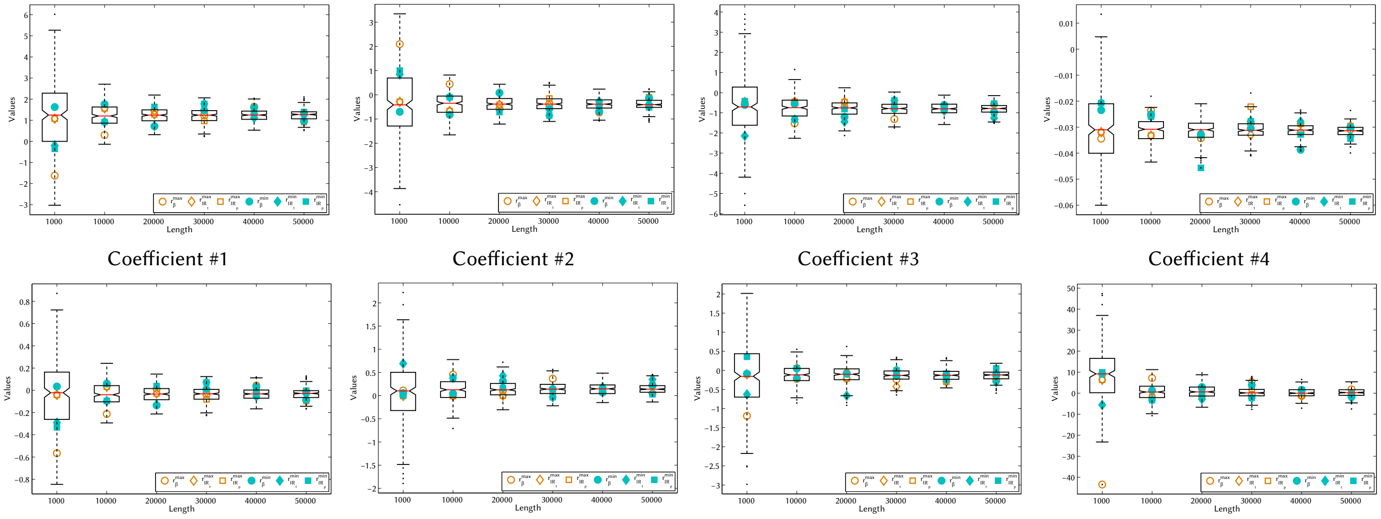


Coefficient #1

Coefficient #2

Coefficient #3

Fig. 1. Boxplots of functional form coefficients of parameterized expectations. SN model. r_C^{min}/r_C^{max} refer to replications minimizing/maximizing L² distance to reference values ($C = \tilde{\beta}, \widetilde{IR}_t$ and $\widetilde{IR}_p$). (1) Monte Carlo experiment: (a) 250 independent replications are computed for each length and model, (b) L2 distance between subsequent β-vectors obtained is required to be less than 10-5, (c) Reference [15] approach is used to initialise the algorithm.



Coefficient #1

Coefficient #2

Coefficient #3

Coefficient #4

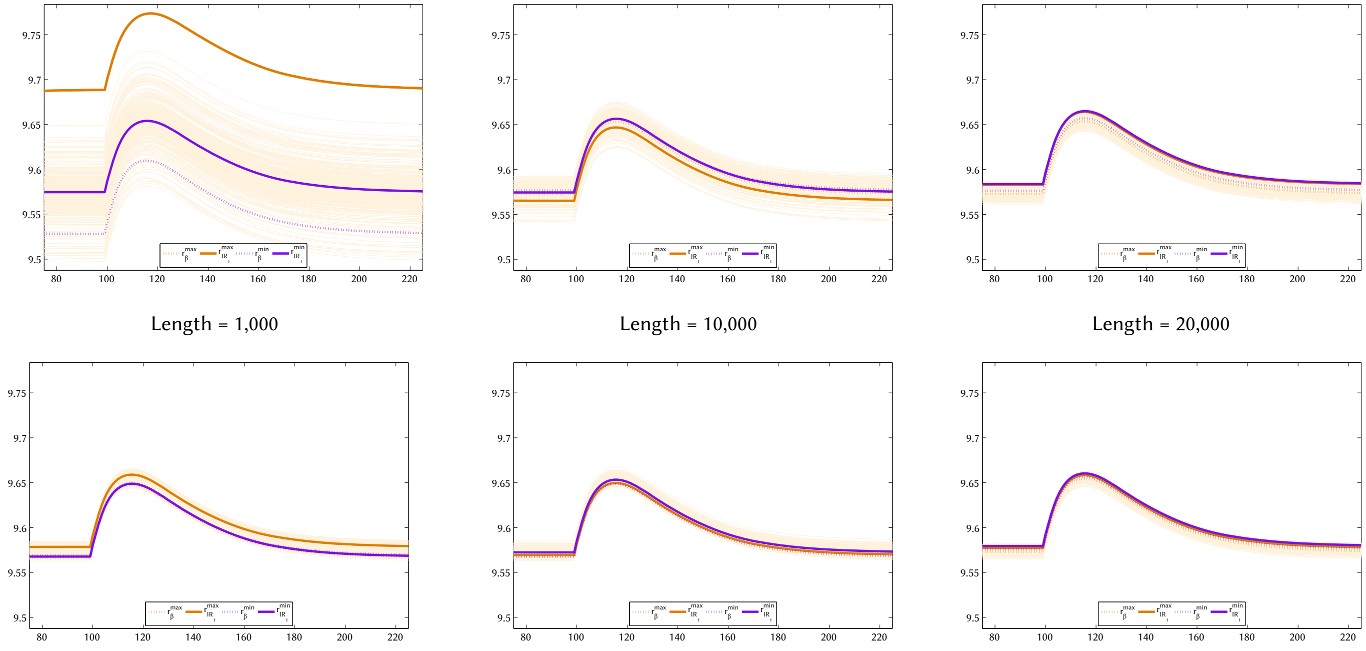
Coefficient #5

Coefficient #6

Coefficient #7

Coefficient #8

Fig. 2. Boxplots of functional form coefficients of parameterized expectations. CH model. r_C^{min}/r_C^{max} refer to replications minimizing/maximizing L² distance to reference values ($C = \tilde{\beta}, \widetilde{IR}_t$ and $\widetilde{IR}_p$). (1) Fig. 1. notes applies here.



Length = 1,000

Length = 10,000

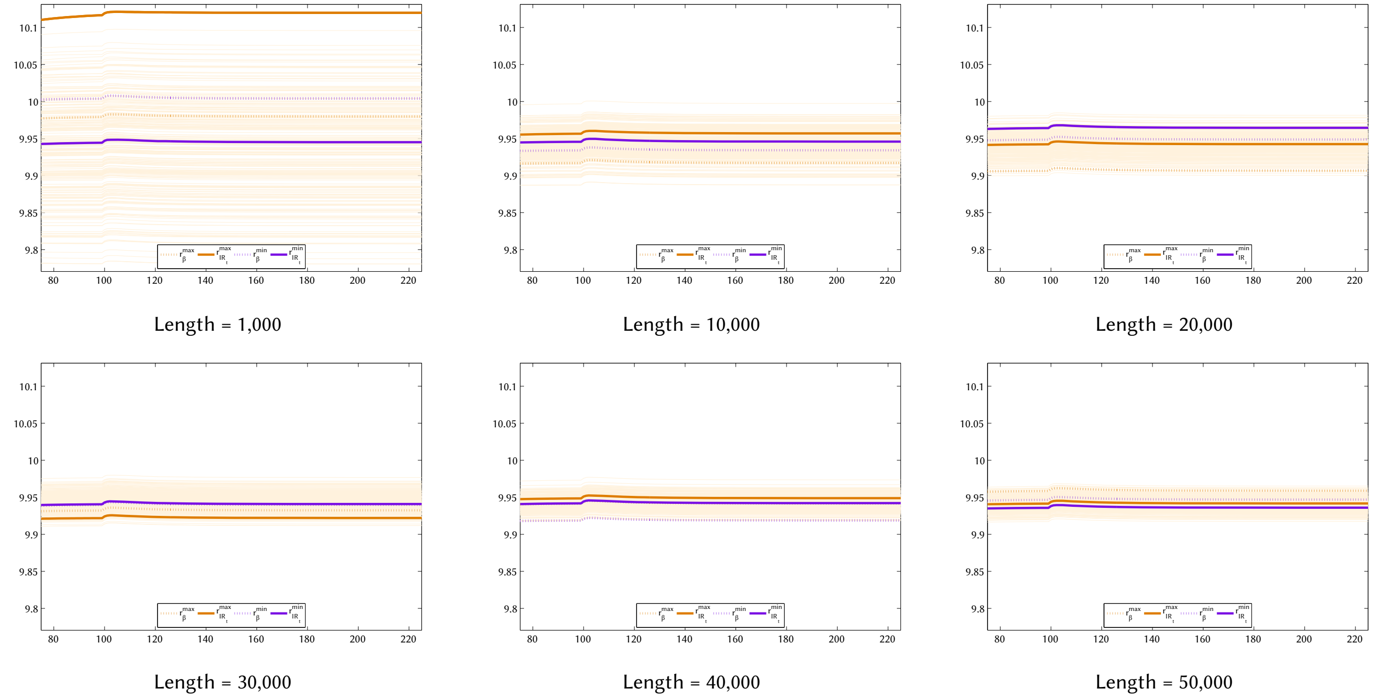
Length = 20,000

Length = 30,000

Length = 40,000

Length = 50,000

Fig. 3. Responses of k to a transitory technology shock. SN model. r_C^{min}/r_C^{max} refer to replications minimizing/maximizing L² distance to reference values ($C = \tilde{\beta}, \widetilde{IR}_t$ and $\widetilde{IR}_p$). (1) Fig. 1. notes applies here.



Length = 1,000

Length = 10,000

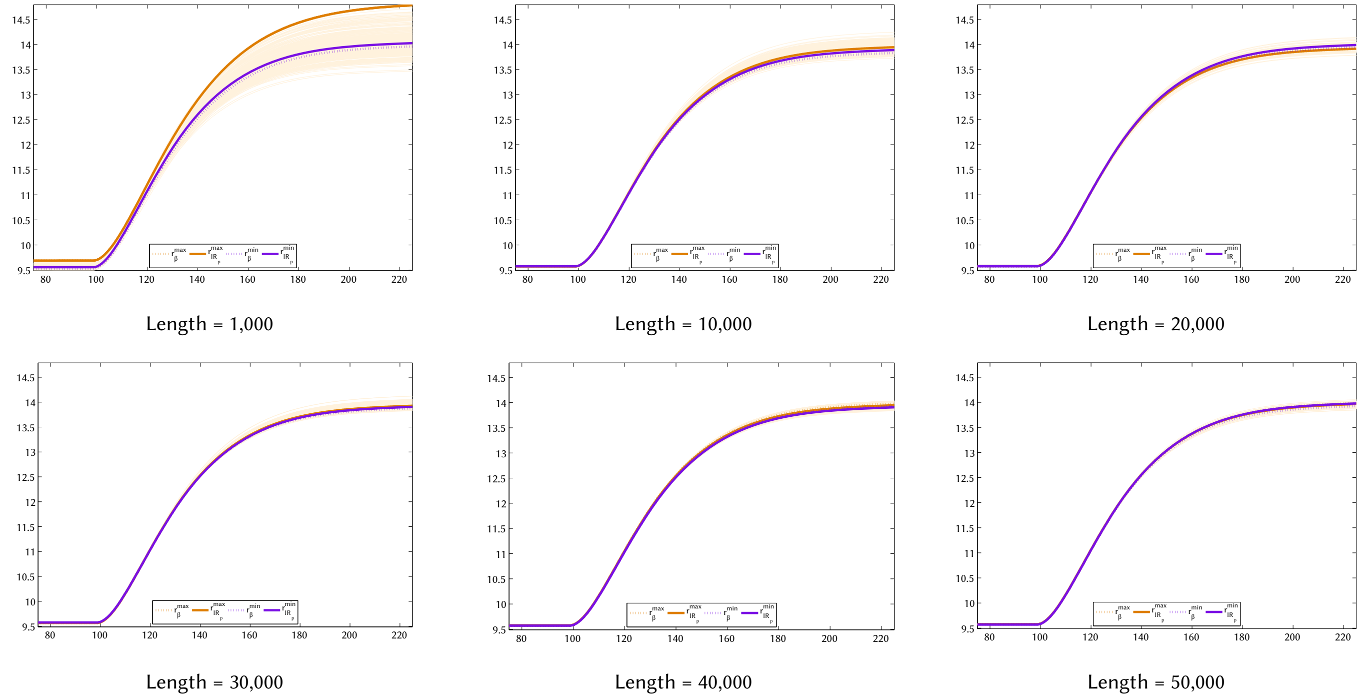
Length = 20,000

Length = 30,000

Length = 40,000

Length = 50,000

Fig. 4. Responses of k to a transitory technology shock. CH model. r_C^{min}/r_C^{max} refer to replications minimizing/maximizing L^2 distance to reference values ($C = \tilde{\beta}, \widetilde{IR}_t$ *and* $\widetilde{IR}_p$). (1) Fig. 1. notes applies here.



Length = 1,000

Length = 10,000

Length = 20,000

Length = 30,000

Length = 40,000

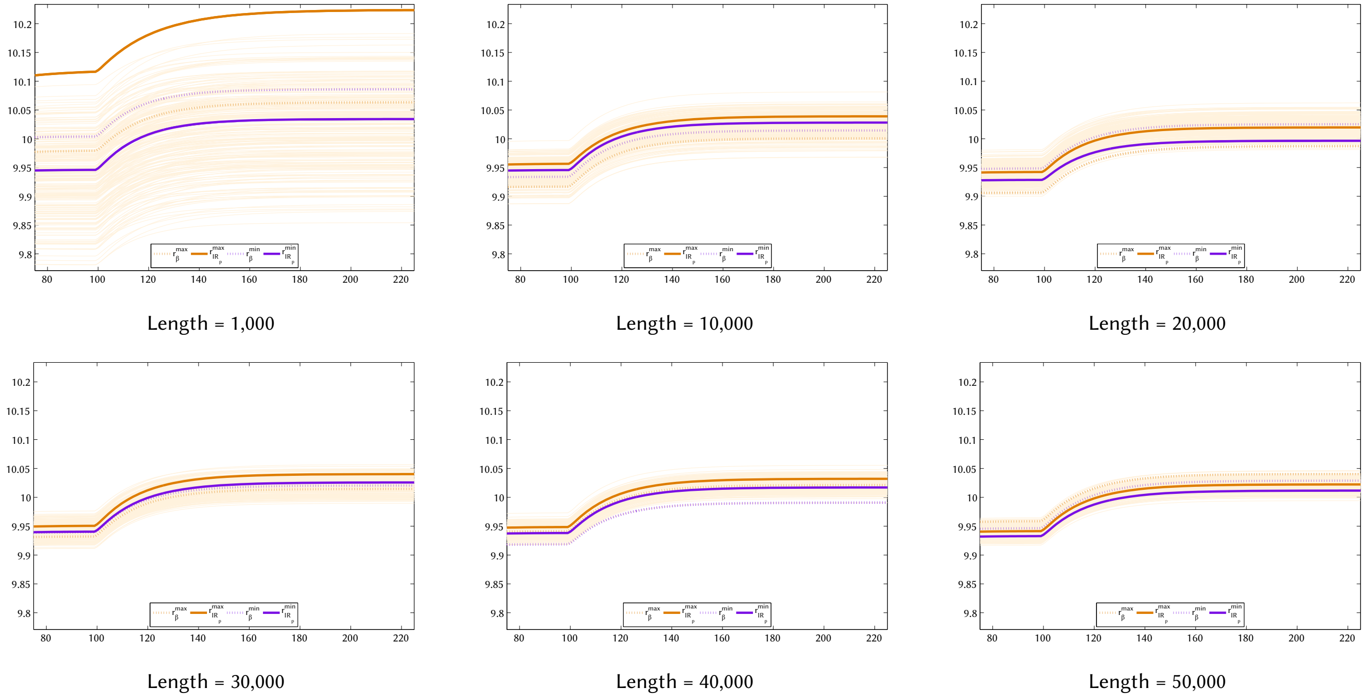
Length = 50,000

Fig. 5. Responses of k to a permanent technology shock. SN model. r_C^{min}/r_C^{max} refer to replications minimizing/maximizing L^2 distance to reference values ($C = \tilde{\beta}, \widetilde{IR}_t$ *and* $\widetilde{IR}_p$). (1) Fig. 1. notes applies here.

Firstly, we observe how all responses are qualitatively equal but we have tested that they are statistically different.[5] The most interesting result, robust across models and types of shock, is the coincidence (tested statistically) of median responses for different lengths. Again, in line with the asymptotic properties of the algorithm, the level of uncertainty is reduced if longer replications are considered. Moreover, the criterion based on median coefficients results non-informative in terms of their responses. This finding confirms, again, the non-coincident choices we would achieve for criteria based on coefficients and those based on impulse-response functions.

These criteria do not only help to choose between different solutions but also to measure the robustness of the estimated coefficients. Firstly, the boxplot figures provide summary statistics relative to estimated values. Secondly, the impulse-response functions can be used to construct bands of confidence around the median response, which surely might add robustness to one specific analysis.

[5] For the sake of brevity, the results are not included here. They are available upon request



Length = 1,000

Length = 10,000

Length = 20,000

Length = 30,000

Length = 40,000

Length = 50,000

Fig. 6. Responses of k to a permanent technology shock. CH model. r_C^{min}/r_C^{max} refer to replications minimizing/maximizing L² distance to reference values ($C = \tilde{\beta}, \widetilde{IR}_t$ and $\widetilde{IR}_p$). (1) Fig. 1. notes applies here.

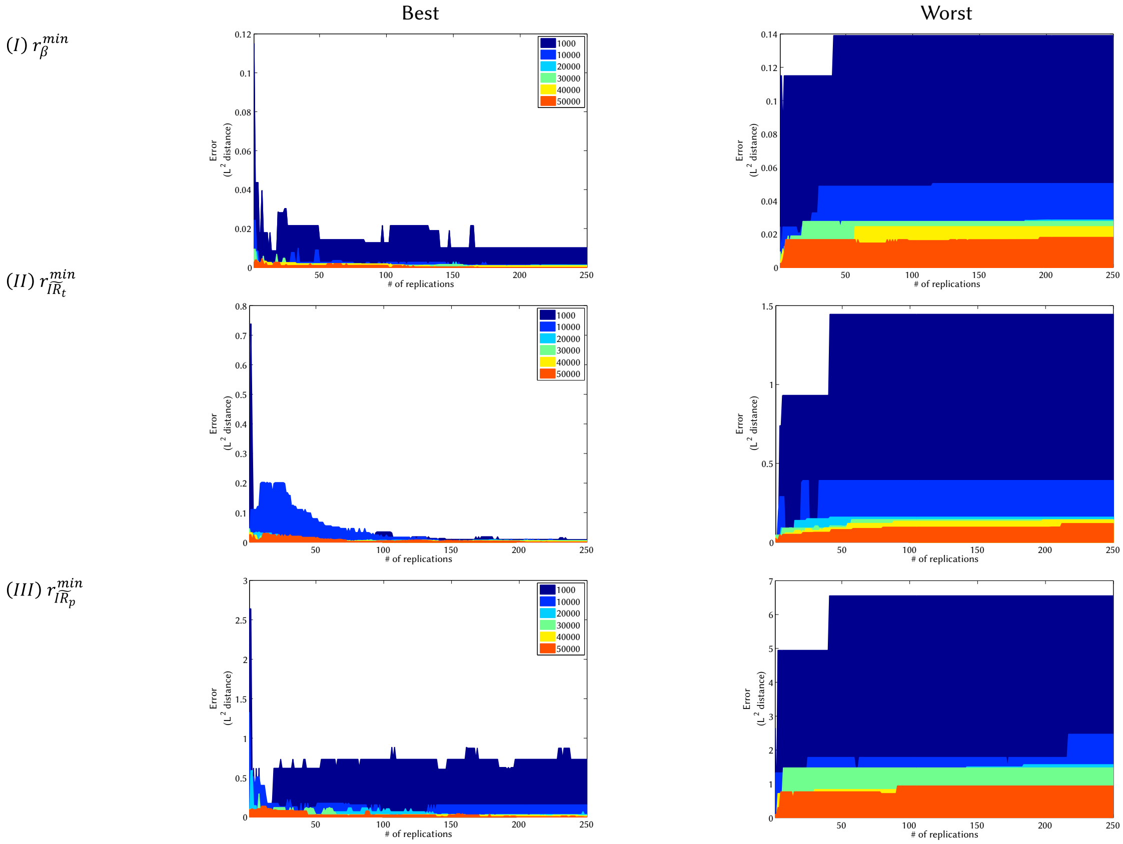


Fig. 7. Relative error with respect to the true solution (Minimal distance to reference values for T=50,000). SN model. (II) $r_{\tilde{C}}^{min}$ refers to replications minimizing L² distance to reference values. (1) Fig. 1. notes applies here.

Another relevant issue to be explored is the sensitivity of these criteria to the number of replications used. In this regard, we compute L² distance (*error*) to the replication we would choose according to each criterion but considering a varying number of replications, starting from the assumption that longest length replications are closer to the *true* solution. The resulting errors are shown in Fig. 7 and Fig. 8. Left (right) panels show how *error* evolves with the number of replications and the sample size. It can be seen that 75-100 replications

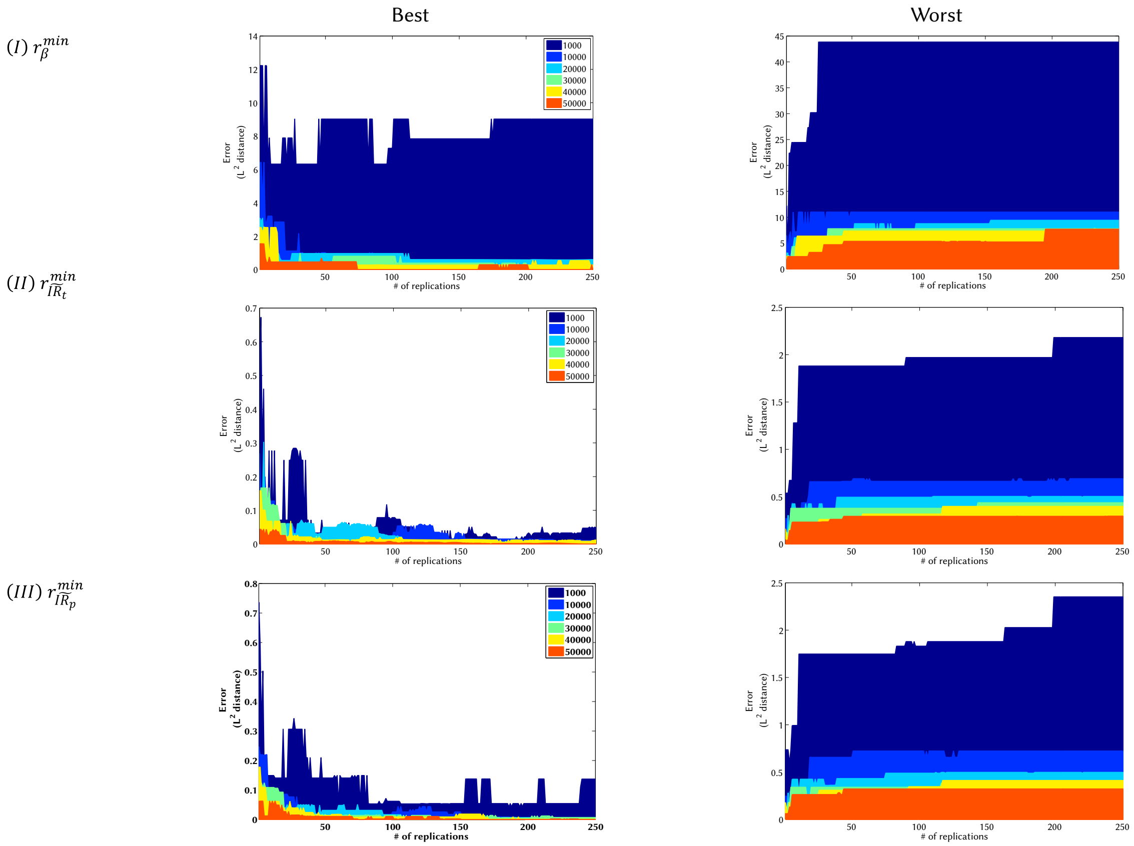


Fig. 8. Relative error with respect to the true solution (Minimal distance to reference values for T=50,000). CH model. (II) $r_{\tilde{C}}^{min}$ refers to replications minimizing L² distance to reference values. (1) Fig. 1. notes applies here.

are enough to minimize the error and generate the existing variance. With respect to the replication length, no significant improvements in the minimization choice are achieved from considering lengths greater than 10,000.[6] They also prove that gains in dispersion (a lower distance from the "true" solution to replication maximizing distance to reference values) are obtained mainly from 1,000 to 20,000. After then, error mainly remains at the same values. The latter result mainly confirms our findings regarding the variability of coefficients distributions.

VI. Conclusions

We find significant differences among different solutions which clearly establish the relevance of this issue (increasing with the level of complexity of the model). Additionally, robust results are obtained due to the summary statistics we compute from the distributions achieved. By contrast, if only one sufficiently long simulation is considered (what people have commonly done so far), there is no guarantee that a representative solution is obtained (there is heterogeneity among solutions even for *T=50,000*).

Moreover, median values are almost equal for different simulation lengths and, therefore, there is no bias in using them. However, the variance among solutions is clearly higher when shorter replications are considered. On the contrary, different criteria rarely accord with their choices. Hence, any decision related to the criterion to be adopted may have consequences on one specific application. However, criteria based exclusively on estimated coefficients ignore the economic performance of these solutions, which represent a significant drawback. Note that there is no guarantee of selecting a non-extreme solution if one is finally interested in, for example, observing how will respond our model to a shock (impulse-response functions).

All in all, the median response criterion appears to be the most suitable criterion for several reasons: (i) there is no bias in using shorter simulations (it allows one to face much lower computational costs), (ii) it shows how different solutions behave when a transitory/permanent shock is applied, and (iii) it provides a band of confidence around the final choice (the level of uncertainty is measured).

Finally, we discuss the number of replications needed to capture the main properties of the distributions under observation. In our exercise, 75-100 replications are enough to minimize the error and generate the existing variance due to the use of shorter replications.

Acknowledgment

I thank Javier Pérez for his contributions to previous versions of this document and Jesús Rodríguez-López and Alfonso R. Sánchez for helpful comments and discussions. Author acknowledges the financial support of the Spanish Ministry of Science (PID2019-105517RB-I00 and 2006-04803), Ministry of Economy and Competitiveness (ECO2012-37572) and the Regional Government of Andalusia (SEJ 1512).

References

[1] L. Christiano, and J.M. Fisher, "Algorithms for Solving Dynamic Models with Occasionally Binding Constraints," *Journal of Economic Dynamics and Control*, vol. 24, pp. 1179-1232, 2000, doi: 10.1016/S0165-1889(99)00016-0.

[2] W. den Haan and A. Marcet, "Solving the Stochastic Growth Model by Parameterized Expectations," *Journal of Business and Economic Statistics*,

[6] We have checked that there are no relevant differences among 5000 and 10000. This information is available upon request.

vol. 8, pp. 31-34, 1990, doi: 10.1080/07350015.1990.10509770.

[3] W. den Haan and A. Marcet, "Accuracy in Simulations," *Review of Economic Studies*, vol. 61, pp. 3-17, 1994, doi: 10.2307/2297873.

[4] B.C. Eaves and K. Schmedders, "General equilibrium models and homotopy methods," *Journal of Economic Dynamics and Control*, vol. 23, pp. 1249-1279, 1999, doi: 10.1016/S0165-1889(98)00073-6.

[5] P. Fackler, "A Matlab solver for Nonlinear Rational Expectations Models," *Computational Economics*, vol. 26, pp. 173-181, 2005, doi: 10.1007/s10614-005-1784-z.

[6] C.B. Garcia and W.I. Zangwill, "*Pathways to solutions, fixed points, and equilibria,"* Prentice- Hall, 1981.

[7] S. Maliar and L. Maliar, "Parameterized expectations algorithm and the moving bounds," *Journal of Business and Economic Statistics*, vol. 1, pp. 88-92, 2003, doi: 10.1198/073500102288618793.

[8] S. Maliar and L. Maliar, "Parameterized Expectations Algorithm: How to Solve for Labor Easily," *Computational Economics*, vol. 25, no. 3, pp. 269-274, 2005, doi: 10.1007/s10614-005-2224-9.

[9] M. Creel, "Using Parallelization to Solve a Macroeconomic Model: A Parallel Parameterized Expectations Algorithm," *Computational Economics*, vol. 32, no. 4, 2008, doi: 10.1007/s10614-008-9142-6.

[10] J.B. Taylor and H. Uhlig, "Solving Nonlinear Stochastic Growth Models: A Comparison of Alternative Solution Methods," *Journal of Business and Economic Statistics*, vol. 8, pp. 1–17, 1990, doi: 10.1080/07350015.1990.10509766.

[11] A. Marcet and D.A. Marshall, "Solving Nonlinear Rational Expectations Models by Parameterized Expectations: Convergence to Stationary Solutions," *Institute for Empirical Macroeconomics Discussion Paper* 91, 1994, May.

[12] T.F. Cooley, and G. Hansen, "The Inflation Tax in a Real Business Cycle Model," *American Economic Review*, vol. 79, pp. 733-748, 1989, doi: 10.4324/9780203070710.ch11.

[13] R.Y. Liu, "On a Notion of Data Depth Based on Random Simplices," *The Annals of Statistics*, vol. 18, pp. 405-414, 1990, doi: 10.1214/aos/1176347507.

[14] J.J. Pérez and A.J. Sánchez-Fuentes, "Alternatives to initialize the Parameterized Expectations Algorithm," *Economics Letters*, vol. 102, no. 2, pp. 116-118, 2009, doi: 10.1016/j.econlet.2008.11.017.

[15] J.J. Pérez, "A Log-linear Homotopy Approach to Initialize the Parameterized Expectations Algorithm," *Computational Economics*, vol. 24, pp. 59-75, 2004, doi: 10.1023/B:CSEM.0000038893.70411.f5.

[16] Y. Zuo and R. Serfling, "General Notions of Statistical Depth Function," *The Annals of Statistics*, vol. 28, pp. 461-482, 2000, doi: 10.1214/aos/1016218226.

A. Jesus Sanchez-Fuentes

He holds a Bachelor in Mathematics from University of Seville and a Ph.D in Economics from Pablo de Olavide University (with distinctions). He currently works as Associate professor at Complutense University of Madrid where has been appointed as Academic secretary of the Complutense Institute of International Studies (ICEI-UCM). He also heads the UCM research group "Family policies", edits the electronic journal on educative innovation "e-pública" and collaborates with the research group Governance and Economics research Network as Research Affiliate. His research areas are mainly public economics and computational economics.

A Comparative Analysis of Machine Learning Models for Banking News Extraction by Multiclass Classification With Imbalanced Datasets of Financial News: Challenges and Solutions

Varun Dogra[1], Sahil Verma[2], Kavita[2], NZ Jhanjhi[3], Uttam Ghosh[4], Dac-Nhuong Le[5,6]*

[1] School of Computer Science and Engineering, Lovely Professional University (India)
[2] Department of Computer Science and Engineering, Chandigarh University, Mohali (India)
[3] School of Computer Science and Engineering, Taylor's University (Malaysia)
[4] Department of Computer Science and Data Science, Meharry School of Applied Computational Sciences, Nashville, TN (USA)
[5] School of Computer Science, Duy Tan University, Danang, 550000 (Vietnam)
[6] Institute of Research and Development, Duy Tan University, Danang, 550000 (Vietnam)

Received 13 November 2020 | Accepted 19 January 2022 | Published 8 February 2022

LA UNIVERSIDAD
EN INTERNET

Abstract

Online portals provide an enormous amount of news articles every day. Over the years, numerous studies have concluded that news events have a significant impact on forecasting and interpreting the movement of stock prices. The creation of a framework for storing news-articles and collecting information for specific domains is an important and untested problem for the Indian stock market. When online news portals produce financial news articles about many subjects simultaneously, finding news articles that are important to the specific domain is nontrivial. A critical component of the aforementioned system should, therefore, include one module for extracting and storing news articles, and another module for classifying these text documents into a specific domain(s). In the current study, we have performed extensive experiments to classify the financial news articles into the predefined four classes Banking, Non-Banking, Governmental, and Global. The idea of multi-class classification was to extract the Banking news and its most correlated news articles from the pool of financial news articles scraped from various web news portals. The news articles divided into the mentioned classes were imbalanced. Imbalance data is a big difficulty with most classifier learning algorithms. However, as recent works suggest, class imbalances are not in themselves a problem, and degradation in performance is often correlated with certain variables relevant to data distribution, such as the existence in noisy and ambiguous instances in the adjacent class boundaries. A variety of solutions to addressing data imbalances have been proposed recently, over-sampling, down-sampling, and ensemble approach. We have presented the various challenges that occur with data imbalances in multiclass classification and solutions in dealing with these challenges. The paper has also shown a comparison of the performances of various machine learning models with imbalanced data and data balances using sampling and ensemble techniques. From the result, it's clear that the performance of Random Forest classifier with data balances using the over-sampling technique SMOTE is best in terms of precision, recall, F-1, and accuracy. From the ensemble classifiers, the Balanced Bagging classifier has shown similar results as of the Random Forest classifier with SMOTE. Random forest classifier's accuracy, however, was 100% and it was 99% with the Balanced Bagging classifier.

Keywords

Class Imbalance, Down-Sampling, Ensemble Approaches, Machine Learning, N-grams, Over-Sampling Techniques, TFIDF.

DOI: 10.9781/ijimai.2022.02.002

I. Introduction

In the equity market, stocks or funds belong to the different business sectors. And sector-based news has become an inseparable part of the management of financial assets, with news-driven stock and bond markets explosively growing. Fund managers take advantage of this reality and make use of sector-oriented news to select individual stocks to diversify their investment portfolios to optimize returns. There is no such structured framework available for classifying the news on specific sectors of someone's interest. This problem is increasing by the day, necessitating a system for news classification methodology for specific sectors.

Machine learning (ML) techniques have demonstrated impressive performance in the resolution of real-life classification problems in

* Corresponding author.
E-mail address: ledacnhuong@duytan.edu.vn

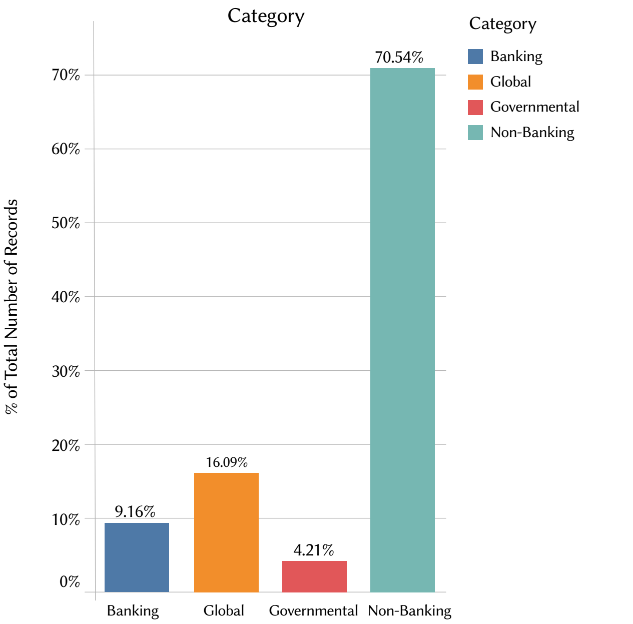
many different areas such as financial markets [1], medical diagnosis [2], vehicle traffic examination [3], fraud detection [4]. There are plenty of document classification systems in the commercial world. For instance, usually, the news stories are grouped by topics [5], medical images are tagged by disease categories [6], and many products are branded according to categories [7]. Different methods of statistical and machine learning are implemented in text labeling, where one of the predefined labels is automatically assigned to a given item of the unlabeled pool of textual articles.

However, the vast majority of articles on the internet about text classification are binary text classification [8] such as email filtering [9], political preferences [10], sentiment analysis [11], etc. Our real-world problem is in most cases much more complex than the binary classification. More formally, if some d is a document in the whole set of documents D and C is the set of all categories i.e. $C=\{c_1, c_2,c_3,\ldots, c_n,\}$ the classification of text assigns one category c_i to the document d. Such a classification function with more than two classes is known as multiclass classification; for example, identify a set of news categories as business, political, economic or entertainment.

In our paper, we're interested in isolating news on the banking sector and its most associated domains from the pool of financial news articles. We feel that 'banking news' of any nation is most correlated with their 'governmental news-events ' that covers news on government initiatives for good governance, state or national elections, change or new development of governmental policies, and 'global' financial news that covers global trade, changes in currency-commodities prices, and global sentiments. So, we have a 4-class classification problem of a set of news articles to extract banking, and its most correlated news i.e. Government, and Global from entire financial news articles. We decide to label the news articles into banking, governmental, global, and non-banking classes with a total of 10000 instances. The non-banking news covers all the financial news scrapped from various new portals divergent from these 3 categories (banking, governmental and global). The news reports on different categories are usually imbalanced. The distribution of the news articles in our dataset is shown in Fig. 1. The news articles are manually labeled into these four classes. [12] mentions that labeling is normally done manually by human experts (or users), which is a time-consuming and labor-intensive process but it results in higher accuracy due to expert knowledge being involved in labeling text articles with appropriate. In the process, we label a set of representative news articles for each class. The labelers are experts in the financial domain and financial markets. A team of three experts is used to perform feature selection to identify important or representative words for each class used in a 4-class classification, followed by inspecting each text document and label it to the respective class based on representative words for each class. An agreement is made with the experts to label the given instances of the news articles. The process is used to derive a set of documents from entire unlabeled documents for each class to form the initial training package. The different machine learning techniques are then applied to build and compare the classifiers. The whole process is explained in the later part of the paper in sections 3-4.

A. Multiclass Classification

For machine learning, the problem of classifying instances into three or more classes in multiclass classification. Although some classification algorithms of course allow the use of more than two classes, some are by definition binary algorithms; however, a variety of strategies may transform these into multi-classification. In a multiclass classification problem, some classes may be represented with only a few samples (called the minority class), and the rest falls into the other class (called the majority class). The data disparity in machine learning creates difficulties in conducting data analytics in virtually all fields of real-world problems. The problem of classifying textual news articles is a two-step process. In our experiment, in the first step, the documents are collected from various websites like Bloomberg, Financial Express, and Moneycontrol using web scrapping code written in Python. It is followed by partitioned news articles into their respective category of banking, non-banking, global, and governmental using manual labeling. In the next step, the news articles are trained and tested using machine learning approaches to achieve the classification goal for a new sample of news articles. A comparative analysis is performed based on the results of the experiment to rate the tested machine learning algorithms in descending order so they can be used to evaluate news classification tasks with imbalanced datasets. We are not detailing the process of downloading news from the various sources in the paper.



% of Total Number of Records for each Category. Color shows details about Category.

Fig. 1. The distribution of news article instances amongst 4-classes.

In turn, multiclass classification can be divided into three groups:

- Native classifiers: These include most common classifiers such as Support Vector Machines (SVM), Classification and Regression Trees (CART), KNN, Naïve Bayes (NB), and multi-layer output nodes i.e. Neural Nets.
- Multi-class wrappers: These hybrid classifiers reduce the problem to smaller chunks that can then be solved with different binary classifiers.
- Hierarchical Classifiers: Using a tree-based architecture this group uses hierarchical methods to partition output space into target class nodes.

B. Learning From Imbalanced Dataset

A dataset is considered class-imbalanced if the number of examples that represent each class is not equal. Dealing with an imbalanced dataset has been a popular subject in the research study of classifying news articles. The conventional machine learning algorithms may introduce biases while dealing with imbalanced datasets [1]. The accuracy of many classification algorithms is considered to suffer from imbalances in the data (i.e. when the distribution of the examples is significantly distorted across classes) [13]. Most binary text classification applications are of this kind, with the negative examples far outnumbered positive examples of the class of interest [2]. Many classifiers assume that examples are evenly distributed among classes and assume an equal cost of misclassification. For example, someone works in an organization and is asked to create a model that predicts whether news belongs to class A, based on the distribution of news in classes A and B at your side. He chooses to use his favorite classifier, train it on data, and before he knows it, he gets an accuracy of 95%. Without further testing, he wants to use the model. A couple of days later he underlines the model's uselessness. Indeed, from the time it was used to gather news, the model he created did not find any news belonging to class A. He figures out after some investigations that there is only about 5 percent of the news produced in the pool that belongs to Class A and that the model always responds to Class "B," resulting in 95 percent accuracy. The kind of "guileless" findings that he obtained were due to the imbalanced dataset with which he works. The goal of this paper is to examine the various methods that can be used with imbalanced groups to tackle classification problems.

In the imbalanced data set, basically with this problem, a classifier's output leans to be partial towards certain classes (majority class) [14]. In Natural Language Processing (NLP) and Machine Learning in general, the problems of imbalanced classification, under which the number of items in each class for a classification process differs extensively and the capacity to generalize on dissimilar data remained critical issues [15]. Most classification data set do not have precisely the same number of instances in each class but a slight variation is often insignificant. There are problems where class inequality is believed to not just normal.

Also, classifiers are typically built to optimize precision, which in the situation of imbalanced training data is not a reasonable metric for determining effectiveness. Therefore, we are presenting the comparison of various machine learning classification techniques which might result in high accuracy even with imbalanced datasets, however, it is worth mentioning certain challenges we find to deal with imbalanced data and evaluating certain measures along with accuracy to evaluate performance. Also, we conduct machine learning on documents to perform multi-classification, where the data sample belongs to one of the multiple categories exactly.

The readers will also come to know the following key points after they have studied this paper:

- Imbalanced classification is the classification issue when the training dataset has an uneven distribution of the classes. As a result, appropriate sampling techniques must be implemented to balance the distribution by taking into consideration the various characteristics and the balanced performance of all of them.
- The class distribution imbalance may vary, but a serious imbalance is more difficult for modeling and may require advanced techniques. It is possible to introduce an efficient hybrid ensemble classifier architecture that incorporates density-based under-sampling or over-sampling and cost-effective methods by examining state-of-the-art solutions using a multi-objective optimization algorithm.
- Most real-world classification problems, such as scam detection, news headlines categorization, and churn prediction, have an imbalanced class distribution. Certain issues should be addressed when constructing multi-class classifiers in the case of class imbalances.

The paper's structure is as follows. In Section 2 we present a review of several current literature methods that handle the classification of imbalanced datasets for text classification. In section 3 we present our framework of classifying news articles along with challenges and possible solutions for the classification of imbalanced datasets. Sections 4 presents the comparative study of different techniques along with the experimental outcomes. Section 5 summarizes the paper and presents the future direction in the area of classification of imbalanced datasets.

II. Literature Review

We will present the necessary review in text classification and imbalanced learning in the subsequent subsections. We also assess the state-of-art research involving both the learning of imbalances and multiclass text classification.

A. Machine Learning for Text Classification

Here, we present the relevant literature work in the area of text classification using approaches to machine learning. Most of the preceding research had effective results using supervised methods of learning [7], [9], [16]. The following sub-sections present the literature work on feature extraction, selection, representation, and classification using learning models.

1. Document Representation

The efficiency of machine learning approaches largely depends on the option of representation of the data on which they would be implemented. For this purpose, most of the practical work in implementing machine learning algorithms runs further into the creation of pre-processing pathways and data conversion that leads to the representation of data that can help efficient machine learning. These representations or attribute development is essential, yet labor-intensive, and illustrates the vulnerability of current learning algorithms: their weakness to isolate and arrange the data discriminatively. However, the goal is clear when it comes to classification; we want to reduce the number of misclassifications upon testing data and overcoming the mentioned challenges in our framework.

Several machine learning implementations within the text field use bag-of-words representation where terms are defined as dimensions with word frequencies corresponding values. Normalized representation of the word frequencies is used by many applications as the dimensional values. One of the significant techniques of describing a document is Bag of Word (BoW). Use the frequency count of every term throughout the text, the BoW is used to form a vector describing document. This method of representation of documents is called a Vector Space Model [17]. However, the relative frequencies of terms often vary widely, which contributes to the differential meaning of the different words in classification applications [18]. With the varying lengths of various text documents, one needs to normalize when measuring distances between them. To solve these issues, term weighting methods are used to assign correct weights to the word for improving text classification efficiency [19]. Term weighting has long been developed in machine learning in the form of term frequency times inverse document frequency i.e. tfidf [20]. [21] suggests techniques to improve the TF-IDF scores to improve the representation of the term spreading between classes. Such practices may be used in various services where bag-of-word-based TF-IDF features are used. Equation (1) is given as:

$$tfidf(t_i, d_j) = tf(t_i, d_j) \times log(N|N(t_i)) \quad (1)$$

Here, N represents the overall number of documents and $N(t_i)$ denotes the number of documents in which the term *ti* occurs in the collection of documents. $tf(t_i, d_j)$, it represents the number of times term *ti* occurs in document *dj*. The newer version is mentioned in (2):

$$w_{i,j} = \frac{tfidf(t_i,d_j)}{\sqrt{\sum_{k=1}^{|T|} tfidf(t_k,d_j)^2}} \quad (2)$$

|T| represents the unique terms available in the collection of documents,

$$tf(t_i, d_j) = 1 + log\left(n(t_i, d_j)\right), if n(t_i, d_j) > 0, otherwise 0 \quad (3)$$

The outline in (2) is concerned with the words that belong to document d_j.

The importance of the standard term weighting outlines in (1), (2) is that three basic principles of word frequency distribution have been integrated into a pool of documents.

1. No less important are uncommon terms than a regular terms-*idf* hypothesis.
2. Numerous presences of a word in a text are no less relevant compared to the presumption of a single appearance-*tf*.
3. Long documents are no less necessary for the equivalent amount of term matching than short documents – the assumption of normalization.

The big drawback of this model is that it results in a large sparse matrix, which poses a high-dimensionality problem. The design of such high-dimensional feature spaces is usually inadequate in the number of items to represent adequately. The reduction of dimensionality is therefore a significant problem for a variety of applications. The literature has suggested several methods for the reduction of dimensionality [3], [22], [23]. For such representations, for instance, linear support-vector machines are comparatively effective [24]; whereas other techniques like Decision trees have to be built and modified with attention to allow their proper usage [25]. When a decision tree induction method computes a decision tree that depends very much on arbitrary features of the training examples, and works well only on trained data, but badly on unknown data, the data becomes overfit. There is a way to reduce the chance of overfitting by choosing the perfect subspace for the function at each node [26]. Cross-validation is an important prevention method to tackle overfitting. We segment the data into sub-sets k, called folds, for regular k-fold cross-validation. We then train the algorithm iteratively on folds of k-1, thus using the remainder of the fold as the test set [27].

In several studies, word n-grams were used effectively [21]. N-gram feature sets include the usage of feature selection approaches to obtain correct attributes subsets. Word n-grams contain bag-of-words (BOWs) and word n-grams in higher-order (e.g. bigrams, trigrams). [28] uses modified n-grams by integrating syntactic information on n-gram relationships. In most document classification activities, this n-gram model is implemented, and almost always boosts precision. This is because the n-gram model allows us to take the sequences of terms into account, as opposed to what will require to do just by using single words (unigrams). Looking into the benefits of the n-grams feature selection, in this paper, a rich collection of n-gram features that encompassed several fixed and variable n-gram categories is studied for classifying textual news articles.

2. Feature Selection

The selection of features serves as a crucial technique for reducing input data space dimensionality to minimize the computational cost. It was designed as a natural sub-part of the process of classification for many learning algorithms. Generally, three feature selection methods i.e. filter method, wrapper method, and embedded method achieve the objective of selecting important features. The ultimate goal of feature selection is always to find the collection of the best features out of the entire dataset to obtain improved classification results. Among all of the feature selection methods, information gain, chi-square, and Gini index have been used effectively [18], [29], [30]. These methods have shown promising results for classification [31]. CHI square reflects one of the more traditional feature selection strategies. In statistics, the CHI square test is used to analyze the independence of two instances. The instances, X and Y, are taken as separate if:

$$p(XY) = p(X)\, p(Y) \tag{4}$$

These two instances result in a particular word and class occurring respectively in the collection of text features. It can be calculated as given in equation (5):

$$Chi2(t, C) = \sum_{t,C\,\in\{0,1\}} \frac{\left(N_{t,C} - E_{t,C}\right)^2}{E_{t,C}} \tag{5}$$

Here, N is termed an observed frequency and E is the expected frequency for every term state t and class C. CHI square would be the function of how often the expected value E counts and N counts observed to deviate from one another. A high value of CHI square means that the independence supposition is wrong. If these two instances are related, then the term existence increases the probability of the class existence. This determines the weighted average score for all classes and then chooses the maximum score between all classes. In this paper, as in (6) given by [29], the former method is ideal to globalize the CHI square value for all classes. Here $P(C_i)$ is the likelihood of a class and $Chi2(t, C_i)$ is the unique Chi^2 value of a term t.

$$Chi^2(t) = \sum_{i=1}^{M} P(C_i).\, Chi^2(t, C_i) \tag{6}$$

Another effective method has been used by researchers i.e. Information Gain. This assesses the overall knowledge that the existence or absence of a word allows one to make the right classification judgment for every class [32]. In other words, it can be used in the selection of features by assessing each variable's gain in the target variable sense. The measurement between the two random variables is considered mutual information.

$$IG(t) =-\sum_{i=1}^{M} P(c) \log P(c) + P(t) \sum_{i=1}^{M} P(c/t) \log P(c/t) + P(\bar{t}) \tag{7}$$

In equation (7), the total classes are represented by M, probability of class c is represented by $P(c)$, the presence and absence of term t are denoted by $P(t)$ and $P(\bar{t})$, $P(c|t)$ and $P(c|\bar{t})$ are class c possibilities provided the existence and absence of a term t.

The other filter method which has been effectively used is the Gini Index [20]. In general, it has simpler computations than the other methods. It can be calculated as given in equation (8):

$$GI(t) = \sum_{i=1}^{M} P(t/C_i)^2 P(C_i/t)^2 \tag{8}$$

In (8), $P(t/C_i)$ is the likelihood of a term t provided that the class C_i is present. $P(C_i/t)$ is a class C_i probability given the presence of term t.

3. Classification Models

Classification is a supervised technique of machine learning wherein the computer algorithm learns from the data it receives as inputs and then uses the experience to classify new data. This data collection may be purely binary or multi-class classification. Types of classification tasks include voice recognition, handwriting recognition, scam detection, news labeling, etc. There has been several machine learning discovered from time to time with different approach and application. One of the models is *Naive Bayes*, simple to build and use for an extremely large volume of data. The classifier Naive Bayes claims that every other feature is unrelated to the inclusion of a specific feature in a class. Even though these characteristics depend on each other or the presence of the other characteristics, each of these properties contributes to the likelihood independently. It can be calculated as given in equation (9) and (10):

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \tag{9}$$

$$P(c|x) = P(x_1|c) \times P(x_2|c) \times \ldots \times P(x_n|c) \times P(c) \quad (10)$$

Here c refers to class and x represents inputs. Given the data x, $P(c|x)$ is mentioned as the posterior probability of c, $P(x|c)$ probability of input value x provided hypothesis was true, $P(c)$ represents the prior probability of c, and $P(x)$ is the prior probability of x.

[33] uses Naïve Bayesian classifier along with two feature evaluation metrics to multi-class text datasets i.e. multi-class Odds Ratio (MOR) and Class Discriminating Measure (CDM) to achieve the best feature selecting results. The other k-nearest-neighbors classifier algorithm takes up a lot of labeled points and uses them to know how to classify certain items. It looks at the points nearest to the new point to identify a new point, so whatever label most neighbors have is the new point label. [16] uses the neighbor-weighted *K-nearest neighbor* algorithm achieving significant performance gains in the classification of an imbalanced data set.

The statistical method, *Logistic Regression*, is used for evaluating a data set in which a result is calculated by one or more independent variables. Uses the probability log-odds of an event that is a linear combination of independent or prediction variables. Logistic Regression uses the Sigmoid activation function which results in either 0 or 1. It can be calculated as given in equation (11):

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (11)$$

Here, z represents the input variable.

It is proven superior to other binary classification such as *KNN*, as it also describes quantitatively the factors leading to classification [34]. The goal is to identify the best fit model to explain the relationship between the dichotomous value attribute and a series of independent variables. *Decision Tree* algorithm gives significant results for treating both categorical and numerical data. In the form of classification or regression models, the decision tree builds a tree structure. It splits down a collection of data into smaller and smaller subsets, thus constructing a linked decision tree incrementally. The tree splitting uses Chi-square, Gini-Index, and Information gain methods. A decision tree with improved chi-square feature selection outperforms in terms of recall for multiclass text classification [35].

The various classifiers being studied in the different applications have shown varied results. The authors have been proposed ensemble methods to further improve classification accuracy measures. Ensemble learning is the mechanism by which several models are systematically created and merged to solve a specific computational intelligence problem. *Random forests* are an ensemble learning system for classification, regression, and other functions that operates by creating a multitude of decision trees during training and providing class mode (classification) or mean forecasting (regression) of the individual trees. [36] uses ensemble methods for keyword extraction where Random Forest shows promising results. The authors have been improving such methods for effective text classification [37].

In the current scenario where data has been converting to big data, *Neural Networks* have been the most studied algorithms for text classification. A neural network is a type of layer-organized units (neurons) that transforms some output into an input vector. Every unit can take input, impose a function on it, and pass the output to the next layer. The networks are commonly known as feed-forward: a unit feeds its output to all the units on the next layer but no input is given to the previous layer. Weights are added to the signals that travel from one unit to another, and it is these weights that are adjusted during the training phase to fit a neural network to a specific problem. [38] proposes three distinct frameworks for sharing information with task-specific and shared layers to model text, based on *recurrent neural networks*. These deep learning algorithms' successes depend on their ability to model complex and nonlinear interactions within the data. Finding suitable architectures for these models, however, has been a problem for researchers addressing leveraging.

B. Techniques for Dealing With Imbalanced Data

We will illustrate in this section the various techniques which have been experienced so far by researchers for training a model to perform well against highly imbalanced data sets. The authors mentioned that where it comes to text classification, the normal distribution of textual data is often unbalanced. To better differentiate documents into minor categories, they used a basic probability-based word weighting scheme to solve the problem [39]. Many real-world text classification tasks, according to the authors, require unbalanced training instances. However, in the text domain, the methods introduced to resolve the imbalanced description have not been consistently tested. They conducted a survey based on the taxonomy of strategies suggested for imbalanced classification, such as resampling and instance weighting, among others [40]. The following sub-sections cover the literature of various techniques used so far to deal the text classification with imbalanced data sets.

1. Data Level Technique

Dealing with imbalanced data sets requires techniques such as enhancing classification algorithms or balancing the training data classes until the machine learning algorithm provides the data as input. The primary goal of balancing classes is either to raise the frequency of the minority class or to decrease the frequency of the majority class. This is provided for all classes to get roughly the same number of instances.

- Under-sampling aids in optimizing class allocation by randomly eliminating instances of majority classes. This is achieved when the majority and minority class cases are balanced completely. Evolutionary undersampling outperforms the non-evolutionary models by increasing the degree of imbalance [41]. They describe a performance function that incorporates two values: the classification factor aligned with both the sub-set of training instances and the percentage of reduction associated with the training set of the same sub-set of instances. A novel under-sampling technique is implemented, called cluster-based instance selection, which incorporates clustering analysis with instance selection [42]. The clustering analysis framework groups identical data samples of the majority class dataset into subclasses, while the instance selection framework extracts out unaccountable data samples from each subclass. It is also proven that under-sampling with KNN is the most powerful approach [43].
- Over-sampling raises the amount of minority class instances by arbitrarily replicating them to make the minority class more represented in the study. The author suggests a Random Walk Over-Sampling method by generating synthetic samples by randomly walking from real data to match different class samples [44]. This sampling method is designed to address the imbalanced grouping of data by producing some samples of a synthetic minority class. The synthetic samples, that properly follow the initial minority training set and extend the minority class boundaries, are coupled with the actual samples to make a more efficient full dataset, and the entire is used to build unbiased classifiers. Unfortunately, traditional over-sampling approaches have shown their respective shortcomings, such as causing serious over-generalization or not effectively improving the class imbalance in data space, while facing the more challenging problem as opposed to the binary class imbalance scenario. The author proposes a synthetic minority oversampling algorithm based on k-nearest neighbors (k-NN), called SMOM, for handling multi-class imbalance problems [20].

SMOM is a method to prevent over-generalization since safer neighboring directions are more likely to be chosen to produce synthetic instances. It is also suggested that combine sampling be rendered by combining the techniques of SMOTE and Tomek with SVM as the method of binary classification [45]. SMOTE is a useful over-sampling technique for increasing the number of positive classes incorporating sample drawing methods by replicating the data randomly so that the number of positive classes is equal to that of the negative class. [46] has performed multiclass classification with equal distribution of the data among various classes using SMOTE, owing to the introduction of synthetic instances which increased the number of training samples to distribute the data equally among 10 different labels. Tomek links method is under-sampling, which works by decreasing negative class numbers. However, in some extreme cases mixing sampling methods are no stronger than utilizing Tomek link methods.

2. Algorithms-Based Decomposition Techniques

The technique must first use decomposition strategies to transform the original multi-class data into binary subsets.

- *One-vs-all* is a strategy that requires training N independent binary classifiers, each programmed to identify a specific class. All those N classifiers are collectively used to classify multiple classes. With multi-class imbalanced data, an algorithm called One-vs-All with Data Balancing (OAA-DB) is built to enhance the classification performance [47]. It is mentioned that the OAA-DB algorithm can boost classification efficiency for imbalanced multi-class data without decreasing the overall classification accuracy. In other words, for every class, One-vs-All trains a single classifier, treating the existing class as the minority one and the remaining classes as a majority.
- *One-vs-One* trains a binary classifier for each potential pair of classes, ignoring examples that are not part of the pair classes. To resolve the multi-class imbalance classification problems, an exhaustive empirical study is proposed to investigate the possibility of improving the one-vs-one scheme through the application of binary ensemble learning approaches [48].
- *One-Against-Higher-Order* (OAHO) is an explicitly designed decomposition process for unequaled sets of data. OAHO first divides class by decreasing the number of samples [49]. OAHO sequentially marks the current class as 'positive class' and all the remaining classes with lower ranks as 'negative classes,' then trains a binary classifier.
- *All-in-One* uses One-vs-All along with One-vs-One, it first uses One-vs-All sub-classifiers to find the top two most probable categories for each test case, and then use the corresponding One-Vs-One sub-classified to decide the final result [50].

3. Algorithms-Based Ensemble Techniques

The main purpose of the ensemble methodology is to improve single classifier efficiency. The method involves constructing from the original data numerous two-stage classifiers and then aggregating their predictions.

a) Boosting-Based Techniques

One strategy which can be used to increase classification efficiency is boosting. Although several data sampling techniques are explicitly developed to fix the issue of class imbalance, boosting is a technique that can increase the efficiency of any weak classifier. *Ada Boost* iteratively constructs a model ensemble, which is an adaptive boosting strategy that combines many weak and inaccurate rules to build a predictive rule that is highly effective. During each iteration, case weights are changed to properly classify the instances in the next iteration that were wrongly classified during the current iteration. Upon completion, all models developed to take part in a weighted vote to identify unlabeled cases. Such a strategy is especially useful when grappling with class inequality as in successive implementations the minority class instances are more likely to be misclassified and thus assigned larger weights. In other words, it's a binary classification algorithm that combines many weak classifiers to create a stronger classifier [4]. Boosting can be achieved either by "reweighing" or "resampling". At each step, the changed example weights are transferred directly to the base learner while boosting by reweighing. Not all learning algorithms are designed to integrate example weights into their decision-making systems, however. This is a class that uses the AdaBoost Ml method to boost a nominal classifier which can only address nominal class problems. It is given in equation (12):

$$f(x) = sign(\sum_{m=1}^{M} \theta_m f_m(x)) \tag{12}$$

Here, $f(x)$ represents m^{th} weak classifier and θ_m is the corresponding weight.

This often improves the performance dramatically but sometimes overfits [51]. *Gradient boosting* is an approach that generates a set of weak regression trees by introducing iteratively a new one which further strengthens the learning goal by optimizing an arbitrary differentiable loss function [52]. Gradient Boosting builds the first learner to predict the samples on the training dataset and calculates the loss. And use that loss in the second stage to build an improved learner. The recent implementation of this boosting method called *XGBoost* combines the principles of computational efficiency. The paper presents a scalable end-to-end tree boosting system *XGBoost* that is widely used by data scientists to perform state-of-the-art machine learning outcomes [52].

b) Bagging-Based Techniques

Bootstrap aggregation, also known as *bagging*, is an ensemble meta-algorithm for machine learning that aims to enhance the stability and accuracy of classification algorithms. The standard algorithm requires the development of specific bootstrap training items, 'n' with substitution. Then train the algorithm on each bootstrapped algorithm separately, then aggregate the forecasts at the end. The authors present online bagging and boosting versions that require only one pass through the training data [53]. Random Forests is an ensemble classifier composed of several decision trees and generating the class which is the class output mode for individual trees. In this way, an RF ensemble classifier works more than a single tree from the classification results perspective [54]. The authors suggested ensemble classifiers focused on original principles such as learning cluster boundaries by the base classifiers and mapping cluster confidences to a class decision using a fusion classification [55]. The classified data set is divided into several clusters and is fed into several distinctive base classifiers. Cluster boundaries are identified to base classifiers and cluster confidence vectors are built. A second stage fusion classifier blends class decisions with confidences and maps of the clusters. This ensemble classifier restructured the learning environment for the base classifiers and promoted successful learning.

4. Other Techniques

Despite their effectiveness, however, sampling methods add complexity and the selection of required parameters. To address these problems, the author suggests a modern decision tree strategy named *Hellinger Distance Decision Trees* (HDDT), which allows the use of distance from Hellinger as the criteria for splitting. For probability and statistics, the Hellinger distance is used to measure the correlation of two distributions of probabilities. The authors use a Hellinger weighted ensemble of HDDTs to combat definition drift and improve the accuracy of single classifiers [56].

Error Correcting Output codes, ECOC is a common multi-class learning tool that works by breaking down the multi-class task into a set of binary class subtasks (dichotomies) and creating a binary classifier from each dichotomy. Both the dichotomy classifiers evaluate a test instance and then assign it to the nearest class in code space. A suitable code matrix, an effective learning strategy, and a decoding strategy highlighting minority classes are needed to enable ECOC to tackle multi-class imbalances. The authors propose the imECOC approach that operates on dichotomies to deal with both the imbalance between class and the imbalance within a class [57]. ImECOC assigns dichotomy weights and uses weighted decoding distances where optimum dichotomy weights are derived through reducing weighted loss in terms of minority classes.

The authors suggest merging weighted One-vs-One voting with a Winnow dynamic combiner customized to the program for the data stream. This will allow weights for classifiers to be dynamically modified, boosting the power of those competent in the current state of the stream [17]. *DOVO* simply adjusts the weights for classified objects returned via an active learning approach that enables even more consistent weights and lower processing costs. From those in the perspective of operation recognition, each action shall be taken over a given period. The proposed weighting procedure thereby enables to rapidly increase the significance of qualified classifiers to identify this particular behavior immediately after it has been identified by the active learning methodology and to sustain the significant importance of these related classifiers throughout its length.

C. Existing Solutions or Software for Classification With Imbalanced Datasets

A program, *KEEL* [58], provides a customized algorithm for the problem of classification with class imbalances. *Multi-IM* draws its basis from the probabilistic relational methodology (PRMsIM), developed to learn from imbalanced data for the problem of two categories [59]. *Imbalanced-learn*; A Python toolbox for resolving imbalanced results [60].

We use the following framework to evaluate the accuracy output of various ML algorithms and to validate our implementations in the classification of multi-class imbalance data on financial news datasets.

III. Framework and Working of Financial News Classification System: Challenges and Solutions of Data Imbalances

Text classification is crucial for information extraction and summarization, text retrieval, and question-answering in general. Using machine learning algorithms, the authors demonstrated the text classification process [19]. Following the approach, we developed a structure shown in Fig. 2. to distinguish the banking and other related sector-oriented news items from financial news posts. It involves three stages, including the data pre-processing phase, the training phase of the classifiers, and a comparative estimation of the performance phase of the classifiers. The phases are discussed in brief in the sub-sections along with certain challenges and solutions are given by researchers.

However, when faced with imbalanced multi-class results, we can drop output on one class quickly when attempting to get output on another class. A clearer analysis of the essence of the issue of class imbalance is required, as one should recognize in what realms class imbalance most impedes the output of traditional multi-class classifiers while developing a system suitable to this topic. Although most of the problems addressed in the preceded section can be applied to these multi-class concerns, the banking and other related news extraction from the financial news domain. We are identifying the following vital research directions for the future.

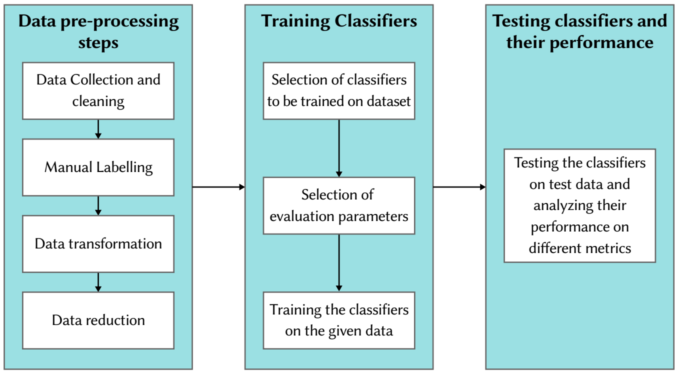

Data pre-processing steps
Data Collection and cleaning
Manual Labelling
Data transformation
Data reduction
Training Classifiers
Selection of classifiers to be trained on dataset
Selection of evaluation parameters
Training the classifiers on the given data
Testing classifiers and their performance
Testing the classifiers on test data and analyzing their performance on different metrics

Fig. 2. Multiclass classification of Financial News.

1. Data Pre-processing

Data preprocessing is a method used to transform the raw data into an effective and functional format. Effective pre-processing of text data is critical to achieving an appropriate output and better text classification quality [61].

Challenge-A: The task of preprocessing data here may be much more critical than in the case of binary issues. Possible difficulties can be easily identified: class overlap can occur in more than two classes, class label noise can influence the issue, and class boundaries may not be specific. Therefore, effective data cleaning and sampling techniques must be implemented to take into consideration the various characteristics of the classes and the balanced performance of all of them [62].

Solution-1: The problem of noise present in the data in the case of imbalanced distributions is incredibly difficult. Distortions may dramatically deteriorate classifier efficiency, particularly in the case of minority examples. New data cleaning methods need to be used to manage the existence of overlapping and chaotic samples which can also lead to worsening efficiency of the classifier. We might conceive projections into different spaces where the overlap is alleviated or basic examples are eliminated as mentioned in 3.1.3. However, measures are needed to assess whether a provided overlapping example can be excluded without discrimination to one class. A study of the effect on the real imbalance between classes is quite important in the case of label noise. Measures are therefore required to determine whether a given overlapping example can be discarded without compromising one of the classes. False labeling may lead to increasing the imbalance or disguise actual disproportions. This situation is handled with sustained methods for sensing and filtering noise, as well as handling and relabeling strategies for such examples as mentioned in 1.

Solution-2: Analysis of the kind of examples found in each class and their connections with other classes is interesting. Measuring each sample's difficulty here isn't straightforward, as it may adjust to various classes. For instance, for classes Banking and Governmental, news related to a collective decision on negative GDP outlook and modification on repo rate by RBI may be of borderline type while at the same time being a safe example when considering the remaining classes. Therefore, we have preferred a more flexible classification i.e. SMOTE. SMOTE functions by choosing similar examples in the vector space, drawing a line through the examples in the vector space, and drawing a new example at a point in the line.

Solution-3: New sampling approaches are needed for issues of multiple classes. Simple re-balancing is not a proper approach towards the largest or smallest class. We need to establish precise methods for adapting the sampling procedures to both the individual class property and their mutual relationships. [6] has provided the ensemble methods to deal with class imbalance classification, ADASYNBagging, and RSYNBagging. The ADASYN and RSYN were based on over-sampling and under-sampling techniques respectively. These were combined with a bagging algorithm to integrate the advantages of both algorithms. Another paper has provided a hybrid model to get a random sample from an unknown population. When compared with a random sample, a non-random sample could not provide better representative inferential statistics. Hence, to overcome this problem, Snoran Sampling Method was developed by [63]. We have not implemented these techniques in our paper. What sampling strategies would function best with the learning of the ensemble to boost class inequality, however, is highly dependent on problem domains.

2. Data Collection

To continue this, we gathered data by *scrapping* news from public news sources such as Bloomberg, Financial Express, Money Control, and Times of India using python-written code. As a result, we have been collected more than 10000 instances of financial news articles from the year 2017 to 2020. The news articles belong to different sectors or market segments. These are then pre-processed such that the machine learning algorithms may learn from the training dataset and adapt them in an acceptable way to the testing data collection. Therefore, these are pre-processed for the machine learning models to be explored from the training sample and implemented in an appropriate format to the test data set.

3. Labeling

The first step in the pre-processing phase is to *label* the news from 4 classes to which they belong to the specific sector. 4-classes are named as Banking, Global, Governmental, and Non-Banking. We prefer manual labeling [64] of the news articles with the help of experts of the financial domain where overlapping examples were preferred to discard without damaging one of the classes. Table I mentions the instance of each class as follows:

Table I. Sample of news articles from different sources and classes

Source	News article	Class
Source1[1]	The Kolkata-based private sector lender Bandhan Bank surpassed the market capitalization of all listed PSU banks except State Bank of India upon blockbuster stock market debut on Tuesday after floating India's biggest bank IPO earlier this month.	Banking
Source2[2]	For India, the current account deficit is within the comfort zone although it has widened and the GDP growth is heading towards 7.5-7.7 percent.	Governmental
Source3 [3]	The U.S. Federal Reserve has cut its benchmark interest rate by a half-point-the biggest reduction, and the first outside of scheduled meetings since the 2008 crisis year.	Global
Source4 [4]	The Nifty50 formed a bearish candle for the sixth consecutive day in a row and analysts feel that it will be hard for the index to breach the 200-DEMA in a hurry.	Non-Banking

4. Data Cleaning

They are then *cleaned* because the data can have several sections that are insignificant and missing. Data cleaning is done to handle that portion. It includes absent managing data, noisy data, etc. It helps the machine learning algorithms to efficiently grasp and operate on them.

5. Data Transformation

The next step, *data transformation*, is taken to turn the data into appropriate forms suited to the mining process, and the text of news articles therein is converted into measures with quantitative values by constructing a vector *set of features.* Since data mining is a technique used for managing enormous quantities of data. In these instances, research became harder when operating with a huge volume of data. To get rid of that, we use the strategy of *data reduction*. This seeks to increase the capacity of storage and reduce the expense of data collection and analysis. In other words, in the last step of this stage, the feature vector is normalized and scaled to prevent an *unbalanced dataset.*

A. Training Classifiers

Training is the practice of having text that is considered to belong to the defined classes and creating a classifier based on that known text. The basic concept is that the classifier accepts a collection of *training data* describing established instances of classes and uses the information obtained from the training data to determine the classes other unknown content belongs to, by conducting statistical analysis of training data. We can also use the classifier to derive information on your data based on the statistical analysis carried out during the training process. First, we identify the classes on a collection of training data, and then the classifier uses these classes to evaluate and decide the classification of other data. When the classifier assesses the data, it uses two often contradictory metrics to help decide if the content found in the new data belongs in or outside a class. *Precision*, is the likelihood that what has been labeled as being is actually in that class. High precision may come at the cost of missing certain results whose terms match those of other outcomes in other groups. *Recall*, the likelihood that an object is listed as being in that class in fact in a class. High recall may come at the cost of integrating outcomes from other classes whose terms match those of target class results. We need to find the right balance with high precision and high recall while we are tuning our classifier. The balance focuses on what our priorities and criteria are for implementation. We need to train the classifier with *sample data* that describes members of all the classes to find the best thresholds for our data. Finding good training samples is very critical because the nature of the training can directly influence the quality of the classification. The samples should be statistically valid for each class and should include samples that include both solid class examples and samples near the class boundary.

Challenge-B: The strong potential resides in the complexity of multi-class, distort-insensitive classifiers. They will permit multi-class complications to be handled without referring to strategies for resampling when algorithm-level approaches are used to counter class imbalances. So one may wonder if other prominent classifiers can be adapted in this case [62].

Solution-1: Certain issues should be addressed when constructing multi-level classifiers in the case of class imbalances. A broader study is required of how numerous unbalanced data sets influence decision boundaries in classifiers. Based on [65] Hellinger distance has proved useful in cases of class imbalance. Since accuracy may offer a distorted picture of success on unbalanced data, current stream classifiers are focused on accuracy that is hampered by minority class output on unbalanced streams, resulting in low recall levels of minority classes. A split based on Hellinger Distance will give a high score to a split separating the classes in the best way relative to the parent population.

[1] www.financialexpress.com

[2] www.moneycontrol.com

[3] www.bloombergquint.com

[4] www.moneycontrol.com

When utilizing Hellinger, it is possible to obtain a statistically relevant change in the recall level on imbalanced data sources, with a reasonable rise in the false positive rate.

Solution-2: Other solutions with potential robustness to the imbalance, such as methods based on density, need to be explored. [66] have provided a more thorough review of the cluster oversampling based on density and in terms of density-dependent clustering under-sampling techniques. Their findings suggest the strategy will boost the classifier's predictive efficiency. It also yields the best in the precision average.

Solution-3: While modern methods of learning with imbalances are suggested to tackle the question of data imbalances, they have certain limitations; under-sampling methods lose essential details, and cost-sensitive methods are prone to outliers and noise. [67] has provided an efficient hybrid ensemble classifier architecture incorporating density-based under-sampling and cost-effective approaches by investigating state-of-the-art solutions using an algorithm for multi-objective optimization. First, they developed a density-based under-sampling method to select informative samples with probability-based data transformation from the original training data, which enables multiple subsets to be obtained following a balanced class-wide distribution. Second, they have used the cost-sensitive approach of classification to address the problem of information incompleteness by modifying weights in minority groups misclassified, rather than majority samples. Finally, they implemented a multi-objective optimization method and used sample-to-sample relations to auto-modify the classification outcome utilizing an ensemble classification system [68-80].

B. Testing Classifiers and Their Performance

We run the trained classifier on unknown news articles to check a classifier to decide which classes each news article belongs to. The goal of this stage is to check the performance of the classifiers on the training set and to see if they detect the training correctly. The classifiers considered will be graded according to their effectiveness in detecting the appropriate class. In the later section, we will test various classifiers on the unseen news articles and compare the performance of each.

IV. Working of Financial News Classification System

Throughout this section, we describe first the experimental method used to train the classifiers and then demonstrate their success in the classification of news articles into four separate classes. It should be noted here that most text classification algorithms are prone to the form and design of the dataset, depending on factors such as class size, class disparity (number of samples per class), feature scaling, number of training samples, number of features, etc. Besides, different algorithms follow different approaches to solving problems of multi-class classification which also affects their performance. So, we have faced some challenges and, to address these challenges, we have made sure that the available data from which each classifier will learn is distributed equally for each class.

A. Experimental Set Up to Train the Classifiers

We used the Tableau prep tool for the data cleaning and preprocessing operations, while the desktop tool was used for the data visualization. The classification tests were performed on Python 3.8 utilizing numerous Python-supported libraries to incorporate machine learning and deep learning algorithms. With a split of 75% and 25% respectively, the total of 10,000 news articles is divided into training and test data. The news articles are related to 4 different classes as mentioned in the introductory section. The data was imbalanced. So, to balance the data various sampling techniques were used. As stated in the introductory section, the news articles are linked to 4 different classes. In nature, the data had been imbalanced. Therefore, different sampling strategies were used to balance the data among classes as discussed in section 4.2. The machine and deep algorithms were further implemented on data for classification using scikit-learn and imblearn libraries of Python. Scikit-learn offers a package named the TfidfVectorizer for the extraction of functionality from text documents. This class is responsible for both vectorizing text documents (news articles) into vectors of word features and transforming them of the term vectors in the scores of TfIdf. We also vectorized the dataset during the experiments using the N-gram approach, with unigrams, bigrams, and tri-grams.

B. Results and Discussion

We have carried out several experiments on our pre-processed data collection utilizing conventional machine learning algorithms detailed in the section preceding. The key purpose of these experiments is to determine the right classifier that gives the best performance. Every classifier's output concerning classification is calculated using the metrics Precision, Recall, and F_1-score. The accuracies are obtained with both train-test split and 5-fold cross-validation for all classifiers. The outcomes of the chosen classifiers are described in the sub-sections that follow.

For the traditional machine learning algorithms, TF-IDF features of 1-gram, 2-gram, and 3-gram were used. The detailed experiments on the financial news datasets were carried out.

1. Results of Multiclass Classification With Data-Imbalances

Table II lists the results of each of the classifiers where data is being vectorized using the N-gram TF-IDF feature with data imbalances across classes. From the different classifiers Decision Tree {criterion='gini' to measure qualility of split, splitter='best', max_depth=2 for maximum depth of tree, random_state=1 is the seed for random number generator}, Linear SVC {C=1 regularization parameter, multi_class='y'}, Logistic Regression {C=1 regularization parameter, random_state=0}, Multinomial Naïve Bayes {alpha=1.0 smoothing parameter}, Random Forest {n_estmators=100 for number of trees, random_state=1 will always produce same results with same parameters and training data, max_depth=3 for maximum depth of the tree}, and Multilayer Perceptron {solver='lbfgs' for weight optimization, alpha=0.0001 L2 penality, learning_rate='constant', hidden_layer_sizes=(5,2), random_state=1}, Random Forest performed best with accuracy 88% as shown in Table III. The Random Forest

TABLE II. Results for the Classifiers for Different Classes With Data Imbalances

Classifier	N-Gram	Banking			Global			Non-Banking			Governmental		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Decision Tree	1, 2, 3	0.93	0.96	0.94	0.76	0.68	0.72	0.90	0.95	0.92	0.80	0.31	0.44
Linear SVC	1, 2, 3	1.00	0.77	0.87	0.86	0.61	0.71	0.86	0.98	0.91	1.00	0.38	0.56
Logistic Regression	1, 2, 3	1.00	0.31	0.47	0.88	0.34	0.49	0.76	0.99	0.86	0.00	0.00	0.00
Multinomial NB	1, 2, 3	0.86	0.23	0.36	0.73	0.54	0.62	0.79	0.97	0.87	0.00	0.00	0.00
Random Forest	1, 2, 3	0.93	1.00	0.96	0.89	0.59	0.71	0.88	0.98	0.92	1.00	0.23	0.38
Multi-layer Perceptron	1, 2, 3	1.00	0.69	0.82	0.78	0.68	0.73	0.86	0.96	0.91	1.00	0.31	0.47

achieved the F_1-score 0.96, 0.71, 0.92, 0.38 for classes Banking, Global, Non-Banking, and Governmental respectively. The comparison of all the mentioned classifiers for 4-different classes is visualized in Fig. 3-6. Table III shows that the accuracy comes out to be 78%-88% range for all machine learning algorithms with train-test split and cross-validation.

TABLE III. Accuracy of the Classifiers With Imbalanced Data

Classifier	Accuracy(Train/Test)	Cross-Validation
Decision Tree	0.87	0.87
Linear SVC	0.87	0.87
Logistic Regression	0.78	0.78
Multinomial NB	0.78	0.78
Random Forest	0.88	0.88
Multi-layer Perceptron	0.87	0.87

However, Table II shows that the recall of the minority classes is very less. The Logistic Regression and Multinomial NB has shown 0% precision and recall for the minority class i.e. Governmental. This is visualized in Fig. 5. At the same time, the precision and recall for the other classes have shown high precision and recall. It shows that machine learning models are more biased towards the majority class. So, we need to apply imbalanced data handling techniques.

2. Results of Multiclass Classification With Data Balance Using Data-Level Technique: Random Over-Sampling With Replacement

The Resampling takes place with the exclusion of the minority class, increasing the sample number to equal that of the majority class. Tables IV and V lists the results of each of the classifiers where data is being vectorized using the N-gram TF-IDF feature with data balanced using random over-sampling technique across classes.

From the different classifiers Decision Tree, Linear SVC, Logistic Regression, Multinomial Naïve Bayes, Random Forest, and Multilayer Perceptron with accuracy for all classes with balanced datasets using up-sampling, the Random Forest again performed best with accuracy 99% as shown in Table V. The Random Forest achieved the F_1-score 1.00, 0.98, 0.98, 1.00 for classes Banking, Global, Non-Banking, and Governmental respectively. The comparison of all the mentioned classifiers for 4-different classes is shown in Tables IV and V. It is observed that with data balances the precision and recall have also

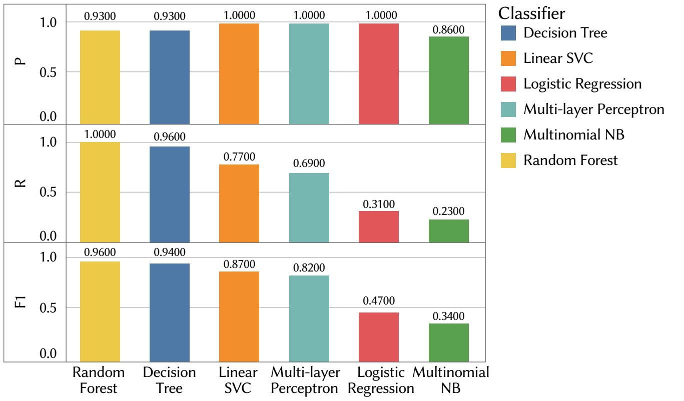


Fig. 3. Performance metrics P, R, F-1 for various classifiers for Banking Class.

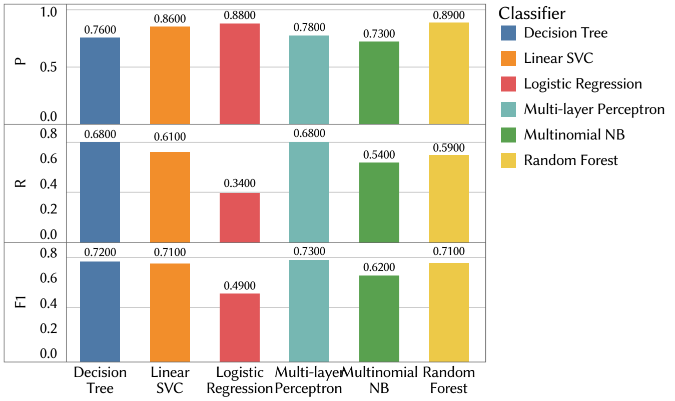


Fig. 4. Performance metrics P, R, F-1 for various classifiers for Global Class.

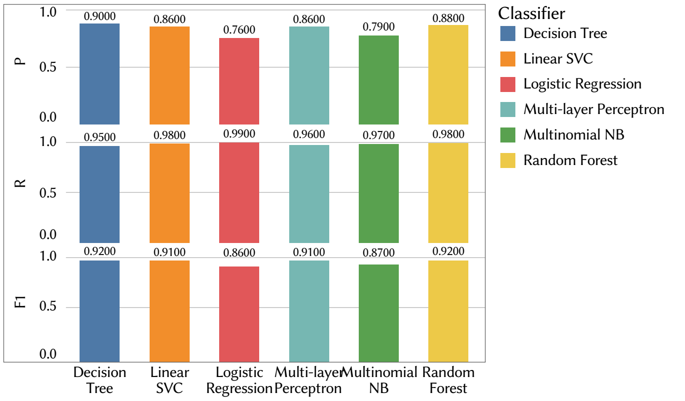


Fig. 5. Performance metrics P, R, F-1 for various classifiers for Non-Banking Class.

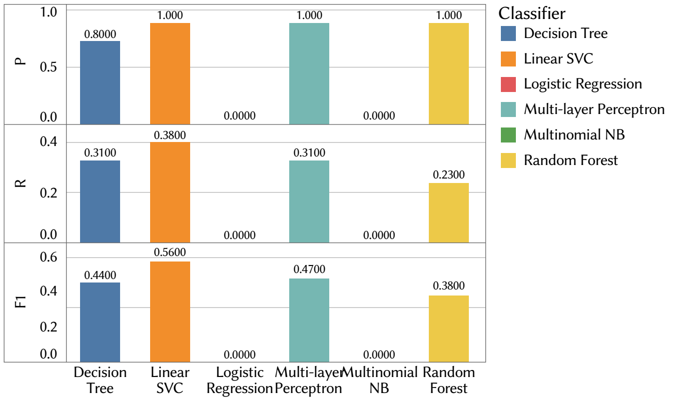


Fig. 6. Performance metrics P, R, F-1 for various classifiers for Governmental Class.

TABLE IV. Results for the Classifiers for Different Classes With Balanced Data Using Up-Sampling

Classifier	N-Gram	Banking			Global			Non-Banking			Governmental		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Decision Tree	1, 2, 3	0.99	1.00	0.99	0.95	0.99	0.97	0.99	0.94	0.97	0.99	1.00	1.00
Linear SVC	1, 2, 3	0.98	1.00	0.99	0.98	0.99	0.98	0.99	0.97	0.98	0.99	1.00	1.00
Logistic Regression	1, 2, 3	0.98	1.00	0.99	0.93	1.00	0.96	1.00	0.91	0.95	0.99	1.00	1.00
Multinomial NB	1, 2, 3	0.97	0.96	0.97	0.93	0.97	0.95	0.93	0.83	0.88	0.94	1.00	0.97
Random Forest	1, 2, 3	0.99	1.00	1.00	0.98	0.99	0.98	0.99	0.98	0.98	1.00	1.00	1.00
Multi-layer Perceptron	1, 2, 3	0.99	1.00	0.99	0.94	0.99	0.97	0.99	0.93	0.96	1.00	1.00	1.00

improved for every classifier. And the accuracy of the classifiers varies between 94% to 100% and it is visualized in Fig. 7.

TABLE V. Accuracy of the Classifiers With Balanced Data Using Up-Sampling

Classifier	Accuracy(Train/Test)	Cross-Validation
Decision Tree	0.98	0.983
Linear SVC	0.98	0.982
Logistic Regression	0.98	0.975
Multinomial NB	0.94	0.948
Random Forest	0.99	0.996
Multi-layer Perceptron	0.98	0.985

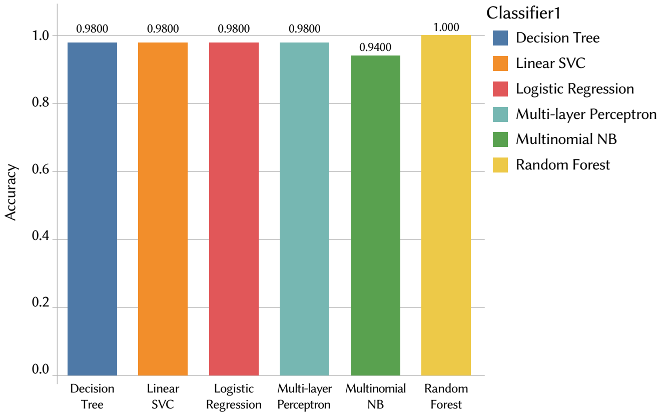


Fig. 7. Accuracy for various classifiers with balanced classes using Up-Sampling.

3. Results of Multiclass Classification With Data Balance Using Data-Level Technique: Random Down-Sampling Without Replacement

This is done by resampling the majority class without replacement, setting the number of samples corresponding to that of the minority class. Table VI, VII lists the results of each of the classifiers where data is being vectorized using the N-gram TF-IDF feature with data balanced using down-sampling technique across classes.

From the different classifiers Decision Tree, Linear SVC, Logistic Regression, Multinomial Naïve Bayes, Random Forest, and Multilayer Perceptron with accuracy for all classes with balanced datasets using down-sampling, the Random Forest again performed best with an accuracy of 80% as shown in Table VII.

The Random Forest achieved the F_1-score 0.95, 0.83, 0.73, 0.70 for classes Banking, Global, Non-Banking, and Governmental respectively. The comparison of all the mentioned classifiers for 4-different classes is shown in Tables VI and VII. The accuracy of the classifiers has degraded with data balances using down-sampling as compared to up-sampling. And the accuracy of the classifiers varies between 67% to 80% and it is visualized in Fig. 8.

TABLE VII. Accuracy of the Classifiers With Balanced Data Using Down-Sampling

Classifier	Accuracy(Train/Test)	Cross-Validation
Decision Tree	0.69	0.695
Linear SVC	0.76	0.764
Logistic Regression	0.71	0.720
Multinomial NB	0.67	0.750
Random Forest	0.80	0.803
Multi-layer Perceptron	0.69	0.692

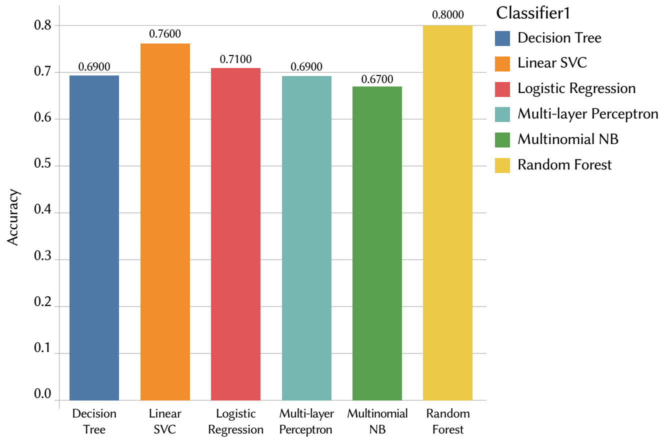


Fig. 8. Accuracy for various classifiers with balanced classes using Down-Sampling.

4. Results of Multiclass Classification With Data Balance Using Data-Level Technique: Hybrid Over-Sampling Technique SMOTE

SMOTE helps to balance the representation of the classes by replicating randomly through minority class examples. SMOTE synthesizes new instances within existing instances of minority classes. This produces the virtual train records by linear interpolation for the minority class. For each case, these synthetic training records are created by a random selection of one or more k-nearest neighbors in the minority class. The data is reconstructed after the oversampling process, and the classification models are implemented for the processing data. Table VIII lists the results of each of the classifiers where data is being vectorized using the N-gram TF-IDF feature with data balanced using the over-sampling technique SMOTE across classes.

From the different classifiers Decision Tree, Linear SVC, Logistic Regression, Multinomial Naïve Bayes, Random Forest, and Multilayer Perceptron with accuracy for all classes with balanced datasets using up-sampling, the Random Forest again performed best with accuracy 100% as shown in Table IX. The Random Forest achieved the F_1-score 0.99, 1.00, 0.99, 1.00 for classes Banking, Global, Non-Banking, and Governmental respectively. The comparison of all the mentioned classifiers for 4-different classes is shown in Tables VIII and IX. It is observed that with data balances using SMOTE the precision and recall have also improved for every classifier. And the accuracy of the classifiers varies between 94% to 100% as visualized in Fig. 9.

TABLE VI. Results for the Classifiers for Different Classes With Balanced Data Using Down-Sampling

Classifier	N-Gram	Banking			Global			Non-Banking			Governmental		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Decision Tree	1, 2, 3	1.00	0.91	0.95	0.56	0.83	0.67	0.71	0.38	0.50	0.60	0.67	0.63
Linear SVC	1, 2, 3	1.00	0.91	0.95	0.75	0.80	0.77	0.78	0.67	0.72	0.67	0.67	0.67
Logistic Regression	1, 2, 3	1.00	0.82	0.90	0.69	0.75	0.72	0.70	0.54	0.61	0.54	0.78	0.64
Multinomial NB	1, 2, 3	0.82	0.82	0.82	0.69	0.75	0.72	0.67	0.31	0.42	0.53	0.89	0.67
Random Forest	1, 2, 3	1.00	0.91	0.95	0.83	0.83	0.83	0.89	0.62	0.73	0.57	0.89	0.70
Multi-layer Perceptron	1, 2, 3	0.88	0.64	0.74	0.71	0.83	0.77	0.62	0.62	0.62	0.60	0.67	0.63

TABLE VIII. Results for the Classifiers for Different Classes With Balanced Data Using SMOTE

Classifier	N-Gram	Banking			Global			Non-Banking			Governmental		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Decision Tree	1, 2, 3	0.99	0.93	0.96	0.99	1.00	1.00	0.95	0.99	0.97	0.98	1.00	0.99
Linear SVC	1, 2, 3	0.99	0.92	0.96	0.98	1.00	0.99	0.94	0.99	0.97	0.99	1.00	1.00
Logistic Regression	1, 2, 3	1.00	0.91	0.95	0.98	1.00	0.99	0.93	1.00	0.96	0.99	1.00	1.00
Multinomial NB	1, 2, 3	0.92	0.85	0.88	0.97	0.96	0.97	0.94	0.96	0.95	0.94	1.00	0.97
Random Forest	1, 2, 3	0.99	0.99	0.99	0.99	1.00	1.00	0.99	0.99	0.99	1.00	1.00	1.00
Multi-layer Perceptron	1, 2, 3	0.99	0.93	0.96	0.99	1.00	0.99	0.94	0.99	0.97	1.00	1.00	1.00

TABLE IX. Accuracy of Classifiers With Balanced Data Using SMOTE Up-Sampling

Classifier	Accuracy(Train/Test)	Cross-Validation
Decision Tree	0.98	0.981
Linear SVC	0.98	0.948
Logistic Regression	0.98	0.972
Multinomial NB	0.94	0.948
Random Forest	1.00	0.995
Multi-layer Perceptron	0.98	0.986

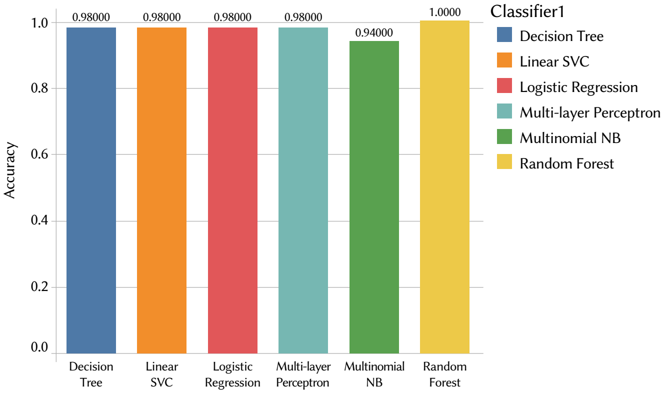


Fig. 9. Accuracy for various classifiers with balanced classes using SMOTE Up-Sampling.

5. Results of Multiclass Classification With Data Balance Using Data-Level Technique: Over-Sampling Technique ADASYN

ADASYN (Adaptive synthetic sampling approach) algorithm builds on the methodology of SMOTE. This uses a weighted distribution for specific examples of minority classes due to their degree of learning capacity, whereas more sophisticated data is generated for examples of minority classes that are more difficult to understand. The key idea of the ADASYN algorithm is to use a density distribution as a parameter to automatically calculate the number of synthetic samples that each minority data example requires to be generated. The data is reconstructed after the oversampling process, and the classification models are implemented for the processing data. Table X lists the results of each of the classifiers where data is being vectorized using the N-gram TF-IDF feature with data balanced using the over-sampling technique ADASYN across classes.

From the different classifiers Decision Tree, Linear SVC, Logistic Regression, Multinomial Naïve Bayes, Random Forest, and Multilayer Perceptron with accuracy for all classes with balanced datasets using up-sampling, the Random Forest again performed best with accuracy 91% as shown in Table XI. The Random Forest achieved the F_1-score 0.94, 0.79, 0.94, 0.53 for classes Banking, Global, Non-Banking, and Governmental respectively. The comparison of all the mentioned classifiers for 4-different classes is shown in Tables X and XI. It is observed that with data balances using ADASYN the precision and recall have also downgraded as compared to SMOTE based up-sampling for every classifier. And the accuracy of the classifiers varies between 72% to 91% as visualized in Fig. 10.

TABLE XI. Accuracy of the Classifiers With Balanced Data Using ADASYN Up-Sampling

Classifier	Accuracy(Train/Test)	Cross-Validation
Decision Tree	0.87	0.872
Linear SVC	0.87	0.865
Logistic Regression	0.88	0.881
Multinomial NB	0.72	0.725
Random Forest	0.91	0.914
Multi-layer Perceptron	0.86	0.863

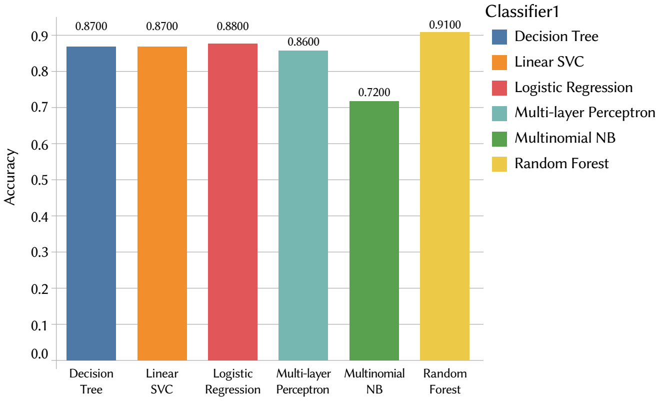


Fig. 10. Accuracy for various classifiers with balanced classes using ADASYN Up-Sampling.

TABLE X. Results for the Classifiers for Different Classes With Balanced Data Using ADASYN

Classifier	N-Gram	Banking			Global			Non-Banking			Governmental		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Decision Tree	1, 2, 3	0.89	0.96	0.93	0.82	0.66	0.73	0.91	0.94	0.92	0.42	0.38	0.40
Linear SVC	1, 2, 3	0.96	0.85	0.90	0.76	0.71	0.73	0.89	0.95	0.92	0.62	0.38	0.48
Logistic Regression	1, 2, 3	0.96	0.85	0.90	0.79	0.76	0.73	0.90	0.95	0.92	0.60	0.46	0.52
Multinomial NB	1, 2, 3	0.50	0.85	0.63	0.57	0.93	0.70	0.99	0.65	0.78	0.29	0.77	0.43
Random Forest	1, 2, 3	0.93	0.96	0.94	0.91	0.71	0.79	0.91	0.98	0.94	0.88	0.38	0.53
Multi-layer Perceptron	1, 2, 3	0.94	0.65	0.77	0.73	0.73	0.73	0.88	0.95	0.92	0.88	0.38	0.53

TABLE XII. Results for the Classifiers for Different Classes With Balanced Data Using Near-Miss

Classifier	N-Gram	Banking			Global			Non-Banking			Governmental		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Decision Tree	1, 2, 3	0.68	1.00	0.81	0.23	0.24	0.24	0.82	0.70	0.75	0.25	0.54	0.34
Linear SVC	1, 2, 3	0.41	0.96	0.57	0.30	0.56	0.39	0.88	0.45	0.59	0.27	0.69	0.39
Logistic Regression	1, 2, 3	0.43	0.96	0.60	0.29	0.56	0.38	0.94	0.17	0.28	0.12	0.92	0.22
Multinomial NB	1, 2, 3	0.32	0.88	0.47	0.30	0.59	0.40	0.91	0.32	0.47	0.20	0.77	0.32
Random Forest	1, 2, 3	0.91	0.77	0.83	0.67	0.20	0.30	0.82	0.97	0.89	0.60	0.46	0.52
Multi-layer Perceptron	1, 2, 3	0.46	0.81	0.58	0.57	0.61	0.59	0.91	0.72	0.80	0.28	0.62	0.38

6. Results of Multiclass Classification With Data Balances Using Data-Level Technique: Down-sampling Technique Near-Miss

The NearMiss Algorithm under-sampled the majority class's instances and made them equivalent to the minority class. The majority classes, here, were reduced to the minimum number as of minority class so that all classes would have the same number of records. The data is reconstructed after the down-sampling process using the Near-Miss method, and the classification models are implemented for the processing data. Table XII lists the results of each of the classifiers where data is being vectorized using the N-gram TF-IDF feature with data balanced using the down-sampling technique Near-Miss across classes.

From the different classifiers Decision Tree, Linear SVC, Logistic Regression, Multinomial Naïve Bayes, Random Forest, and Multilayer Perceptron with accuracy for all classes with balanced datasets using down-sampling, the Random Forest again performed best with an accuracy of 81% as shown in Table XIII. The Random Forest achieved the F_1-score 0.83, 0.30, 0.89, 0.52 for classes Banking, Global, Non-Banking, and Governmental respectively. The comparison of all the mentioned classifiers for 4-different classes is shown in Table XII and XIII. The accuracy of the classifiers has degraded with data balances using down-sampling with the Near-Miss approach as compared to all other up-sampling approaches as visualized in Fig. 11.

TABLE XIII. Accuracy of the Classifiers With Balanced Data Using Near-Miss Down-Sampling

Classifier	Accuracy(Train/Test)	Cross-Validation
Decision Tree	0.65	0.652
Linear SVC	0.53	0.526
Logistic Regression	0.34	0.344
Multinomial NB	0.43	0.431
Random Forest	0.81	0.814
Multi-layer Perceptron	0.70	0.704

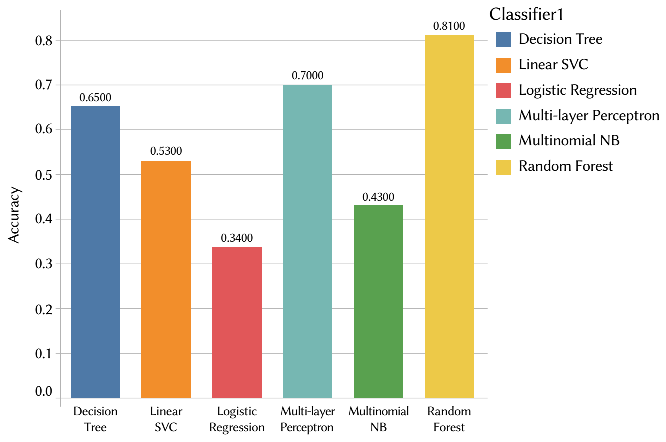


Fig. 11. Accuracy for various classifiers with balanced classes using Near-Miss Down-Sampling.

7. Results of Multiclass Classification With Data Balance Using Ensemble Classifiers

Ensemble models are meta-algorithms incorporating many strategies in machine learning into one predictive model to minimize variance (bagging), bias (boosting), or strengthen predictions (stacking). Bagging methods build multiple estimators on various randomly chosen subsets of data in ensemble classifiers. The classifier is called BaggingClassifier in scikit-learn. This classifier, however, does not require a balancing of the data sub-set. So, this classifier would support the plurality groups when training on imbalanced data set. **BalancedBaggingClassifier** requires each subset of data to be resampled until any of the ensemble estimators are equipped. In brief, the performance of an EasyEnsemble sampler is paired with an ensemble of classifiers (i.e., BaggingClassifier). Hence the BalancedBaggingClassifier requires the same parameters as the BaggingClassifier scikit-learn. Additionally, there are two additional parameters to monitor the actions of the random under-sampler, sampling strategy, and substitution. **BalancedRandomForestClassifier** is another ensemble method that provides a balanced bootstrap sample for each tree in the forest. **RUSBoostClassifier** sub-sample the data collection randomly before executing a boosting iteration. A particular method in the bagging classifier which uses AdaBoost as learners is named EasyEnsemble. The **EasyEnsembleClassifier** allows AdaBoost learners to be trained on appropriate samples of bootstrap. Table XIV lists the results of each of these ensemble classifiers for the various classes. And Table XV shows the accuracy of these ensemble classifiers.

From the different ensemble classifiers BalancedBaggingClassifier, BalancedRandomForestClassifier, RUSBoostClassifier, EasyEnsemble Classifier with accuracy for all classes, the BalancedBaggingClassifier performed best with an accuracy of 99% as shown in Table XV. The BalancedBaggingClassifier achieved the F_1-score 0.97, 1.00, 0.98, 1.00 for classes Banking, Global, Non-Banking and Governmental respectively. The comparison of all the mentioned ensemble classifiers for 4-different classes is shown in Table XIV and XV. The accuracy of the BalancedBaggingClassifier has resulted in 99% which is quite similar to the result of multiclassification using Random-Forest Classifier with SMOTE sampling i.e. 100%. The accuracy of the Random Forest classifier with a random up-sampling approach for data balances is also 99%. The comparison of the accuracies of classifiers across all approaches is visualized in Fig. 12. The accuracy of classifiers with down-sampling using the Near-Miss approach is worst amongst all.

The accuracy, precision, recall, and F-1 of Random-Forest Classifier with SMOTE sampling is very good in terms of multiclass news classification. However, under Governmental and Banking classes (minor classes in original), the precision of Random Forest with SOMTE overlapped with the precision of Random Forest with a random up-sampling approach. The comparison of the Precision of classifiers with each approach across all mentioned classes is visualized in Fig. 13. Some of the key explanations for the low performance of some of the classifiers, including Linear SVC and Multinomial naïve Bayes, is that a huge number of features don't fit well for them. Earlier it has been stated that Multinomial Naive Bayes' output is very weak when

TABLE XIV. Results for the Ensemble Classifiers for Different Classes

Classifier	N-Gram	Banking			Global			Non-Banking			Governmental		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
BalancedBaggingClassifier	1, 2, 3	0.98	0.97	0.97	0.99	1.00	1.00	0.98	0.98	0.98	0.99	1.00	1.00
BalancedRandomForestClzssifier	1, 2, 3	0.86	0.64	0.73	0.99	0.90	0.94	0.66	0.88	0.76	0.84	0.87	0.86
RUSBoostClassifier	1, 2, 3	0.33	1.00	0.50	0.94	0.34	0.50	0.08	0.05	0.06	0.00	0.00	0.00
EasyEnsembleClassifier	1, 2, 3	0.87	0.93	0.90	0.57	0.44	0.49	0.26	0.85	0.40	0.92	0.83	0.87

the dataset faces class imbalance problems. The result has shown that the efficiency of the RUSBoostClaasifer ensemble algorithm is very poor when it comes to the multi-class classification of text with noisy data and class imbalance.

TABLE XV. Accuracy of the Ensemble Classifiers

Classifier	Accuracy (Train/Test)	Cross-Validation
BalancedBaggingClassifier	0.99	0.991
BalancedRandomForestClassifier	0.82	0.823
RUSBoostClassifier	0.34	0.344
EasyEnsembleClassifier	0.78	0.781

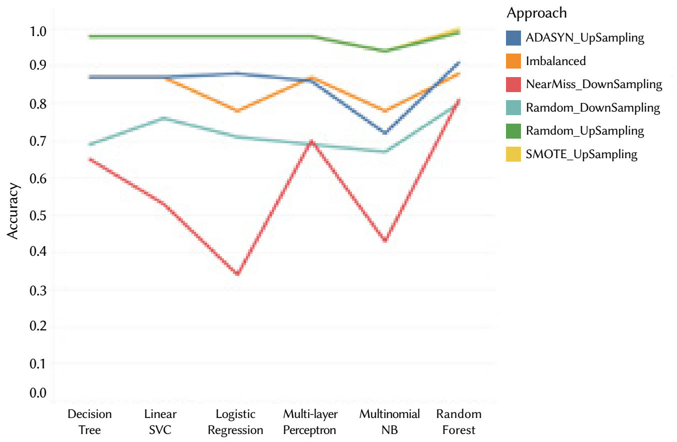


Fig. 12. Comparison of accuracies with Classifiers across different approaches.

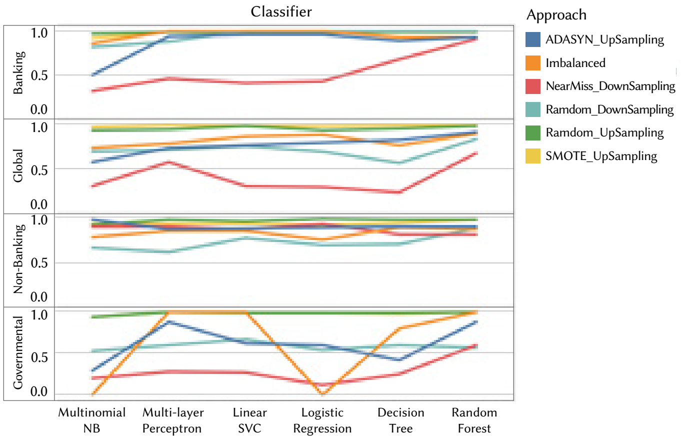


Fig. 13 Comparison of Precision with Classifiers under each class across different approaches.

It is clear from the Fig. 14., the recall of the classifier Random Forest with data balanced across classes using random up-sampling and SMOTE is increased as compared to down-sampling techniques random down-sampling and Near-Miss. The comparison of recall across all approaches with different classifiers under each class is visualized in Fig. 14.

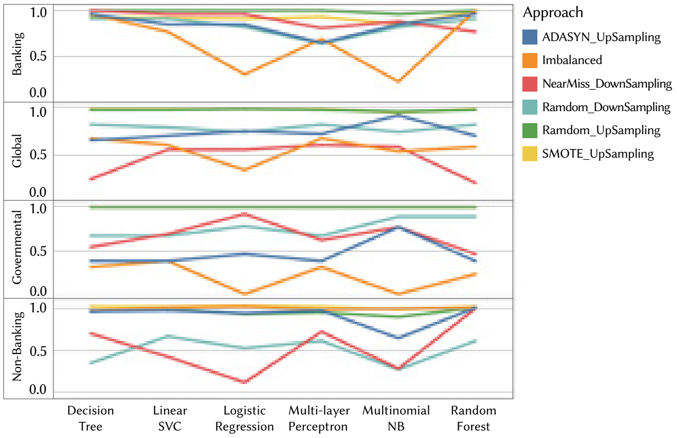


Fig. 14. Comparison of Recall with Classifiers under each class across different approaches.

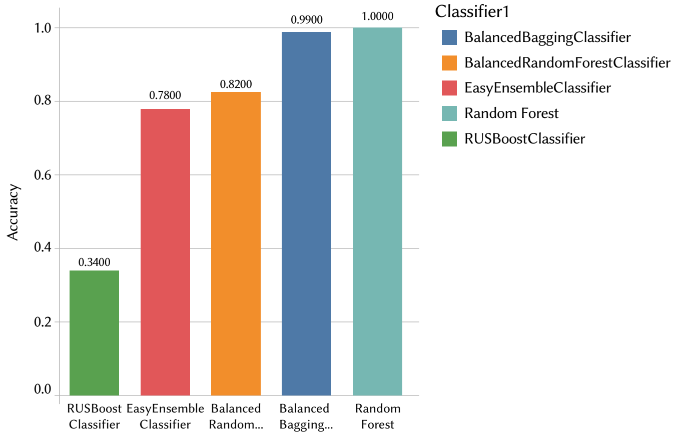


Fig. 15. Ensemble classifiers vs Random Forest (SMOTE).

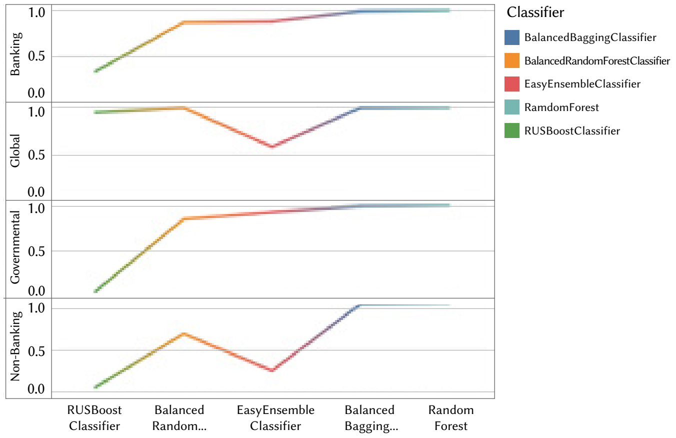


Fig. 16. Precision of ensemble classifiers vs Random Forest (SMOTE).

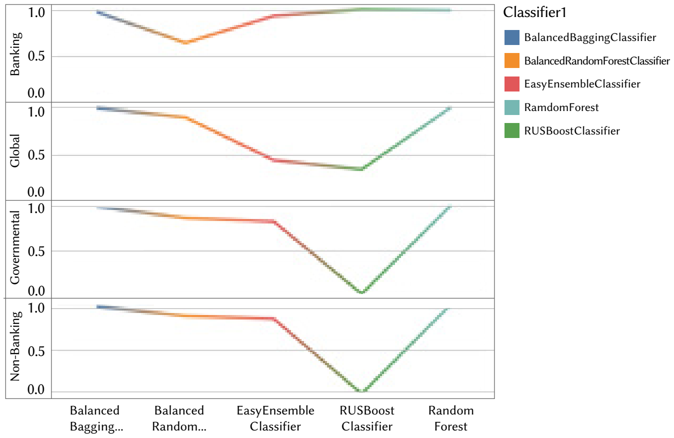


Fig. 17. Recall of ensemble classifiers vs Random Forest (SMOTE).

The accuracy of the ensemble classifiers is compared with Random Forest with SMOTE and it is visualized in Fig. 15. The accuracy of multi-class financial news classification using Random Forest with data balanced using SMOTE is higher as compared to all other ensemble classifiers discussed in the previous section. It is slightly greater than BalancedBaggingClassifier. The precision and recall of Random Forest with data balanced using SMOTE across all classes are higher as compared to all other ensemble classifiers and it is visualized in Fig. 16. and 17 respectively.

V. Conclusion and Future Direction

This paper aims to extract banking news from the pool of articles on financial news. This multi-class Financial News classification will help to get news on the banking domain. The development of a system for gathering banking news and other relevant domains is a major and untested problem for the Indian stock market. We're interested in seeking news from Indian banks, the Indian government, and the global. We take a structured approach to divide the news into realms of our choosing, grouping the news articles into 4 classes. The news articles are gathered from numerous online news sources and labeled to derive the banking and other related news to achieve the paper's goal. To automate the classification process, 5 traditional machine learning classifiers, 1 neural network classifier, and 4 ensemble classifiers are used to classify the news articles into 4 classes (Banking, Governmental, Global, and Non-Banking). Since our data set faces the class imbalance issue, we used many methods to align the data set between classes, and the classifier output is evaluated using the original imbalanced and balanced data set. We used precision, recall, F-1, and accuracy parameters to evaluate the classification models. It is evident from results that Random Forest with balanced data using SMOTE achieved the highest accuracy of 100% whereas other models have lower classification accuracy even with 34%. Based on our results, our trained classification model can be used to classify the news into other specific domains by training the model on data-sets of those domains. The labeling of the dataset is done manually at the current stage of our study, with the help of the domain experts. In our future research, including those listed in this paper, we may also use certain recently introduced methods and frameworks for classifying data with a larger volume.

References

[1] Atkins, Adam, Mahesan Niranjan, and Enrico Gerding. "Financial news predicts stock market volatility better than close price," *The Journal of Finance and Data Science* 4, no. 2, pp. 120-137, 2018.

[2] Belainine, Billal, Alexsandro Fonseca, and Fatiha Sadat. "Named entity recognition and hashtag decomposition to improve the classification of tweets," In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pp. 102-111. 2016.

[3] da Costa Albuquerque, Fábio, Marco A. Casanova, Jose Antonio F. de Macedo, Marcelo Tilio M. de Carvalho, and Chiara Renso. "A proactive application to monitor truck fleets," In *2013 IEEE 14th International Conference on Mobile Data Management*, vol. 1, pp. 301-304. IEEE, 2013.

[4] D. McDonald, H. Chen, and R. Schumaker. "Transforming Open-Source Documents to Terror Networks: The Arizona TerrorNet," In *AAAI Spring Symposium: AI Technologies for Homeland Security*, pp. 62-69, 2005.

[5] C.P. Wei, and Y.H. Lee. "Event detection from online news documents for supporting environmental scanning," Decision Support Systems 36, pp. 385-401, 2004.

[6] M.H. Steinberg. "Clinical trials in sickle cell disease: adopting the combination chemotherapy paradigm," *American Journal of Hematology* 83, no. 1, pp. 1-3, 2008.

[7] S. Xiong, K. Wang, D. Ji, B. Wang. "A short text sentiment-topic model for product reviews," *Neurocomputing* 297, pp. 94-102, 2018.

[8] Abbasi, Ahmed, Stephen France, Zhu Zhang, and Hsinchun Chen. "Selecting attributes for sentiment classification using feature relation networks," *IEEE Transactions on Knowledge and Data Engineering* 23, no. 3, pp. 447-462, 2010.

[9] Aggarwal, Charu C. "Machine Learning for Text: An Introduction," In *Machine Learning for Text*, pp. 1-16. Springer, Cham, 2018.

[10] Ahmed, Sajid, Asif Mahbub, Farshid Rayhan, Rafsan Jani, Swakkhar Shatabda, and Dewan Md Farid. "Hybrid methods for class imbalance learning employing bagging with sampling techniques," In *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, pp. 1-5. IEEE, 2017.

[11] Alcalá-Fdez, Jesús, Luciano Sanchez, Salvador Garcia, Maria Jose del Jesus, Sebastian Ventura, Josep Maria Garrell, José Otero *et al.* "KEEL: a software tool to assess evolutionary algorithms for data mining problems," *Soft Computing* 13, no. 3, pp. 307-318, 2009.

[12] Armanfard, Narges, James P. Reilly, and Majid Komeili. "Local feature selection for data classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38, no. 6, pp. 1217-1227, 2015.

[13] Bahassine, Said, Abdellah Madani, and Mohamed Kissi. "An improved Chi-sqaure feature selection for Arabic text classification using decision tree," In *2016 11th International Conference on Intelligent Systems: Theories and Applications (SITA)*, pp. 1-5. IEEE, 2016.

[14] Cao, Peng, Dazhe Zhao, and Osmar Zaiane. "An optimized cost-sensitive SVM for imbalanced data learning," In *Pacific-Asia conference on knowledge discovery and data mining*, pp. 280-292. Springer, Berlin, Heidelberg, 2013.

[15] Chen, Jingnian, Houkuan Huang, Shengfeng Tian, and Youli Qu. "Feature selection for text classification with Naïve Bayes," *Expert Systems with Applications* 36, no. 3, pp. 5432-5435, 2009.

[16] S. Kumar, Ravishankar, and S. Verma. "Context Aware Dynamic Permission Model: A Retrospect of Privacy and Security in Android System," in 2018 International Conference on Intelligent Circuits and Systems , IEEE Xplore, Phagwara, India, pp. 324-329, 2018.

[17] T. Sabbah, A. Selamat, M.H. Selamat, F.S. Al-Anzi, E.H. Viedma, O. Krejcar, and H. Fujita. "Modified frequency-based term weighting schemes for text classification," Applied Soft Computing 58, pp. 193–206, 2017.

[18] B. Vijayalakshmi, K. Ramar, NZ Jhanjhi, S. Verma, M. Kaliappan, et.al. "An Attention Based Deep Learning Model For Traffic Flow Prediction Using Spatio Temporal Features Towards Sustainable Smart City," *International Journal of Communication Systems*, 34, pp. 1-14 ,2020.

[19] S. Schmidt, S. Schnitzer, and C. Rensing. "Text classification based filters for a domain-specific search engine," Computers in Industry 78, pp. 70–79, 2016.

[20] Y. Liu, H.T. Loh, and A. Sun. "Imbalanced text classification: A term weighting approach," Expert System Applications 36, pp. 690–701, 2013.

[21] Ghosh, Samujjwal, and Maunendra Sankar Desarkar. "Class specific TF-IDF boosting for short-text classification: Application to short-texts generated during disasters," In *Companion Proceedings of the The Web Conference 2018*, pp. 1629-1637. 2018.

[22] Dal Pozzolo, Andrea, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, and Gianluca Bontempi. "Credit card fraud detection: a realistic modeling

and a novel learning strategy," *IEEE Transactions on Neural Networks and Learning Systems* 29, no. 8, pp. 3784-3797, 2017.

[23] Das, Sanjiv Ranjan. "Text and context: Language analytics in finance," *Foundations and Trends® in Finance* 8, no. 3, pp. 145-261, 2014.

[24] I. Batra, S. Verma and Kavita, and M. Alazab. "A Lightweight IoT based Security Framework for Inventory Automation Using Wireless Sensor Network," *International Journal of Communication Systems* 33, pp.1-16, 2019.

[25] Elagamy, Mazen Nabil, Clare Stanier, and Bernadette Sharp. "Stock market random forest-text mining system mining critical indicators of stock market movements," In *2018 2nd International Conference on Natural Language and Speech Processing (ICNLSP)*, pp. 1-8. IEEE, 2018.

[26] García, Salvador, and Francisco Herrera. "Evolutionary undersampling for classification with imbalanced datasets: Proposals and taxonomy," *Evolutionary Computation* 17, no. 3, pp. 275-306, 2009.

[27] Ghanem, Amal S., Svetha Venkatesh, and Geoff West. "Multi-class pattern classification in imbalanced data," In *2010 20th International Conference on Pattern Recognition*, pp. 2881-2884. IEEE, 2010.

[28] Gomez, Juan Carlos, and Marie-Francine Moens. "PCA document reconstruction for email classification," *Computational Statistics & Data Analysis* 56, no. 3, pp. 741-751, 2012.

[29] Granitto, Pablo M., Cesare Furlanello, Franco Biasioli, and Flavia Gasperi. "Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products," *Chemometrics and Intelligent Laboratory Systems* 83, no. 2, pp. 83-90, 2006.

[30] He, Haibo, and Edwardo A. Garcia. "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering* 21, no. 9, pp. 1263-1284, 2009.

[31] Jeatrakul, Piyasak, and Kok Wai Wong. "Enhancing classification performance of multi-class imbalanced data using the OAA-DB algorithm," In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8. IEEE, 2012.

[32] Jin, Xin, Anbang Xu, Rongfang Bie, and Ping Guo. "Machine learning techniques and chi-square feature selection for cancer classification using SAGE gene expression profiles," In *International Workshop on Data Mining for Biomedical Applications*, pp. 106-115. Springer, Berlin, Heidelberg, 2006.

[33] H. Kaur, H.S. Pannu, and A.K. Malhi. "A systematic review on imbalanced data challenges in machine learning: Applications and solutions," *ACM Computing Surveys (CSUR)* 52, no. 4, pp. 1-36, 2019.

[34] L. Khreisat. "Arabic Text Classification Using N-Gram Frequency Statistics A Comparative Study," In *Conference on Data Mining* (*DMIN* 2006), pp. 78-82, 2006.

[35] S.B. Kotsiantis. "Decision trees: a recent overview," *Artificial Intelligence Review* 39, no. 4, pp. 261-283, 2013.

[36] B. Krawczyk. "Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence* 5, no. 4, pp. 221-232, 2016.

[37] I. Batra, S. Verma, Kavita, U. Ghosh, J. J. P. C. Rodrigues, *et al.* "Hybrid Logical Security Framework for Privacy Preservation in the Green Internet of Things," MDPI-Sustainability 12, no. 14, pp. 5542, 2020.

[38] J. Lee, I. Yu, J. Park, D.W. Kim. "Memetic feature selection for multilabel text categorization using label frequency difference," *Information Sciences* 485, pp. 263-280, 2019.

[39] G. Lemaître, F. Nogueira, and C.K. Aridas. "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," *The Journal of Machine Learning Research* 18, no. 1, pp. 559-563, 2017.

[40] Jing, Li-Ping, Hou-Kuan Huang, and Hong-Bo Shi. "Improved feature selection approach TFIDF in text mining," In *Proceedings. International Conference on Machine Learning and Cybernetics*, vol. 2, pp. 944-946. IEEE, 2002.

[41] G. Liang, C. Zhang. "A comparative study of sampling methods and algorithms for imbalanced time series classification," In *Australasian Joint Conference on Artificial Intelligence*, pp. 637-648. Springer, Berlin, Heidelberg, 2012.

[42] M. A. Jan, B. Dong, S. R. U. Jan, Z. Tazzn, S. Verma, *et al.* "A Comprehensive Survey on Machine Learning-based Big Data Analytics for IoT-enabled Smart Healthcare System," *Mobile Networks and Applications* 26, pp.234-252, Springer, 2021.

[43] P. Liu, X. Qiu, and H. Xuanjing. "Recurrent neural network for text classification with multi-task learning," In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pp.2873-2879, 2016.

[44] X. Liu, Q. Li, and Z. Zhou. "Learning imbalanced multi-class data with optimal dichotomy weights," In *2013 IEEE 13th International Conference on Data Mining*, pp. 478-487. IEEE, 2013.

[45] R.J. Lyon, J.M. Brooke, J.D. Knowles, and B.W. Stappers. "Hellinger distance trees for imbalanced streams," in *2014 22nd International Conference on Pattern Recognition*, pp. 1969-1974. IEEE, 2014.

[46] D. Fatta, Giuseppe, A. Fiannaca, R. Rizzo, A. Urso, M. R. Berthold, and S. Gaglio. "Context-Aware Visual Exploration of Molecular Datab," In *Sixth IEEE International Conference on Data Mining-Workshops (ICDMW'06)*, pp. 136-141. IEEE, 2006.

[47] A. Makazhanov, and D. Rafiei, "Predicting the political preference of Twitter users," *Social Network Analysis and Mining* - ASONAM '13, pp. 298–305, 2013.

[48] K. Mathew, and B. Issac. "Intelligent spam classification for mobile text message," In *Proceedings of 2011 International Conference on Computer Science and Network Technology*, vol. 1, pp. 101-105. IEEE, 2011 .

[49] C. Zhang, J. Bi, S. Xu, E. Ramentol, G. Fan, B. Qiao, and H. Fujita. "Multi-Imbalance: An open-source software for multi-class imbalance learning," *Knowledge Based System* 174, pp. 137–143, 2019.

[50] A. Mazyad, F. Teytaud, and C. Fonlupt. "A comparative study on term weighting schemes for text classification", in *Lecture Notes in Computer Science*, Springer Verlag, pp. 100–108, 2018.

[51] A. Moreo, A. Esuli, and F. Sebastiani. "Distributional random oversampling for imbalanced text classification," In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 805-808, 2016.

[52] A. Onan, S. Korukoğlu, and H. Bulut. "Ensemble of keyword extraction methods and classifiers in text classification," *Expert System Applications* 57, pp. 232–247, 2016.

[53] N.C. Oza, and S. J. Russell. "Online bagging and boosting," In *International Workshop on Artificial Intelligence and Statistics*, pp. 229-236., 2001.

[54] A. Özçift. "Random forests ensemble classifier trained with data resampling strategy to improve cardiac arrhythmia diagnosis," *Computers in Biology and Medicine* 41, no. 5, pp. 265-271, 2011.

[55] V.N. Phu, V.T.N. Tran, V.T.N. Chau, N.D. Dat, and K.L.D. Duy. "A decision tree using ID3 algorithm for English semantic analysis," *International Journal of Speech Technology* 20, no. 3, pp. 593-613, 2017.

[56] T. Pranckevičius, and V. Marcinkevičius. "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic Journal of Modern Computing* 5, no. 2, pp. 221, 2017.

[57] M. Raza, F.K. Hussain, O.K. Hussain, M. Zhao, and Z. ur Rehman. "A comparative analysis of machine learning models for quality pillar assessment of SaaS services by multi-class text classification of users' reviews," *Future Generation Computer Systems* 101, pp. 341–371, 2017.

[58] F. Khan, A. Shahnazir, N. Ayazsb, S. Khan, S. Verma, and Kavita. "A Resource Efficient hybrid Proxy Mobile IPv6 extension for Next Generation IoT Networks," *IEEE Internet of Things Journal*, 2021, 10.1109/JIOT.2021.3058982

[59] A. P. Singh, A. K. Luhach, S. Agnihotri, N. R. Sahu, D. S. Roy, NZ Jhanjhi, S. Verma, Kavita, and U. Ghosh. "A Novel Patient-Centric Architectural Framework for Blockchain-Enabled Healthcare Applications," IEEE-Transaction on Industrail Informatics 17, no. 8, pp. 5779 – 5789, 2020, 10.1109/TII.2020.3037889.

[60] R.E. Schapire, Y. Singer, and A. Singhal. "Boosting and Rocchio applied to text filtering," In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 215-223. 1998.

[61] R.P. Schumaker, and H. Chen. "Textual analysis of stock market prediction using breaking financial news: The AZFin text system," *ACM Transactions on Information Systems (TOIS)* 27, no. 2, pp. 1-19, 2009.

[62] R.A. Stein, P.A. Jaques, and J.F. Valiati. "An analysis of hierarchical text classification using word embeddings," *Information Sciences* 471, pp. 216–232, 2019.

[63] S. Tan. "Neighbor-weighted K-nearest neighbor for unbalanced text corpus," *Expert System Applications* 28, pp. 667–671, 2005.

[64] H. Tayyar Madabushi, E. Kochkina, and M. Castelle. "Cost-Sensitive

BERT for Generalisable Sentence Classification on Imbalanced Data," *arXiv preprint arXiv:2003.11563*, pp. 125–134, 2020.

[65] C.F. Tsai, W.C. Lin, Y.H. Hu, and G.T. Yao. "Under-sampling class imbalanced datasets by combining clustering analysis and instance selection," *Information Sciences* 477, pp. 47–54, 2019.

[66] A.K. Uysal, and S. Gunal. "A novel probabilistic feature selection method for text classification," *Knowledge-Based Systems* 36, pp. 226–235, 2012.

[67] B. Verma, and A. Rahman. "Cluster-oriented ensemble classifier: Impact of multicluster characterization on ensemble classifier learning," *IEEE Transactions on Knowledge and Data Engineering* 24, no. 4, pp. 605–618, 2012.

[68] M.K. Verma, D.K. Xaxa, and S. Verma. "DBCS: density based cluster sampling for solving imbalanced classification problem," In *2017 International conference of Electronics, Communication and Aerospace Technology (ICECA)*, vol. 1, pp. 156-161. IEEE, 2017.

[69] G. Yang, M. A. Jan, A. U. Rehman, M. Babar, and M. M. Aimal. "Interoperability and Data Storage in Internet of Multimedia Things: Investigating Current Trends, Research Challenges and Future Directions," *IEEE Access* 8, pp. 124382 – 124401, 2020.

[70] V. Dogra. "Banking news-events representation and classification with a novel hybrid model using DistilBERT and rule-based features," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)* 12, no. 10, pp. 3039-3054, 2021.

[71] J. Yan, B. Zhang, N. Liu, S. Yan, Q. Cheng, W. Fan, Q. Yang, W. Xi, and Z. Chen. "Effective and efficient dimensionality reduction for large-scale and streaming data preprocessing," *IEEE Transactions on Knowledge and Data Engineering* 18, no. 3, pp. 320–332, 2006.

[72] J. Yang, Y. Liu, X. Zhu, Z. Liu, and X. Zhang. "A new feature selection based on comprehensive measurement both in inter-category and intra-category for text categorization," *Information Processing Management* 48, pp. 741–754, 2012.

[73] K. Yang, Z. Yu, S. Member, X. Wen, W. Cao, S. Member, C.L.P. Chen, H. Wong, and J. You. "Hybrid Classifier Ensemble for Imbalanced Data," *IEEE Transactions on Neural Networks and Learning Systems* 31, no. 4, pp. 1–14, 2019.

[74] H. Zhang, and M. Li. "RWO-Sampling: A random walk over-sampling approach to imbalanced data classification," *Information Fusion* 20, pp. 99–116, 2014.

[75] A. S. Ashour, S. Beagum, N. Dey, A. S. Ashour, D. S. Pistolla, , G. N. Nguyen,... and F. Shi. "Light microscopy image de-noising using optimized LPA-ICI filter," *Neural Computing and Applications* 29, no. 12, pp. 1517-1533, 2018.

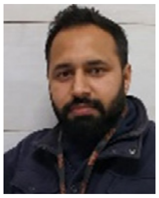
[76] S. Doss, J. Paranthaman, S. Gopalakrishnan, A. Duraisamy, S. Pal *et al.* "Memetic optimization with cryptographic encryption for secure medical data transmission in iot-based distributed systems," *Computers, Materials & Continua* 66, no.2, pp. 1577–1594, 2021.

[77] D. N. Le. "A new ant algorithm for optimal service selection with end-to-end QoS constraints," *Journal of Internet Technology 18*, no.5, pp. 1017-1030, 2017.

[78] Z. Zhang, B. Krawczyk, S. Garcìa, A. Rosales-Pérez, and F. Herrera, "Empowering one-vs-one decomposition with ensemble learning for multi-class imbalanced data," *Knowledge Based System* 106, pp. 251–263, 2016.

[79] Z. Sabir, K. Nisar, M. A. Z. Raja, M. R. Haque, M. Umar, A. A. A. Ibrahim, and D. N. Le. "IoT technology enabled heuristic model with Morlet wavelet neural network for numerical treatment of heterogeneous mosquito release ecosystem," *IEEE Access 9*, pp. 132897-132913, 2021.

[80] T. Zhu, Y. Lin, and Y. Liu. "Synthetic minority oversampling technique for multiclass imbalance problems," Pattern Recognition 72, pp. 327–340, 2017.



Varun Dogra

Varun Dogra has been pursuing a Ph.D. in Computer Applications at Lovely Professional University, Phagwara, Punjab, India. He has Bachelor in Science and Masters in Computer Applications. He has also been working as Assistant Professor in the School of Computer Science and Engineering, Lovely Professional University. He has to have 14 years of experience in teaching/ industry. He has published papers in reputed journals and presented papers in International conferences. He has also been reviewed research papers of Scopus/ WoS indexed journals. His area of research covers Artificial Intelligence, Natural Language Processing, Data Science, and Financial Markets.

Sahil Verma

Sahil Verma (Senior Member IEEE, ACM, IAENG) is Ph. D in Computer Science and Engineering. He is an Associate Professor and (A.) Directior in Chandigarh University, Mohali, India. He has published many research articles in reputed journals/publishers like IEEE, Wiley, Springer, ACM, Elsevier, MDPI etc. He has published papers in reputed top-cited journals like IEEE Transaction in Industrial Informatics, IEEE Transaction on Network Science and Engineering, IEEE Internet of Things Journals, ACM Transaction on Internet Technology, CMC, IEEE Access, MONET Elsevier, HCIS Springer, MTAP Springer, MDPI Sensors, Symmetry and many more. He is reviewer of top-cited journals like IEEE Transaction on Intelligent Transport Systems, IEEE Transactions on Network Science and Engineering, IEEE Access, Neural Computing and Applications Springer, Human-centric Computing and Information Sciences Springer, Mobile Networks and Applications Springer, Journal of Information Security and Applications Elsevier, Mobile Information Systems Hindawi, International Journal of Communication Systems Wiley, Security and Communication Networks Hindawi etc. Dr. Verma is also had professional membership of many reputed organisations like IEEE, ACM, IAENG. His tenure led to an overall Excellence in Education, Research, Infrastructure and Systemic Development of Organization. His current focus is to enhance the Quality of Education through Strategic Quality Initiatives. He has visited many countries like: Austria, Czech Republic, Germany, Switzerland, France, Italy and Thailand for exploring research and development, establishment of labs and for the collaboration with foreign universities (students exchange programs, faculty exchange programs etc.

Kavita Verma

Kavita Verma is Ph. D in Computer Science and Engineering. She is an Associate Professor at Chandigarh University, Mohali, India. She has published papers in reputed journals like IEEE Transaction in Industrial Informatics, IEEE Transaction on Network Science and Engineering, IEEE Internet of Things Journals, ACM Transaction on Internet Technology, CMC, IEEE Access, MONET Elsevier, HCIS Springer, MTAP Springer, MDPI Sensors, Symmetry and many more. She is also a reviewer of top-cited journals like IEEE Transaction on Intelligent Transport Systems, IEEE Transactions on Network Science and Engineering, IEEE Access, Neural Computing, and Applications Springer, Human-centric Computing and Information Sciences Springer, Mobile Networks and Applications Springer, Journal of Information Security and Applications Elsevier, Mobile Information Systems Hindawi, International Journal of Communication Systems Wiley, Security and Communication Networks Hindawi, etc. Dr. Kavita Verma has professional membership of many reputed organizations like SMIEEE, MACM, MIAENG, MISCA.

Noor Zaman Jhanjhi

Noor Zaman Jhanjhi (NZ Jhanjhi) is currently working as Associate Professor, Director Center for Smart society 5.0 [CSS5], and Cluster Head for Cybersecurity cluster, at School of Computer Science and Engineering, Faculty of Innovation and Technology, Taylor's University, Malaysia. He is supervising a great number of Postgraduate students, mainly in cybersecurity for Data Science. The cybersecurity research cluster has extensive research collaboration globally with several institutions and professionals. Dr Jhanjhi is Associate Editor and Editorial Assistant Board for several reputable journals, including IEEE Access Journal, PeerJ Computer Science, PC member for several IEEE conferences worldwide, and guest editor for the reputed indexed journals. Active reviewer for a series of top tier journals has been awarded globally as a top 1% reviewer by Publons (Web of Science). He has been awarded as outstanding Associate Editor by IEEE Access for the year 2020. He has high indexed publications in WoS/ISI/SCI/Scopus, and his collective research Impact factor is more than 350 points as of the first half of 2021. He has international Patents on his account, edited/authored more than 30 plus research books published by world-class publishers. He has great experience supervising and co-supervising postgraduate students. An ample number of PhD and Master students graduated under his supervision. He is an external PhD/Master thesis examiner/evaluator for several universities globally. He has completed more than 22 international funded research grants successfully. He has served as Keynote speaker for several international conferences, presented several Webinars worldwide, chaired international conference sessions. His research areas include Cybersecurity, IoT security, Wireless security, Data Science, Software Engineering, UAVs.

Uttam Ghosh

Uttam Ghosh is currently working as Associate Professor of Cybersecurity in Meharry School of Applied Computer Science, Nashville, TN, USA. He has been over 10 years of research and development experience in secure wireless and wired communications, Software defined networking, CPS Security. His area of research covers multiple domains like Cyber Physical system Security, Mobile Ad hoc Networks, Wireless Sensor Networks, Software-Defined Networking, Cloud Computing, Distributed Algorithms, and Internet of Things(IoT). He has published many research articles in reputed journals/publishers. He is also a reviewer of top-cited journals. He has a professional membership of reputed organizations like SMIEEE, Sigma Xi, ACM, IEEE, AAAS, ASEE.

Dac-Nhuong Le

Dac-Nhuong Le has an MSc and PhD in computer science from Vietnam National University, Vietnam in 2009, and 2015, respectively. He is an Associate Professor on Computer Science, Deputy Head of the Faculty of Information Technology, Haiphong University, Vietnam. He has a total academic teaching experience of 20+ years in computer science. He has more than 80+ publications in the reputed international conferences, journals, and book chapter contributions (Indexed by SCIE, SSCI, ESCI, Scopus). His areas of research are in the field of intelligence computing, multi-objective optimization, network security, cloud computing, virtual reality/argument reality. Recently, he has been on the technique program committee, the technique reviews, the track chair for international conferences under Springer-ASIC/LNAI/CISC Series. Presently, he is serving on the editorial board of international journals and edited/authored 20+ computer science books published by Springer, Wiley, CRC Press, Bentham Publishers.

An Ensemble Classifier for Stock Trend Prediction Using Sentence-Level Chinese News Sentiment and Technical Indicators

Chun-Hao Chen[1]*, Po-Yeh Chen[2], Jerry Chun-Wei Lin[3]*

[1] Department of Information and Finance Management, National Taipei University of Technology, Taipei (Taiwan)
[2] Department of Computer Science and Information Engineering, Tamkang University, Taipei (Taiwan)
[3] Department of Computer Science, Electrical Engineering, and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen (Norway)

Received 22 November 2021 | Accepted 28 January 2022 | Published 11 February 2022

unir
LA UNIVERSIDAD
EN INTERNET

Abstract

In the financial market, predicting stock trends based on stock market news is a challenging task, and researchers are devoted to developing forecasting models. From the existing literature, the performance of the forecasting model is better when news sentiment and technical analysis are considered than when only one of them is used. However, analyzing news sentiment for trend forecasting is a difficult task, especially for Chinese news, because it is unstructured data and extracting the most important features is difficult. Moreover, positive or negative news does not always affect stock prices in a certain way. Therefore, in this paper, we propose an approach to build an ensemble classifier using sentiment in Chinese news at sentence level and technical indicators to predict stock trends. In the training stages, we first divide each news item into a set of sentences. TextRank and word2vec are then used to generate a predefined number of key sentences. The sentiment scores of these key sentences are computed using the given financial lexicon. The sentiment values of the key phrases, the three values of the technical indicators and the stock trend label are merged as a training instance. Based on the sentiment values of the key sets, the corpora are divided into positive and negative news datasets. The two datasets formed are then used to build positive and negative stock trend prediction models using the support vector machine. To increase the reliability of the prediction model, a third classifier is created using the Bollinger Bands. These three classifiers are combined to form an ensemble classifier. In the testing phase, a voting mechanism is used with the trained ensemble classifier to make the final decision based on the trading signals generated by the three classifiers. Finally, experiments were conducted on five years of news and stock prices of one company to show the effectiveness of the proposed approach, and results show that the accuracy and P / L ratio of the proposed approach are 61% and 4.0821 are better than the existing approach.

Keywords

Chinese News Analysis, Sentiment Analysis, Stock Trend Prediction, Support Vector Machine, Technique Indicators.

DOI: 10.9781/ijimai.2022.02.004

I. Introduction

Financial data analysis is an attractive research area for researchers and many topics could be studied, e.g., portfolio management and optimization [1]–[3], trading strategy discovery and optimization [4], [5], news sentiment analysis [6], [7]. One of the challenging research topics is the prediction of stock trends [8]–[10]. For example, Chen et al. proposed a graph feature-based neural network model that uses constructed stock networks, the corresponding feature matrices, trading data, and technical indicators to predict stock trends [8]. Considering trading transaction data, Long et al. proposed a framework that merges trading and market information for stock prediction using an attention-based bidirectional network for long-term memory [10].

The literature shows that financial news sentiment has an impact on stock prices [11], [12]. Through experiments with historical news articles of Hong Kong stock market, Li et al. found that the prediction models with sentiment analysis are better than the bag-of-words model in validation and test data set [11]. Loeffler et al. analyzed the tone of voice in Moody's rating reports and pointed out that the short-term effects of tone of voice in reports with negative ratings affect investors' mood and attention differently [12]. To determine the impact of news sentiment on financial returns, Kelly et al. found that annual returns are better when news sentiment is considered in a trading strategy than the buy-and-hold strategy for the stock and oil markets [13].

For predicting stock trends, several approaches have been presented to solve this problem using different techniques, including neural networks [8], [9], [14], rough set [15], support vector machine (SVM) [10], genetic algorithms (GA) [16] and others [17]. Many algorithms that consider news sentiment have also been proposed for predicting stock trends [18]–[20]. For example, Hao et al. proposed a fuzzy twin

* Corresponding author.

E-mail addresses: chchen@ntut.edu.tw (C. H. Chen), jerrylin@ieee.org (J. C.W. Lin).

support vector machine based on the hidden topic model and the emotional information of news to predict stock trends [18]. Wen et al. presented an SVM model for predicting stock price movements that takes into account not only the content of news but also the information hidden in the relationships between news [19].

From the literature, we have seen that there are still two problems that need to be considered in order to build a more practical predictive model: (1) The first problem is that the release of positive news does not mean that the stock price will increase in the next trading days. On the contrary, negative news does not always mean that the stock price will fall. Therefore, the first problem to solve is how to use the sentiment from financial news to build multiple forecasting models for trend prediction. (2) To extract sentiment from the corpus, a common method is to search for keywords from the entire corpus and then calculate the sentiment value based on the generated keywords. However, searching keywords from key phrases might provide a better representative set of keywords than those from the whole corpus. So the second question is how to generate the key phrases and find a representative set of keywords for the classification attributes from them.

To address the aforementioned problems, we propose a three-stage algorithm to create an ensemble classifier that uses sentiment in Chinese news at the sentence level and technical indicators to predict stock trends. In the data preprocessing stage, each news item is used to find a set of keywords using TextRank, and divided into a set of sentences. Then, the keywords generated from the sentences are compared with the keywords from the corpus using word2vec to find the predefined number of key sentences. The sentiment values of the generated key phrases are calculated using the given financial lexicon. The sentiment values of the key phrases and the news title, as well as the two technical indicators, are used as classification attributes along with the trend label generated from the difference between the closing and opening prices on the news release day to form a training instance. Based on the sentiment values of the key records, the training instance is stored in the training datasets for positive and negative news. In the modeling phase, the two datasets are used to build models for positive and negative stock trend predictions using the Support Vector Machine (SVM). To increase the reliability of the generated trading singles, a classifier based on the Bollinger bands is also created. These three classifiers are combined to form an ensemble classifier. In the prediction phase, the constructed ensemble classifier is used to predict trading signals for trading. To demonstrate the merits of the proposed approach, experiments were conducted on real financial news datasets. The contributions of this work are listed as follows:

- This paper proposes a three-phase framework for predicting stock trends using an ensemble classifier that incorporates news sentiment analysis and technical indicators.
- Using sentence-level sentiment analysis to build models for predicting positive and negative stock trends outperforms using a keyword-based approach in terms of returns.
- We also find that the news release day is highly associated with stock prices on the next trading day.

The remaining sections of this paper are organized as follows: Section II reviews the relevant literature. Section III explains the proposed framework and algorithm. In Section IV, experimental evaluations are given and discussed. Finally, Section V describes the conclusions and future work.

II. Literature Review

In this section, we first introduce related approaches to predicting stock trends in Section A. Then, related approaches to corpus analysis are described in Section B.

A. Stock Trend Prediction

The prediction of stock trends for investment is a challenging and widely studied research topic, as several factors need to be considered simultaneously when building the predictive model. Currently, there are several approaches to solve this problem using different techniques, including neural networks [8], [9], [14], rough set [15], support vector machine [10], genetic algorithms [16], and others [17], [21]. For example, Chen et al. proposed a Graph Convolutional Features based Convolutional Neural Network (GC-CNN) model for predicting stock trends [8]. It used the correlations and characteristics of stocks to generate stock market information. Then, trading data and technical indicators were used to observe stock information. Then, the stock market information and stock information were converted into images, which served as a training dataset for generating GC-CNN. Zhang et al. proposed an approach, called the status box method, to predict the change direction of stock prices [10]. It packs a period of stock points into three kinds of boxes to represent the stock status. Then, the specific feature extraction approach was developed to derive characteristic attributes. Finally, AdaBoost algorithm, Support Vector Machine and genetic algorithm were integrated to construct the classifiers to predict the stock price performance. Using keywords from financial news and technical indicators as classification attributes, Chen et al. proposed a genetic-based model to predict the stock trend using Support Vector Machine and genetic algorithm [16]. In this approach, keywords were first extracted from financial news. Then, more meaningful keywords were generated by combining 2 words. Feature selection was used to keep important keywords. The classifier was created based on the generated dataset using the support vector machine. To reduce the risk of trading, the technical indicators were used to increase the reliability of the trading signal proposed by the classifier. Since determining the hyperparameters of the predictive model and the technical indicators used is an optimization task, the genetic algorithm was used to find the appropriate hyperparameters. To deal with the profit bias in the model, Liu et al. proposed an effective metric called mean profit rate (MPR), which can be used to measure the return of the model and the correlation between the metric values to evaluate the stock trend prediction classifier [17]. Chen et al. proposed an approach for finding trading signals using long short-term memory neural network (LSTM) and genetic algorithms (GA) [22]. LSTM was used to learn fluctuations of stock prices and GA was then employed to find trading signals. Besides, the Kelly criterion was also employed to calculate optimal investment score for minimizing losses and maximizing returns. Take leading indicators into consideration, Wu et al. proposed the long short-term memory with leading indicators (LSTMLI) for prediction and showed that the prediction error can be reduced [23], and the LSTM-GA approach was also proposed based on LSTMLI [24]. In addition, based on the noisy equity state representation, Huang et al. proposed an algorithm for stock prediction using the recurrent neural network [25].

Since the literature shows that news sentiment has an impact on investors and investment goals [11], [12], many algorithms have also been proposed for predicting stock trends [18], [19]. For example, Long et al. proposed the semantic and structural (S&S) kernel for Support Vector Machine to predict the stock trend, considering not only the content of the news but also the information structures between the news [19]. The text point graph and keyword graph were first created from the given financial news. The S&S kernel was then defined over the two graphs and used to build the predictive model. To deal with and reduce the influence of outliers, Hao et al. proposed a fuzzy twin support vector machine classifier that used the emotional information and hidden topic model from the financial news to predict stock trends [18]. First, the high-level expression features were captured from the given financial news, including keywords, topic and sentiment distribution vectors. Then, the labels for the training instances were generated from the stock price data. Finally, the fuzzy twin support vector machine classifier was created using the prepared training instances.

Since investor sentiment can also have an impact on the stock market, many algorithms have been proposed to predict market trends [26], [27]. Using user-generated online content on the stock news board, Li et al. compared different classifiers that considered investor sentiment in predicting stock prices and found that the long-term memory model had higher predictive power than others [27]. In addition, Derakhshan et al. proposed a graphical part-of-speech model to extract user opinions based on social network datasets to predict stock price movements [26].

B. Corpus Analysis

This section describes the relevant approaches to corpus analysis. First, the approaches for different types of applications are discussed, e.g., short text sentiment [28], [29], Chinese corpus analysis [30]–[32]. Based on the probabilistic linguistic term sets (PLTSs) and relevance theory, Song et al. proposed a framework for analyzing sentiment in short texts [28]. They designed a text representation model, called Word2PLTS, which uses PLTSs and relevance theory. Using Word2PLTS, each word was converted into a vector to show the possibilities for sentiment polarity. Finally, the sentiment and polarity classification system was developed using support vector machines for sentiment analysis of short texts.

For sentiment analysis in Chinese microblogs, Wu et al. first created several sentiment dictionaries, including original sentiment, emoji, and other dictionaries, to increase the coverage of sentiment words in Chinese microblogs. To improve the accuracy of sentiment analysis of Chinese microblogs, the semantic rule sets were further derived to represent the possible sentence patterns and information between sentences [29]. Wang et al. proposed a mixed character-word architecture using the semantic information of the intra-word characters to solve the problem of compositional Chinese sentence representation [32]. Two main strategies were employed to achieve the goal. The first uses the mask gate on characters to observe the relationship between them in a word. The second one applied the max-pooling operation on the words to find an optimal mixture of the compositional Chinese sentences. The results showed that the architecture was better than the character-based and word-based models. Wang et al. proposed an algorithm to observe the opinion orientation of Chinese news based on the word embedding and syntax rules [31]. To determine the orientation value of words, the cosine similarity between words and sentiment dictionaries was calculated using word2vec. The generated word vectors were then used to extract key phrases. The syntax rules and word vector similarity were used to analyze the alignment of the document based on the key phrases. Wang et al. proposed a Chinese parsing approach that uses the semantic string-matching sliding agreement (SMOSS) [30]. Their approach is divided into a training phase and a parsing phase. In the training phase, the tree node keywords were encoded in a treebank according to the semantic codes in the synonym dictionary. Then, the semantic templates were extracted from these tree nodes using sliding windows to build the semantic template library. In the parsing phase, the built semantic template library was used to extract the semantic code strings for better syntax parsing performance.

In recent years, many deep learning based approaches have also been proposed for analyzing sentiment in the corpus, including neural networks [33], [34], bidirectional long-term memory (BiLSTM) [35], recurrent neural networks (RNN) [36], the bidirectional encoder representations of transformers (BERT) [37], and reinforcement learning [38]. To extract sentiment polarities from a small training corpus incorporating external knowledge, Chen et al. proposed a framework for sentiment analysis of Chinese reviews using a neural network [33]. In their approach, context features were extracted from review sentences. Then, the aspect-opinion pairs and their sentiment polarities were retrieved from the given sentiment knowledge graph as external knowledge. By using the generated training dataset, the created model was able to provide better results in analyzing sentiment from the limited review dataset. In order to consider the semantic and sentimental information of words to capture the contextual information, Xu et al. proposed an improved word representation model that integrates the sentimental information into the TF-ITF method to generate the word vectors [35]. Using the word representation model, a classifier was constructed using bidirectional long-term memory (BiLSTM) based on the generated working vectors to obtain effective sentiment analysis results. In order to provide producers with a useful tool to learn about consumers' needs and related aspects of online platforms, Yang et al. first defined a task called multi-entity aspect-based sentiment analysis (ME-ABSA) and proposed a method considering context, entity, and aspect memory to construct the classifier using a neural network [36]. In their approach, the network consisted of three layers, namely the interaction layer, the positional attention layer, and the RNN with attention layer, to form a context memory. The created model was then used to predict the sentiment polarity. To observe the implicit sentiments and the contagion process, Daudert proposed a sentiment analysis approach using a customized feed-forward neural network [34]. Based on different types of data structures, text preprocessing was performed. Then, the relationships between the records in the corpus were represented using a graph structure. In the graph, the features and constraints were used to form the vertices and edges between the vertices. The transformer-based model was used to extract more textual information from the corpus. The derived graph and text information were then used to construct the adapted feed-forward neural network. To analyze the sentiment of tweets, Pota et al. proposed an approach using the bidirectional encoder representations of transformers (BERT) [37]]. In their approach, the data were first preprocessed to remove noise data. In addition, they attempted to utilize information hidden in the tweets, including emojis, hashtags, etc. The processed instances were then used to build the model using BERT. Ma et al. proposed a phonetically enriched text representation using reinforcement learning for Chinese sentiment analysis [38]. The main concept of their approach was to consider the two features, deep phonemic orthography and intonation variations, and fuse them with textual information to obtain an effective Chinese sentiment analysis. As a result, an algorithm called Disambiguate intonation for sentiment analysis was proposed using reinforcement learning. They also pointed out that the performance of their approach was better than the character-level approaches.

III. Proposed Ensemble Classifier for Stock Prediction Algorithm

In this section, we describe the proposed approach to build an ensemble classifier for predicting stock trends based on Chinese news sentiment and sentence-level technical indicators. In the following, Section A describes the three-phase framework of the proposed approach. The three phases of the proposed approach, including the preprocessing of data, the construction of the ensemble classifier, and the prediction phase, are described in Sections B, C, and D.

A. Flowchart of Proposed Approach

The three-stage framework of the proposed approach for building the ensemble classifier for predicting stock trends, incorporating sentence-level sentiment analysis of Chinese news and technical indicators, is shown in in Fig. 1.

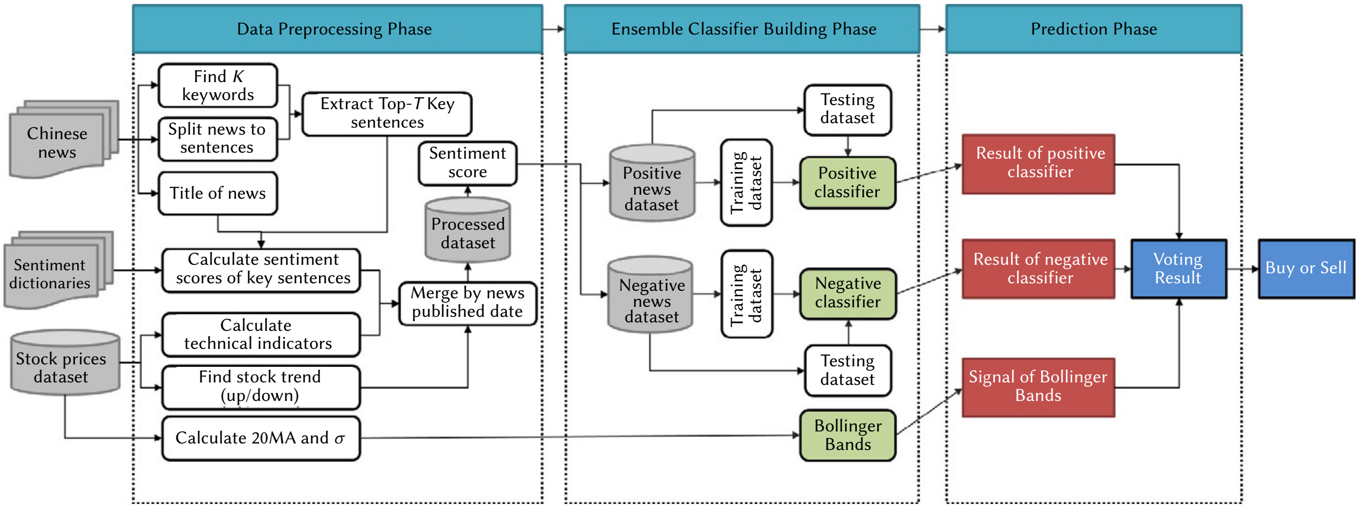


Fig. 1. Three-phase framework of the proposed approach.

Algorithm 1: Data preprocessing

```
Input: Chinese News $N = \{n1, ..., n_x\}$, stock prices dataset $S = \{s_1, ..., s_y\}$, sentiment dictionaries $Dict^A = \{dict^P, dict^N\}$, number of keywords $K$, number of top key sentences $T$, and stock trend period $P$.
Output: Positive training dataset $posTrainDataset$, and negative training dataset $negTrainDataset$.
$posTrainDataset \leftarrow null$;
$negTrainDataset \leftarrow null$;
for data $\in N$ do
    $title, content, publishedDate \leftarrow getRelatedInformation(data)$;
    $newsKeywords \leftarrow generateNewsKeywords(content, K)$;
    $sentenceSet \longleftarrow splitNewsToSentences(content)$;
    $wordSentenceSet \longleftarrow splitSentenceToWords(sentenceSet)$;
    $WIS \longleftarrow calculateWordImportanceScore(wordSentenceSet, newsKeywords)$;
    $SIS \longleftarrow calculateSentenceScore(WIS)$;
    $keySentenceSet \longleftarrow keySentencExtraction(SIS, T) \cup title$;
    $KSW \longleftarrow splitKeySentenceToWords(keySentenceSet)$;
    $WOS \longleftarrow calculateWordOrientationScore(DictA, KSW)$;
    $SOS, numPosKS, numNegKS \longleftarrow calculateSentenceOrientationScore(WOS)$;
    $KValue, DValue, RSI \longleftarrow calculateTechnicalIndicators(S, publishedDate)$;
    $trendLabel \longleftarrow generateStokTrend(S, P, publishedDate)$;
    $generatedInstance \longleftarrow mergeInformationByDate(SOS, KValue, DValue, RSI, trendLabel, publishedDate)$;
    if $numPosKS \geq numNegKS$ then
        $posTrainDataset \longleftarrow posTrainDataset \cup generatedInstance$;
    else
        $negTrainDataset \longleftarrow negTrainDataset \cup generatedInstance$;
return $posTrainDataset, negTrainDataset$.
```

Fig. 1 shows that the proposed approach consists of three phases, namely data preprocessing, ensemble classifier creation, and prediction phase. In the data preprocessing phases, each Chinese news is first used to extract the K keywords and split them into a set of sentences. The similarity value of each sentence with the K keywords is calculated using word2vector. Based on the similarity values, the top T key phrases are selected. Using the sentiment dictionaries, the sentiment scores of the key phrases and the news title are calculated as classification attributes. In addition, the stock price dataset is used to calculate the technical indicators, including the K, D, and RSI values, as classification attributes, and the stock trend of a certain period as the class label. The generated $T + 1$ sentiment values, the K, D, and RSI values, and the stock trend designation are combined into a training instance based on the news release date and stored in the processed dataset. Based on the sentiment values of the news, they are split into two data sets. If the number of records with positive sentiment is greater than the number of records with negative sentiment, the training instance is added to the positive news dataset. In contrast, it is assigned to the negative news dataset. In the phase where the ensemble classifier is created, the positive and negative classifiers are trained using the generated training datasets for positive and negative news. In addition, the 20-day moving average and stock series variance are used to construct the Bollinger bands as the third classifier. Finally, the three classifiers are used as an ensemble classifier to predict the stock trend by voting in the prediction phase. The following sections describe the details of the three phases.

B. Data Preprocessing

In the data preprocessing phase, which is the key phase of the proposed approach, the main objective is to generate positive and negative training instances through text analysis, technical indicators and sentiment orientation analysis based on the given Chinese news, stock prices and sentiment dictionaries to create the classifiers in the second phase. Following the earlier approach [31], the pseudocode for data preprocessing is shown in Algorithm 1.

TABLE I. A Processed Instance for the Proposed Approach

SOS_T	SOS_1	SOS_2	SOS_3	SOS_4	SOS_5	SOS_6	SOS_7	K	D	RSI	*Trend*
2.69	3.15	1.74	2.96	1.94	2.91	3.11	3.09	60.27	51.88	85.71	1

Algorithm 1 shows that the preprocessing phase of the data consists of two parts, namely the extraction of key sentences (lines 4 to 10) and the analysis of sentiment in the news (lines 11 to 20). For the extraction of key phrases, the related information is generated from each news item, including the title, content, and publication date (line 4). The news *newskeywords* are then generated according to the specified parameter *K* and the content of the news (line 5). The content of each news item is split into sentences to create *sentenceSet* based on the predefined punctuation (line 6). Each sentence in *sentenceSet* is then converted into a set of words by jieba to obtain *wordSentenceSet* (line 7). The generated *newskeywords* and *wordsentenceSet* are used to calculate the word importance score WIS for each sentence in the *sentenceSet* (line 8). The sentence importance score SIS is calculated for each sentence using *WIS* (line 9). Key sentences are then selected based on SIS and the number of top key sentences *T* to obtain *keySentenceSet* (line 10).

For news sentiment analysis, each key sentence is divided into a set of keywords and stored in *KWS* (line 11) to calculate the sentiment value of each sentence. The prepared dictionaries and the *KWS* are used to compute the word orientation score *WOS* (line 12). The sentence orientation score *SOS* is then computed using the generated *WOS* (line 13), and the number of positive and negative key phrases *numPosKS* and *numNegKS* for each news is also recorded. To create a more reliable classifier, the technical indicators, including *KD* and *RSI*, are used as additional classification attributes and are calculated using the stock price series *S* and the published date of the news *publishedDate* (line 14). For each news, the trend label is generated based on the publication date of the stock price series S and the predefined stock price period *P* (line 15). Then the generated *SOS*, *KValue*, *DValue*, *RSI*, and *trendLabel* are merged according to the publication date *publishedDate* as a trading instance (line 16). If the number of positive key sets is greater than the number of negative key sets, the generated training instance is stored in *posTrain* record. Otherwise, it is stored in *negTrainDataset* (lines 17 to 20). Finally, the generated *posTrainDatasets* and *negTrainDatasets* are returned for the second phase (line 21). The details of the data preprocessing phase are described below.

1. Key Sentence Extraction

For the given Chinese news set $N = \{n_1, ..., n_x\}$, the key phrase extraction for a news can be divided into the following steps. First, the title, content, and publication date are extracted from each news n_i and represented as title, content, and *publishedDate*. Then, *TextRank* is used to extract *K* keywords from the content of the news item, denoted as $newsKeywords = \{kw_1, kw_2, ..., kw_k\}$. To find important sentences, the news content is split using punctuation marks such as {“,”, “.”, “;”} into sentences to create a set of sentences $sentenceSet = \{sent_1, sent_2, ..., sent_n\}$. Each sentence $sent_i$ in *sentenceSet* is used by `Jieba` to generate a set of words, noted as $sw_i = = \{w_1, w_2, ..., w_m\}$. In the same way, the word sets for all sentences are formed as $wordSentenceSet = = \{sw_1, sw_2, ..., sw_n\}$. For the $sent_i$ sentence, the similarity between it and the news content is calculated by the similarity of each word in sw_j and the *newsKeywords* using word2vec and denoted as *wordSentenceScore* $WIS_j = \{score_1^j, score_2^j, ..., score_m^j\}$, where $score_m^j$ is the score of the *f*-th word in the $sent_j$ sentence to the keywords extracted from the news content and is calculated by the following equation 1:

$$score_f^j = \max\{sim(w_f, kw_1), sim(w_f, kw_2), \ldots, sim(w_f, kw_k)\} \tag{1}$$

Then, the score of the sentence $sentScore_j$ is calculated by average of the scores in the WIS_j using the equation 2:

$$sentScore_j = \frac{\sum_{f=1}^{m} score_f^j}{m} \tag{2}$$

In equation 2, the score of all sentences can be calculated and represented as $SIS = \{sentScore_1, sentScore_2, ..., sentScore_n\}$. The top-*T* sentences based on the scores of sentences and the news title are selected as the key sentences to form the set $keySentenceSet = \{senTitle, kSent_1, ..., kSent_T\}$ for news sentiment analysis.

2. News Sentiment Analysis

In this section, the details of news sentiment analysis are stated. For every news, the sentiment analysis can be processed by the following four steps, including the words extraction for key sentences, the word orientation score calculation for key sentences, the sentence orientation score calculation for key sentences, and the news orientation generation.

In the first step, each sentence $kSent_j$ in the *keySentenceSet* is split into a set of words and expressed as $kSent_j = \{ksw_{j1}, ksw_{j2}, ..., ksw_{jl}\}$, where ksw_{ji} represents the *i*-th word of the key sentence, and *L* is the number of words. In the second step, using the given dictionaries $Dict^A$, which consist of the positive dictionary $Dict^P$ and the negative dictionary $Dict^N$, the word orientation score wos_{ji} of ksw_{ji} is calculated according to the following equation 3 as:

$$wos_{ji} = \tau \cdot MAX(SIM(ksw_{ji}, w^O)) \tag{3}$$

where $w^0 \in Dict^A$, and τ is set as 1 if $w^O \notin Dict^P$; otherwise, it is set as -1 if $w^O \in Dict^N$.

Based on the equation 3, this means that the wos_{ji} is set to the maximum similarity with the words in the dictionaries, and the word2vec is used to measure the similarity of two words in *SIM*( ). If w^O is included in $Dict^P$, then wos_{ji} is a negative value, otherwise it is a positive value. After calculating all the key phrases, this can be represented as $WOS = \{kSentWOS_1, ..., kSentWOS_j, ..., kSentWOS_{T+1}\}$, where $kSentWOS_j = \{wos_{j1}, ..., wos_{ji}, ..., wos_{jL}\}$. Meanwhile, the number of positive and negative sentences, *numPosKS* and *numNegKS*, are also counted. In the third step, the sentence orientation value sos_j of a key sentence $kSent_j$ can be calculated using the following equation 4:

$$sos_j = \frac{\sum_{i=1}^{L} w\,os_{ji}}{L} \tag{4}$$

where L is the number of words of the key sentence $kSent_j$. Finally, the sentence orientation scores of all key sentences are calculated and denoted as $SOS = \{sos_1, ..., sos_j, ..., sos_{T+1}\}$. Then, according to the stock price series *S* and the published date of news *publishedDate*, the *K*, *D*, *RSI* values, and the stock trend label are generated. After that, the *SOS*, *KValue*, *DValue*, *RSI*, and trendLabel are merged according to the news published date as a trading instance. For example in Table I, the first eight columns are the sentiment scores of the key sentences and the last one is the class label.

In the news orientation generation, for every news, if the number of positive sentences is larger than the number of negative sentences, it is identified as a positive training instance. Otherwise, it is a negative training instance. In other words, according to the counted number of positive and negative sentences, the positive news training dataset *posNTrainDataset* and the negative news training dataset *negNTrainDataset* can be formed for next phase.

```
Algorithm 2: Ensemble classifier building, ClassifierBuildingProcedure()
    Input: Stock prices dataset $S = \{s_1, ..., s_y\}$, the positive news training dataset posNTrainDataset, the negative news training dataset negNTrainDataset, number of keywords $K$, number of top key sentences $T$, testind dataset proportion testProportion, and moving average days maDay.
    Output: Stock trend ensemble classifier sTrendEnsembleclassifier.
1   posXTrain, posXTest, posYTrainLabel, posYTestLabel ← trainTestSplit(posNTrainDataset, testProportion);
2   negXTrain, negXTest, negYTrainLabel, negYTestLabel ← trainTestSplit(negNTrainDataset, testProportion);
3   accuracy ← [ ];
4   posNewClassifier ← buildClassifier (posXTrain, posYTrainLabel);
5   predictPosY ← newClassifierPrediction (posNewClassifier, posXTest);
6   accuracy ⟵ classifierAccuracyScore (predictPosY, posYTestLabel);
7   negNewClassifier ⟵ buildClassifier (negXTrain, negYTrainLabel);
8   predictNegY ⟵ newClassifierPrediction (negNewClassifier, negXTest);
9   accuracy ⟵ classifierAccuracyScore (predictNegY, posYTestLabel);
10  bbandClassifier ⟵ generateBbandClassifier (S, maDay);
11  sTrendEnsembleclassifier ⟵ posNewClassifier ∪ negNewClassifier ∪ bbandClassifier;
12  return sTrendEnsembleclassifier.
```

```
Algorithm 3: Prediction, PredictionProcedure()
    Input: Stock prices dataset $S = \{s_1, ..., s_y\}$, Chinese News $S = \{n_1, ..., n_x\}$, sentiment dictionaries $Dict^A = \{dict^P, dict^N\}$, stock trend ensemble classifier sTrendEnsembleclassifier, number of keywords $K$, number of top key sentences $T$, and tradingTreshold $\alpha$.
    Output: Prediction $P = \{p_1, ..., p_x\}, p \in \{0, 1\}$.
1   preductionResult ← [ ];
2   processedTestingDataset ← dataPreprocessing(S, N, $Dict^A$);
3 for data ∈ processedTestingDataset do
4       tradingSingal1 ← sTrendEnsembleclassifier.posClassifier(data);
5       tradingSingal2 ← sTrendEnsembleclassifier.negClassifier(data);
6       tradingSingal3 ← sTrendEnsembleclassifier.bbClassifier(S);
7       if $tradingSingal1 + tradingSingal2 + tradingSingal3 \geq \alpha$ then
8               preductionResult.append(1); // Represent a buying signal
9       else
10              preductionResult.append(0); // Represent do nothing
21 return preductionResult.
```

C. Building Ensemble Classifier

In the proposed method, three classifiers are used to construct the ensemble classifier. The pseudo code of the second phase is shown in Algorithm 2.

From Algorithm 2, the ensemble classifier building procedure divides the preprocessed datasets, *posNTrainDataset negNTrainDataset*, into training and test datasets in the first phase according to the predefined test dataset proportion *testProportion* (lines 1 to 2). After the training and test datasets are created, the positive and negative classifiers are constructed using the Support Vector Machine (SVM) and their accuracies are stored in the accuracy list (lines 3 to 9). Then, the stock series S and the moving average days *maDay* are used to compute the Bollinger upper channel as the third classifier (line 10). Finally, the three classifiers are merged and the stock trend ensemble classifier *sTrendEnsembleclassifier* is returned (lines 11 to 12). In other words, the ensemble classifier is designed to take not only news contents and sentiments but also stock prices into consideration for generating more reliable trading signals for investors. For example, when a positive news is published, the prediction results of the first two classifiers based on the content of the news, *K*, *D* and *RSI* values can be used together to identify the trend of the target, which is better than the prediction by only one classifier. Hence, the first problem of the news-based forecasting model mentioned in the previous section can be solved. In addition, to make the trading signal more reliable, the technical indicator Bollinger bands is selected as the third classifier. If the stock price is greater than the upper Bollinger channel, a positive trading signal is generated. Otherwise, it generates a neutral trading signal.

D. Prediction Phase

After the stock trend ensemble classifier is built, it can be used for prediction. The pseudocode for the prediction phase is shown in Algorithm 3.

From Algorithm 3, in the prediction phase, when a news is released, using the same data preprocessing method, the key sentiment scores, *KValue*, *DValue*, and *RSI* are generated (line 2), and the processed data are used as input to the positive and negative classifiers to generate two trading signals, *tSignal1* and *tSignal2* (lines 4 to 5). Then, the stock price of the news release date is sent to the third classifier to generate the other trading signal, *tSignal3* (line 6). The final prediction is based on the voting of the three trading signals. If the voting score is greater than the trading threshold, a buy signal is generated. Because three classifiers are used in the ensemble classifier, when two out of three classifiers are suggesting buying, then the buying signal is outputted as the prediction result. On the contrary, when two of them suggesting selling, then selling signal is provided as the prediction result. Based on the signals, the return of every trading can be calculated.

IV. Experimental Evaluation

In this section, we first describe the experimental data and settings in Section A. Then, in Sections B through D, the results of the proposed approach are examined with respect to different training intervals, different parameter settings, and in comparison to the existing approach.

A. Experimental Data and Settings

As for the financial corpus, we used real data for experiments to test the effectiveness of the proposed approach. The news information was obtained from one of the most popular financial websites covering the stock market in Taiwan. We selected relevant news about Catcher Technology Co, Ltd. and obtained 5,191 news articles from May 11, 2014 to January 11, 2020. However, the content of some news is not what we want. For example, the content of the news articles consists only of company-related data tables without any other text descriptions, or the length of the article is too short to extract a reasonable number of key sentences. So after we receive the news, we perform a screening action. The conditions for screening are as follows: (1) The length of the article must be more than 200 words. (2) The content must contain more than numerical data. After screening, there were 2,331 news articles left.

As for the stock price series, according to the time interval of the news, the time interval of our stock acquisition extends from May 1, 2014 to January 30, 2020, and the data was collected from the Taiwan Stock Exchange. The closing price of the stock used in this study is shown in Fig. 2.

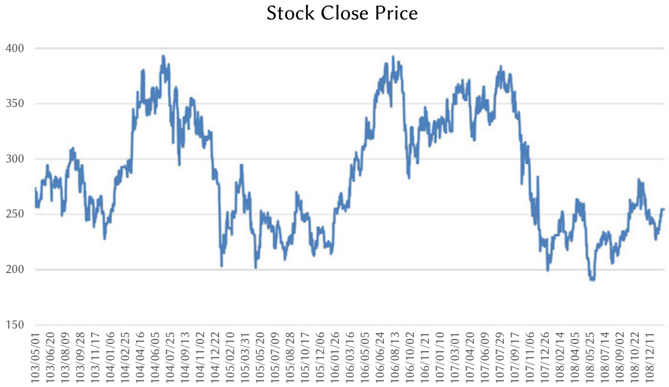


Fig. 2. The closing price of the stock used in the experiments.

As for the sentiment dictionary, according to literature [39], the effect of using finance-related sentiment dictionaries in finance is better than that of general sentiment dictionaries. The sentiment dictionary used in this paper is a financial sentiment dictionary established by Loughran et al. [40], which includes a positive word dictionary and a negative word dictionary. Since the original text is in English, it needs to be translated and partially selected before use. The final positive dictionary and negative dictionary contain 352 and 2,332 words, respectively.

In the following experiments, we use standard evaluation indicators, including accuracy, recall, precision, and F-score. Accuracy can reflect the overall judgment of the model and can be calculated as follows by equation 5:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

where *true positive* (*TP*) refers to the number of data whose predicted trend is positive and the trend of those data is actually positive, *true negative* (*TN*) refers to the data whose trend is predicted to be negative and the trend of those data is actually negative. *false positive* (*FP*) refers to the data whose predicted trend is positive but the actual trend is negative. *false negative* (*FN*) represents the data whose predicted trend is negative but the actual trend is positive. The precision, recall and F-score are respectively defined in equation 6, equation 7, and equation 8:

$$precision = \frac{TP}{TP + FP} \tag{6}$$

$$recall = \frac{TP}{TP + FN} \tag{7}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{8}$$

where the *precision* is defined as the proportion of data that is recognized as a positive category. The recall rate is in all samples of positive categories, what is the proportion of correctly identified as positive categories. There is a close relationship between precision and recall. When the precision is low, the recall rate is usually higher; but when the precision is high, the recall rate is usually lower. Therefore, in practical applications, F-score is usually used to find a balance between the two. In addition, the profit-to-loss ratio (P/L ratio), which is calculated by the ratio of average profit to average loss, is defined in equation 9:

$$profit\_loss = \frac{avg\_profit}{avg\_loss} \tag{9}$$

B. Evaluation on Different Training Intervals

Experiments with different lengths of training data to test whether the length of training time has an effect on prediction. Five different training periods are used in the experiments, including a 5-year training period (2014-2019), a 4-year training period (2015-2019), a 3-year training period (2016-2019), a 2-year training period (2017-2019), and a 1-year training period (2019). The test period is one year (2020). Table II shows the number of training and testing instances.

Moreover, the experiments are performed in a fixed way in terms of parameters. The number of keywords used to extract the key sentences is set to 10, and the number of key sentences T is set to 7 for testing. The prediction time is the same day, i.e., M is set to 0. In other words, the opening price minus the closing price is used to set the label. If the number is positive, it means an uptrend and is labeled "1", otherwise, it means a downward trend and is labeled as "0". The comparison results of the proposed approach with different number of classifiers for different training intervals are shown in Table III.

From Table III, we can make two observations: (1) Overall, the accuracy of the proposed approach using three classifiers is better than that using only one positive or negative classifier, especially when the 3-year training period is used to train the model. (2) From the profit-to-loss ratios, we can also see that the P/L ration of the proposed approach with a 5-year training period is 4.082, which is the highest compared to the others. Based on the results, we can say that the ensemble classifier derived with the proposed approach is effective. Note that the accuracies of the ensemble classifier, the negative model, and the positive model using 5-year dataset for training and 1-year dataset for testing are 0.6, 0.59, and 0.61, the results also indicate that there is no overfitting issue. Of course, when the number of training instances is too small, the overfitting problem may be occurred.

TABLE II. Number of Training and Testing Instances in Different Intervals

Training periods					Testing periods	Total
5 years	**4 years**	**3 years**	**2 years**	**1 year**	**1 year**	**2 years**
2014-2019	2015-2019	2016-2019	2017-2019	2019	2020	2014-2020
2,124	1,850	1,603	1,221	581	207	2,331

TABLE III. Comparison Results of the Proposed Approach in Terms of Various Criteria Using Different Training Intervals

5 years	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.64	0.58	0.45	0.51	0.58	0.62	0.6	
0	0.69	0.45	0.54	0.59	0.71	0.65	0.65	0.6	0.62	
Accuracy	0.6			0.59			0.61			4.0821
4 years	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.63	0.57	0.3	0.39	0.55	0.53	0.54	
0	0.68	0.43	0.53	0.56	0.8	0.66	0.61	0.63	0.62	
Accuracy	0.59			0.56			0.58			3.4097
3 years	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.51	0.8	0.62	0.57	0.3	0.39	0.57	0.55	0.56	
0	0.67	0.34	0.45	0.56	0.8	0.66	0.62	0.64	0.63	
Accuracy	0.55			0.56			0.6			3.6884
2 years	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.51	0.79	0.62	0.56	0.38	0.45	0.54	0.56	0.55	
0	0.66	0.35	0.46	0.57	0.73	0.64	0.6	0.58	0.59	
Accuracy	0.55			0.56			0.57			2.9368
1 year	Positive Model (PM)			Negative Model (NM)			PM+NM+BBAND			P/L Ratio
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.48	0.86	0.62	0.55	0.4	0.46	0.53	0.53	0.53	
0	0.64	0.22	0.32	0.57	0.71	0.63	0.59	0.59	0.59	
Accuracy	0.51			0.56			0.56			2.5783

TABLE IV. Comparison Results of the Proposed Approach in Terms of Various Criteria Using Different Number of Keywords

		Training data					Testing data
		5 years	4 years	3 years	2 years	1 year	1 year
	Total	2125	1851	1604	1222	582	206
K = 10	Positive	1345	1164	1020	811	413	135
	Negative	780	687	584	411	169	71
K = 8	Positive	1318	1148	1004	786	403	137
	Negative	807	703	600	436	179	69
K = 6	Positive	1343	1163	1015	803	403	137
	Negative	782	688	589	419	179	69
K = 4	Positive	1343	1163	1015	803	403	137
	Negative	782	688	589	419	179	69

C. Evaluation on Different Parameter Settings

In this section, the proposed approach is tested with different parameter settings. The parameters include the number of keywords *K* extracted from the key sentence *K*, the number of key sentences *T*, the prediction time point *M*, and the standard deviation *γ* for Bollinger bands. Therefore, a total of four experiments were conducted in this subsection.

First, experiments were conducted to show the influence of the number of keywords on the proposed approach. Other parameters such as the number of key phrases *T*, prediction time *M*, training interval *Y*, and standard deviation *γ* are set to 7, 0, 5, and 1, respectively. When the number of keywords *K* is different, the results for the number of positive and negative news will also different. Table IV shows that the number of positive and negative news can be obtained with different number of keywords from 4 to 10.

Table IV shows that when the number of keywords is less than 6, the number of positive and negative news generated is the same, indicating that there are too few keywords to perform effective sentiment analysis, so that a state of convergence occurs. The experimental results of the proposed approach with different number of keywords are shown in Table V.

From Table V, we can see that the best P/L ratio is 4.0821 when the number of keywords for the derived ensemble classifier was set to 10 by the proposed approach. The results also show that the number of keywords indeed has an impact on the construction of the classifier.

Experiments were then conducted to show the influence of the number of key sentences on the proposed approach. Other parameters, including the number of words *K*, prediction period *M*, training interval *Y*, and standard deviation *γ* are set to 10, 0, 5, and 1, respectively. The experimental results of the proposed approach with different number of key sentences from 4 to 7 are shown in Table VI.

Table VI shows that the P/L ratio and accuracy of the proposed approach are highest when the number of key sentences is set to 7. When the number of key sentence were set to 4 to 6, it can be seen that the results seem to be somewhat unstable in terms of P/L ratio.

Next, experiments were conducted to show the influence of prediction periods on the proposed approach. Other parameters such as the number of words *K*, the number of key sentences *T*, the training

TABLE V. Comparison Results of the Proposed Approach in Terms of Various Criteria Using Different Number of Keywords

K = 10	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.64	0.58	0.45	0.51	0.58	0.62	0.6	
0	0.69	0.45	0.54	0.59	0.71	0.65	0.65	0.6	0.62	
Accuracy	0.6			0.59			0.61			4.0821
K = 8	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.58	0.82	0.68	0.6	0.43	0.5	0.57	0.6	0.59	
0	0.78	0.53	0.63	0.55	0.71	0.62	0.64	0.61	0.62	
Accuracy	0.66			0.57			0.61			3.2891
K = 6	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.56	0.8	0.66	0.64	0.39	0.480.56	0.55	0.56		
0	0.77	0.52	0.62	0.53	0.76	0.62	0.62	0.63	0.62	
Accuracy	0.64			0.57			0.59			3.9313
K = 4	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.56	0.8	0.66	0.64	0.39	0.48	0.56	0.55	0.56	
0	0.77	0.52	0.62	0.53	0.76	0.62	0.62	0.63	0.62	
Accuracy	0.64			0.57			0.59			3.9313

TABLE VI. Comparison Results of the Proposed Approach in Terms of Various Criteria Using Different Number of Key Sentences

T = 7	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.64	0.58	0.45	0.51	0.58	0.62	0.6	
0	0.69	0.45	0.54	0.59	0.71	0.65	0.65	0.6	0.62	
Accuracy	0.6			0.59			0.61			4.0821
T = 6	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.55	0.75	0.64	0.58	0.38	0.45	0.58	0.57	0.58	
0	0.69	0.48	0.56	0.58	0.76	0.65	0.63	0.64	0.63	
Accuracy	0.6			0.58			0.61			3.6
T = 5	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.55	0.75	0.64	0.58	0.28	0.37	0.57	0.53	0.55	
0	0.69	0.48	0.56	0.56	0.82	0.67	0.61	0.65	0.63	
Accuracy	0.6			0.56			0.59			3.8661
T = 4	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.56	0.77	0.65	0.54	0.33	0.41	0.58	0.51	0.54	
0	0.7	0.48	0.57	0.56	0.76	0.64	0.61	0.67	0.64	
Accuracy	0.61			0.55			0.6			3.8225

TABLE VII. Comparison Results of the Proposed Approach in Terms of Various Criteria Using Different Number of Key Sentences

M = 0	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.54	0.77	0.64	0.58	0.45	0.51	0.58	0.62	0.6	
0	0.69	0.45	0.54	0.59	0.71	0.65	0.65	0.6	0.62	
Accuracy	0.6			0.59			0.61			4.0821
M = 3	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.32	0.14	0.2	0.64	0.64	0.64	0.52	0.37	0.43	
0	0.5	0.74	0.6	0.7	0.7	0.7	0.5	0.66	0.57	
Accuracy	0.46			0.67			0.51			1.0247
M = 5	**Positive Model (PM)**			**Negative Model (NM)**			**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
1	0.52	0.41	0.46	0.54	0.88	0.67	0.58	0.63	0.6	
0	0.57	0.68	0.62	0.71	0.28	0.4	0.56	0.52	0.54	
Accuracy	0.55			0.53			0.57			1.5192

TABLE VIII. Comparison Results of the Proposed Approach in Terms of Various Criteria Using Different γ Values

γ = 1	**BBAND**			**P/L Ratio**	**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1		Precision	Recall	F1	
1	0.6	0.52	0.56		0.58	0.61	0.6	
0	0.63	0.7	0.66		0.66	0.61	0.63	
Accuracy	0.62			3.2681	0.61			4.0821
γ = 2	**BBAND**			**P/L Ratio**	**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1		Precision	Recall	F1	
1	0.84	0.32	0.47		0.58	0.54	0.56	
0	0.62	0.95	0.75		0.62	0.66	0.64	
Accuracy	0.66			4.2631	0.61			4.2833
γ = 3	**BBAND**			**P/L Ratio**	**PM+NM+BBAND**			**P/L Ratio**
	Precision	Recall	F1		Precision	Recall	F1	
1	1	0.49	0.66		0.58	0.53	0.55	
0	0.54	1	0.7		0.62	0.66	0.64	
Accuracy	0.55			N/A	0.6			4.1916

TABLE IX. Comparison Results of the Proposed Approach and GASTP in Terms of Various Criteria Using Different *M* Values

M = 0	**Proposed Approach**			**P/L Ratio**	**GASTP**			**P/L Ratio**
	Precision	Recall	F1		Precision	Recall	F1	
1	0.58	0.62	0.6		0.58	0.42	0.49	
0	0.65	0.6	0.62		0.55	0.7	0.61	
Accuracy	0.61			4.0821	0.56			3.725
M = 3	**Proposed Approach**			**P/L Ratio**	**GASTP**			**P/L Ratio**
	Precision	Recall	F1		Precision	Recall	F1	
1	0.52	0.37	0.43		0.64	0.2	0.31	
0	0.5	0.66	0.57		0.51	0.88	0.65	
Accuracy	0.54			1.0247	0.53			2.6666
M = 5	**Proposed Approach**			**P/L Ratio**	**GASTP**			**P/L Ratio**
	Precision	Recall	F1		Precision	Recall	F1	
1	0.58	0.63	0.6		0.6	0.53	0.56	
0	0.56	0.52	0.54		0.54	0.61	0.57	
Accuracy	0.57			1.5192	0.57			1.2921

interval *Y*, and the standard deviation *γ* were set to 10, 7, 5, and 1, respectively. Table VII shows the experimental results of the algorithm at different prediction periods, which are 0, 3, and 5.

From Table VII, we can easily see that the P/L ratio of the proposed approach at a prediction period of 0 is 4.0821, which is the highest and better than the other two settings. Moreover, although the accuracy of the negative model is the highest at a prediction period of 3, the P/L ratio is the worst. From these experimental results, we can conclude that the effect of predicting the rise and fall of the day is the best. In other words, the financial news is suitable for predicting short-term stock trends.

Finally, experiments were conducted to show the influence of the weight of the standard deviation of the Bollinger bands on the proposed approach. Other parameters, such as the number of words *K*, the number of key sentences *T*, the prediction time period *M*, and the training interval *Y* were set to 10, 7, 0, and 5, respectively. Table VIII shows the experimental results of the proposed algorithm at different *γ*-values by 0, 3, and 5.

Table VIII shows that the proposed approach can achieve the best values for the P/L ratio when γ was set to 2. When *γ* was set to 3, the generated trading signal from the Bollinger Bands is "do not buy" and consequently the P/L ratio is zero. However, with the derived ensemble classifier, an acceptable P/L ratio can still be achieved, which means that the ensemble classifier is better than using a single classifier.

D. Comparisons With the State-of-the-art Models

In this section, experiments were conducted to show the results of the comparison between the proposed approach which is a sentence-level approach and the previous approach [16], where a keyword-based method, technical indicators and genetic algorithms were considered together to adjust the parameters for generating trading signals, called GASPT. The results of the comparison are shown in Table IX.

Table IX shows that the proposed approach is better than GASTP in terms of P/L ratio when the prediction period was set to 0. However, when *M* was set to 3, GASTP is better than the proposed approach. Based on the experimental results, we can conclude that the proposed approach is suitable and effective for short-term trading. Besides, in the proposed approach, the time information is important for generating training datasets, including the classification attributes and labels. It may also be utilized to reduce the training dataset when large amount of news are given. In other words, it is worthy to design a novel algorithm to increase the efficiency of the proposed approach in the future.

V. Conclusion and Future Work

This paper proposes a three-stage framework, including data preprocessing, ensemble classifier creation, and prediction phases, for predicting stock trends based on sentence-level sentiment analysis of Chinese news and technical indicators. In first phase, the goal is to generate the training instances from given corpus and stock prices. Then, based on the sentiment orientation of instances, the positive and negative classifiers are constructed. To increase the reliability of the prediction result, Bollinger bands are also considered as another classifier. The three classifiers are merged as a final ensemble classifier to generate trading signals. Experiments were also conducted on the real data set. The experimental results show that: (1) The longer the training period, the higher the accuracy and profit-to-loss ratio; (2) The number of keywords used to find the key sentences should be larger than a threshold value for the classifier to achieve more stable results; (3) The constructed ensemble classifier is suitable for predicting the stock trend on the news release day; (4) Compared with the previous key-level based prediction approach, the proposed sentence-level based approach is effective in terms of accuracy and the P/L ratio. In the future, the proposed approach can be further developed in the following directions. For example, the optimization algorithm can be applied to it to automatically find a suitable parameter setting, transfer learning can also be used to select useful instances from the given sources to improve the accuracy of the classifier, or weight mechanism can be designed for the positive and negative models to derive a better performance.

References

[1] M. Drenovak, V. Ranković, B. Urošević, R. Jelic, "Mean- maximum drawdown optimization of buy-and-hold portfolios using a multi-objective evolutionary algorithm," *Finance Research Letters*, p. 102328, 2021.

[2] T. E. Simos, S. D. Mourtas, V. N. Katsikis, "Time- varying black-litterman portfolio optimization using a bio-inspired approach and neuronets," *Applied Soft Computing*, p. 107767, 2021.

[3] S. T. Tayalı, "A novel backtesting methodology for clustering in mean–variance portfolio optimization," *Knowledge-Based Systems*, vol. 209, p. 106454, 2021.

[4] W. Ding, K. Mazouz, Q. Wang, "Volatility timing, sentiment, and the short-term profitability of vix-based cross-sectional trading strategies," *Journal of Empirical Finance*, vol. 63, pp. 42–56, 2021.

[5] X. Wu, H. Chen, J. Wang, L. Troiano, V. Loia, H. Fujita, "Adaptive stock trading strategies with deep reinforcement learning methods," *Information Sciences*, vol. 538, pp. 142–158, 2020.

[6] A. K. Banerjee, A. Dionisio, H. K. Pradhan, B. Mahapatra, "Hunting the quicksilver: Using textual news and causality analysis to predict market volatility," *International Review of Financial Analysis*, p. 101848, 2021.

[7] N. Pröllochs, S. Feuerriegel, D. Neumann, "Negation scope detection in sentiment analysis: Decision support for news-driven trading," *Decision Support Systems*, vol. 88, pp. 67–75, 2016.

[8] W. Chen, M. Jiang, W. G. Zhang, Z. Chen, "A novel graph convolutional feature based convolutional neural network for stock trend prediction," *Information Sciences*, vol. 556, pp. 67–94, 2021.

[9] J. Long, Z. Chen, W. He, T. Wu, J. Ren, "An integrated framework of deep learning and knowledge graph for prediction of stock price trend: An application in chinese stock exchange market," *Applied Soft Computing*, vol. 91, p. 106205, 2020.

[10] X. Zhang, A. Li, R. Pan, "Stock trend prediction based on a new status box method and adaboost probabilistic support vector machine," *Applied Soft Computing*, vol. 49, pp. 385–398, 2016.

[11] X. Li, H. Xie, L. Chen, J. Wang, X. Deng, "News impact on stock price return via sentiment analysis," *Knowledge-Based Systems*, vol. 69, pp. 14–23, 2014.

[12] G. Löffler, L. Norden, A. Rieber, "Negative news and the stock market impact of tone in rating reports," *Journal of Banking & Finance*, vol. 133, p. 106256, 2021.

[13] S. Kelly, K. Ahmad, "Estimating the impact of domain-specific news sentiment on financial assets," *Knowledge-Based Systems*, vol. 150, pp. 116–126, 2018.

[14] A. Thakkar, K. Chaudhari, "Predicting stock trend using an integrated term frequency–inverse document frequency-based feature weight matrix with neural networks," *Applied Soft Computing*, vol. 96, p. 106684, 2020.

[15] L. Lei, "Wavelet neural network prediction method of stock price trend based on rough set attribute reduction," *Applied Soft Computing*, vol. 62, pp. 923– 932, 2018.

[16] C. H. Chen, P. Shih, G. Srivastava, S. T. Hung, J. C. W. Lin, "Evolutionary trading signal prediction model optimization based on chinese news and technical indicators in the internet of things," *IEEE Internet of Things Journal*, p. Early access, 2021.

[17] G. Liu, X. Wang, "A new metric for individual stock trend prediction," *Engineering Applications of Artificial Intelligence*, vol. 82, pp. 1–12, 2019.

[18] P. Y. Hao, C. F. Kung, C. Y. Chang, J. B. Ou, "Predicting stock price trends based on financial news articles and using a novel twin support vector machine with fuzzy hyperplane," *Applied Soft Computing*, vol. 98, p. 106806, 2021.

[19] W. Long, L. Song, Y. Tian, "A new graphic kernel method of stock price trend prediction based on financial news semantic and structural similarity," *Expert Systems with Applications*, vol. 118, pp. 411–424, 2019.

[20] P. Rajendiran, P. L. K. Priyadarsini, "Survival study on stock market prediction techniques using sentimental analysis," *Materials Today: Proceedings*, p. Early access, 2021.

[21] C. Martín, D. Quintana, P. Isasi, "Dynamic generation of investment recommendations using grammatical evolution," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, pp. 104–111, 2021.

[22] L. Chen, L. Sun, C. M. Chen, M. E. Wu, J. M. T. Wu, "Stock trading system based on machine learning and kelly criterion in internet of things," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1– 9, 2021.

[23] J. M. T. Wu, L. Sun, G. Srivastava, J. C. W. Lin, "A long short-term memory network stock price prediction with leading indicators," *Big Data*, vol. 9, pp. 343–357, 2021.

[24] J. M. T. Wu, S. Sun, L., G., J. C. W. Lin, "A novel synergetic lstm-ga stock trading suggestion system in internet of things," *Mobile Information Systems*, vol. 2021, pp. 1–15, 2021.

[25] H. Huang, X. Liu, Y. Zhang, F. C., "News-driven stock prediction via noisy equity state representation," *Neurocomputing*, vol. 470, pp. 66–75, 2022.

[26] A. Derakhshan, H. Beigy, "Sentiment analysis on stock social media for stock price movement prediction," *Engineering Applications of Artificial Intelligence*, vol. 85, pp. 569–578, 2019.

[27] Y. Li, H. Bu, J. Li, J. Wu, "The role of text-extracted investor sentiment in chinese stock price prediction with the enhancement of deep learning," *International Journal of Forecasting*, vol. 36, pp. 1541–1562, 2020.

[28] C. Song, X. Wang, P. Cheng, J. Wang, L. Li, "Sacpc: A framework based on probabilistic linguistic terms for short text sentiment analysis," *Knowledge-Based Systems*, vol. 194, p. 105572, 2020.

[29] J. Wu, K. Lu, S. Su, S. Wang, "Chinese micro- blog sentiment analysis based on multiple sentiment dictionaries and semantic rule sets," *IEEE Access*, vol. 7, pp. 183924–183939, 2019.

[30] W. Wang, D. Huang, J. Cao, "Chinese syntax parsing based on sliding match of semantic string," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 7, pp. 1–14, 2020.

[31] P. Wang, Y. Luo, Z. Chen, L. He, Z. Zhang, "Orientation analysis for chinese news based on word embedding and syntax rules," *IEEE Access*, vol. 7, pp. 159888– 15898, 2019.

[32] S. Wang, J. Zhang, C. Zong, "Empirical exploring word-character relationship for chinese sentence representation," *ACM Transactions on Asian and Low- Resource Language Information Processing*, vol. 17, pp. 1–18, 2019.

[33] F. Chen, Y. Huang, "Knowledge-enhanced neural networks for sentiment analysis of chinese reviews," *Neurocomputing*, vol. 368, pp. 51–58, 2019.

[34] T. Daudert, "Exploiting textual and relationship information for fine-grained financial sentiment analysis," *Knowledge-Based Systems*, p. 107389, 2021.

[35] G. Xu, Y. Meng, X. Qiu, Z. Yu, X. Wu, "Sentiment analysis of comment texts based on bilstm," *IEEE Access*, vol. 7, pp. 51522–51532, 2019.

[36] J. Yang, R. Yang, H. Lu, C. Wang, J. Xie, "Multi- entity aspect-based sentiment analysis with context, entity, aspect memory and dependency information," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 18, pp. 1–22, 2019.

[37] M. Pota, H. Ventura, M. amd Fujita, M. Esposito, "Multilingual evaluation of pre-processing for bert- based sentiment analysis of tweets," *Expert Systems with Applications*, vol. 181, p. 115119, 2021.

[38] H. Peng, Y. Ma, S. Poria, Y. Li, E. Cambria, "Phonetic- enriched text representation for chinese sentiment analysis with reinforcement learning," *Information Fusion*, vol. 70, pp. 88–99, 2021.

[39] A. Li, P. Wu, W. Wang, "Incorporating stock prices and news sentiments for stock market prediction: A case of hong kong," *Information Processing & Management*, vol. 57, p. 102212, 2020.

[40] T. Loughran, B. Mcdonald, "When is a liability not a liability? textual analysis, dictionaries, and 10-ks," *Cognitive Computation*, pp. 1167–1176, 2011.

Chun-Hao Chen

Chun-Hao Chen is an associate professor at Department of Information and Finance Management at National Taipei University of Technology, Taipei, Taiwan. Dr. Chen received his Ph.D. degree with major in computer science and information engineering from National Cheng Kung University, Taiwan, in 2008. He has a wide variety of research interests covering data mining, time series, machine learning, evolutionary algorithms, and fuzzy theory. Research topics cover portfolio selection, trading strategy, business data analysis, time series pattern discovery, etc. He serves as the associate editor of the International Journal of Data Science and Pattern Recognition, and IEEE Access. He is also a member of IEEE.

Po-Yeh Chen

Po-Yeh Chen received her M.S. degree in Computer Science from the Department of Computer Science and Information Engineering at Tamkang University, Taiwan, in 2020. He now is a technical engineer of Comwave International Ltd. His research interests include text mining and classification with a focus on financial sector.

Jerry Chun-Wei Lin

Jerry Chun-Wei Lin received his Ph.D. from the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan in 2010. He is currently a full Professor with the Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway. He has published more than 500+ research articles in refereed journals (IEEE TKDE, IEEE TFS, IEEE TNNLS, IEEE TCYB, IEEE TII, IEEE TITS, IEEE TNSE, IEEE TETCI, IEEE SysJ, IEEE SensJ, IEEE IOTJ, ACM TKDD, ACM TDS, ACM TMIS, ACM TOIT, ACM TIST, ACM TOSN) and international conferences (IEEE ICDE, IEEE ICDM, PKDD, PAKDD), 15 edited books, as well as 33 patents (held and filed, 3 US patents). His research interests include data mining and analytics, natural language processing (NLP), soft computing, IoTs, bioinformatics, artificial intelligence/machine learning, and privacy preserving and security technologies. He is the Editor-in-Chief of the International Journal of Data Science and Pattern Recognition, the Guest Editor/Associate Editor for several IEEE/ACM journals such as IEEE TFS, IEEE TII, IEEE TIST, IEEE JBHI, ACM TMIS, ACM TOIT, ACM TALLIP, and ACM JDIQ. He has recognized as the most cited Chinese Researcher respectively in 2018, 2019, and 2020 by Scopus/Elsevier. He is the Fellow of IET (FIET), ACM Distingushed Member (Scientist), and IEEE Senior Member.

AWS PredSpot: Machine Learning for Predicting the Price of Spot Instances in AWS Cloud

Alejandro Baldominos*, Yago Saez, David Quintana, Pedro Isasi

Computer Science and Engineering Department, Universidad Carlos III de Madrid, Leganés, Madrid (Spain)

Received 30 July 2020 | Accepted 25 January 2022 | Published 8 February 2022

LA UNIVERSIDAD
EN INTERNET

Abstract

Elastic Cloud Compute (EC2) is one of the most well-known services provided by Amazon for provisioning cloud computing resources, also known as instances. Besides the classical on-demand scheme, where users purchase compute capacity at a fixed cost, EC2 supports so-called spot instances, which are offered following a bidding scheme, where users can save up to 90% of the cost of the on-demand instance. EC2 spot instances can be a useful alternative for attaining an important reduction in infrastructure cost, but designing bidding policies can be a difficult task, since bidding under their cost will either prevent users from provisioning instances or losing those that they already own. Towards this extent, accurate forecasting of spot instance prices can be of an outstanding interest for designing working bidding policies. In this paper, we propose the use of different machine learning techniques to estimate the future price of EC2 spot instances. These include linear, ridge and lasso regressions, multilayer perceptrons, K-nearest neighbors, extra trees and random forests. The obtained performance varies significantly between instances types, and root mean squared errors ranges between values very close to zero up to values over 60 in some of the most expensive instances. Still, we can see that for most of the instances, forecasting performance is remarkably good, encouraging further research in this field of study.

Keywords

Cloud Computing, Machine Learning, Prediction, Price Forecasting.

DOI: 10.9781/ijimai.2022.02.003

I. Introduction

Amazon Web Services (AWS) is an Amazon ecosystem comprising a large number of cloud services. This ecosystem is in a process of continuous growth, with new services or functionalities added every few months.

One of the most well-known AWS services is EC2 (Elastic Cloud Compute), an application that provides Infrastructure-as-a-Service (IaaS) for cloud computing. These services allow users to launch on-demand instances (virtual machines) in order to satisfy certain computational needs. This option is interesting when a user or company has a variable computing load, thus avoiding the need to acquire specific infrastructure whose administration and maintenance can become very expensive.

Besides on-demand instances, EC2 allows users to bid for computing capacity that is not in use. This enables users to establish a maximum bidding price and, in case they be the winner of the bid, then they are able to use the corresponding computational capacity. In EC2, these instances are called “spot instances”. The hourly cost of spot instances can be significantly lower than on-demand instances; however, the instance will only belong to the user as long as the bid is higher than the spot price. In other case, the instance will be terminated and the user will not be able to access it anymore.

AWS allows to query the price of spot instances in real time [1]. Additionally, users can study the historic evolution of EC2 instances of a certain type, up to three months in the past, as it can be seen in Fig. 1.

In this paper, we aim at designing and developing a system able to predict the future price of a spot instance in EC2, with the final objective of easing the optimization of the bidding procedure. To do so, we will rely on historic information in the spot instances prices.

The remainder of this document is structured as follows: in Section II we present some basic concepts which are key to understand the current proposal, in Section III we will briefly describe the state of the art and some related work.

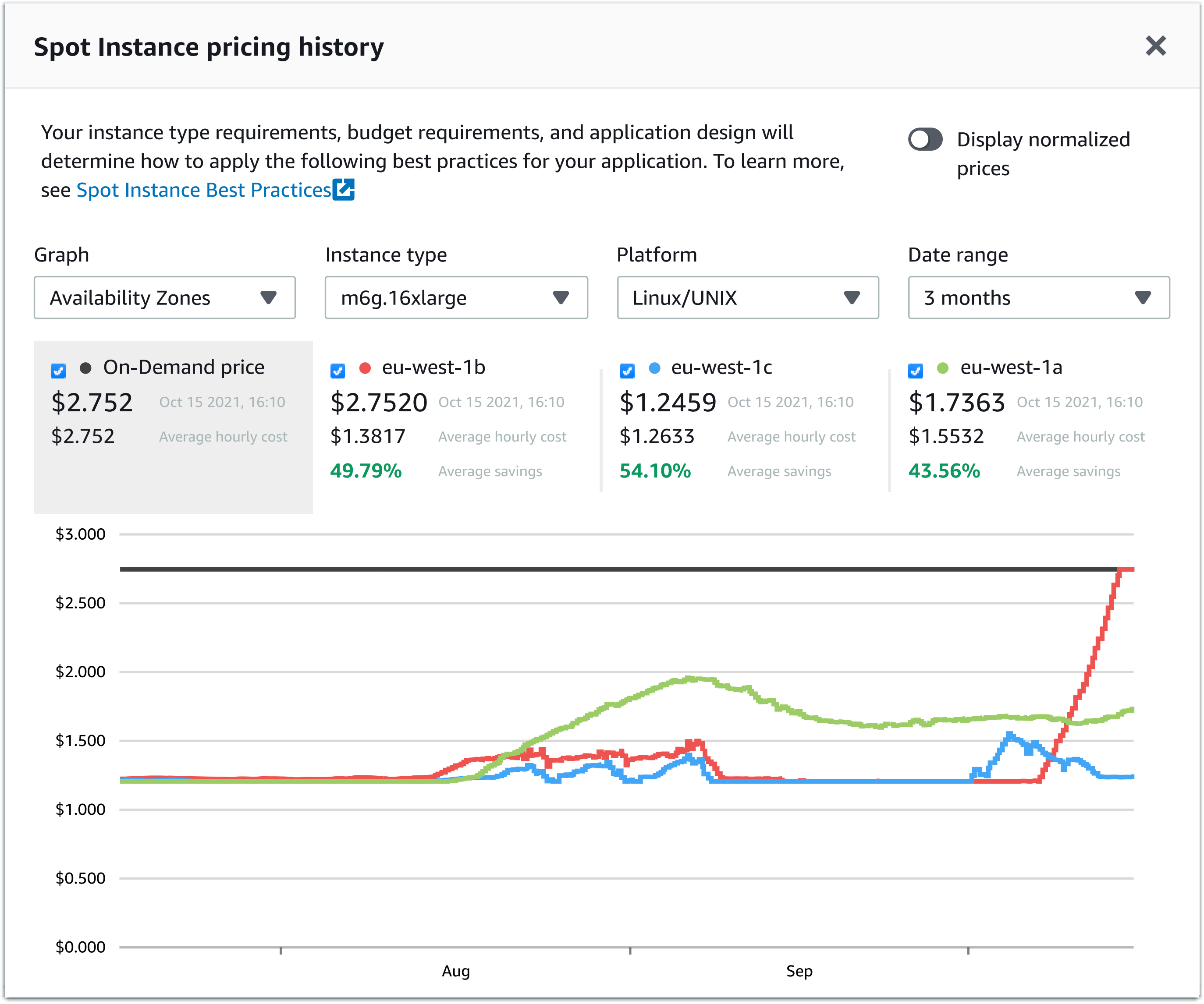
Then, in Section IV,we will identify different data sources, explaining the acquisition process, and in Section V we will describe the cleansing and processing stages.

Later, in Section VI we will detail the procedure for learning regression models that fit the instance prices and in Section VII we will provide quality metrics to assess the performance of the learned prediction models and discuss the results obtained.

Finally, in Section VIII we will provide some conclusive remarks regarding the work performed in this paper as well as suggest lines of future research. In , we will present the prediction system delivered as a service, describing the infrastructure underlying the prediction system and an API for accessing it.

* Corresponding author.

E-mail address: abaldomi@inf.uc3m.es



Spot Instance pricing history ✕

Your instance type requirements, budget requirements, and application design will determine how to apply the following best practices for your application. To learn more, see Spot Instance Best Practices

○ Display normalized prices

Graph: Availability Zones ▼

Instance type: m6g.16xlarge ▼

Platform: Linux/UNIX ▼

Date range: 3 months ▼

☑ ● On-Demand price		☑ ● eu-west-1b		☑ ● eu-west-1c		☑ ● eu-west-1a	
$2.752	Oct 15 2021, 16:10	$2.7520	Oct 15 2021, 16:10	$1.2459	Oct 15 2021, 16:10	$1.7363	Oct 15 2021, 16:10
$2.752	Average hourly cost	$1.3817	Average hourly cost	$1.2633	Average hourly cost	$1.5532	Average hourly cost
		49.79%	Average savings	54.10%	Average savings	43.56%	Average savings

Fig. 1. Panel showing the evolution of the EC2 spot instance price in the AWS console, for a specific instance type and different availability zones, over a period of three months. On-demand (non spot) price is shown in the black line.

II. Structure of the Paper

Before proceeding with the description of the proposal, it is important to introduce some relevant concepts that are required to understand which factors affect the instance price (both in the case of on-demand and spot instances). These factors are the following:

- Region: it refers to the Amazon datacenter where the instance will be launched. Some examples of regions are the following: North Virginia (us-east-1), Ohio (us-east-2), North California (us-west-1), Canada (ca-central-1), Ireland (eu-west-1), etc.
- Availability zone: it is a more precise area within the region. It is identified with a letter after the region codename, for instance, region “us-east-1” contains zones from “us-east-1a” to “us-east-1f”. Instances, even those of the same type, can see their cost affected depending on their availability zone.
- Type and size: the instance type determines the compute capabilities it provides. Most often, the instance types adheres to the following convention: <type>.<size>. For example, an instance p3.16xlarge is an instance of type P3 (general use GPU computing) and of size 16xlarge, meaning in this case that it contains 8 GPUs. Amazon provides an updated listing with all the different instance types and their specifications [2]. This factor is the one that affects the most the instance cost.
- Operating system: also called “product” the instance cost can vary depending on whether it runs a Windows environment or a UNIX/Linux one.

III. Related Work

The problem of forecasting the prices of EC2 spot instances is of clear interest, since it allows companies of different sizes to work on optimal bidding strategies that can optimize economic resources spent on cloud computing infrastructure. For this reason, this problem has been observed mostly from two perspectives. The first perspective relies on the study of EC2 spot pricing as an economic problem, using approaches based on econometrics or other financial tools to design bidding models. The second perspective, which is the one followed in this paper, relies on techniques of computational intelligence to frame the approach as a supervised learning problem.

One early work which aims at reverse engineering the EC2 spot pricing scheme is provided by Ben-Yehuda et al. [3], where they build a model concluding that the prices are not fully market-driven, but are most of the times generated randomly within a small range of values. They do not suggest a bidding strategy as such, but rather work on a thorough economic analysis of the pricing models.

Another statistical analysis of the pricing scheme of EC2 spot instances is provided much more recently by Portella et al. [4]. Again, they do not focus on a bidding strategy or price forecasting, but obtain a useful conclusion from the analysis: by bidding at 30% of the on-demand price, availability of over a 90% can be attained, although the specifics vary based on the instance type. Another work by Lumpe et al. [5] focuses as well on a descriptive statistical analysis and econometric study of EC2 spot prices, with authors also devising a bidding strategy that minimises the bidding cost while guaranteeing a certain probability of availability over a defined threshold.

Another early work by Tian et al. [6] suggests a decision model for provisioning computing resources in EC2 by combining different schemes, combining spot instances with the classical on-demand model. In the paper, they introduce a model able to predict the demand and spot prices are expected to vary as a result. Interestingly, they do not focus only on price forecasting, but also in how to diversify instances to deal with potential loss of spot instances (in case their actual price exceeds the bidding price).

Tang et al. [7] address the problem of tackling an optimal bidding strategy. In this case, authors use Markov decision processes, and prove a theorem by which any sequence of bidding decisions can be obtained by a dual-option strategy, either bidding the maximum spot price or giving up at each time. In a more recent work [8], the authors apply this strategy under service-level agreement constraints. In both works, the authors do not focus on price forecasting as an intermediate task to build the bidding policy.

Chhetri et al. [9] have studied the streamlined EC2 spot markets, a different model where prices are softened by using long-term trends in demand and supply. They combine econometric indices as well as computational techniques (logistic regression and principal component analysis) to perform their study. Authors extract interesting conclusions from their analysis: median spot prices have grown in the streamlined model, and sophisticated bidding strategies are less useful in this pricing model. Also, they suggest how to perform bidding price estimation.

When focusing on spot price forecasting, Chhetri et al. [10] use time-series decomposition and look-backs, attaining results that compare or slightly outperform other more classical approaches. For the evaluation, they constrain to eight Microsoft Windows-based instance types in the Sydney region, attaining root mean squared errors that achieve values of 0.559 in c3.xlarge instances.

Another approach using regression random forests has been provided by Khandelwal et al. [11], where they learn models to perform one-day and one-week ahead forecasting. They state that this technique outperforms other methods, reporting a mean absolute percentage error of less than 10% for one-day and less than 15% for one-week forecasting.

A more recent approach has been proposed by Lancon et al. [12], where they use long short-term memory neural networks and claim a reduction of 95% in mean average percentage error as when compared to a baseline model. Unfortunately, absolute errors do not seem to be reported in the paper. We also found this problem in a recent contribution by Malik and Bagmar [13]. These authors discuss a technique to analyze and predict the spot prices for instances using random forests. The authors report mean average percentage errors in the range from 0.15% to 56.2% depending on the instance type.

Recently, Chittora and Gupta [14] explored the feasibility of relying on 2-layer stacked LSTM model for this task using 3 months of spot price data for 5 instances. The results for next day spot price forecast show mean absolute percentage errors under 10% and root mean squared errors below 20%, outperforming the standard LSTM and the 3-layer LSTM considered as alternatives.

Finally, Liu et al. [15] benchmarked kNN regression against linear regression, support vector machine, random forest, multi-layer perceptron and gcForest using the $MAPE_{5\%}$, which represents the number of results whose absolute percentage error is less than or equal to 5% as a percentage of the number of total results, as performance metric. According to their results, kNN regression offered the best performace with a $MAPE_{5\%}$ up to 94% in 1-day-ahead prediction and 94.06% in 1-week-ahead, respectively.

In this work, we will carry out an extensive comparison of diverse machine learning techniques towards forecasting of EC2 spot prices. To the best of our knowledge, this is the most detailed work when it comes to tackling a larger number of EC2 instance types and reporting results in a separate manner for each of them, as well as comprehensive due to the large number of techniques tested.

IV. Data Sources

In this section we will present the different data sources used in order to train and validate the regression models for the price prediction of EC2 spot instances.

A. Approach

The instance price data is modeled as a time series. This series can be seen as a sequence of values for each instant of time, existing a different series for each region, instance type and operating system. Each value in the time serieswill contain a timestamp indicating the moment of time towhich it refers, as well as the instance price at that time.

The data in the time series can be obtained via two different approaches: recovering them from historic archives or querying the prices in real time. In the first case, we would be talking of previously captured data, stored for their later recovery. Meanwhile, in the second case we would refer to new data that is changing as time happens.

The availability of historic data is useful for feeding the models with a large amount of values (e.g., corresponding to several years). Conversely, access to real-time data is useful to provide feedback to the model and updating it periodically to ensure that its predictions are updated to the current characteristics of the time series.

B. History Data Sources

In this work we have used two different data sources providing information of archived historical data of EC2 spot instances prices.

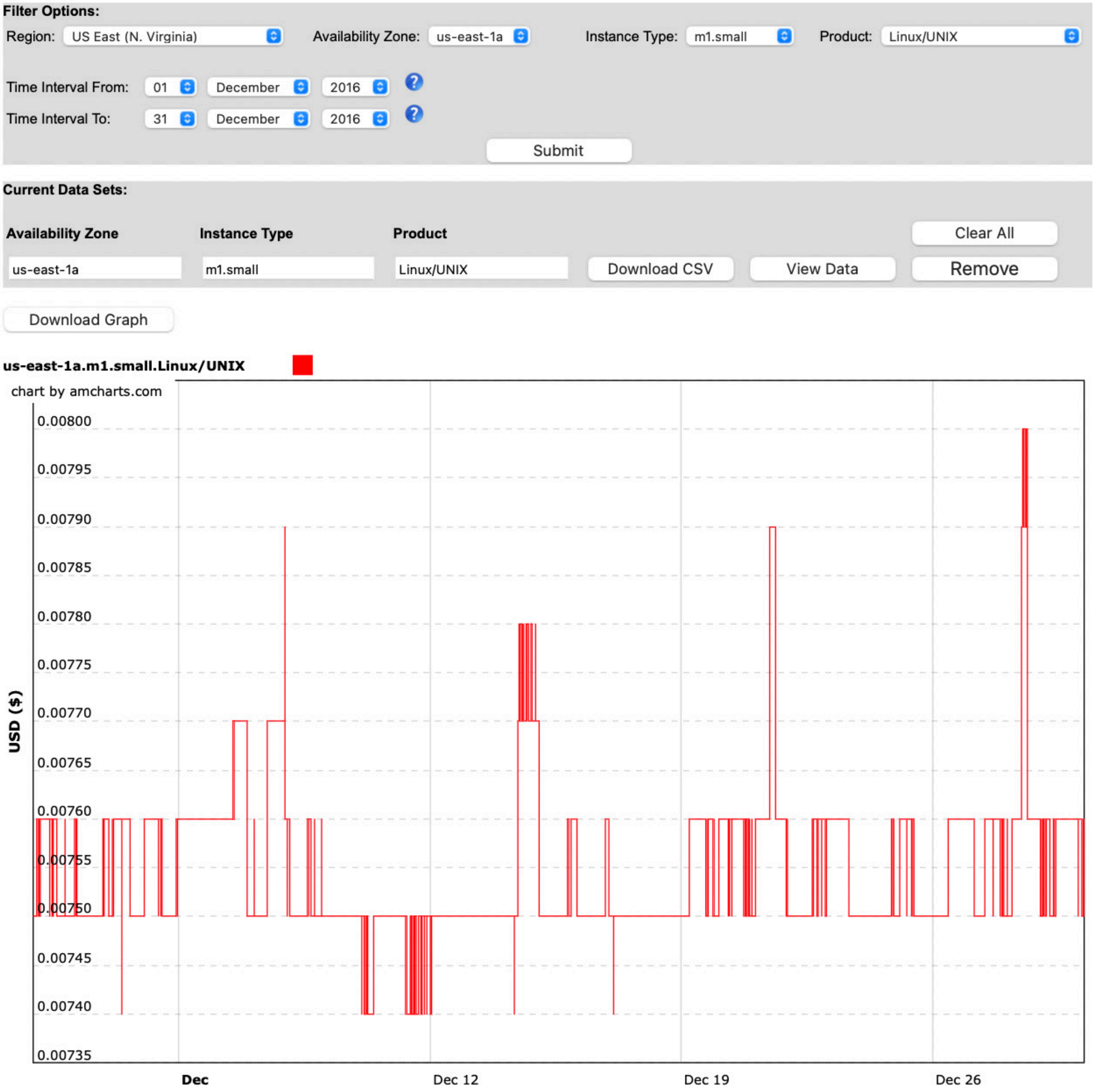
1. AWS Spot Pricing Market

Dataset provided by the Data Science Awards 2017 competition, which is publicly available for download in Kaggle. This involves a CSV file for each of the regions [16]. The structure of these CSV files is shown in Table I.

TABLE I. Structure of AWS Spot Pricing Market CSV Files

Timestamp	Type	OS	Zone	Price
2017-05-06 17:29:01	c4.large	Linux	ca-central-1a	0.0139
2017-05-06 17:29:01	m4.4xlarge	Windows	ca-central-1b	0.8328

This dataset is very complete as it gathers many different types of possible instances, for every operating system and comprising eleven AWS regions. However, the main drawback of this dataset has to do



Filter Options:

Region: US East (N. Virginia)
Availability Zone: us-east-1a
Instance Type: m1.small
Product: Linux/UNIX

Time Interval From: 01 December 2016
Time Interval To: 31 December 2016

Current Data Sets:

Availability Zone	Instance Type	Product
us-east-1a	m1.small	Linux/UNIX

Fig. 2. Graphical user interface of Spot Price Archive.

with its limited of the time period considered, since it only includes instance prices between February and May 2017, not providing time series longer than three months.

This is of course a problem when trying to capture seasonal patterns. To illustrate this with an example, it could be reasonable to hypothesize that in summer season instances are cheaper, because of the low demand (there are fewer companies actively requiring cloud computing services). Conversely, there could be demand peaks at other moments, such as back-to-work period or Christmas campaigns.

Moreover, there could exist inter-annual trends, such as a decrease in the average instance cost, which cannot be detected since the dataset only comprises data from 2017.

2. Spot Price Archive

Spot Price Archive is a historic data archive of EC2 spot instances prices provided by Western Sydney University, Australia. The archive provides a graphical interface for accessing data [17] (see Fig. 2) and was developed for a project aiming at modeling the spot instances price [18].

This dataset is much more complete regarding the length of time series, since it provides data for all years comprised between 2009 and 2016. As a drawback, it imposes some limitations over the previous dataset, since it only comprises certain regions and instance types.

In particular, Data Science Awards dataset provided prices for 68 instance types and 11 regions, whereas Spot Price Archive only contains 15 instance types in 8 regions. Besides, the number of availability zones is also more limited. For example, region us-east-1 comprises six zones (from us-east-1a to us-east-1f), from which the former dataset gathers five and the latter only four.

Despite of being more restricted, this historic dataset will be used to improve the prediction performance over those instance types and regions included in the dataset, providing more information about seasonal behaviors along the year as well as inter-annual trends.

Data from this dataset can be downloaded in CSV files. The acquisition process can be automated because URLs for the CSV files always follow the same convention, with the following base URL:

```
http://spot.scem.uws.edu.au/ec2si/Download.jsp
```

This URL accepts the following query parameters, which can be specified in the GET request: Zone (the availability zone), Type (the instance type), Product (the operating system), IntervalFrom (the start date, formatted as "yyyy-mm-dd 00:00:00.0" and IntervalTo (the end date, with the same format).

The CSV files obtained are structured as shown in Table II. It can be seen howthis structure is equivalent to the previous dataset, since the values in the columns corresponding to the availability zone, the instance type and the operating system are known beforehand.

TABLE II. Structure of Spot Price Archive CSV Files

Timestamp	Price
2012-11-05 12:00:00	0.006

C. Real-Time Data Sources

Historic data allows us to learn regression models that can take into consideration inter-annual trends and seasonal factors. Nevertheless, it is important to periodically feedback these models in order to keep them upgraded and get useful and accurate predictions over time. In this work we consider the use of one source of real-time data.

1. EC2 API

The most convenient way to obtain real-time data is to use EC2's API, which has an endpoint (*describe_spot_price_history*) [19] that returns the history of prices from the current time up to 90 days into the past. By calling this endpoint, we can obtain real-time data.

The endpoint returns a JSON-encoded object with the following structure, as described in the specification:

```
{
  "SpotPriceHistory" : [
    "AvailabilityZone"   : <zone>,
    "InstanceType"       : <type>,
    "ProductDescription" : <os>,
    "Timestamp"          : datetime(yyyy, m, d),
    "SpotPrice"          : <price>
  ], ...
}
```

These fields identify the instance type, the zone and the operating system, as well as the timestamp and the price. Therefore, we will be able to easily transform this data into CSV files with the format that we had seen previously in the case of historic data.

V. Data Processing

Before training the regression models for instance price prediction, we will perform some basic processing of the data in order to extract relevant features that can be useful for training the models, as well as to standardize the output format of the different data sources.

A. Feature Selection

After acquiring the data, the available attributes are those characterizing the instance (availability zone, type and operating system) and the timestamp.

The instance features comprise categorical data which will not be subject to any additional processing. However, in the case of the timestamp, it is stored as a text string, which is not particularly useful. We will use it to retrieve some features which can be of interest:

- Year: the year can be relevant to discover inter-annual trends, as whether the price of a certain instance type decreases as new instances are released.
- Month: the month is an important feature to detect intraannual seasonal trends, such as whether the price is lower in summer months, when demand may be lower due to the holidays season.
- Day of month: it is difficult to know whether this is a relevant feature, but it could be in the case of intra-monthly trends.
- Day of week: it can be interesting to detect whether prices change on weekdays versus weekends.
- Hour: it can be of interest because the cost could change based on days as opposed to nights. For this reason, this feature could also depend on the instance region. We have omitted the minute and second since they do not seem to be relevant features affecting the instance price.

B. Output Format Standardization

As we saw in the previous section, all data from different sources is equivalent when it comes to the features (columns). When it comes to rows, they also follow the same format, which is the one provided by Amazon EC2 API: a row is only shown when there is a change in the instance price. This means that the difference between two consecutive timestamps is not constant, and also that there is not explicit information about the price at every timestamp, although this information can be easily inferred.

In this standardization, we reduce the resolution of the timestamp from seconds to hours, as we explained previously. In some cases, the prices can slightly vary within the same hour, in which case we compute the median of the different values. We have decided to use the median instead of the mean to avoid adding values that did not appear originally in the input data.

Finally, we fill the non-existing rows between the start and end date. Since rows only exist when there is a change in the price, newrows will have as the price the last value available immediately before the time of the row being added.

VI. Model Learning

In this section we will explain the machine learning techniques used to train the regression models for instance price prediction. First, we will discuss some design decisions regarding the training process, and later we will detail the training procedure.

A. Design Decisions

When learning a regression model from a time series such as the one described in this paper, we can mainly choose among two different approaches.

In the first approach, we would use the information available in the time series to predict the next value, which is unknown. In this case, the attributes are formed by the price in the *n* previous times. In other words, given $(t_0, t_1, ..., t_n)$, we want to predict the value at time t_{n+1}.

This approach, despite being very common, does not seem the most appropriate for solving the problem. The first reason is that the quality of predictions degrades when we want to predict values that are far in the future, since we would be using as input some features whose value is unknown and are just an estimation (this means that for predicting t_{n+m}, we will need values (t_{n+1}, t_{n+2}, ..., t_{n+m-1}), which are not known). The second reason, which is domain-dependent, is that this time series is very static, and in some cases an instance can hold the same price during hours or even days. For this reason, a prediction model would like turn to keep the price invariant, missing all the times that the price is actually updated.

The second approach, which we have deemed more interesting in this work, is to introduce as input the parameters that were previously described: year, month, day of month, day of week and hour, as well as availability zone, instance type and operating system. In this case, knowing the values in the time series immediately before the desired prediction time instant is irrelevant, since they are not used for the prediction. As a consequence of this, the approach has the advantage that the quality of the prediction is not affected by how far in the future the desired value is.

Finally, we have decided to train a model for each instance type. This is due to the fact that there is a significant heterogeneity between different instance types, and one model could find difficulties when dealing with such an amount of diverse data. Nevertheless, given a fixed instance type, the price is much more stable, even though it still can vary significantly depending on the availability zone, the operating system, and the date and time.

B. Learning Procedure

In order to train the models, we will use the Scikit-learn library for Python [20].

The first step is to use one-hot-encoding (OHE) to convert categorical attributes into binary features in order to establish a data format which can be accepted by this library. As an example, if we had a sample whose operating system is "Windows", the OHE encoding would generate three binary features, from which feature "os_windows" would be set to one and features "os_linux-unix" and "os_suse-linux" would be zero.

When it comes to the process of learning a regression model from the available data, there are numerous techniques that could be used, and it can be difficult to determine beforehand which of such techniques could work best. In fact, it could happen that a technique works the best on a certain instance type, but be outperformed by other techniques in other types. In this case, we could not claim that a technique is the best performer.

Because of this, we will test different machine learning techniques with each type of instance. In particular, these techniques are the following:

- Linear regression: a standard algorithm that will learn a hyperplane fitting input data with the least square error.
- Linear regression with ridge regularization: same as the previous one yet imposing a penalty in the size of the regression coefficients.
- Linear regression with lasso regularization: same as the first one, but preferring solutions with fewer parameters.
- Multilayer perceptron: a neural network that can act as a universal function approximator. In the chosen setup, it comprises one hidden layer with 100 units.
- K-nearest neighbors: a geometric model where the K closest instances to the one being predicted are retrieved, and their outputs are averaged to provide a prediction. In the chosen setup, K is set to five.
- Extra Trees: an ensemble grouping several models and weighting their outputs. In particular, in this case these models will be regression trees, where each one will be trained using a random sample of the data and a random subset of the features. In the current setup, the ensemble will be formed of 10 models.
- Random Forests: similar to Extra Trees, yet with less randomness when it comes to choose the attributes to build the decision trees.
- AdaBoost: an ensemble where a regression model is first fitted over the original data and then additional models are trained over this data, but giving a higher weight to instances poorly estimated by previous models.
- Bagging: an ensemble where several regression models are trained over different samples of data.

We have chosen these techniques since they capture a large diversity of the machine learning techniques. For instance, linear regression is a simple model aiming at learning a line within the space of features and, along with the variations using ridge and lasso regularization, they are a good representative of linear models. The multilayer perceptron is the best representative of a feed-forward neural network. K-nearest neighbors is a simple model based on analogy, that is able to capture complex frontiers of decision in the space of features, under the hypothesis that similar instances will have a similar output. Finally, we have tried different ensembles based on decision trees, which are models able to learn rules for making a decision based on the features' values. We have only tested ensembles of decision trees since individual trees will often take longer to train and will rarely obtain better results, as they are more prone to overfitting.

Table III summarize the hyperparameters used for these techniques. These hyperparameters have been chosen after a prior stage of sensitivity analysis.

TABLE III. Hyperparameters of the Different ML Techniques

Technique	Parameter	Value
Ridge / Lasso	Regularization strength	1
MLP	Number of layers Number of units Activation function Optimizer	1 100 ReLU Adam
KNN	Number of neighbors (K) Distance metric	5 Euclidean
RF / ET / Bagging	Number of models	10
AdaBoost	Number of models	50

Once the techniques are chosen, we will follow the next procedure: First, we will split the dataset for each instance type into a training set and a test set. Instead of performing a random division of data, we have decided that data from September 2017 be assigned to the test set, with previous information assigned to the training set.

Later, for each instance type we will train each model ten times. This decision is motivated by the fact that most of the previously described techniques are stochastic, and therefore one single run could introduce an important bias. Finally, for each instance type we will serialize the best model, so we can recover it later when aiming to predict the price of a spot instance in the future.

VII. Model Evaluation

To validate the different models, we will compute three quality metrics for the best model obtained and compare it to a baseline. Such baseline will correspond to the performance of a naive regression model that would always predict the average price. The most improvement over such baseline, the best performance of the model.

The metrics reported in this work are the following:

- Root Mean Squared Error (RMSE) is the square root of the mean of squared errors. Therefore, being yi the real price for sample *i* and

($\hat{y}_i$) the price estimated by the regression model, RMSE is computed as in (1).

$$RMSE = \sqrt{\frac{\sum_{i\in N}\left(y_i - \hat{y}_i\right)^2}{|N|}} \tag{1}$$

The closest this value is to zero, the most accurate the model predictions will be.

- Explained Variance Score (EVS) computes to which extent a regression model captures the distribution of the original data. It is computed as shown in (2).

$$EVS = 1 - \frac{Var\{y - \hat{y}\}}{Var\{y\}} \tag{2}$$

The value will be better as it approaches one. In the case of the baseline, since the mean is always returned, then EVS will be zero.

- R^2 Score, or coefficient of determination, measures to what extent the model will estimate future samples, and is computed as shown in (3).

$$R2 = 1 - \frac{\sum_{i\in N}\left(y_i - \hat{y}_i\right)^2}{\sum_{i\in N}\left(y_i - \bar{y}_i\right)^2} \tag{3}$$

Again, the value will be better as it approaches one. For the baseline, since y_i is the mean value, R^2 score will be zero. Notwithstanding, a model arbitrarily worse could have a negative score in this metric.

Table IV shows the results of the evaluation in terms of the previously defined metrics, also showing the technique leading to the best obtained model. In the case of the baseline, only the RMSE is shown, since EVS and R^2 score are always zero. The last column displays an estimation of the model quality depending on its R^2 value.

A. Discussion

As it can be seen, results vary significantly depending on the instance type. For some types, such as *c3*, *r3*, *m1*, *m3* or *i3*, we have achieved models that are able to successfully predict prices for almost the entire family. As we suggested earlier, the best model can vary from one instance type to another, although the multilayer perceptron or ensembles often behave well in many cases.

Conversely, some instance types obtain poor results, even if RMSE always improves over the baseline. This happens because the instance type has a small spot offer, therefore turning the market price more unpredictable. This effect is clearly seen in some instance types.

For example, *f1.16xlarge* instances have an on-demand cost of 13.2 dollars per hour, but a baseline RMSE over 65 dollars. The reason is that in a scenario with few offer, the price can be set to the maximum established by AWS, which is ten times the on-demand price, i.e., 132 dollars. This can severely affect the time series, leading to a bumpy landscape that can harden the process or learning a regression model. In such cases, an alternative would be needed to improve these models performance.

Regarding machine learning techniques, there is not a clear winner that shows an outstanding prediction capability for all of the diversity of EC2 instance types. Generally speaking, linear regression is not dominant except for a small set of instances, regardless of whether regularization is used or not. This seems to indicate that most instance types have spot prices that do not have a linear dependency on the input features. Also, KNN is not displaying a good performance except for a couple of instance types. We can also see how MLP seems to be the model of choice for those instance types where prediction is more difficult and leads to worse result. This can be due to the fact that MLP is able to approximate the series of prices better than any other model, but it is still insufficient for considering the result as a good prediction. Conversely, ensembles of decision trees are found most often among the best models to achieve successful regression. Given this information, it seems that price prediction is rarely a linear problem, except for a few cases of instance types.

VIII. Conclusions

In this paper we have described all the steps carried out to tackle the problem of predicting EC2 spot instance prices. In this problem, we are interested in knowing the price of a certain spot instance at some point in the future, in order to be able to bid consequently. In order to solve this problem, we have used two different historical datasets, as well as data extracted in real time from the EC2 API, providing data from the last three months. Once data is retrieved, we have transformed then in order to extract relevant features from the timestamp and have later trained a regression model for instance type. The rationale beyond training separate models based on the instance types is that there is a very high variability in the prices depending on the type.

In particular, we have used Scikit-learn to test different regression techniques and selected those improving the quality metrics for each instance type: RMSE, EVS and R2. When looking at the results, we notice that some instance types obtain almost perfect models, whereas in others the baseline (a base prediction of the average price) was barely outperformed. This difference can be explained at least partially due to the characteristics of the instance.

Finally, we have developed a Prediction-as-a-Service system which we have deployed in the cloud. The infrastructure underlying this service as well as the API documentation is described in the appendix.

In order to further improve this work, we could add the support for more instance types and availability zones, by retrieving enough data from the EC2 API. Also, we could introduce more features to the problem, taking into account that these features must be known beforehand for those instances we want to predict. An example of such attribute could be whether the day is a national holiday in the region where we want to predict the price.

Appendix: Prediction-as-a-Service

In this appendix, we will detail the backend infrastructure required for storing the models, keeping them updated and enabling real-time prediction using a public endpoint (web service), as well as describe the interface for using the prediction system as a service.

A. Infrastructure

The CSV data and serialized models will be stored in the cloud, in an S3 bucket.

In a periodic fashion, a batch process will update the models. To do so, it will create an EC2 instance that will download the data from S3, include the most recent data using the EC2 API and finally retrain the machine learning model with the new data. This model will be stored in S3 replacing the previous version.

To provide the prediction service, we have deployed the implementation of our API over AWS Lambda. This cloud service allows us to run code in a serverless infrastructure, i.e., without requiring us to manually deal with the server resources, and providing an URL that could be used by the API clients. Moreover, AWS guarantees the service scalability, therefore allowing large number of concurrent requests without increasing the latency or response times.

B. API

The endpoint, available in AWS Lambda and accessible through AWS API Gateway is the following:

TABLE IV. Performance of the Machine Learning Models for Each Instance Type

Type	Baseline RMSE	Technique	RMSE	Best EVS	R^2	Result
t1.micro	0.049	ExtraTrees	0.025	0.741	0.741	+
m1.small	0.117	ExtraTrees	0.005	0.998	0.998	+ +
m1.medium	0.235	RandomForest	0.007	0.999	0.999	+ +
m1.large	0.473	ExtraTrees	0.063	0.982	0.982	+ +
m1.xlarge	0.936	ExtraTrees	0.024	0.999	0.999	+ +
m2.xlarge	0.389	MLP	0.374	0.079	0.079	– –
m2.2xlarge	0.933	Lasso	0.933	0	0	– –
m2.4xlarge	2.707	ExtraTrees	1.620	0.644	0.642	+
m3.medium	0.267	ExtraTrees	0.025	0.991	0.991	+ +
m3.large	0.465	RandomForest	0.098	0.956	0.955	+ +
m3.xlarge	0.927	RandomForest	0.180	0.962	0.962	+ +
m3.2xlarge	1.889	AdaBoost	0.471	0.941	0.938	+ +
m4.large	0.052	KNN	0.006	0.988	0.987	+ +
m4.xlarge	0.110	AdaBoost	0.068	0.634	0.602	+
m4.2xlarge	0.193	AdaBoost	0.067	0.899	0.871	++
m4.4xlarge	2.818	MLP	2.238	0.357	0.357	–
m4.10xlarge	11.703	MLP	7.989	0.558	0.533	+
m4.16xlarge	16.422	MLP	10.793	0.562	0.561	+
c1.medium	0.093	MLP	0.087	0.129	0.129	– –
c1.xlarge	1.723	ExtraTrees	1.212	0.503	0.498	–
c3.large	0.491	RandomForest	0.030	0.996	0.996	+ +
c3.xlarge	1.163	RandomForest	0.014	1	1	+ +
c3.2xlarge	2.117	RandomForest	0.596	0.921	0.920	+ +
c3.4xlarge	4.360	Ridge	2.106	0.762	0.762	+ +
c3.8xlarge	8.990	MLP	2.492	0.932	0.923	+ +
c4.large	0.303	Ridge	0.094	0.904	0.903	+ +
c4.xlarge	0.950	ExtraTrees	0.131	0.981	0.981	+ +
c4.2xlarge	1.122	MLP	0.439	0.847	0.844	+ +
c4.4xlarge	2.989	MLP	2.014	0.549	0.522	+
c4.8xlarge	6.106	KNN	3.412	0.674	0.674	+
x1.16xlarge	44.522	MLP	26.098	0.656	0.656	+
x1.32xlarge	104.019	MP	57.215	0.682	0.675	+
r3.large	0.478	RandomForest	0.141	0.912	0.912	+ +
r3.xlarge	1.198	ExtraTrees	0.061	0.997	0.997	+ +
r3.2xlarge	1.761	ExtraTrees	0.773	0.807	0.807	+ +
r3.4xlarge	4.375	ExtraTrees	1.578	0.869	0.869	+ +
r3.8xlarge	11.143	ExtraTrees	3.108	0.922	0.9	+ +
r4.large	0.046	Ridge	0.007	0.977	0.977	+
r4.xlarge	0.126	Ridge	0.099	0.389	0.388	–
r4.2xlarge	0.725	MLP	0.692	0.089	0.085	– –
r4.4xlarge	3.161	AdaBoost	2.670	0.321	0.285	–
r4.8xlarge	8.448	MLP	6.560	0.398	0.391	–
r4.16xlarge	26.507	MLP	13.310	0.757	0.748	+
p2.xlarge	0.116	LinearRegression	0.105	0.195	0.181	– –
p2.8xlarge	55.079	MLP	27.525	0.751	0.732	+
p2.16xlarge	86.914	MLP	57.599	0.531	0.492	–
g2.2xlarge	0.9	Lasso	0.9	0	0	– –
g2.8xlarge	16.048	MLP	11.418	0.411	0.401	–
cg1.4xlarge	2.120	RandomForest	0	1	1	+ +
f1.2xlarge	0.142	MLP	0.102	0.155	0.142	– –
f1.16xlarge	65.664	MLP	65.393	0	0	– –
i2.xlarge	2.469	RandomForest	0.802	0.894	0.892	+ +
i2.2xlarge	4.741	AdaBoost	3.224	0.598	0.530	+
i2.4xlarge	11.903	Bagging	6.743	0.673	0.673	+
i2.8xlarge	20.549	MLP	11.261	0.707	0.697	+
i3.large	0.590	ExtraTrees	0.153	0.933	0.933	+ +
i3.xlarge	1.010	RandomForest	0.323	0.898	0.897	+ +
i3.2xlarge	1.949	Bagging	0.802	0.831	0.831	+ +
i3.4xlarge	5.789	MLP	4.281	0.454	0.453	–
i3.8xlarge	17.312	ExtraTrees	6.288	0.868	0.867	+ +
i3.16xlarge	34.811	MLP	13.566	0.851	0.847	+ +
cc2.8xlarge	9.748	MLP	8.623	0.100	0.051	– –
d2.xlarge	0.242	Lasso	0.242	0	0	– –
d2.2xlarge	4.871	MLP	4.232	0.207	0.206	– –
d2.4xlarge	8.198	LinearRegression	7.021	0.263	0.262	–
d2.8xlarge	17.020	MLP	12.643	0.451	0.448	–
h1.4xlarge	12.074	MLP	6.681	0.692	0.692	+
cr1.8xlarge	15.786	MLP	2.073	0.983	0.983	+ +

The last column shows the quality of the model according to its coefficient of determination, which is directly correlated with the other quality metrics. In particular, the legend for this column is the following: –– means that the model is very poor (R^2 <0.25), – means that the model is poor ($0.25 \leq R^2 < 0.5$), + means that the model is reasonably good ($0.5 \leq R^2 < 0.75$), and finally ++ means that the model is very good ($R^2 \geq 0.75$)).

```
https://slcswaq0e2.execute-api.us-east-1.amazonaws.com
/dsawards/predict
```

This endpoint must be accessed through a POST request, with a JSON body including the following parameters:

- type: instance type (required).
- os: instance operating system (required).
- datetime: time desired for the prediction, which must be in the format "yyyy-mm-dd hh" (required).
- regions: regions for which a prediction should be returned. A list with one or more regions can be specified, and the service will return the prediction for all zones in each region. This parameter is optional and, if not specified, then all regions will be considered.

An example of a well formed body in the API call would be the following:

```
{
"type" : "c3.xlarge",
"os" : "Linux/UNIX",
"datetime" : "2017-09-13 13",
"regions" : ["us-east-1"]
}
```

Such a valid request will return a JSON object whose keys are the availability zones and the values are the estimated prices. For instance, for the previous call:

```
{
"us-east-1a" : 2.109, "us-east-1b" : 0.155,
"us-east-1c" : 0.155, "us-east-1d" : 0.155,
"us-east-1e" : 0.155, "us-east-1f" : 0.155,
}
```

Acknowledgment

Authors would like to thank Data Science Awards committee for proposing the competition.

This work has been supported by the Madrid Government (Comunidad de Madrid-Spain) under the Multiannual Agreement with UC3M in the line of Excellence of University Professors (EPUC3MXX), and in the context of the V PRICIT (Regional Programme of Research and Technological Innovation).

References

[1] Amazon Web Services, "Amazon EC2 Spot Instances Pricing." Accessed: Oct. 15, 2021, [Online]. Available: https://aws.amazon.com/ec2/spot/pricing.

[2] Amazon Web Services, "Amazon EC2 Instance Types." Accessed: Oct. 15, 2021, [Online]. Available: https://aws.amazon.com/ec2/instance-types.

[3] O. A. Ben-Yehuda, M. Ben-Yehuda, A. Schuster, D. Tsafrir, "Deconstructing Amazon EC2 spot instance pricing," *ACM Transactions on Economics and Computation*, vol. 1, no. 3, p. 16, 2013.

[4] G. Portella, G. N. Rodrigues, E. Nakano, A. C. Melo, "Statistical analysis of Amazon EC2 cloud pricing models," *Concurrency and Computation. Practice and Experience*, vol. 31, no. 18, p. e4451, 2018.

[5] M. Lumpe, M. B. Chhetri, Q. B. Vo, R. Kowalcyk, "On estimating minimum bids for Amazon EC2 spot instances," in *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, Madrid, Spain, 2017, IEEE.

[6] C. Tian, Y. Wang, F. Qi, B. Yin, "Decision model for provisioning virtual resources in Amazon EC2," in *2012 8th Intl. Conf. Network and Service Management and 2012 Workshop on Systems Virtualization Management*, Las Vegas, NV, USA, 2012, IEEE.

[7] S. Tang, J. Yuan, X.-Y. Li, "Towards optimal bidding strategy for Amazon EC2 cloud spot instance," in *2012 IEEE Fifth International Conference on Cloud Computing*, Honolulu, HI, USA, 2012, IEEE.

[8] S. Tang, J. Yuan, C. Wang, X.-Y. Li, "A framework for Amazon EC2 bidding strategy under SLA constraints," *CIEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 1, pp. 2–11, 2014.

[9] M. B. Chhetri, M. Lumpe, Q. B. Vo, R. Kowalczyk, "To bid or not to bid in streamlined EC2 spot markets," in *2018 IEEE International Conference on Services Computing*, San Francisco, CA, USA, 2018, IEEE.

[10] M. B. Chhetri, M. Lumpe, Q. B. Vo, R. Kowalczyk, "On forecasting Amazon EC2 spot prices using time-series decomposition with hybrid look-backs," in *2017 IEEE International Conference on Edge Computing*, Honolulu, HI, USA, 2017, IEEE.

[11] V. Khandelwal, A. Chaturvedi, C. P. Gupta, "Amazon EC2 spot price prediction using regression random forests," *IEEE Transactions on Cloud Computing*, vol. 8, no. 1, pp. 59–72, 2020.

[12] J. Lancon, Y. Kunwar, D. Stroud, M. McGee, R. Slater, "AWS EC2 instance spot price forecasting using LSTM networks," *SMU Data Science Review*, vol. 2, no. 2, p. 8, 2019.

[13] M. Malik, N. Bagmar, "Forecasting price of amazon spot instances using machine learning," *International Journal of Artificial Intelligence and Machine Learning*, vol. 11, pp. 71–82, 07 2021.

[14] V. Chittora, C. P. Gupta, "Dynamic spot price forecasting using stacked lstm networks," in *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, 2020, pp. 1080–1085.

[15] W. Liu, P. Wang, Y. Meng, C. Zhao, Z. Zhang, "Cloud spot instance price prediction using knn regression," *Human-centric Computing and Information Sciences*, vol. 10, no. 1, p. 34, 2020.

[16] Kaggle, "AWS Spot Pricing Market Dataset." Accessed: Oct. 15, 2021, [Online]. Available: https://www. kaggle.com/noqcks/aws-spot-pricing-market.

[17] Western Sydney University, "Spot Price Archive." [Online]. Available: http://spot.scem.uws.edu.au/ ec2si.

[18] B. Javadi, R. Thulasiram, R. Buyya, "Statistical Modeling of Spot Instance Prices in Public Cloud Environments," in *4th IEEE/ACM International Conference on Utility and Cloud Computing*, Melbourne, Australia, 2011, 2011, IEEE.

[19] Amazon Web Services, "EC2-Boto 3 Docs." Accessed: Oct. 15, 2021, [Online]. Available: http://boto3.readthedocs.io/en/latest/ reference/services/ec2.html#EC2.Client. describe_spot_price_history.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

Alejandro Baldominos

Alejandro Baldominos holds a Ph.D. in Computer Science and Technology by Universidad Carlos III de Madrid, where he is currently working as a researcher in the Evolutionary Algorithms, Neural Networks and Artificial Intelligence group. His current research line involves the application of evolutionary computation to the evolution of the topology of deep neural networks. Additionally, he has also published several papers in journals and international conferences regarding the application of machine learning to diverse real-world fields.

Yago Saez

Yago Saez received the degree in computer engineering in 1999. He got his Ph.D. in Computer Science from the Universidad Politécnica de Madrid, Spain, in 2005. Since 2007 till 2015 he was vice-head of the Computer Science Department from the Carlos III University of Madrid, where he got a tenure and is nowadays associate professor. He belongs to the Evolutionary Computation, Neural Networks and Artificial Intelligence research group (EVANNAI) and member of the IEEE Computational Finance and Economics Technical committee.

David Quintana

David Quintana holds Bachelor degrees in Business Administration and Computer Science. He has an M.S. in Intelligent Systems from Universidad Carlos III de Madrid and a Ph.D. in Finance from Universidad Pontificia Comillas (ICADE). He is currently Associate Professor at the Computer Science Department at Universidad Carlos III de Madrid. There, he is part the bio-inspired algorithms group EVANNAI. His current research interests are mainly focused on applications of Computational Intelligence in finance and economics.

Pedro Isasi

Pedro Isasi is Graduate and Doctor in Computer science by the Polytechnic University of Madrid since 1994. Currently, he is University professor and head of the Evolutionary Computation and Neural Networks Laboratory in the Carlos III University of Madrid. His research is centered in the field of the artificial intelligence, focusing on problems of Classification, Optimization and Machine Learning, fundamentally in Evolutionary Systems, Metaheuristics and artificial neural networks.

Comparative Analysis of Building Insurance Prediction Using Some Machine Learning Algorithms

Chukwuebuka Joseph Ejiyi, Zhen Qin*, Abdulhaq Adetunji Salako, Monday Nkanta Happy, Grace Ugochi Nneji, Chiagoziem Chima Ukwuoma, Ijeoma Amuche Chikwendu, Ji Gen*

University of Electronic Science and Technology of China, Chengdu (China)

Received 8 October 2021 | Accepted 2 February 2022 | Published 17 February 2022

LA UNIVERSIDAD
EN INTERNET

Abstract

In finance and management, insurance is a product that tends to reduce or eliminate in totality or partially the loss caused due to different risks. Various factors affect house insurance claims, some of which contribute to formulating insurance policies including specific features that the house has. Machine Learning (ML) when brought into the field of insurance would enable seamless formulation of insurance policies with a better performance which will also save time. Various classification algorithms have been used since they have a long history and have also got some modifications for optimum functionality. To illustrate the performance of each of the ML algorithms that we used here, we analyzed an insurance dataset drawn from Zindi Africa competition which is said to be from Olusola Insurance Company in Lagos Nigeria. This study therefore, compares the performance of Logistic Regression (LR), Decision Tree (DT), K-Nearest Neighbor (KNN), Kernel Support Vector Machine (kSVM), Naïve Bayes (NB), and Random Forest (RF) Regressors on a dataset got from Zindi.africa competition and their performances are checked using not only accuracy and precision metrics but also recall, and F1 score metrics, all displayed on the confusion matrix. The accuracy result shows that logistic regression and Kernel SVM both gave 78% but kSVM outperformed LR in precision with a percentage of 70.8% for kSVM and 64.8% for LR showing that kSVM offered the best result.

Keywords

Claim, Insurance, Machine Learning, Prediction, Regressors.

DOI: 10.9781/ijimai.2022.02.005

I. Introduction

Insurance is described as a risk management strategy used to hedge against the risk of accidents. Usually, the underwriter provides the insurance while the policyholder buys the said insurance. The policyholder receives an insurance policy with specifications on the circumstances and conditions under which the underwriter will give a stipulated or required compensation to the insured as they continue to be in good standing with the insurance company (that is, they pay their premium). If there is an experience of a loss by the policyholder which is potentially included in the insurance policy, he/she puts forward or files a claim to the underwriter [1], [2]. People have many reasons for insuring their property, but one of the major and basic reasons for insurance is usually because it gives a sense of safety and security

Lagos State, one of the states in Nigeria that is popular for commerce and industry has an insurance company named Olusola Insurance Company (OIC). Being among the most renowned and famous insurance companies available in Lagos state that is into building insurance and not only that but also being one of the oldest. The recent recurrent collapse of buildings in the state has become a big concern to the landlords as well as this insurance company because of their insurance policy. Therefore, the OIC sought a way to be able to theorize by prediction if a policyholder will file a claim in case of the collapse of his building or not and if such an individual will be qualified for the insurance. They made available the dataset collected differently from various sources at different times at the site of Zindi Africa. We analyzed the data for the prediction of claims or no claims with respect to the insurance. We have the task to design and build a model that is applicable for prediction with which it could be determined if a particular building will have or is supposed to have an insurance claim during the specified period or not. Since machine learning algorithms such as LR, RF, kSVM, KNN, NB, and DT can be used, we compared them with the motive of finding out the best predictive algorithm with respect to this data made available by OIC.

Each of the classification algorithms has been used for various predictions. [3], [4], [5] among others used logistic regression only and have seen it to be very efficient. [6], [7], [8] are some of the works that employed the prediction algorithm the Random forest and it was quite good in performance. [9], [10], [11] are some of the researchers that used the Decision tree for their prediction. For example, Amra and Maghari analyzed students' performance using KNN and Naïve Bayesian algorithms and discovered that Naïve Bayes with an accuracy of 93.2% performed much better than KNN with an accuracy of 63.5% [12].

For this comparison, we used Exploratory Data Analysis (EDA) first to help find out the underlying pattern, discover and spot irregularities, frame the possible hypothesis, and check assumptions with the aim to find a good fitting model. The model will then be anchored on the

* Corresponding author.

E-mail addresses: qinzhen@uestc.edu.cn (Z. Qin), jgeng@uestc.edu.cn (J. Gen).

specified building characteristics. After which the target variable, the claim – that is, if the building in question has a minimum of one claim during the said insured period or if it does not have. After the EDA, the data is preprocessed to make the data "parseable" by the machine. Feature selection and then model training and evaluation followed. We used LR, RF, kSVM, KNN, NB, and DT algorithms for the regression analysis, having in mind that the accuracy will be checked, to ascertain the performance of each regressor we used a confusion matrix and determine the best predictive algorithm among them and then compared their performance.

The other parts of this paper have been organized thus: Section II presents the related works for all the models for prediction. The background or what may be called overview was covered in section III. The model design and experimental analysis were the focus of section IV. The result and discussion were covered in section V before the paper was concluded in section VI.

II. Related Works

Bhat and Gandhi reported that the ANN-based framework is gaining popularity since it performs well in prediction [13]. They implemented a normalization technique to improve the performance of ANN and also used the analysis of correlation coefficient to find and determine the best input to the ANN framework and gave a better performance than the statistical model. Various regression techniques have been proposed which can be used for prediction apart from Logistic regression. In [14], Multiple Linear Regression (MLR) is said to be a kind used in which more than one attributes are for prediction. In [15], [16], Ridge regression (RR) and LASSO regressions (LR) were used; here, Ridge regression does the regularization of the regression coefficient while LASSO Regression uses L1 penalty, being the only thing that differentiates it from RR. In [17] a regression form called the Elastic Net (ER) was used as a penalization method. Mislan *et. al.* initiated a Back Propagation Neural Network model with double hidden layers for rainfall prediction [16]. In [18], the NB classification algorithm was used for classification before prediction. In [19] - [20] used Artificial Neural Network theory for prediction. The Linear Regression model was also used in [20], [21].

Some other tools employed for prediction by other authors are as follows; Geographically Weighted Regression. In [22], Bayesian Linear Regression [23]. Support Vector Machine Regression (SVMR) was used by Paniagua Tineo *et. Al.* [19] to predict daily maximum temperature. From their work, the SVMR algorithm was concluded to be able to produce accurate temperature predictions after 24 hours. Onan [24] identified web classification as a research direction with great importance in data science. The author presented different feature selections as well as four different ensemble learning methods on the basis of four different base learners(NB, KNN, C4.5 algorithm together with FURIA algorithm). The author concluded that in web page classification, ensemble learning, as well as feature selection, has the capacity to enhance the predictive ability of classifiers. In another work by Onan *et al* [25], they reported that in order to reduce the training time and also develop a robust and efficient classification model, feature selection is necessary. [26] and [27] achieve good performance of classification on language using various classification models employing feature selection methods that are suitable for such classifications.

Many other algorithms have been employed and several other models proposed and tested for various predictions as reported by [28]. Abhishek *et. Al.* [29] proposed and implemented a model that predicted temperature using a backpropagation neural network. Shobha *et. Al.* [30] used a clustering-based analysis approach to monitor whether based on meteorological data. The authors studied data from agricultural meteorological patterns collected from the Meteorological Centre of Bengaluru district and determined very important agricultural parameters like minimum temperature, maximum temperature, relative humidity, rainfall, and pan evaporation using not only K-Means but also hierarchical clustering which are extremely critical for agriculture. Gill *et. Al.* [31] proposed a model with a backpropagation employing a genetic algorithm. From their work, it became observable that genetic algorithms can be effectively used for prediction alongside backpropagation neural networks. Each work employed EDA to enable better performance of each algorithm that was used.

A lot of things have been predicted, like rainfall, weather, etc. using various algorithms. Paniagua-Tineo *et. Al.* [32] worked on maximum daily temperature prediction by employing Support Vector Machine Regression (SVMR). It was concluded by them, that the SVMR algorithm can produce accurate temperature prediction after 24 hours. Abdel-Aal *et. Al.* proposed an abductive networks approach for the prediction of temperature on an hourly basis [33], which was able to predict the temperature after an hour and also after a day. A modified type of Support Vector Machine (SVM) called Multi-view Least Squares (LS-SVM) regression for black-box temperature prediction was proposed by Houthuys *et. Al.* [34]. The goal of multi-view LS-SVM is to improve the model's performance by taking information from all views into account as there is an appreciable number of observations in black-box weather forecasting.

Prediction of short-term wind power with the aid of empirical mode decomposition-based GA-SYR was experimented by Xie *et. Al.* [35]. First, the wind speed data from NWP is decomposed into the EMD components, including multiple intrinsic mode functions (IMFs) as well as one residue. Thereafter, a Genetic Algorithm Support Vector Regression model (GA-SVR) is used to build models of all components. Similarly, another method referred to as short-term wind power forecasting was implemented by Peng *et. Al.* [36] using numerical weather prediction and error correction methods. [37] Zhang *et. Al.* used SVM by training multiclass predictors and adjusting SVM parameters using Particle Swarm Optimization (PSO). In [38] Papantoniou *et. Al* utilized data obtained from about four European cities to display or introduce the implementation and then evaluation of different neural network-based algorithms used in identification.

Da-Chun Wu *et al* [39] used ANN to forecast or predict air compressor load of different compressors at different times and under different conditions. They also investigated the prediction of the electrical demand peak with ANNs and SVM and discovered that integration of ANNs to SVM gave a significant improvement to the accuracy of the prediction. Priyadarshini Patil *et al* [40] compared the performance of SVM, RF, and ANN in potato blight disease and discovered that the ANN performed better than SVM and RF. From their experiment, ANN gave an accuracy of 92% which was better than 84% and 79% respectively of SVM and RF.

Xiaohu T *et al* [41] used the DT algorithm to predict the winning team in the Chinese super league and got an accuracy of 57.7% when other factors are put into consideration. In another prediction, Nwulu [42] used DT to predict the price of crude oil from data gathered which covered about 24 years. According to his work, DT outperformed other models that he used and had a less computational period. They [43] basically used DT to predict churn as well as KNN and DT gave an accuracy of about 93% which is a good accuracy [43].

Raj *et al* [44] concluded from their comparison that SVM has better performance (accuracy of 82%) when compared to Naïve Bayes (62.5%) when they compared the SVM and NB classifiers used in diabetes prediction. Bayindir *et al* used an NB classifier to predict the daily energy generated from an installed photovoltaic system [45] and an accuracy of 82.2% was obtained.

Salim *et al* predicted the timely graduation of students in Indonesia using KNN because of KNN's robustness on noisy data and its ability to train on a large dataset [46]. With the advent and rise of the technology of machine learning, it has been used in the prediction of flight delays as well [47]. Machine learning algorithms that have been used for this kind of prediction include random forest, decision tree, logistic regression, SVM as well as K-nearest neighbor algorithms [48], [49], [50].

Linear regression has been identified as the basic regression model that has been used for prediction, it takes into account the variation that exists between the variables called independent and those that are dependent also according to [51]. These researchers [51] compared the regression models which will have the capacity to predict graduate admission and found out that linear regression outperformed other models on their dataset. The models compared include Linear regression, Support Vector Machine, Decision Tree Regression, and Random Forest regression. Linear Regression gave an output of 0.00480149 for MSE and 0.72486310 for R2 which is the best among the models [51].

In all these, limited literature was found with respect to insurance prediction. And with the growth in the need for the insurance of properties such as cars, houses, and others, data is needed to be able to build models that will be useful for insurance prediction.

III. Background

Machine Learning (ML) has over time been viewed as a branch or more precisely a subcategory of Artificial Intelligence (AI) that learns computer algorithms and improves via experience [48], [52]. It has grown and become very popular, it has also found usefulness in many fields not occasionally but on daily basis [49], [53]. ML uses training data or sample data to build models which they use to make decisions or predictions in this case without further programming. So ML gives the system all it needs to learn and understand by itself and consequently gives or makes a prediction for the unknown outputs [50]. Machine learning algorithm has its performance depends on the training success, dataset availability, data preprocessing, selection of attributes among others.

Regression analysis is primarily used for two conceptually distinct purposes. First, for prediction as well as forecasting, where its application has a remarkable connection with the field of ML. which second identified usefulness, in some situations, to hypothesize the underlying relationships that exist between the variables that are independent and those that are dependent. It is very important to note that regressions by themselves only bring to light the relationships that are found between a variable that is dependent and another set that is a collection of variables that are independent and in a fixed dataset. To use regressions for prediction or to hypothesize causal relationships, respectively, a researcher must ensure to carefully bring to light the reason why the existing relationships are assumed to have predictive power for a new context or why a relationship between two variables is thought to possess a causal interpretation. The latter is considered crucially important especially when researchers hope to make an estimation of the causal relationships using observational data [54]. The techniques used in the study are introduced in the following subsections.

A. Artificial Neural Networks (ANNs)

Artificial Neural Networks (ANNs) originally inspired by the nervous system of animals, process data in a manner similar to the brain of mammals. They have the capacity to forecast difficult problems. by computationally learning a set of input data and consequently giving out an output that is desirable. ANN is usually defined as a framework instead of an algorithm because it acts as the basis for many machine learning algorithms [55], for complex data (input) processing. It has got many applications in different fields and so we are applying it to this dataset for prediction.

ANN has been identified as a tool that has found usefulness for both regression and classification since it can model systems that have a non-linear relationship [56]. One peculiar thing about ANN is that it is structured such that after training, it has the capacity to give outputs that are reliable and will do that quite fast even if the data is noisy or in cases where some information is missing from the data [57]. ANN in its structure has hidden layers that can grow depending on the network size, this growth helps the network memorize more of the data but it increases the training time [58].

B. Linear Classifier

Linear Regression (LR) is estimated with the Maximum Likelihood Estimation (MLE) approach and provides constant output. The MLE is just"a "likelihood" maximization method which is a function that measures the parameters that seem likely to give the observed data and accepts a joint probability mass function. Statistically speaking, MLE sets the mean and variance as parameters in measuring the particular parametric values for a given model. This set of parameters can be adopted to predict the data required in a normal distribution and also accepts a joint probability mass function [59]. A function known as the logistic function or sigmoid function shown in (1) gives an "S" shaped curve which takes any real number value and maps the number into the interval of 0 and 1. The predicted value becomes 1 (one) in case the predicted values go to positive infinity and 0 (zero) if it is negative infinity. Logistic regression is majorly used for classification but is also useful in solving regression problems since it shows good performance.

$$f(x) = \frac{1}{1+e^{-x}} \tag{1}$$

C. Decision Tree Classifier

Decision Tree (DT) is considered one of the easiest and most popular classification algorithms to learn and interpret. It can be applied in solving classification and regression problems. The decision tree looks more like a flow chart that has the structure as a tree would have in which the features are represented with internal node, while the branches and the leaf node represent the decision rule and the leaf node respectively. But unlike the normal tree where the root is at the base, the uppermost part of the decision tree is where the root node is located. It learns to partition using the attribute values in a recursive manner otherwise called recursive partitioning. The decision tree as the name implies helps in decision making. Its complexity is a function of the number of attributes and records in the dataset [60]. The decision tree makes predictions in a tree-like manner just as people would make decisions when faced with certain challenges especially when there are two options to decide from. When one option is taken, you may have to make other choices based on the one you have decided on and it continues until the result hoped for or expected is got. This is basically the framework of the decision tree [41] and classification is based on characteristics. The algorithm is relatively easy to implement because of how easy it is to understand, and according to Xiaohu T *et al* [41], it is applicable for data analysis and forecasting. Xiaohu T *et al* [41] have also earlier reported that DT is based on instances and that it is referred to as an "inductive learning algorithm." DT has also been shown to comprise of some major steps which are feature selection, generation of the decision tree and finally pruning of the tree. Some of the identified advantages of DT are easy calculation and workload, simple and easy to understand, interpret, analyze, and a high degree of accuracy [61]. The basic and fundamental idea behind the decision tree algorithm has been identified to be recursive partitioning which is a statistical technique for the analysis of multivariable [42].

Like in the Agricultural point of view where trees are more popular and pruning is done for more productivity, pruning is also usually applied to DT to improve the algorithm's performance [42]. The noise which is one of the characteristics of raw data is said to be easily managed by the DT algorithm [43] because DT has the ability to avoid overfitting by pruning.

D. Random Forest Classifier

Random Forest (RF) on the other hand is also a supervised learning algorithm also used for both regression and classification. Just as the name (forest) suggests, it is supposed to be composed of many trees. The robustness of the forest will then be a function of the number of trees in it. RF functions by creating decision trees on selected data samples on a random basis. Then it usually gets a prediction from every tree and selects the solution that is the best by voting. By this, it provides a very nice indicator of the important features. So, a random forest algorithm is seen as a collection of many decision tree classifiers, each decision tree is got using an appropriate characteristic selection gauge like information gain. Individual trees are dependent on an independent random sample and each tree votes the most frequent class, unlike the regression where the mean of the entire tree results is assumed to be the final output. The random forest has been proven to offer a good feature selection indicator [62]. Its functioning is divided into the following steps; Samples are selected randomly from a given dataset, a decision tree is then constructed for each sample, getting a prediction from individual trees. After that, a vote is performed for each predicted output or result and then the prediction with the highest voted is selected as the final prediction [63].

In all these, Random forest differs from decision tree in the following ways; Random forest keeps from overfitting whereas Decision tree may run into it. It, therefore, follows that RF manages the challenge of overfitting more than DT. Random forest is composed of multiple decision trees, although the Decision tree is faster computationally. It is not difficult to infer that the Decision tree is easier to interpret when compared to the Dandom forest.

E. K-Nearest Neighbor Classifier

Here in K-Nearest Neighbor (KNN), an unknown data point is categorized into its nearest neighbor which is already defined and determined. The nearest neighbor is figured out by k-value which works out the actual neighbors and at the same time the classes that belong to a particular data point [12]. On some occasions, it requires not only one nearest neighbor to ascertain the particular datapoint's class. In KNN it is usually necessary for data points to be in memory at runtime, it is also called "memory-based technique" [12]. There were some improvements proposed by some researchers on the pioneer KNN but the computational complexity and memory requirements have remained unchanged. Nevertheless, the memory requirements can be managed well if there is a reduction in the size of the dataset used, thereby reducing comprehensively the repeated training sample pattern. Some data points that are perceived to have no effects on the result are the ones that are more advisable to remove. The nearest feature line, ball tree, tunable metric, k-d tree, principal axis search tree, and orthogonal search tree are some of the algorithms that have been identified to bring increment to the speed of KNN [12].

F. Naïve Bayes (NB) Classifier

Naïve Bayes (NB) as a machine learning algorithm is designed in such a way that it can accomplish classification tasks. It has been reasoned that its popularity is credited to the fact that it can be written into code quite easily with less time. NB has also been identified as an algorithm that can be implemented in real-time prediction and organizations find it very useful in bringing quick answers to users' request(s). NB classifier is basically anchored on the theorem proposed by Bayes, as such, it is seen as a conditional probabilistic classifier [64]. It is also denoted as the Generative learning model on some occasions [65]. The algorithm is called Naïve Bayes because the existence of a particular feature does not depend on the existence of another feature which is the same principle in conditional probability [44] and it is very helpful for a very large dataset classification. NB performs well and is said to be most suitable for data with high dimensionality [12]. Some of the real-world scenarios where NB has found application are in Recommendation System, Real-Time Prediction, Multiclass prediction, Sentiment Analysis, Text Classification, and the popular Spam Filtering.

The general principle of NB hinges on conditional probability and understanding it is paramount to understanding the NB algorithm.

G. Support Vector Classifier

Support Vector Machine (SVM) is linear naturally [66] and is known to support Linear regression as well as non-linear regression, this feature made it possible for SVM to be referred to as Support Vector Regression as well. As a model which is classified as a supervised machine learning model, it can to make predictions from the earlier learned data. It has been suggested that SVM gives good results when the dataset to analyze is not much. Support Vector Machine a discriminative classifier is expressed by a separating line or hyperplane. Given training data, it outputs a hyperplane that categorizes the new data set. Kernel Support Vector Machine tends to discover the best hyperplane that separates data in a Hilbert space. This best hyperplane is selected to maximize the margins between the classes (usually two, since Kernel SVM is a binary classifier). The kernel functions [66] of the SVM are activated or brought into the light so as to make the linearity of the SVM classifier got by dot product to nonlinearity. The Kernel SVMs allow the hyperplane's extreme margin to adapt in a feature space that has been transformed and has the advantage of taking care of classification difficulties over conventional SVM [67]. SVM aggregates the two classes that are to be classified using support vectors which are the extreme data points separated by hyperplane [44]. In this context, the hyperplane stands as the classification between the objects.

IV. Model Design and Experimental Analysis

A. Model Design

Our work looks at the comparison of the following machine learning algorithms Logistic regression, Decision Tree, Kernel SVM, Random Forest, Naïve Bayes, and K-Nearest neighbor. The basic information of the algorithms and the principles of how they function have been discussed in the previous section. The algorithms were given the same input data which have the same ratio of training and testing data. The data were preprocessed as explained in section IV*B* to reduce noise and undesirable characteristics before they were divided randomly into training and testing sets. The training set was used for the training of the algorithm and the performance of the algorithms tested with the testing sets.

The diagram above Fig. 1 is the flowchart of the model we used for this work.

B. Dataset

The dataset we used was got from zindi.africa [68] the data comprises collected information from 2012 to 2016 which are the years of observation. It was said that for the period of observation, the dataset was collected by various people. The variables in the dataset with their descriptions are shown in Table. I.

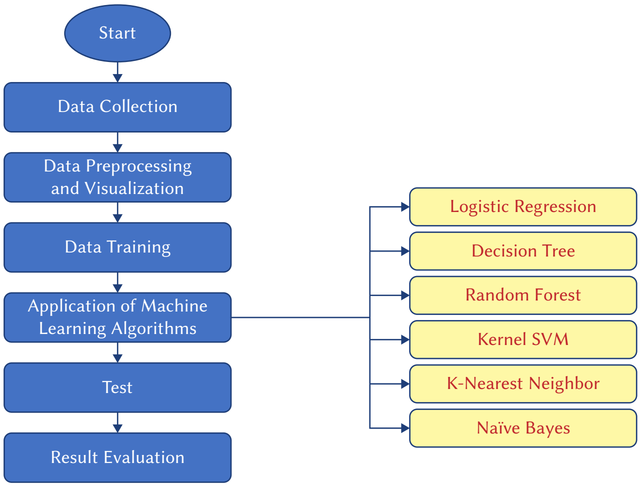


Fig. 1. Block diagram of the model.

TABLE I. Variables and Their Description

No.	Variable	Description
1	Customer ID	Identification number for the Policyholder
2	Year of observation	Year of observation for the insured policy
3	Insured period	Duration of insurance policy in Olusola Insurance (Eg: Full-year insurance, Policy Duration= 1; 6 months = 0.5
4	Residential	Whether the building is residential or not
5	Building painted	Whether the building is painted or not painted (N-Painted, V-Not Painted)
6	Building fenced	Whether the building is fenced or not fenced (N-Fenced, V-Not Fenced)
7	Garden	Whether the building has at least one garden or not (V-has garden; O-no garden).
8	Settlement	The area where the building is located. (R-rural area; U- urban area)
9	Building dimension	Size of the insured building in m^2
10	Building type	The type of building (Type 1, 2, 3, 4)
11	Date of occupancy	Date or Year the building was first occupied
12	Number of windows	Number of windows in the building
13	Geo-code	Geographical location code of the insured building
14	Claim	Target variable. (0: no claim, 1: at least one claim over the insured period).

C. Tools Used

The following tools categorized as python libraries were used for the implementation of the algorithms: The Seaborn that helps with the generation of heatmaps, the Scikit learn/sklearn which helps to ensure that the algorithms are implemented, the Pandas helps in operations that are data-related while the Matplotlib is for plotting of the various required plots. The Jupyter notebook was also used for the writing of the python codes which were consequently executed on Keras/ TensorFlow framework.

D. Data Preprocessing and Visualizations

After data collection, the next step is data preprocessing followed by Data Integration, Data Transformation, and then reduction [69]. Data preprocessing is an essential and basic step in the process of knowledge discovery; because the data obtained from the logs may be incomplete, noisy, or inconsistent [70]. The most promising attributes of quality data include completeness, consistency, and timeliness. The performance of a mining algorithm depends on the quality of the data. But, the real-world data is incomplete and uncertain. The incompleteness of the data can be easily identified and its elimination is sometimes acceptable. But, identification of the inconsistencies in the data is very difficult and even a very negligible amount of inconsistency in data degrades the performance of the mining algorithm at a very high rate. The existence of inconsistencies in the training data affects the performance of the mining algorithm and the removal of such inconsistencies improves the performance of the mining algorithm [70].

Data visualization, on the other hand, aims to transmit the data clearly and effectively via graphical representation. Data visualization has found extensive use in many scenarios like reporting at work, task-progress tracking among others. One of the most popularly used advantages of the visualization technique is in discovering data relationships that may be hard to identify or observe by merely looking at the raw data [71]. These days, people have also used data visualization to design funny and interesting graphics.

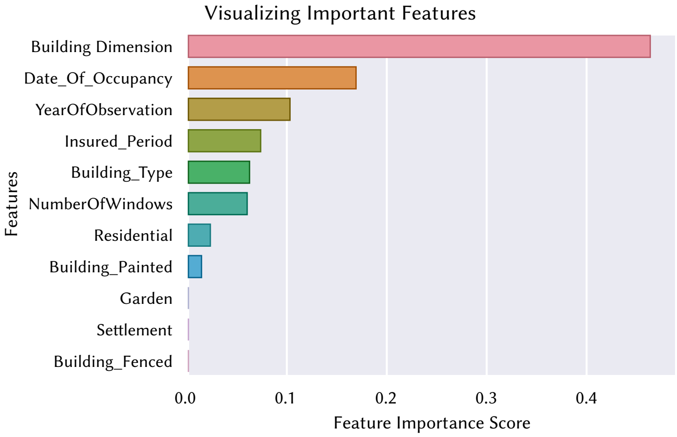


Fig. 2. Feature Importance score.

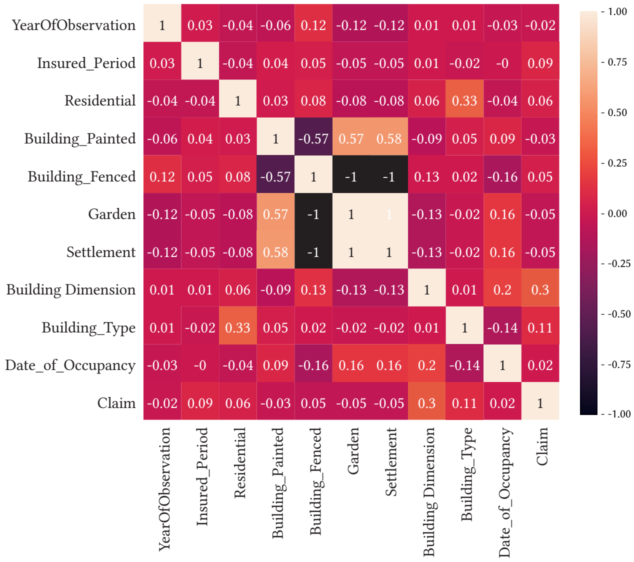
We generated feature importance of the train data using Gini importance and Fig. 2 above shows the importance from the highest to the lowest. And the correlation matrix is shown in Fig. 3.

Since we have to design a system that will predict the probability of having at least one claim over the period that the building was under insurance. The model will be accessed based on the features of the building. The target variable, in this case, being claim:

- 1 if the building in question has at least one claim over the said period of insurance.
- 0 if the said building does not have a claim over the said period of insurance.

The data collected from the site was then preprocessed. The data were divided into a ratio of 7:3 training (70%) and testing (30%). Only the decision tree and the random forests were tested with 20% of the dataset and trained with 80% of them. The dataset was given a good description of the variables for each of the columns. Selection of the data or division of the dataset was done randomly and the models' performance was checked afterward.

Since the data is noisy, we use methods of dealing with noisy data to fill in the data. From the data set both of the training and testing, 4 of the variables contain noisy data (null values). Those variables are; Garden, Building dimension, Date of occupancy, and Geo-code for both the training and testing dataset. The data were normalized and the vacant data was filled with modal values for the variables Geo-code and Garden, and mean value for the Building dimension and Date of occupancy.



	YearOfObservation	Insured_Period	Residential	Building_Painted	Building_Fenced	Garden	Settlement	Building Dimension	Building_Type	Date_of_Occupancy	Claim
YearOfObservation	1	0.03	-0.04	-0.06	0.12	-0.12	-0.12	0.01	0.01	-0.03	-0.02
Insured_Period	0.03	1	-0.04	0.04	0.05	-0.05	-0.05	0.01	-0.02	-0	0.09
Residential	-0.04	-0.04	1	0.03	0.08	-0.08	-0.08	0.06	0.33	-0.04	0.06
Building_Painted	-0.06	0.04	0.03	1	-0.57	0.57	0.58	-0.09	0.05	0.09	-0.03
Building_Fenced	0.12	0.05	0.08	-0.57	1	-1	-1	0.13	0.02	-0.16	0.05
Garden	-0.12	-0.05	-0.08	0.57	-1	1	1	-0.13	-0.02	0.16	-0.05
Settlement	-0.12	-0.05	-0.08	0.58	-1	1	1	-0.13	-0.02	0.16	-0.05
Building Dimension	0.01	0.01	0.06	-0.09	0.13	-0.13	-0.13	1	0.01	0.2	0.3
Building_Type	0.01	-0.02	0.33	0.05	0.02	-0.02	-0.02	0.01	1	-0.14	0.11
Date_of_Occupancy	-0.03	-0	-0.04	0.09	-0.16	0.16	0.16	0.2	-0.14	1	0.02
Claim	-0.02	0.09	0.06	-0.03	0.05	-0.05	-0.05	0.3	0.11	0.02	1

Fig. 3. Figure Correlation matrix.

The input parameters used for the experimental phase and purpose are shown in Table II. We also included the k-fold cross-validation strategy used. We used the k-fold with the best performance for each algorithm. For other models, we used 10 but the decision tree and the random forest seemed to be overfitting with the same value of k-fold, as such we used 5 for them.

TABLE II. Input Parameters

Model name	K-fold	Input parameters

Decision Tree	K = 5	Criterion = entropy
		Splitter = best
		Random_state = 0
		Max_dept = none

KNN	K = 10	Number of neighbours = 5
		Weights = uniform
		Leaf-size = 30
		Algorithm = Kd tree

Logistic Regression	K= 10	Tolerance = 1e-4
		Class weight = balanced
		Solver = Newton cg
		Random state = 0

Kernel SVM	K=10	Kernel = RBF
		Random_state = 0
		Tolerance = 1e-4
		Class weight = balanced

Naïve Bayes'	K=10	Type = Gaussian NB
		Priors = none
		Var_smoothing = 1e-9

Random Forest	K= 5	Number of trees = 10
		Criterion = entropy
		Random_state = 0

E. Performance Evaluation Criteria

The basis for the comparison of these algorithms anchors on some known performance criteria some of which are outlined below with a short description for each of them. We choose these criteria because they are popular and easy to understand in addition to the fact that they can be applied to all the algorithms for easier comparison.

Accuracy: This gives the estimate of the percentage of the actual rate values of all classes. A higher accuracy shows better performance (higher is better). Accuracy simply put is a ratio of appropriately predicted observation to the total observations. The formula for the calculation of accuracy is shown in (2).

$$\text{Accuracy} = (TP + TN) / (TP + FP + FN + TN) \quad (2)$$

Precision: This determines the percentage rate of the true positive values for the relevant elements to the irrelevant ones. As with accuracy, higher the precision percentage, higher relevant results were retrieved than the irrelevant ones (higher is better). The formula for the estimation of precision is shown in (3).

$$\text{Precision} = TP / (TP + FP) \quad (3)$$

Recall: Also referred to as Sensitivity Measure or True Positive Rate is seen as the fraction of a true positive rate of relevant values. A higher ratio means higher retrieved relevant elements (higher is better). The method used for the estimation of recall is shown in (4).

$$\text{Recall} = TP / (TP + FN) \quad (4)$$

F-Measure (or F1 score): It is a weighted mean of both precision and recall. The upper threshold value of the F1 score is usually 1, which stands for the best score, and the lower threshold value 0, which means the worst score (higher is better). The formula for the estimation of the F1 score is shown in (5).

$$\text{F1Score} = 2^* (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision}) \quad (5)$$

Confusion Matrix

The result from the matrix permits us to have more detailed information about the results for the algorithms used, by making available the number truly predicted positive and negative from the results.

True Positives (TP): These refer to the correctly predicted values that are positive, meaning that the value of the class is actually yes and the model predicted the class also as yes. In this case, the model predicted that a house has a claim and it actually does. The True Positive values estimated from the models we used are shown in Table IV.

True Negatives (TN): These refer to the predicted negative values that were predicted correctly, meaning that the value of the class is no actually and the model predicted the class also as no. In the case of our study, the model predicted no claim when the house does not have any claim. The True negative values estimated from the models we used are shown in Table IV.

False Positives (FP): This is sometimes called Type I error and occurs when the class is no in the real sense while the model predicted it as yes. Taking inference from our study, if the model predicts a claim when it is supposed to be no claim. The False Positive values estimated from the models we used are shown in Table IV.

False Negatives (FN): This one is sometimes called Type II error and occurs when the class is yes in an actual sense but the model predicted it as no. When viewed in the sense of our study, when a model predicts no claim but it is supposed to have a claim. The False Negative values estimated from the models we used are shown in Table IV.

Precision and Recall can all be calculated with the above-got parameters as shown from equations (2) to (5).

V. Result and Discussion

This work looked at the capability of the aforementioned machine learning algorithm's ability to predict claim or no claim on the insurance according to the provided dataset from Olusola Insurance Company. The used input parameters are displayed in Table II and the results in Tables III and IV.

The data was collected by different sources or people according to [68] and at different points. That made the data have some abnormalities, like missing information. This is so because the span of the year of the collection was a bit long and some data may not have been considered important at some point. As time went by, some became relevant and more emphasis was laid on them. Also, some people may not be willing to give some details or may have forgotten some data which might have amounted to the missing of these data. Not minding the cause of the missing data, during the data preprocessing these were taken care of as explained in section IV*B*. The dataset was split into two for training and testing in the ratio of 7:3 for most of the models (KNN, KSVM, LR, and NB) and 8:2 for DT and RF because of the k-fold value used for them.

TABLE III. Model Results

Model name	Accuracy	Precision	Recall	F1
Decision Tree	0.680	0.311	0.329	0.320
KNN	0.757	0.444	0.257	0.325
Kernel SVM	0.788	0.708	0.123	0.210
Logistic Regression	0.788	0.648	0.158	0.254
Naïve Bayes’	0.773	0.507	0.278	0.359
Random Forest	0.756	0.444	0.278	0.342

From Table III above, the Logistic Regression and kSVM both gave an accuracy of 78.8% which is better than others. For a better assessment, the precisions are considered in which kSVM gave a better performance than logistic regression. But both in the recall and F1, LR has relatively higher performance than kSVM. Although NB has a very close accuracy (77.3%) to kSVM and LR and also close precision value to them too, its recall was joint second to the best, and it produced the best F1 score. The model evaluation table shows explicitly that kSVM gave a better performance than the other models.

The logistic regression model performed well possibly because of its ability to learn and perform optimally if the value of the variable is categorical and requests binary output as in this case. The kSVM also performed well too although the data available for learning can be considered small because kSVM gives a good performance and optimally for cases that are linearly separable.

We will take a good look at the confusion matrix which is a convenient presentation of the measured accuracy of a model with two or more classes. Table III presents the values for the components of the confusion matrix. For each of the models.

The kSVM with a true positive value of 1243 gave the highest number of true positives, representing about 80% of the data. It is quite closely followed by LR with 1230 representing about 76% of the data. Although their true negative values were quite low when compared to others, they are seen to have outperformed the other algorithms with kSVM giving the best prediction which is meticulously followed by LR. We have noted above the possible reason(s) for the high performances of the kSVM and LR.

TABLE IV. Confusion Matrix Table

Model name	True Positive (TP)	False Positive (FP)	False Negative (FN)	True Negative (TN)
Decision Tree	990	272	251	123
KNN	1142	120	278	96
Kernel SVM	1243	19	251	46
Logistic Regression	1230	32	315	59
Naïve Bayes’	1161	101	270	104
Random Forest	1132	130	270	130

In this section too, we tend to give a broader view of the models used using SHAP (SHapley Additive exPlanations) [72] value is an emergence from the concept of Shapeley and it works at increasing the transparency of models. The Shapely value tends to construct an additive explanation model considered as “contributors.” [47] the predictor or model generates a value of prediction for every sample while the SHAP value is described as the value given to every feature in the sample. In any case, where the SHAP value indicates negative, it is a pointer that the feature influences the prediction by lowering the predicted value; but in the case where it is positive, it shows that the influence of that feature will bring an increase to the predicted value in the prediction. When the baseline or the expected value which is the mean of the output of the model over the training data is estimated, the predicted value becomes the sum of this obtained base line and the contribution value of every feature available in the sample.

Using the SHAP, the random forest predictor was plotted in Fig. 6. From the SHAP value plot, we decipher the positive as well as the negative relationships that exist between the predictor and the target variables. From the figure, the target variables are on the y-axis and the impact on the x-axis shows that the building dimension is the variable with the highest impact on the predictor. Other information that can be drawn from the above plot are

- Feature importance which is placed in the descending order of importance in the figure, the least important variable feature with respect to the predictor is the building painted.
- The impact is displayed by the horizontal location. This is correlative with the feature importance and the location signifies whether the effect produced by the value has the association with the prediction in a higher or lower form.
- Original value depicted by the colors on the plot. The variables with high values are shown in red and the ones with low values are shown in blue for this observation.
- Correlation: from the plot, it is observed that the building dimension has a level of correlation rated as high and has both positive and negative impact on the predictor as shown by the red and blue colors. But whereas the building dimension is leading in the positive impact, the insured period is taking the lead on the negative impact. This will make us agree that the building dimension has a positive correlation with the target variable while the insured period has a negative correlation with the same. For random forest model.

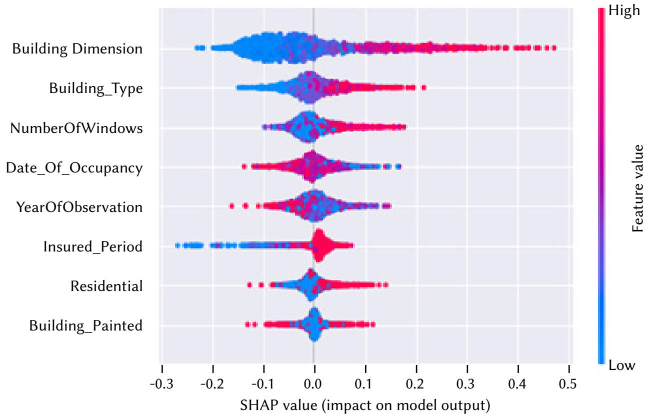


Fig. 4. SHAP value plot for Random Forest.

From Fig. 4 it is observable intuitively that the building dimension is considered to have the most important impact on the model output which is followed by the building type. Although the building type has the highest positive impact and second only to insured_period in the negative impact. The ones with no or negligible impacts were not covered in the plot. From the foregoing, it is easy to agree that the type of building will go a long way to determine whether a client will file for a claim or not in case of unforeseen contingencies when the house is under insurance.

To estimate how important each feature is to the predictor or model as shown in Fig 5 the SHAP estimates the mean absolute value got from the SHAP values from every feature presenting it in a form that the horizontal or y-axis stands for the feature importance while the row or x-axis stands for a feature. The plot in Fig 5. shows variable importance giving a list of the variables in the descending order of their importance to the predictor or model. The variables located at the top have more contribution to the model than those located at the lower part of the plot. From our context, the building dimension has the greatest impact on the predictor while the building painted has the least impact. Those with the greatest impact and as such have a higher power of predictivity. The plot was obtained from the random forest algorithm.

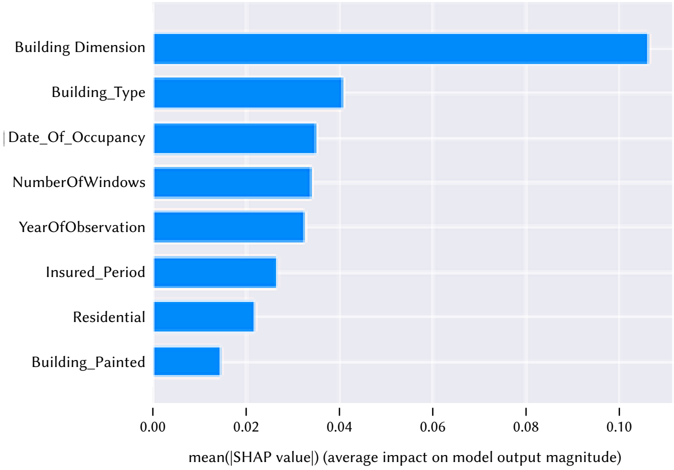


Fig. 5. SHAP value plot for variable importance.

Individual SHAP value plot was made as shown in Fig. 6 to show local interpretability which helps to enhance the transparency of the model – Random forest. When some of the variables were taken at random from the table of values of the variables and plotted using SHAP, the plot obtained is shown in fig. 6.

Any of the rows chosen at random is plotted and the output value is the prediction obtained from that particular row. The base value represents the average of the output of the model over the training data. The different colors (Red/Blue) represents features that influence the prediction to the positive side that is giving a higher prediction (to the right) and influencing the prediction to the negative side giving a lower prediction on the model. The red color with the arrow pointing to the right shows a higher prediction while the blue color with the arrow that points to the left shows a lower prediction.

Other models that we used for the prediction gave a similar result when analyzed used SHAP value showing that the building dimension contributed more to the predictability of the model followed by the building type. It follows the same trend in the variable importance.

Limitations of the models we compared are that their activities are only constrained within the dataset they were trained on like most machine learning algorithms. And the strength is that it gives an opportunity to people who may be confused on which model to employ to know the model that may be able to perform better in a particular scenario. Another thing we can consider as the study's strength is that it was employed in a scenario of a dataset that is considered small with lots of abnormalities. This shows that even in cases of a relatively small dataset with abnormalities, the dataset can be preprocessed and be used for prediction.

VI. Conclusion

This study was carried out on different machine learning algorithms on the insurance dataset from Zindi.africa [68] to get a prediction of whether a customer will have claims or apply for claims over his/her property or not based on the attributes of the building as explained earlier. We preprocessed the data, carried out all the necessary data engineering, implemented the algorithm on python, and obtained the results as shown in the previous section. Some of the algorithms have special qualities and situations for optimum performance, NB for instance is to be chosen in real-time prediction situations and where there are multi classes to be predicted. The DT has special abilities in handling noisy data as well as being able to avoid overfitting when pruning is applied in DT, apart from it being easy to implement and interpret. SVM has shown to be appropriate in a situation where the data available is really small for any reason. The results were analyzed in light of the GINI index and the SHAP values.

The result from the GINI index showed that the Kernel Support Vector Machine outperformed the other algorithms and its performance was followed by that of Logistic regression. Although the results from the algorithms are closely related perhaps because of the amount of data provided but in doubt of the result got from one, one can use the next one with higher with better accuracy, thereby reducing the human effort and time to do the task manually. When analyzed using the SHAP values, it was discovered that the feature with the highest influence on the predictors was the building dimension. This was shown in Fig 4. Consequently, the plots from the SHAP analysis showed that for each row in the distribution, the

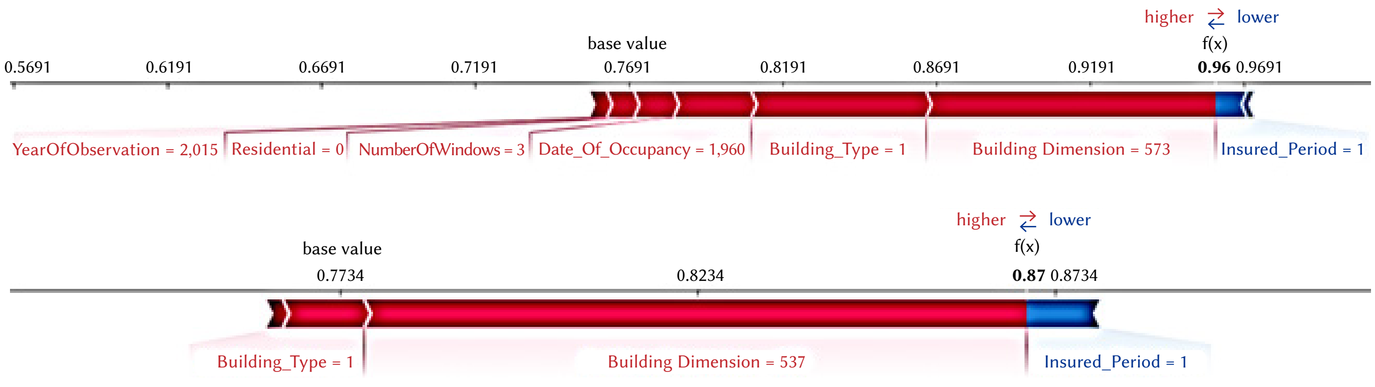


Fig. 6. Individual SHAP value plot for Random Forest.

building dimension has a great deal of influence t the prediction followed by the building type.

Some other algorithms have been theorized to have shown good performance on certain conditions of the dataset for example large dataset, noisy data, etc, it is, therefore, important to put into consideration the amount of dataset available and some other things surrounding it before one concludes on the classification algorithm to use.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (No.62072074, No.62076054, No.62027827, No.61902054, No.62002047), the Frontier Science and Technology Innovation Projects of National Key R&D Program (No.2019QY1405), the Sichuan Science and Technology Innovation Platform and Talent Plan (No.2020JDJQ0020) and the Sichuan Science and Technology Support Plan (No. 2020YFSY0010).

References

[1] H. Sufriyana, Y. W. Wu, and E. C. Y. Su, "Artificial intelligence-assisted prediction of preeclampsia: Development and external validation of a nationwide health insurance dataset of the BPJS Kesehatan in Indonesia," *EBioMedicine*, vol. 54, 2020, doi: 10.1016/j.ebiom.2020.102710.

[2] Y. Huang and S. Meng, "A Bayesian nonparametric model and its application in insurance loss prediction," *Insurance: Mathematics and Economics*, vol. 93, pp. 84–94, 2020, doi: 10.1016/j.insmatheco.2020.04.010.

[3] P. Li, S. Li, T. Bi, and Y. Liu, "Telecom customer churn prediction method based on cluster stratified sampling logistic regression," *IET Conference Publications*, vol. 2014, no. CP660, pp. 282–287, 2014, doi: 10.1049/CP.2014.1576.

[4] Z. Kai-Hui, L. Lei, and L. Peng, "Customer churn prediction based on cluster stratified sampling logistic regression," *International Journal of Digital Content Technology and its Applications*, 2011, doi: 10.4156/jdcta.vol5.issue10.45.

[5] L. Tao, D. Zhu, L. Yan, and P. Zhang, "The traffic accident hotspot prediction: Based on the logistic regression method," *ICTIS 2015 - 3rd International Conference on Transportation Information and Safety, Proceedings*, pp. 107–110, Aug. 2015, doi: 10.1109/ICTIS.2015.7232194.

[6] H. Lan and Y. Pan, "A crowdsourcing quality prediction model based on random forests," *Proceedings - 18th IEEE/ACIS International Conference on Computer and Information Science, ICIS 2019*, pp. 315–319, Jun. 2019, doi: 10.1109/ICIS46139.2019.8940306.

[7] X. Ye, X. Wu, and Y. Guo, "Real-time Quality Prediction of Casting Billet Based on Random Forest Algorithm," *Proceedings of the 2018 IEEE International Conference on Progress in Informatics and Computing, PIC 2018*, pp. 140–143, Jul. 2018, doi: 10.1109/PIC.2018.8706306.

[8] Y. Liu and H. Wu, "Prediction of road traffic congestion based on random forest," *Proceedings - 2017 10th International Symposium on Computational Intelligence and Design, ISCID 2017*, vol. 2, pp. 361–364, Feb. 2018, doi: 10.1109/ISCID.2017.216.

[9] J. Guo, H. Liu, Y. Luan, and Y. Wu, "Application of birth defect prediction model based on c5.0 decision tree algorithm," *Proceedings - IEEE 2018 International Congress on Cybermatics: 2018 IEEE Conferences on Internet of Things, Green Computing and Communications, Cyber, Physical and Social Computing, Smart Data, Blockchain, Computer and Information Technology, iThings/Gree*, 2018, doi: 10.1109/Cybermatics_2018.2018.00310.

[10] X. Hu, Y. Yang, L. Chen, and S. Zhu, "Research on a Customer Churn Combination Prediction Model Based on Decision Tree and Neural Network," *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics, ICCCBDA 2020*, pp. 129–132, 2020, doi: 10.1109/ICCCBDA49378.2020.9095611.

[11] R. K. Gupta, S. S. Lathwal, A. P. Ruhil, T. K. Mohanty, and Y. Singh, "Lameness prediction in Karan fries cross-bred cows using decision tree models," *2015 International Conference on Computing for Sustainable Global Development, INDIACom 2015*, 2015.

[12] I. A. A. Amra and A. Y. A. Maghari, "Students performance prediction using KNN and Naïve Bayesian," *ICIT 2017 - 8th International Conference on Information Technology, Proceedings*, 2017, doi: 10.1109/ICITECH.2017.8079967.

[13] G. A. Bhatt and P. R. Gandhi, "Statistical and ANN based prediction of wind power with uncertainty," *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, vol. 2019-April, no. Icoei, pp. 622–627, 2019, doi: 10.1109/icoei.2019.8862551.

[14] A. Yusof and S. Ismail, "Multiple Regressions in Analysing House Price Variations," *Communications of the IBIMA*, 2012, doi: 10.5171/2012.383101.

[15] M. A. Babyak, "What You See May Not Be What You Get: A Brief, Nontechnical Introduction to Overfitting in Regression-Type Models," *Psychosomatic Medicine*, 2004, doi: 10.1097/00006842-200405000-00021.

[16] Mislan, Haviluddin, S. Hardwinarto, Sumaryono, and M. Aipassa, "Rainfall Monthly Prediction Based on Artificial Neural Network: A Case Study in Tenggarong Station, East Kalimantan - Indonesia," *Procedia Computer Science*, 2015, doi: 10.1016/j.procs.2015.07.528.

[17] A. Chogle, P. Khaire, A. Gaud, and J. Jain, "House Price Forecasting using Data Mining Techniques," *House Price Forecasting using Data Mining Techniques*, 2017, doi: 10.17148/IJARCCE.2017.61216.

[18] D. Sangani, K. Erickson, and M. Al Hasan, "Predicting Zillow Estimation Error Using Linear Regression and Gradient Boosting," *Proceedings - 14th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, MASS 2017*, 2017, doi: 10.1109/MASS.2017.88.

[19] A. Nur, R. Ema, H. Taufiq, and W. Firdaus, "Modeling House Price Prediction using Regression Analysis and Particle Swarm Optimization Case Study : Malang, East Java, Indonesia," *International Journal of Advanced Computer Science and Applications*, 2017, doi: 10.14569/ijacsa.2017.081042.

[20] A. Khalafallah, "Neural Network Based Model for Predicting Housing Market Performance," *Tsinghua Science and Technology*, 2008, doi: 10.1016/S1007-0214(08)70169-X.

[21] N. Bhagat, A. Mohokar, and S. Mane, "House Price Forecasting using Data Mining," *International Journal of Computer Applications*, 2016, doi: 10.5120/ijca2016911775.

[22] S. C. Bourassa, E. Cantoni, and M. Hoesli, "Spatial dependence, housing submarkets, and house price prediction," *Journal of Real Estate Finance and Economics*, 2007, doi: 10.1007/s11146-007-9036-8.

[23] C. Brunsdon, A. S. Fotheringham, and M. E. Charlton, "Geographically weighted regression: a method for exploring spatial nonstationarity," *Geographical Analysis*, 1996, doi: 10.1111/j.1538-4632.1996.tb00936.x.

[24] A. Onan, "Classifier and feature set ensembles for web page classification," *Journal of Information Science*, 2016, doi: 10.1177/0165551515591724.

[25] A. Onan and S. KorukoGlu, "A feature selection model based on genetic rank aggregation for text sentiment classification," *Journal of Information Science*, 2017, doi: 10.1177/0165551515613226.

[26] A. Onan, "An ensemble scheme based on language function analysis and feature engineering for text genre classification," *Journal of Information Science*, 2018, doi: 10.1177/0165551516677911.

[27] A. Onan, S. Korukoğlu, and H. Bulut, "Ensemble of keyword extraction methods and classifiers in text classification," *Expert Systems with Applications*, 2016, doi: 10.1016/j.eswa.2016.03.045.

[28] A. Sharaff and S. R. Roy, "Comparative Analysis of Temperature Prediction Using Regression Methods and Back Propagation Neural Network," *Proceedings of the 2nd International Conference on Trends in Electronics and Informatics, ICOEI 2018*, no. Icoei, pp. 739–742, 2018, doi: 10.1109/ICOEI.2018.8553803.

[29] K. Abhishek, M. P. Singh, S. Ghosh, and A. Anand, "Weather Forecasting Model using Artificial Neural Network," *Procedia Technology*, 2012, doi: 10.1016/j.protcy.2012.05.047.

[30] N. Shobha and T. Asha, "Monitoring weather based meteorological data: Clustering approach for analysis," *Proceedings-IEEE International Conference on Innovative Mechanisms for Industry Applications, ICIMIA 2017 -*, 2017, doi: 10.1109/ICIMIA.2017.7975575.

[31] J. Gill, B. Singh, and S. Singh, "Training back propagation neural networks with genetic algorithm for weather forecasting," *SIISY 2010 - 8th IEEE International Symposium on Intelligent Systems and Informatics*, 2010, doi: 10.1109/SISY.2010.5647319.

[32] A. Paniagua-Tineo, S. Salcedo-Sanz, C. Casanova-Mateo, E. G. Ortiz-García, M. A. Cony, and E. Hernández-Martín, "Prediction of daily maximum temperature using a support vector regression algorithm,"

Renewable Energy, 2011, doi: 10.1016/j.renene.2011.03.030.

[33] R. E. Abdel-Aal, "Hourly temperature forecasting using abductive networks," *Engineering Applications of Artificial Intelligence*, 2004, doi: 10.1016/j.engappai.2004.04.002.

[34] L. Houthuys, Z. Karevan, and J. A. K. Suykens, "Multi-view LS-SVM regression for black-box temperature prediction in weather forecasting," *Proceedings of the International Joint Conference on Neural Networks*, 2017, doi: 10.1109/IJCNN.2017.7965975.

[35] H. Xie, M. Ding, L. Chen, J. An, Z. Chen, and M. Wu, "Short-term wind power prediction by using empirical mode decomposition based GA-SYR," *Chinese Control Conference, CCC*, 2017, doi: 10.23919/ChiCC.2017.8028818.

[36] X. Peng, D. Deng, J. Wen, L. Xiong, S. Feng, and B. Wang, "A very short term wind power forecasting approach based on numerical weather prediction and error correction method," *China International Conference on Electricity Distribution, CICED*, 2016, doi: 10.1109/CICED.2016.7576362.

[37] W. Zhang, H. Zhang, J. Liu, K. Li, D. Yang, and H. Tian, "Weather prediction with multiclass support vector machines in the fault detection of photovoltaic system," *IEEE/CAA Journal of Automatica Sinica*, 2017, doi: 10.1109/JAS.2017.7510562.

[38] S. Papantoniou and D. D. Kolokotsa, "Prediction of outdoor air temperature using neural networks: Application in 4 European cities," *Energy and Buildings*, 2016, doi: 10.1016/j.enbuild.2015.06.054.

[39] D. C. Wu, B. Bahrami Asl, A. Razban, and J. Chen, "Air compressor load forecasting using artificial neural network," *Expert Systems with Applications*, no. October, p. 114209, 2020, doi: 10.1016/j.eswa.2020.114209.

[40] P. Patil, N. Yaligar, and S. Meena, "Comparision of Performance of Classifiers - SVM, RF and ANN in Potato Blight Disease Detection Using Leaf Images," *2017 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2017*, pp. 3–7, 2018, doi: 10.1109/ICCIC.2017.8524301.

[41] X. Tang, Z. Liu, T. Li, W. Wu, and Z. Wei, "The application of decision tree in the prediction of winning team," *Proceedings - 2018 International Conference on Virtual Reality and Intelligent Systems, ICVRIS 2018*, 2018, doi: 10.1109/ICVRIS.2018.00065.

[42] N. I. Nwulu, "A decision trees approach to oil price prediction," *IDAP 2017 - International Artificial Intelligence and Data Processing Symposium*, pp. 0–4, 2017, doi: 10.1109/IDAP.2017.8090313.

[43] M. A. Hassonah, A. Rodan, A. K. Al-Tamimi, and J. Alsakran, "Churn Prediction: A Comparative Study Using KNN and Decision Trees," *ITT 2019 - Information Technology Trends: Emerging Technologies Blockchain and IoT*, 2019, doi: 10.1109/ITT48889.2019.9075077.

[44] R. S. Raj, D. S. Sanjay, M. Kusuma, and S. Sampath, "Comparison of Support Vector Machine and Naïve Bayes Classifiers for Predicting Diabetes," *1st International Conference on Advanced Technologies in Intelligent Control, Environment, Computing and Communication Engineering, ICATIECE 2019*, pp. 41–45, 2019, doi: 10.1109/ICATIECE45860.2019.9063792.

[45] R. Bayindir, M. Yesilbudak, M. Colak, and N. Genc, "A novel application of naive bayes classifier in photovoltaic energy prediction," *Proceedings - 16th IEEE International Conference on Machine Learning and Applications, ICMLA 2017*, vol. 2017-Decem, pp. 523–527, 2017, doi: 10.1109/ICMLA.2017.0-108.

[46] A. P. Salim, K. A. Laksitowening, and I. Asror, "Time Series Prediction on College Graduation Using KNN Algorithm," *2020 8th International Conference on Information and Communication Technology, ICoICT 2020*, 2020, doi: 10.1109/ICoICT49345.2020.9166238.

[47] B. Zhang and D. Ma, "Flight delay prediciton at an airport using maching learning," *Proceedings - 2020 5th International Conference on Electromechanical Control Technology and Transportation, ICECTT 2020*, 2020, doi: 10.1109/ICECTT50890.2020.00128.

[48] H. Khaksar and A. Sheikholeslami, "Airline delay prediction by machine learning algorithms," *Scientia Iranica*, 2019, doi: 10.24200/sci.2017.20020.

[49] L. Belcastro, F. Marozzo, D. Talia, and P. Trunfio, "Using scalable data mining for predicting flight delays," *ACM Transactions on Intelligent Systems and Technology*, 2016, doi: 10.1145/2888402.

[50] S. Choi, Y. J. Kim, S. Briceno, and D. Mavris, "Prediction of weather-induced airline delays based on machine learning algorithms," *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 2016, doi: 10.1109/DASC.2016.7777956.

[51] M. S. Acharya, A. Armaan, and A. S. Antony, "A comparison of regression models for prediction of graduate admissions," *ICCIDS 2019 - 2nd International Conference on Computational Intelligence in Data Science, Proceedings*, 2019, doi: 10.1109/ICCIDS.2019.8862140.

[52] C. J. Ejiyi, O. Bamisile, N. Ugochi, Q. Zhen, N. Ilakoze, and C. Ijeoma, "Systematic Advancement of Yolo Object Detector For Real-Time Detection of Objects," *2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pp. 279–284, Dec. 2021, doi: 10.1109/ICCWAMTIP53232.2021.9674163.

[53] C. J. Ejiyi, J. Deng, T. U. Ejiyi, A. A. Salako, M. B. Ejiyi, and C. G. Anomihe, "Design and Development of Android Application for Educational Institutes," *Journal of Physics: Conference Series*, 2021, doi: 10.1088/1742-6596/1769/1/012066.

[54] R. D. Cook and S. Weisberg, "Criticism and Influence Analysis in Regression," *Sociological Methodology*, 1982, doi: 10.2307/270724.

[55] S. O. Bamisile Olusola, Ariyo Oluwasanmi, Chukwuebuka Joseph Ejiyi, Nasser Yimen, "Comparison of machine learning and deep learning algorithms for hourly global / diffuse solar radiation predictions," *International Journal of Energy Research*, no. January, pp. 1–22, 2021, doi: 10.1002/er.6529.

[56] K. Methaprayoon, C. Yingvivatanapong, W. J. Lee, and J. R. Liao, "An integration of ANN wind power estimation into unit commitment considering the forecasting uncertainty," *IEEE Transactions on Industry Applications*, 2007, doi: 10.1109/TIA.2007.908203.

[57] M. A. F. Azlah, L. S. Chua, F. R. Rahmad, F. I. Abdullah, and S. R. W. Alwi, "Review on techniques for plant leaf classification and recognition," *Computers*. 2019, doi: 10.3390/computers8040077.

[58] A. Ramil, A. J. López, J. S. Pozo-Antonio, and T. Rivas, "A computer vision system for identification of granite-forming minerals based on RGB data and artificial neural networks," *Measurement: Journal of the International Measurement Confederation*, 2018, doi: 10.1016/j.measurement.2017.12.006.

[59] S. Sperandei, "Understanding logistic regression analysis," *Biochemia Medica*, 2014, doi: 10.11613/BM.2014.003.

[60] Y. Y. Song and Y. Lu, "Decision tree methods: applications for classification and prediction," *Shanghai Archives of Psychiatry*, 2015, doi: 10.11919/j.issn.1002-0829.215044.

[61] L. Song, "Research on the application of data mining algorithm based on decision tree," *Metallurgical and Mining Industry*, 2015.

[62] D. T. Ahneman, J. G. Estrada, S. Lin, S. D. Dreher, and A. G. Doyle, "Predicting reaction performance in C–N cross-coupling using machine learning," *Science*, 2018, doi: 10.1126/science.aar5169.

[63] Y. L. Pavlov, "Random forests," *De Gruyter*, 2019, doi: https://doi.org/10.1515/9783110941975.

[64] M. Kantardzic, "Data Mining: Concepts, Models, Methods, and Algorithms: Second Edition," *John Wiley & Sons, Inc., Hoboken, New Jersey*, 2011, doi: 10.1002/9781118029145.

[65] K. L. Priya, M. S. Charan Reddy Kypa, M. M. Sudhan Reddy, and G. R. Mohan Reddy, "A Novel Approach to Predict Diabetes by Using Naive Bayes Classifier," *Proceedings of the 4th International Conference on Trends in Electronics and Informatics, ICOEI 2020*, no. Icoei, pp. 603–607, 2020, doi: 10.1109/ICOEI48184.2020.9142959.

[66] Mahima and N. B. Padmavathi, "Comparative study of kernel SVM and ANN classifiers for brain neoplasm classification," *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies, ICICICT 2017*, 2018, doi: 10.1109/ICICICT1.2017.8342608.

[67] Y. Zhang and L. Wu, "An MR brain images classifier via principal component analysis and kernel support vector machine," *Progress in Electromagnetics Research*, 2012, doi: 10.2528/PIER12061410.

[68] "Competitions - Zindi." https://zindi.africa/competitions (accessed Jul. 17, 2020).

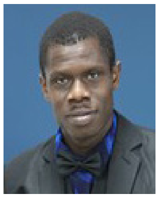
[69] S. Sharma and A. Bhagat, "Data preprocessing algorithm for Web Structure Mining," *Proceedings on 5th International Conference on Eco-Friendly Computing and Communication Systems, ICECCS 2016*, 2017, doi: 10.1109/Eco-friendly.2016.7893249.

[70] S. Samsani, "An RST based efficient preprocessing technique for handling inconsistent data," *2016 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2016*, 2017, doi: 10.1109/ICCIC.2016.7919591.

[71] J. Han, M. Kamber, and J. Pei, "Data Mining: Concepts and Techniques," *3rd Edition Morgan Kaufmann Publishers, Waltham.*, 2012, doi: 10.1016/

C2009-0-61819-5.

[72] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, 2017.



Chukwuebuka Joseph Ejiyi

Chukwuebuka Joseph Ejiyi received his Bachelor's Degree in 2014 from the Federal University of Technology Owerri (FUTO) Nigeria. He went on to obtain a master's degree in Software Engineering at the University of Electronic Science and Technology of China (UESTC) in 2021. . He is currently pursuing a Ph.D. degree with the Schoool of Information and Software Engineering at UESTC Chengdu China. His research interest is in Artificial intelligence, Deep Learning and he is currently working on Object detection using a single-stage neural network as well as Object classification. He also has a strong interest in image analysis especially with regards to medical images.

Zhen Qin

Zhen Qin is currently a professor in the School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC). He received his Ph.D. degree from UESTC in 2012. He was a visiting scholar in the Department of Electrical Engineering and Computer Science at Northwestern University. His research interests include data fusion analysis, mobile social networks, wireless sensor networks, and image processing.

Abdulhaq Adetunji Salako

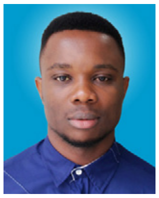
Abdulhaq Adetunji Salako received a bachelor's degree in information technology from the Valley View University (VVU), Accra- Ghana, West Africa, in 2017. He is currently pursuing an M.Sc. degree in computer science and engineering with the University of Electronic Science and Technology of China (UESTC). From 2015 to 2019, he worked under sub-contracts for Wigal Solutions, Ghana, and Teachers Fund of GNAT, Ghana. He is a member of the Data Visualization Research Team - UESTC. His research interests include information visualization, visual analytics, data mining, explainable AI, and NLP.

Happy Nkanta Monday

Happy Nkanta Monday received the B.Tech. degree in agricultural and environmental engineering from the Federal University of Technology, Akure, Nigeria, in 2013 and the M.Eng. degree in electronic science and technology from the University of Electronic Science and Technology of China, in 2018. He is currently pursuing a Ph.D. degree with the school of computer science and engineering, University of Electronic Science and Technology of China. His research interests include computer vision, wavelet, super-resolution, and deep learning.

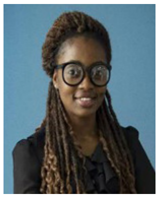
Grace Ugochi Nneji

Grace Ugochi Nneji received the B.Tech. degree in computer science from the Federal University of Technology, Owerri, Nigeria, in 2014 and the M.Eng. degree in software engineering from the University of Electronic Science and Technology of China, in 2019. She is currently pursuing a Ph.D. degree with the school of software engineering, University of Electronic Science and Technology of China. Her research interests include computer vision, re-identification, super-resolution, and deep learning.



Chiagoziem C. Ukwuoma

Chiagoziem C. Ukwuoma received the B.Eng. degree (Mechanical Engineering-Automobile Technology) from the Federal University of Technology Owerri in 2014 and his MSc. degree (Software Engineering) from the University of Electronic Science and Technology of China (UESTC) in 2020. He is currently a Ph.D. student at the University of Electronic Science and Technology of China (UESTC). His research interests include Object Detection and Object Classification.



Ijeoma Amuche Chikwendu

Ijeoma Amuche Chikwendu Received a B.Sc degree in Information management technology at the Federal University of Technology Owerri in 2014 and a Masters degree in Information and Communication Engineering at the University of Electronic Science and Technology of China (UESTC) in 2021. She is currently pursuing a Ph.D. degree at the same University where she obtained her master's degree. Her research interest is Statistical signal processing, Distributed estimation, target tracking, and localization.

Ji Gen

Ji Gen is currently a professor in the School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC). He received his Ph.D. degree from UESTC in 2012. His research interests include system software, information processing. He is a communication evaluation expert of information Department of National Natural Science Foundation of China.

Machine Learning in Business Intelligence 4.0: Cost Control in a Destination Hotel

Fulgencio Sánchez-Torres[1]*, Iván González[2], Cosmin C. Dobrescu[2]

[1] Higher Polytechnic School, Universidad de Alicante, Alicante (Spain)
[2] MAmI Research Lab at Castilla-La Mancha University (Spain)

Received 17 January 2021 | Accepted 4 February 2021 | Published 25 February 2022

unir
LA UNIVERSIDAD
EN INTERNET

Abstract

Cost control is a recurring problem in companies where studies have provided different solutions. The main objective of this research is to propose and validate an alternative to cost control using data science to support decision-making using the business intelligence 4.0 paradigm. The work uses Machine Learning (ML) to support decision-making in company cost-control management. Specifically, we used the ability of hierarchical agglomerative clustering (HAC) algorithms to generate clusters and suggest possible candidate products that could be substituted for other, more cost-effective ones. These candidate products were analyzed by a panel of company experts, facilitating decisions based on business costs. We needed to analyze and modify the company's ecosystem and its associated variables to obtain an adequate data warehouse during the study, which was developed over three years and validated HAC as a support to decision-making in cost control.

Keywords

Business Analysis With Expert Assessment, Business Intelligence 4.0, Candidate Product, ICT Ecosystem, Machine Learning.

DOI: 10.9781/ijimai.2022.02.008

I. Introduction

When we talk about industry 4.0, we associate the term with the fourth industrial revolution, with artificial intelligence as a differentiating element. This work studies the use of data science as a support for business intelligence to control company costs [1], [2], [3], [4].

Cost management in companies is almost always a problem solved from the financial point of view. This work proposes and implements an alternative to cost management based on the analysis of product consumption. To do this, we apply Machine Learning (ML), specifically, hierarchical agglomerative clustering (HAC) algorithms to support decision-making [5], [6], [7], [8], [9]. The research took place in a hotel on the southeastern coast of Spain.

To delimit the investigation and ensure its viability, we studied the company in its environment and context. We analyzed the company's ecosystem and its Information and Communications Technology (eICT) ecosystem, together with the possible Artificial Intelligence environments to be used. Later, we acted on business and technological processes.

Since this is applied research, we incorporated a panel of experts in the areas of: ICT management, purchasing, and food and beverages. These professionals provided deductive knowledge to identify the items to be considered and evaluated the results obtained. This knowledge was combined with the inductive knowledge generated by Machine Learning, increasing the differential value of the research [10], [11].

To ensure the viability of the work and avoid additional problems in the company, the investigation was carried out in phases, where the completion of one allowed the completion of the next

II. Methods

A. Hypothesis and Objectives

To deal with profit ratio problems, the tourism industry focuses on selling as much as possible at as high a price as possible, without addressing organizational difficulties in depth. Food and Beverages (F&B) are an important part of a hotel's operating costs, where the consumption of raw materials is difficult to manage. We have found works based on food costs during a trial period [12] and those focused on room or energy costs associated with hotel management [13]. Most solutions avoid the problem of control by allocating a percentage of the sales produced. Thus, these questions arise: Is applying a percentage of sales the best way to control consumption? Is this the only way to control consumption? This work proposes and develops an alternative to cost management by controlling product consumption instead of controlling financial cost [14], [15] [16], [17], [18], [19].

B. Scope of Work

The tourism sector is a market in need of innovation. The 4.0 paradigm can help digitize its value chain [20]. To do this, it is necessary to analyze companies as a whole, taking into account their environment, context, users, and computer systems to get a complete vision. A company's environment focuses on its surroundings and the data associated with systems other than in-house customer service. Context focuses on the circumstances of the company as an entity and its interaction through business processes [21], [22]. The data generated by these processes must be usable, so it is necessary to ensure that they are correctly produced, complying with the technical

* Corresponding author.

E-mail address: fulgenciosancheztorres@gmail.com

and legal requirements established by the mandatory data protection regulations (RGPD) [23], [24] for companies in the European Union.

We grouped computer systems by the business processes related to the clients of the hotel. To do this, we used two dimensions, one referring to the company and the other to the clients of the hotel. We studied the systems' functionality and interaction, their usefulness in the study, and compliance with the GDPR. We established three degrees: high, medium, and low, to quantify the degree of interaction among the available systems.

C. Business Ecosystem

Using the cost-control approach based on the company's consumption and its associated elements, we focused on the F&B area. Consumption control can be carried out in two ways depending on the warehouse: weekly inventories of the products for production warehouses, in this case, the hotel kitchen, and permanent inventories that record the data in real time for the regulatory warehouse, in this case, the general warehouse of the hotel. This regulatory warehouse control allows management to keep track of direct consumption.

It was necessary to carry out some tasks prior to the study. The relationships between business processes and the data they generate were analyzed [25], [26], [27]. Internal meetings were held with the people involved to avoid rejection. The systems designed to control product consumption were tested, and support measures were implemented regarding regulation warehouses, safety stock, and real-time inventories.

Counting on a panel of experts is a basic part of proposals based on data science to obtain results that are consistent with the problem they address. This has been treated in previous works [28] [29], [30], [31]. Knowing what and how to assess is best carried out by experts [32].

Contextualizing the data, their selection, and the degree to which they could be used by the expert panel defined the variables. We kept in mind the viewing preferences of reference in the sector [33] and when the categorization of the data and items that define them condition their analysis and use [34]. It was crucial to analyze the dimensions of the products based on the needs of the study and not on manufacturers' specifications. For this reason, we created and implemented a new concept for product management called the generic product, which grouped all the products with equivalent dimensions. Each product used had its generic code Fig.1 and a unit of measure that characterized it (liter, kilogram, etc.). This was essential to reconfigure and provide feedback to the system based on need and the analyses carried out over time.

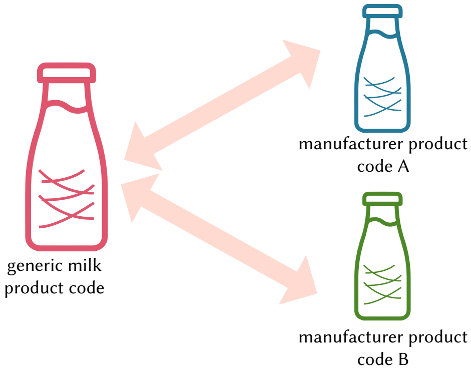


Fig. 1. Generic product.

D. The ICT Used

The technologies used were grouped by their functionalities for data collection and treatment, data analysis, and AI techniques.

1. Data Generation and Access

We started with a double handicap. The ICTs used in the tourism sector normally collect and store data in isolation. Each type of software generates its island of data. Moreover, software manufacturers do not facilitate access to the data directly. Therefore, it was first necessary to ensure valid, accurate, complete, and consistent data over time for the data warehouse. The selected systems were Warehouse Management System, Supply Chain Management, Property Management Systems, Point Of Sale, and Finance System.

We identified the data to use and its associated fields in the database. To obtain the data, we used MSQL Server Management Studio, SoapUI, Postman, and XML and JSON developments we carried out. During the research, it was necessary to add new software functionalities and make changes to the management systems. The modifications were made from the functional and technical points of view, for example, incorporating mobility and barcode systems.

2. AI Environment

The work took place where scientific and expert knowledge converged, making it difficult to distinguish where one began and the other ended. The domain experts performed deductive tasks, and the engineering experts, inductive ones. We needed a work environment that would allow us to evaluate the possibilities of ML, although soon after the work began, we decided to use HAC algorithms. These, by grouping the possible candidate products in an unsupervised way, facilitated business analysis with expert assessment (BAEA), and this knowledge became part of the company's know-how.

The chosen suite had to be used in the company and be especially oriented to data science [35]. We chose Python [36] with the Anaconda distribution as it is an open-source suite conceptually designed for data science and encompasses applications and libraries.

3. AI Techniques

Looking for an alternative to classical management, we avoided techniques such as support vector machines based on training, kNearest algorithms based on supervised learning with training sets, or tree and forest algorithms. Initially, clustering [37] and classification techniques seemed to be the most appropriate, although these can be considered similar to each other when identifying the two behavior patterns. The essential difference is that data classification uses predefined classes to perform the grouping, and clustering identifies similarities between the objects of the data sets by grouping them by their common characteristics.

In our study, input data was available without any type of labeling, from which information was obtained without conditioning the final result. This led us to unsupervised learning. We were looking for common patterns in the items that would give us candidate products to be substituted for more cost-effective products. The decision to substitute these products would be made in the [38] BAEA. This led us to agglomerative clustering algorithms about which there are different works [39], such as those that compare clustering algorithms [40].

E. Implementation

We decided to carry out the work in phases, with the completion of each allowing the following to begin. In this way, we were able to better delimit the study and cause the least possible disturbance to the hotel.

1. Phase 0. Preparing the Data Set

We wanted to work with complete and quality data sets to be able to test the AI environment, variables, and algorithms. The information needed to be consistent over time. Therefore, the data warehouse had to be separated from the business processes of the hotel. To extract the data, we used SQL, XML, and JSON.

Previous data. First, we analyzed needs from the point of view of business logic. For example, to obtain information about product consumption, it was necessary to consider all the warehouses involved in the product's life cycle and the business processes that affected it.

Record selection. Consistent and complete data, including all the fields and records were needed to facilitate assessment.

Consumption control is associated with inventory cycles and can be carried out daily or weekly, depending on the warehouse. This is why we established weeks as control units, defined as Monday to Sunday, based on the merchandise replenishment cycle in the warehouses.

After data processing, we had a CSV file (separated by ;). UTF8 standard, encoding=ISO-8859-1, with the following nomenclature *Proposal phase_number of variables used_variable weighted and records used.csv*. Table I.

TABLE I. Description of File Name Composition

Phase	Phase of the proposal where the file is used
Var	Number of variables included in the file
05	Percentage of the weight of the main variable for the records of the file
FullReg	All records available
Esp	Records weighted by the consumption variable
Cos	Records weighted by the cost variable

Example. *Phase1_3Var_05_EspReg* -> *Phase1*, 3 variables used, weighting the registers at 5 percent of the consumption variable.

Data validation. Validation was carried out first from the business logic perspective to get an early approximation of the suitability of the data. The data can have a correct format but an incorrect value. To do this, graphic representation techniques, frequency distribution, and crossing variables in two and three dimensions were used. We used data from one year applying the *Sturges* rule that permits consistent scaling. This allowed us to identify errors and the business process where they originated, as well as the person responsible for correcting them. Fig. 2 shows a three-dimensional crossover of variables between the unit cost, sum of consumption - sum of cost, where we identified discordant elements requiring action in business processes. We needed the products to be correctly coded using the concept of generic product and its unit of measure. We also needed to identify the variables and values that did not provide value in controlling consumption and cost and that simply added noise to the data.

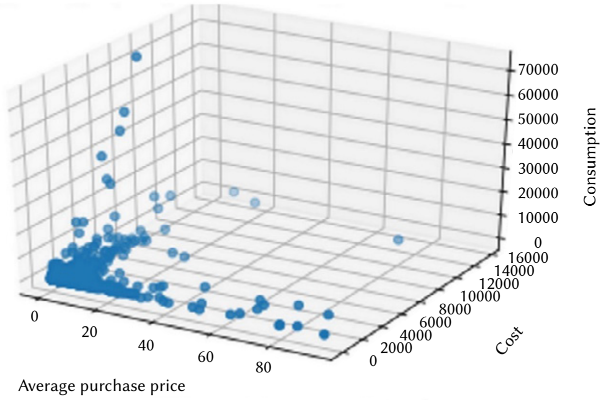


Fig. 2. Representation of variables.

Suitability of the data. The different HAC algorithms generated different solutions for evaluation by the panel of experts. The algorithms based their strategy for generating clusters on the minimum possible distance and maximum similarity between them. At first, to study the validity of the data, we used the single algorithm with Euclidean metric from the *linkage* package of *scipy.clusters.hierarchy* [41].

The three-dimensional representation of clusters and distances Fig. 3 shows us some examples of suitable values in green, and outliers in red. To see the relationships among the possible candidate products, the clusters formed, and the distances between the clusters, we used dendrograms to facilitate the interpretation of the data. A vertical dendrogram facilitates the analysis even more by representing the distances or heights on the "y" axis. An additional problem is the representation of data that does not add value to the study. To solve this problem, we decided to truncate the dendrograms and not graphically represent the values below a certain reference value. This improved the visual analysis Fig. 4. This figure shows the same data set with the weighting of records with the cost variable at 5 percent in the weight of the data set. The analysis allowed us to discard records with values close to zero with a weight in the variable of less than 5 percent of the total weight. It also made it possible to identify clusters linked at great distances that were susceptible to being analyzed directly.

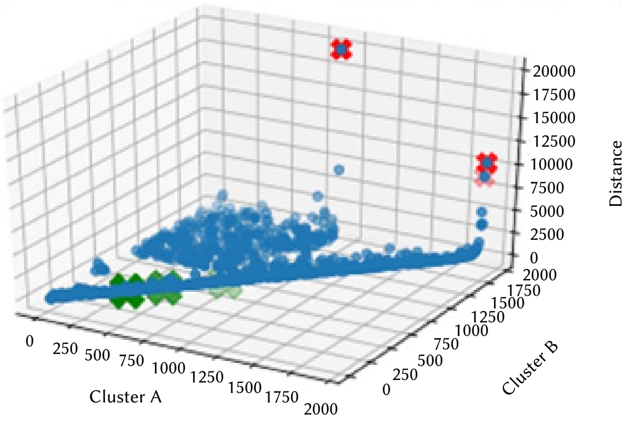


Fig. 3. Representation of clusters and distances.

2. Phase 1. Control Variables and Characterization Items

We continued with the application of the HAC algorithms and, specifically, with the single algorithm with Euclidean metrics to evaluate the possible combinations Table II. The variables under study were product consumption, product cost, and average purchase price. We used new data sets from the year 2017 that met the requirements of the previous stage.

TABLE II. Data Set and Variables Used

data set	consumption	cost	average purchase price	weighted variable
Phase1_ 3Var _FullReg	X	X	X	all records
Phase1_ 3Var _05_EspReg	X	X	X	consumption
Phase1_ 3Var _05_CosReg	X	X	X	cost
Phase1_ 2Var _05_EspReg	X		X	consumption
Phase1_ 2Var _05_CosReg		X	X	cost
Phase1_ 2Var _05_EspCos	X	X		consumption
Phase1_ 2Var _05_CosEsp	X	X		cost

For an objective evaluation, we made a table that reflected the results after applying the algorithm to the different data sets. This table was modified during the proposal by adding the results of each new phase. The data reflected initially were:

- Data sets. The data set used, which reflects the number of variables used and therefore prevails at 5 percent over the others.
- Weighted variable. Significant variable in the data set used.
- Algorithm clusters. Number of clusters generated by the default algorithm.
- Cophenetic coefficient. Cophenetic correlation coefficient.
- Maximum inconsistency value. To see how close it is to the fixed maximum value of 2.
- Number of elbow clusters. Number of clusters generated by observing the elbow method.
- Maximum Acceleration value.
- Minimum cutting distance.
- Maximum cutting distance.
- Number of candidate products. Number of candidate products identified after applying BAEA.

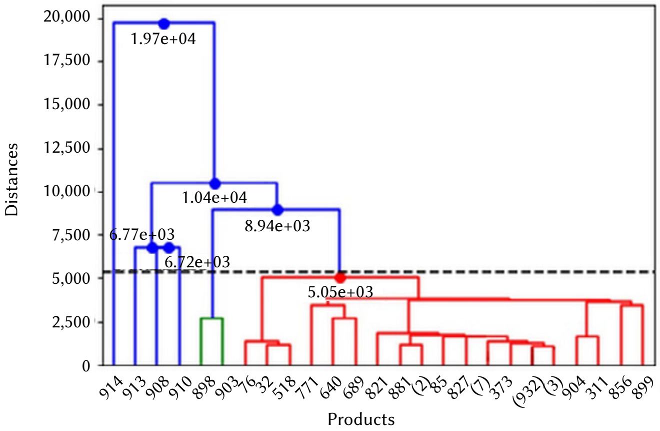


Fig. 4. Truncated vertical dendrogram.

Analyzing the data in Table III, we see that: the cluster number generated by the default algorithm ranges between 3 and 4; the number of candidate products is greater for two variables; the cophenetic coefficient is closer to a value of one for two variables; and the elbow method is not significant in this case. We see that the acceleration is less for two variables. At first, the algorithm behaves better for two variables than for three. At this point in the study, this is not significant because what is of interest is validating the usefulness of the HAC algorithms to propose candidate products to which BAEA can be applied.

3. Phase 2. Selection of the Algorithm

We studied the ability of single, complete, weighted, average, centroid, median, and Ward hierarchical agglomerative algorithms to propose candidate products using the variables consumption and cost. To do this, we used new data sets with two variables, one weighted by the consumption variable and the other by cost. We used new data from the year 2018 and corrected it as necessary. Finally, a maximum inconsistency of two was set.

To bring the research closer to the reality of the hotel, we included additional characteristics of the products from the kitchen and shopping requirements Table IV. We evaluated the combinations among the data set, algorithms, and new characteristics.

- Algorithm: Algorithm used.
- Acceleration cluster: Cluster where the acceleration is triggered.
- Distance difference: Difference between the maximum and minimum distance.
- Outliers: Number of outliers calculated by the default method.

We updated the characterization table V with the new items for all the algorithms, using the *Phase2_2Var_05Espcos* data set to illustrate it. This data set contained the products with a consumption incidence greater than 5 percent with respect to the cost variable. The algorithm that came closest to this hypothesis after applying BAEA was the average algorithm.

We analyzed the graph generated by applying the elbow method Fig. 5. We see the acceleration represented by the yellow line, marked with a light purple arrow where it begins to shoot, and in with a dark purple arrow where it reaches its maximum. The blue line reflects the distances, identifying the maximum and minimum (red and yellow, respectively). To find the interval identifying candidate products, we used the lower and upper cutoff distances. The minimum cutoff distance is after low cluster values (the light green arrow). The maximum cutoff distance is before the last cluster value (dark green arrow).

TABLE III. Machine Learning Assessment

data set	weighted variable	algorithm clusters	coefficient cophenetic	maximum inconsistency value	number of clusters elbow	maximum acceleration value	minimum cutting distance	maximum cutting distance	number of candidate products
Phase1_3Var _FullReg	all records	3	0.8672	1,9098	4	6	3,949	15,050	4
Phase1_ 3Var_05_EspReg	consumption	4	0.8523	1,1522	4	6	751	15,050	5
Phase1_ 3Var_05_CosReg	cost	3	0.8594	1,8418	4	6	258	22,755	5
Phase1_ 2Var_05_EspReg	consumption	3	0.9336	1,9599	4	5	751	15,050	7
Phase1_ 2Var_05_CosReg	cost	4	0.9231	1,9685	4	5	449	22,755	6
Phase1_ 2Var_05_EspCos	consumption	4	0.9260	1,8907	4	5	751	15,050	6
Phase1_ 2Var_05_CosEsp	cost	3	0.9596	1,8418	4	5	258	22,755	6

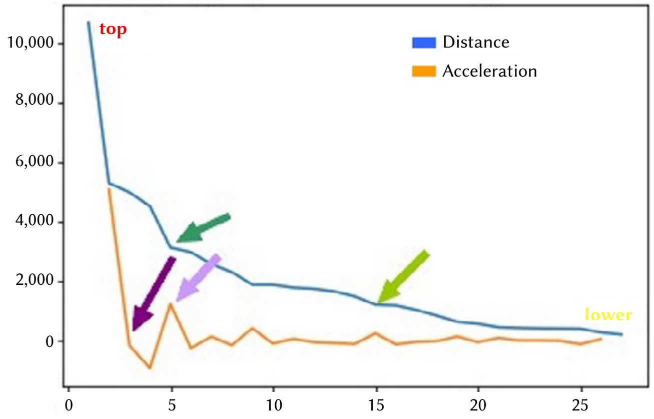


Fig. 5. Elbow method representation.

In the dendrogram Fig. 6, we observe the relationship between clusters, distances, and candidate products. We indicate the candidate products with a purple arrow, an example of two candidate products is shown with a blue circle, and two possible candidate products not valued because they fall outside the minimum distance set are shown with an orange circle.

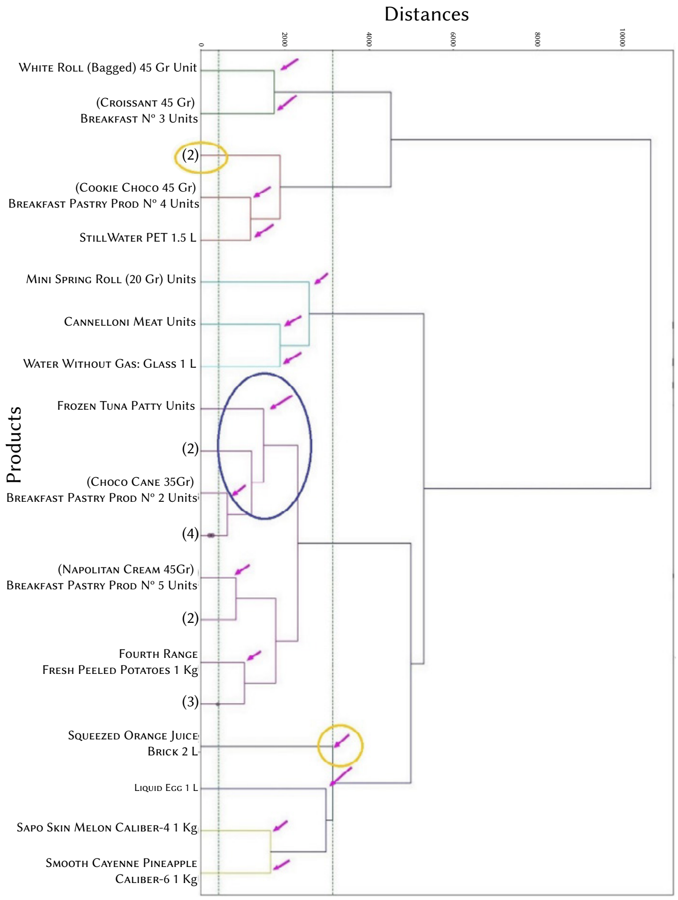

Distances
0
2000
4000
6000
8000
10000
Products
White Roll (Bagged) 45 Gr Unit
(Croissant 45 Gr) Breakfast Nº 3 Units
(2)
(Cookie Choco 45 Gr) Breakfast Pastry Prod Nº 4 Units
StillWater PET 1.5 L
Mini Spring Roll (20 Gr) Units
Cannelloni Meat Units
Water Without Gas: Glass 1 L
Frozen Tuna Patty Units
(2)
(Choco Cane 35Gr) Breakfast Pastry Prod Nº 2 Units
(4)
(Napolitan Cream 45Gr) Breakfast Pastry Prod Nº 5 Units
(2)
Fourth Range Fresh Peeled Potatoes 1 Kg
(3)
Squeezed Orange Juice Brick 2 L
Liquid Egg 1 L
Sapo Skin Melon Caliber-4 1 Kg
Smooth Cayenne Pineapple Caliber-6 1 Kg

Fig 6. Relationship between clusters, distances, and candidate products.

Fig. 7 shows the relationship between the *cost* and *consumption* variables, identifying the same elements from the dendrogram in Fig. 6.

Low acceleration helps us identify the intermediate clusters since the closer we get to the value of one, the more significant it will be in the final part. A large difference between the distances helps us locate the clusters that are included and where the outliers will be identified.

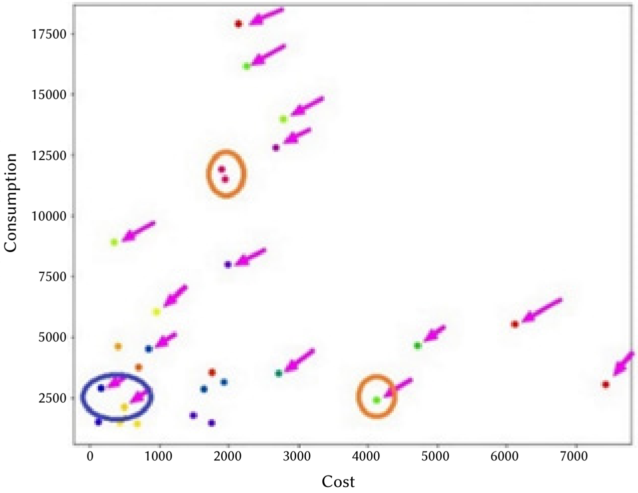


Fig 7. Relationship between variables and candidate products.

4. Intermediate Results

In this phase, we identified the algorithm that proposed more candidate products. We can see the characterization data in Table IV. The data that most interested us was the number of candidate products selected after applying BAEA that were delimited by the difference between the distances. There were other interesting results that helped us choose the algorithm. They are the following: having a cophenetic coefficient closest to one indicates a better correlation with the initial data set; low maximum acceleration at the beginning indicates that it will shoot up in the final part of the dendrogram; and the number of outliers indicates the number of elements that are left out and can be seen directly.

We know that when HAC is applied, there is not just one solution, and it depends on different factors, so we need to approximate an algorithm to each hypothesis. In Table IV, we see the data identified for each hypothesis. Blue is for cost and green for consumption. We highlighted the selected algorithm in bold for each color. The consumption hypothesis and average algorithm are in green, and the cost hypothesis and average algorithm are in blue.

5. Phase 3. Profile Variables

With the inclusion of the profile variables associated with client profiles, we wanted to see the relationship between client type and candidate products to facilitate decision-making based on the hotel's clients [42], [43]. Including financial variables made it possible to study the relationship between consumption and the two types of financial imputations studied, production in the F&B area and total hotel production. Considering the results of phase 2 on the applied algorithms, average for consumption, and median for cost, we addressed the profile variable. To do this, it was necessary to expand the data warehouse with the data for the period between 2018-07-01 and 2019-06-30. The variables were defined as:

- Customer profile.

a) Number of people = Breakfast (all guests have breakfast included).

b) Meals included = clients with lunches and dinners included in the hotel package purchased.

c) Occupation = Number of people + Number of meals included.

- Financial.

a) Billing of the A&B area.

b) Total hotel billing.

Three work scenarios were proposed for the study Table V. One scenario was related to the control of consumption and the variables

TABLE IV. Algorithm Assessment

data Set	weighted variable	algorithm	algorithm clusters	coefficient cophenetic	maximum inconsistency value	number of clusters elbow	acceleration cluster	maximum value acceleration	minimum cutting distance	maximum cutting distance	distance difference	outliers	number of candidate products
Phase2_2Var_05_EspCos	consumption	single	6	0.860057	1.2960	3	3	724	208	3,054	2,846	1	7
Phase2_2Var_05_CosEsp	cost	single	3	0.932709	1.1154	4	4	930	19	4,319	4,300	1	8
Phase2_2Var_05_EspCos	**consumption**	**average**	**3**	**0.904901**	**1.1535**	**2**	**2**	**5,241**	**208**	**10,678**	**10,470**	**0**	**15**
Phase2_2Var_05_CosEsp	cost	average	3	0.942683	1.1546	2	2	6,178	19	12,317	12,298	0	8
Phase2_2Var_05_EspCos	consumption	weighted	3	0.902703	1.5356	3	3	2,188	208	9,813	9,605	0	8
Phase2_2Var_05_CosEsp	cost	weighted	3	0.959757	1.1535	2	2	2,76	19	9,462	9,443	0	7
Phase2_2Var_05_EspCos	consumption	centroid	3	0.904757	1.1539	2	3	5,726	208	10,477	10,269	0	8
Phase2_2Var_05_CosEsp	cost	centroid	3	0.942070	1.1542	2	3	1,589	19	12,22	12,201	0	8
Phase2_2Var_05_EspCos	consumption	median	3	0.904390	1.1537	3	3	4,87	208	9,647	9,439	0	7
Phase2_2Var_05_CosEsp	cost	median	3	0.959928	1.1538	2	3	2,152	19	9,293	9,274	2	8
Phase2_2Var_05_EspCos	consumption	ward	3	0.837265	1.1529	2	3	21,246	208	32,312	32,104	0	6
Phase2_2Var_05_CosEsp	cost	ward	3	0.919233	1.1496	2	2	2,692	19	42,928	42,909	0	6
Phase2_2Var_05_EspCos	consumption	complete	3	0.849150	1.1391	2	3	3,109	208	16,554	16,364	0	6
Phase2_2Var_05_CosEsp	cost	complete	3	0.936782	1.1349	2	3	3,337	19	17,989	17,970	0	5

TABLE V. Associated Profile Variables

number	data set	weighted variable	number people	number pensions	occupation	A&B billing	hotel billing	algorithm
1	Phase3_6Var_05EspCos_Ab_Hot	all records	X	X		X	X	average
2	Phase3_2Var_05Esp_Per	consumption	X					average
3	Phase3_2Var_05Esp_Pen	consumption		X				average
4	Phase3_2Var_05Esp_Ocu	consumption			X			average
5	Phase3_2Var_05Esp_Per_Pen	consumption	X	X				average
6	Phase3_6Var_05CosEsp_Ab_Hot	all records	X	X		X	X	median
7	Phase3_6Var_05CosEsp_Ab	cost				X		median
8	Phase3_6Var_05CosEsp_Hot	cost					X	median

TABLE VI. Profile Assessment

number	algorithm clusters	coefficient	maximum inconsistency value	number of clusters elbow	maximum value acceleration	minimum cutting distance	maximum cutting distance	cluster below cutoff	minimum distance	maximum distance	distance difference	outliers	possible products candidate	candidate products
1	2	0.909555	1.154697	2	4,395	519	5,249	6	152	19,622	19,470	1	17	14
2	2	0.947091	1.145795	3	2,870	54	2,102	11	16	19,537	19,521	2	16	5
3	2	0.947091	1.145795	3	2,870	54	2,102	11	16	19,537	19,521	2	16	5
4	2	0.947091	1.145795	2	2,870	54	2,102	11	16	19,537	19,521	2	15	5
5	2	0.947091	1.145795	2	2,870	54	2,102	11	16	19,537	19,521	2	13	5
6	3	0.925175	1.154313	3	6,914	190	2,528	14	52	17,369	17,317	3	22	12
7	3	0.902039	1.145677	3	1,051	95	1,401	26	1	4,549	4,548	0	16	4
8	3	0.902039	1.145677	3	1,051	95	1,401	25	1	4,549	4,548	0	17	4

people, meals included, and occupation. Another had to do with billing for both F&B and the total hotel, and a third covered all the possibilities, including these together with the data set and algorithms.

The characterization results are shown in Table VI. They associate the cost-control data, on a blue background, with the consumption data on a green background. The different hypotheses refer to the data obtained regardless of the variables and data sets used. The column showing the difference between distances illustrates how close together the results are. The column of possible candidate products indicates the number of clusters between the minimum and maximum cutoff distances. The inclusion of variables associated with customer profiles and billing improved the results, and these are related to the candidate products in Table VII.

TABLE VII. Evaluation Of Candidate Products by Hypothesis

candidate product \ number	1	2	3	4	5	6	7	8	Total
(Milk Buns 40Gr) Prod Breakfast Pastries Nº6 Unit	1	1	1	1	1				5
(Choco Cane 35Gr) Breakfast Pastry Prod Nº2 Unit	1	1	1	1	1				5
(Cookie Choco 45Gr) BreakfastPastry Prod Nº4 Unit	1	1	1	1	1				5
(Magdalena 30Gr) Breakfast Pastry Prod Nº1 Unit	1	1	1	1	1	1			6
(Miguelito 60Gr) Breakfast Pastry Prod Nº5 Unit	1	1	1	1	1				5
Frozen Trunk Tuna 1 Kg (LOINS)						1		1	2
COD FILLET T-1000g Frozen 1 Kg						1	1	1	3
Frozen Pork tenderloin 1 Kg						1			1
Frozen Tuna Patty Units	1	1	1	1	1				5
Swordfish Piece 10/30 Frozen 1 Kg							1		1
Cooked Shrimp 40/60 Frozen 1 Kg						1		1	2
Toad Skin Melon Caliber-4 1 Kg	1					1			2
Artisan Bread Rhombus 30Gr Uds	1								1
White Toasts 45Gr Uds	1	1	1	1	1	1			6
Integral Roll 40Gr Units	1								1
Fourth Range Fresh Peeled Potatoes 1 Kg	1								1
Monalisa Washed Potato 1 Kg	1								1
Smooth Cayenne Pineapple Caliber-6 1 Kg	1					1			2
Frozen Chicken Thigh Fillet "W/Skin" 1 Kg						1			1
Semi-cured Cheese Mixed 1 Kg						1			1
Mini Spring Roll (20G) Units	1	1	1	1	1				5
Fresh Salmon 5/6 1 Kg (LOINS)						1			1
Frozen Salmon 1 Kg							1		1
Processed Cuttlefish 05-1 Kg Frozen 1 Kg						1	1	1	3
Total	14	8	8	8	8	12	4	4	

III. Results

The research produced intermediate results, some of which have already been described, although we will summarize the phases carried out.

Phase 0. Data set. It was focused on obtaining the appropriate data set and selecting the work environment in which to carry out the research.

- We selected the hotel's internal IT systems, Warehouse Management System, Supply Chain Management, Property Management Systems, Point Of Sale, and Finance System to obtain the necessary data. We used MSQL Server Management Studio, SoapUI, Postman, and our own programs made with XML and JSON to extract the data.
- Creation and implementation of the generic product code within the eICT to have comparable data over time.
- We incorporated a procedure to create and debug the data sets. We defined the format and notation for the files (CSV separated by ;) standard UTF8, encoding=ISO-8859-1.
- Validation of the adequacy of the HAC algorithms to develop the research based on a single algorithm with Euclidean metric.

1. Phase 1. Control Variables and Item Characterization

- The variables product consumption and product unit price are set and weighted as basic for the data sets, discarding the average purchase price variable.
- After applying the single algorithm with Euclidean metric to the data sets, we obtained the first version of the valuation table characterizing the algorithms. The table reflects the items: data set, number of variables used, number of algorithm clusters, number of clusters generated by default by the algorithm, number of elbow method clusters, cophenetic coefficient, maximum inconsistency value, maximum acceleration value, maximum cutoff distance, minimum cutoff distance, and BAEA number.

2. Phase 2. Selection of the Algorithm

New items were obtained for the evaluation table, the algorithm used, the number of clusters where acceleration shoots up, the difference between the maximum and minimum distance, and the outliers.

We selected the algorithm for each hypothesis, the average algorithm to control based on consumption, and the median for control based on cost.

3. Phase 3. Profile Variables

In this phase, we obtained the first global comparison between consumption and cost management hypotheses, identified in Table VII on a green and blue background, respectively. To do this, we defined new variables associated with customer profiles, including: number of people, meals included, and occupancy. We also defined new variables related to F&B turnover and the hotel. These new variables generated new characterization items: hypothesis used, lower and upper cutoff distances, number of lower cutoff clusters, maximum acceleration difference, number of visual clusters, and number of candidate product clusters in the segment.

The final result is presented in Table VII as a summary. The candidate products generated by the consumption hypothesis with the median algorithm are in green, and those generated from the cost hypothesis with the average algorithm are in blue. The rows identify the candidate products and the columns, the hypothesis from which it is obtained. The last row and last column show the respective totals.

Some data are worth pointing out. When we used all the variables, more candidate products were generated (columns 1 and 6). The type of product and the number of candidate products varied depending on the hypothesis and the variables used. In green and bold we see that there are products, such as round cupcakes and white muffins, that are identified from the consumption hypothesis regardless of the number of variables, but they are only identified from the cost hypothesis if all the variables are used. In blue and bold, we see products such as cod and shrimp that are identified from the cost hypothesis, but not for all the variables. In brown and bold, we have the special case of pineapple and melon, which are only identified when all the variables are used, regardless of the hypothesis.

IV. Discussion

The intermediate results conditioned the viability of the work and the subsequent phases, making it necessary to modify the business logic of the eICT and gain access to the raw data to generate the data warehouse.

The initial results showed it was necessary to compare products from different manufacturers throughout time. These products, which are equivalent to each other, had to be used as if they were the same product. Therefore, we implemented a generic product code with its unit of measure for each product, which allowed us to standardize the products. This implementation was vital to the research because, without it, the work would not have continued since it would not be possible to systematically purchase the same products over time.

The criteria for the generation of the data sets were established, excluding those non-significant products and defining a format and nomenclature for the files used. These files had to be easily usable in the different work environments.

The work used the ability of HAC algorithms to cluster by focusing on the elements that made up the cluster and using them as candidate products. The need to objectively compare the algorithms, variables, and results led to the creation of a working method and different tables of results. These tables reflect the needs that had to be met to objectively assess the candidate products from the business logic point of view. From this perspective, the panel of experts established the criteria that the candidate products had to meet, making those criteria part of the company's know-how. Throughout the study, the requirements established by the GDPR for the study of customer profiles were met.

The results reflected in the assessment tables reflect a double perspective: that of the panel of experts who, using business logic, established the criteria for the the candidate products and who assessed the results that became part of the company's know-how; and the one associated with business intelligence that was conditioned by the data available in the eICT and had to comply with the RGPD regulating customer profiles.

Compared to other works that address problems in the F&B area through food cost rates for a period of time or costs associated with hotel management, this work considered control from the product consumption management hypothesis and compared it to cost. Control based on consumption was more useful as it provided weekly details of raw material consumption according to the hotel's customer profiles. This made it possible to implement business processes quickly, improving customer management and satisfaction.

Selecting the algorithm closest to each hypothesis led us to expand the assessment items in each phase to align the HAC algorithms with the BAEA. The different algorithms gave different results, but all with a certain degree of validity, so it was not possible to speak of only one solution. There were factors, such as the type of distance used by the algorithm or the inconsistency value we set, that varied the results and represent possibilities for future work.

Carrying out the work in phases and including new items in each one made it possible to identify candidate products of higher quality and closer to customer profiles and consumption. This helped improve management and could lead to studying new variables.

When comparing management from consumption versus cost, Table 7 shows how management from the perspective of consumption identifies more candidate products, and how management from cost identifies those with the highest price per unit. Separating consumption management from financial management helps companies better control consumption and favors the identification of candidate products. This helps control products that are risky for the company and allows problems such as consumption and price fluctuations to be addressed more quickly than with other conventional cost-control systems. Developing a system that recommends candidate products based on business logic could be a future project.

The summary in Table VII shows the candidate products generated by both approaches. In Tables V, VI, and VII, the numbers correspond, and the hypotheses are differentiated by color. The data obtained from consumption with the median algorithm has a green background, and the data obtained from the average cost algorithm has a blue background.

V. Summary and Conclusions

The work proposes and validates an alternative system for managing the costs of raw materials based on consumption. To do this, different ML tools were evaluated to support a company's panel of experts in their decision-making. Specifically, the capabilities of HAC algorithms were used to generate clusters in an unsupervised way based on the similarities and differences between the elements. This produced candidate products that were studied against conventional analysis systems. The research was carried out during three years, using data from four years of the company's eICT.

For implementation, it was necessary to modify the company's eICT and provide it with the necessary items to feed the data warehouse. We had the collaboration of a panel of experts from the areas involved and a suitable tool, Anaconda, as a Python distribution.

To analyze whether cost control from consumption is feasible and comparable to financial control, it is necessary to study the variables and data associated with weekly product consumption control, customer profiles, and financial production in each phase of the study. Having data did not imply that they were adequate. It was necessary to validate them.

The main point of the study was to verify that the initial results generated by ML with the starting data sets allowed business experts to identify possible candidate products and thus help improve their business logic. For this reason, we refined the data set and defined the starting point to evaluate the algorithms using a table that included the results obtained at each moment. The subsequent study of each algorithm and the different data sets led to the expansion of the items evaluated and the characterization table. We could then establish the base algorithm for each hypothesis, the average algorithm for the consumption hypothesis, and the median algorithm for the cost hypothesis.

The evolution of the research led to studying how the variables associated with customer profiles and financial production influenced the results. To do this, the base algorithms of each hypothesis were analyzed with new data sets and this led to the inclusion of new items in the assessment table.

The comparison of the results of the consumption hypothesis and the cost hypothesis reflects that more candidate products are suggested from the consumption management perspective, and this makes it easier for the experts to replace some products with others that are more suitable at any given time. On the other hand, cost-based management is not as versatile, although it clearly identifies the most expensive products. Additionally, the inclusion of variables associated with client profiles will depend on the analysis that is required at each moment.

The proposal presents advances in aspects of both science and business:

- The creation and validation of the generic product concept and the standard consumption measure to equate products with equivalent properties is crucial. This permits processes to be automated and time series to be studied and analyzed. The generic product concept was included with a standard in the software functionalities of a software package of the sector following its validation in this study.
- The use of HAC algorithms is valid as a support tool for decision-making in an area that is difficult to manage, such as F&B in a hotel.
- This study points the way to new work in AI by identifying points of inquiry, such as recommending systems.

At the same time, the inclusion of HAC algorithms in raw material management and control provides a differential value for:

- Facilitating the early detection of anomalies in management and cost by allowing detailed, transparent control.

- Improving the quality of service and reducing costs by valuing products from the new approach provided by HAC algorithms.
- Facilitating and speeding up the work of the people involved.

The application of ML is a viable alternative for the management of costs in companies from the control of product consumption versus classic control. Panels of experts play an important role when implementing the system, identifying items, and validating results. The use of variables linked to different profiles, such as consumption, customer profiles, and financial production allows candidate products to be obtained using a new approach. This new decision-making support scenario makes it easier for experts to identify the items and algorithms that best suit their needs at all times.

References

[1] S. Coleman, "Data science in industry 4.0," *progress in industrial mathematics at ecmi 2018.* Springer, New York City, New York, USA, Springer Publishing Company, 2019, pp. 559-566, doi.org/10.1007/978-3-030-27550-1_71.

[2] P. Foster, F. Tom, *Data Science for Business: What you need to know about data mining and data-analytic thinking. G*ravenstein highway north Sebastopol, USA: O'Reilly Media, Inc, 2013.

[3] C. Flath, N. M. Stein, "Towards a data science toolbox for industrial analytics applications," *Computers in Industry*, vol 94, pp.16-25, 2018, doi.org/10.1016/j.compind.2017.09.003.

[4] M.A Waller, A.; S.E. Fawcett, "Data science, predictive analytics, and big data: a revolution that will transform supply chain design and management," *Journal of Business Logistics*, vol. 34, no 2, pp. 77-84, 2013, doi.org/10.1111/jbl.12010.

[5] S. Athey, "The Impact of Machine Learning on Economics," *The economics of artificial intelligence,* University of Chicago Press, Chicago, IL 60637 USA, pp. 507-552, 2019, doi.org/10.7208/9780226613475.

[6] Y.S. Reshi, R.A. Khan, "Creating business intelligence through machine learning: An Effective business decision making tool," *Information and Knowledge Management*, vol 4, pp. 65-75, 2014.

[7] M. Gopal, *Applied machine learning*, New York, USA, McGraw-Hill Education, 2018.

[8] S. Finlay, *Artificial intelligence and machine learning for business: a no-nonsense guide to data driven technologies*, Lancaster, UK, Lancaster University, 2021.

[9] K. R. Larsen, S. Becker, Automated *machine learning for business,* Oxford, UK, Oxford University Press, 2021.

[10] L. B. Akeem, "Effect of cost control and cost reduction techniques in organizational performance," International Business and Management, vol. 14, no 3, pp. 19-26, 2017, doi.org/10.3968/9686.

[11] Y. Hamuro, et al, "A machine learning algorithm for analyzing string patterns helps to discover simple and interpretable business rules from purchase history," *Progress in Discovery Science,* Springer, Berlin, Germany, pp. 565-575, 2002. doi.org/10.1007/3-540-45884-0_43.

[12] Z. Guo, "Research on the cost control with hotel operation system based on cost management theory," *Journal of Computational and Theoretical Nanoscience,* vol. 13, no 12, pp. 9882-9885, 2016, doi.org/10.1166/jctn.2016.5945.

[13] Q. Y. Yan, H.J. Shen, "Assessing hotel cost control through value engineering: A case study on the budget hotels in a middle-sized city in China," *Asia Pacific Journal of Tourism Research,* vol. 21, no 5, pp. 512-523, 2016, doi.org/10.1080/10941665.2015.1063521.

[14] Z. Wu, P.D. Christofides, "Economic machine-learning-based predictive control of nonlinear systems," *Mathematics*, vol. 7, no 6, pp. 494. 2019, doi.org/10.3390/math7060494.

[15] E. Cengiz, et al, "Do food and beverage cost-control measures increase hotel performance? A case study in Istanbul, Turkey," Journal of Foodservice Business Research, vol. 21, no 6, pp. 610-627, 2018, doi.org/10.1080/15378020.2018.1493893.

[16] M. H. Rafiei, H. Adeli, "Novel machine-learning model for estimating construction costs considering economic variables and indexes," *Journal of construction engineering and management*, vol. 144, no 12, pp. 04018106, 2018 , doi.org/10.1061/(ASCE)CO.1943-7862.0001570.

[17] S. NosratabadiI, et al, "Data science in economics: comprehensive review of advanced machine learning and deep learning methods," *Mathematics*, vol. 8, no 10, pp. 1799. 2020, doi.org/10.3390/math8101799.

[18] J. Sun, "Analysis on Cost Control in Hotel Financial Management," *Destech Transactions on Social Science, Education and Human Science,* Huhhot, China, 2017, doi.org/10.12783/dtssehs/ssme2017/13011.

[19] A. Arbelo, P. Pérez-gómez, M. Arbelo-pérez, "Cost efficiency and its determinants in the hotel industry," *Tourism Economics*, vol. 23, no 5, pp. 1056-1068, 2017, doi.org/10.1177/1354816616656419.

[20] S. Coleman,et al, "How can SMEs benefit from big data? Challenges and a path forward," *Quality and Reliability Engineering International*, vol. 32, no 6, pp. 2151-2164, 2016, doi.org/10.1002/qre.2008.

[21] L. Oliveira, A. Fleury, M.T. Fleury, "Digital power: Value chain upgrading in an age of digitization," *International Business Review*, vol. 30, no 6, pp. 101850, 2021, doi.org/10.1016/j.ibusrev.2021.101850.

[22] O. D. Kazakov, et al, "Development of the concept of management of economic systems processes through construction and calling of machine learning models," *IEEE International Conference-Quality Management, Transport and Information Security, Information Technologies-*IEEE, Saint Petersburg, Russia, pp. 316-321, 2018, doi.org/10.1109/ITMQIS.2018.8524985.

[23] J. L. José, A.V. Borja, "*La adaptación al nuevo marco de protección de datos tras el RGPD y la LOPDGDD,"* Wolters Kluwer, Madrid, España, 2019.

[24] C. Batini, et al, "Methodologies for data quality assessment and improvement," *ACM computing surveys*, vol. 41, no 3, pp. 1-52. 2009, doi.org/10.1145/1541880.1541883.

[25] E. Parra, et al, "A methodology for the classification of quality of requirements using machine learning techniques," *Information and Software Technology*, vol. 67, p. 180-195, 2015, doi.org/10.1016/j.infsof.2015.07.006.

[26] Y. Jin, B. Sendhoff, "Pareto-based multiobjective machine learning: An overview and case studies," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews),* vol. 38, no 3, pp. 397-415, 2008, doi.org/10.1109/TSMCC.2008.919172.

[27] J. C. Chen, et al, "Off to the races: A comparison of machine learning and alternative data for predicting economic indicators," *Big Data for 21st Century Economic Statistics,* University of Chicago Press, Chicago, USA, 2019.

[28] N. Azarenko, "The model of human capital development with innovative characteristics in digital economy," *IOP Conference Series: Materials Science and Engineering,* IOP Publishing 2020, St. Petersburg, Russian Federation, pp. 012032, doi.org/10.1088/1757-899X/940/1/012032.

[29] E. G. Mitchell, et al, "From Reflection to Action: Combining Machine Learning with Expert Knowledge for Nutrition Goal Recommendations," *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems,* pp. 1-17, Yokohama, Japan, doi.org/10.1145/3411764.3445555.

[30] K. D. Roe, et al, "Feature engineering with clinical expert knowledge: A case study assessment of machine learning model complexity and performance," *PloS one,* vol. 15, no 4, pp. e0231300, 2020, doi.org/10.1371/journal.pone.0231300.

[31] J. L. Loyer, et al, "Comparison of machine learning methods applied to the estimation of manufacturing cost of jet engine components," *International Journal of Production Economics,* vol. 178, pp. 109-119, 2016, doi.org/10.1016/j.ijpe.2016.05.006.

[32] H. Ahmed, et al, "Establishing standard rules for choosing best KPIs for an e-commerce business based on google analytics and machine learning technique," *International Journal of Advanced Computer Science and Applications,* vol. 8, no 5, pp. 12-24, 2017, doi.org/10.14569/ijacsa.2017.080570.

[33] F. Sánchez, Y. Hassan-Montero, "Visualization Design Dimensions for Data Science in Tourism and Transport," *Multidisciplinary Digital Publishing Institute Proceedings.* 13th International Conference on Ubiquitous Computing and Ambient "Intelligence UCAmI 2019, Toledo," Castilla la Mancha, Spain 2019, pp. 58, doi.org/10.3390/proceedings2019031058.

[34] A. Cook, P. Wu, K. Mengersen, "Machine learning and visual analytics for consulting business decision support," *2015 Big Data Visual Analytics (BDVA),* Hobart, Tasmania, Ausatralia, 2015, pp. 1-2, doi.org/10.1109/BDVA.2015.7314299.

[35] I. Lee, Y.J. Shin, "Machine learning for enterprises: Applications, algorithm selection, and challenges," *Business Horizons*, vol. 63, no 2, pp. 157-17, 2020, doi.org/10.1016/j.bushor.2019.10.005.

[36] Python Software Foundation. Python Language Reference, version 3.1. Available at http://www.python.org.

[37] P. Berkhin, "A survey of clustering data mining techniques," *Grouping multidimensional data,* Springer, Berlin, Heidelberg, Germany, pp. 25-71, 2006, doi.org/10.1007/3-540-28349-8_2.

[38] A. Fernandez, J. Preciado, A. Prieto, F. Sánchez-Figueroa, J. Gutiérrez, "Compare ML: A Novel Approach to Supporting Preliminary Data Analysis Decision Making," *International Journal of Interactive Multimedia and Artificial Intelligence*, 2021, doi.org/10.9781/ijimai.2021.08.001.

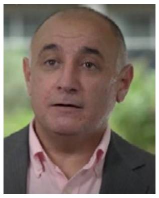
[39] S. Balakrishna, et al, "Incremental hierarchical clustering driven automatic annotations for unifying IoT streaming data," *International Journal Of Interactive Multimedia And Artificial Intelligence*, 2020, doi.org/10.9781/ijimai.2020.03.001.

[40] A. A Navarro, P. M. Ger, "Comparison of clustering algorithms for learning analytics with educational datasets," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 5, no. 2, pp. 9-16, 2018, doi.org/10.9781/ijimai.2018.02.003.

[41] The SciPy community, SciPy documentation, https://docs.scipy.org/doc/scipy/reference/cluster.hierarchy.html

[42] P. Talón-ballestero, et al, "Using big data from customer relationship management information systems to determine the client profile in the hotel sector," *Tourism Management*, vol. 68, pp. 187-197, 2018, doi.org/10.1016/j.tourman.2018.03.017.

[43] C. Kim, K. Chung, "Measuring Customer Satisfaction and Hotel Efficiency Analysis: An Approach Based on Data Envelopment Analysis," *Cornell Hospitality Quarterly*, 2020, doi.org/10.1177/1938965520944914.



Fulgencio Sánchez Torres

Fulgencio Sánchez Torres is currently CIO at Garza Real Hotels. He is a doctoral student in computer science at the University of Alicante and an approved collaborator of the School of Industrial Organization. He has a Master's degree in Big Data and massive data visualization from UNIR and a degree in computer science. He has been an external expert for ICUAL of the Ministry of Education of Spain. He is a researcher at the ICT Technology Center, a researcher at the University of Castilla La Mancha, and at the National Technological University of Argentina, where he conducted a project financed by the International Cooperation Agency, Ministry of Foreign Affairs.

Iván González

Iván González is assistant professor and researcher at the Castilla-La Mancha University (UCLM). He received his M.Sc. (2015) and Ph.D. (2018) degrees in Advanced Computer Technologies from the same University. Member of the MAmI (Modelling Ambient Intelligence) Research Group since 2013, Dr. González has been involved in several research and development of International and National projects and contracts. He has participated in International conferences with 18 publications to date and he is author of 11 JCR research contributions. Member of research networks and scientific platforms related to Ubiquitous Computing and Ambient Intelligence (UBIHEALTH, AIAm, RedAmITIC and GITCE-UTP). He is currently performing research efforts focused on Quantitative Gait Analysis (QGA), Frailty assessment and Mild Cognitive Impairment (MCI) screening through mobile technologies and embodied sensors. His research interests also include Ubiquitous Computing, Smart Health, Smart Environments, Artificial Intelligence, IoT and Sensor Networks. Dr. González has years of experience organizing International conferences and R&D+I activities being one of the main organizers of UCAmI annual conference (since 2014). He has participated in the scientific committee of International conferences (7) and as a regular external reviewer of impact journals from research publishers (MDPI, Springer, Hindawi, SAGE, etc.). Also, he has been guest editor of 2 JCR-indexed special issues and Volume Editor of the UCAmI 2019 MDPI Proceedings. In the Educational field, Dr. González is coordinator of the Computer Engineering Degree at UCLM. His teaching covers the following subjects: Programming Fundamentals I and II, Operating Systems, Concurrent and Real-Time Programming, Multimedia and Human-Computer Interaction.

Cosmin C. Dobrescu

Cosmin C. Dobrescu is PhD candidate in the MAmI research group at the University of Castilla–La Mancha (UCLM). He completed the master's degree in Systems and Control Engineering in 2021 at the National University of Distance Education (UNED). He has also a degree in Computer Engineering with a specialization in computing in 2018 at the UCLM. Constantin obtained a competitive contract as a research support technician in the WeCareLab laboratory at the Institute of Information Technologies and Systems (ITSI), co-financed by the Fondo Social Europeo AEI 2018 (PEJ2018) in 2019. His teaching in 2021/22 includes the practices of Interactive Systems Design subject in the Degree in Computer Engineering. He is primarily interested in the design and development of IoT wearable medical devices aimed at prevention and rehab. His research is specialized in the development of firmware for IoT devices with a variety of sensors and integrated System on a chip SoC. The main challenge in this technology is to achieve maximum energy efficiency when acquiring, preprocessed and storing the generated data. By analyzing data using artificial intelligence, conclusions can be drawn on how to prevent diseases or improve the rehabilitation of people.

Using Customer Knowledge Surveys to Explain Sales of Postgraduate Programs: A Machine Learning Approach

Eva Asensio, Alejandro Almeida*, Aida Galiano, Juan-Manuel Martín-Álvarez*

Universidad Internacional de La Rioja (UNIR), Logroño (Spain)

Received 20 February 2021 | Accepted 24 January 2022 | Published 31 January 2022

LA UNIVERSIDAD
EN INTERNET

Abstract

Universities collect information from each customer that contacts them through their websites and social media profiles. Customer knowledge surveys are the main information-gathering tool used to obtain this information about potential students. In this paper, we propose using the information gained via surveys along with enrolment databases, to group customers into homogeneous clusters in order to identify target customers who are more likely to enroll. The use of such a cluster strategy will increase the probability of converting contacts into customers and will allow the marketing and admission departments to focus on those customers with a greater probability of enrolling, thereby increasing efficiency. The specific characteristics of each cluster and those postgraduate programs that are more likely to be selected are identified. In addition, better insight into customers regarding their enrolment choices thanks to our cluster strategy, will allow universities to personalize their services resulting in greater satisfaction and, consequently, in increased future enrolment.

Keywords

Business Analytics, Business Decision Making, Clustering Methods, Machine Learning.

DOI: 10.9781/ijimai.2022.01.008

I. Introduction

When informing themselves about postgraduate programs in public and private universities, potential students commonly use the Internet [1]. Furthermore, the presence of universities on social networks, such as Instagram or Twitter, allows interaction with potential students.

Many universities have websites and social media profiles that allow them to collect information about potential students. In addition, some of these universities send customer knowledge surveys entitled, for example, "we want to know more about you" to potential students who are interested in the programs on offer. The data generated through these platforms is highly valued [2].

Generally, the use of the Internet facilitates communication between the potential student and postgraduate university services [1]. In many cases, universities offer unrequested information about their programs, for example, via social networks [3]. Therefore, for both universities and potential students, the consensus is that the use of the Internet generates value in the educational market [4].

The connections made through websites or social networks used in combination with customer knowledge surveys which some potential students voluntarily fill out, provide useful sales-optimizing information. In this context, the analysis of the information gathered contributes to building a better customer relationship management strategy [5].

Regarding the use of digital platforms, it must be taken into consideration that potential students are saturated with information from many sources. Immersed in an environment of over-information [6] they do not only receive information about university degrees, and pursuing a postgraduate degree is not even a primary need in many cases [7], [8]. Unsurprisingly in this context, there is a high probability that potential students will not notice or ignore the publicity they receive from universities. Thus, it is very important for the Universities to know their target customers well in order to access them effectively [9]. As mentioned above, postgraduate training may not be a primary need, so knowing customers well is essential to sales maximization [10]-[12].

These challenges demand that university marketing departments consider the fundamental characteristics of those potential students who have made the decision to start the admission process in postgraduate programs under offer [13]. For this reason, it is clearly useful to take into account the available data relating to enrolled postgraduate students and the information provided by those potential students who have answered the customer knowledge survey "we want to know more about you". Nevertheless, being in possession of this data alone does not generate advantages for the universities [14], and presently, few universities apply Business Analytics to generate competitive advantages [15] from the analysis of such data.

Despite data analysis being an important issue for all universities, given the high cost of academic programs, it is mostly private universities which use it for the design of effective strategies to attract

* Corresponding author.

E-mail address: alejandro.almeida@unir.net (A. Almeida), juanmanuel.martin@unir.net (J.-M. Martín-Álvarez).

new customers [8] and to design new programs that satisfy them more. In this sense, learning objectives and methodologies could be modified adapting them better to the characteristics of the customer.

This article analyses the data generated by the "we want to know more about you" surveys that the International University of La Rioja (UNIR) carried out with potential students some of whom finally enrolled. Attention is not limited to enrolled students, since the determining factors of those who finally decided not to enroll is also interesting. The analysis is carried out through the application of Machine Learning techniques, specifically cluster analysis with mixed data, to detect critical characteristics of both categories of students: those who finally enrolled and those who did not.

Machine Learning techniques applied to customer knowledge surveys allow us to identify three targets which are statistically different. This result is achieved by applying a statistical significance test to compare the proportions of enrolments within the distinct clusters. Target customers, or clusters, have differing enrolment probabilities. We identify the characteristics of these clusters and the probability of enrolment and we find that they belong to identifiable segments of the population. We consider that this find is crucial in the adaptation of marketing strategies and product characteristics to each target.

Our objective is to obtain more and better knowledge of the customer, the potential student, detecting the particular characteristics of those who are more likely to enroll in currently offered postgraduate programs. This knowledge will allow the identification of different targets and allow the marketing department to segment efforts and strategies by better adapting them to each target.

This document is structured as follows: section II describes the methodology undertaken; section III outlines the main results. In section IV there is a discussion of these results. Finally, section V draws conclusions.

II. Methodology

We collect data from customers who enquire about university programs and apply a specific methodology to that data. As part of this process, potential students fill out one of the data sources used in this work: a questionnaire. The other main data source comes from enrolments providing information about which students finally signed up.

Each dataset allows the segmentation of students and highlights differences in the conversion rate of each target cluster, hence moving us towards our objective.

A. Data Collection

To collect information about students, universities can use different methods such as the extraction of information from social networks or via knowledge surveys.

In part, the data used in this study is obtained through a survey carried out among students who are interested in a postgraduate program. The objective of the survey is to advise them on making the best possible choices so as to increase their satisfaction with the program. Importantly, this method allows the university to obtain direct and useful information from students without having to extract information from third parties.

This information was collected from May to October of 2020. A total of 16,272 surveys were filled in. The questionnaire poses 17 questions the responses to which provide data such as, response time in addition to other variables related to individual characteristics. It is important to note that not all the students that voluntarily completed the survey enrolled in one of the postgraduate programs.

The second data collection source is enrolment information: information about which of these students finally enrolled is collected. The availability of this information allows us to know which students are enrolled or not. In other words, we can compute the conversion rate, which is the probability of enrolling on a postgraduate program. Furthermore, we discover the characteristics of the students who enroll or not, which facilitates an evaluation of the appropriateness of degree courses to such

B. Data Preparation

To carry out our analysis we need to consider which data will be useful. First, nine variables that provide valuable information to segment the students were selected. Second, we carry out a data-cleaning process eliminating those students that present many null values or some outliers that may distort the analysis. This process and subsequent analysis are carried out in R, the language and environment for statistical computing, and its integrated development environment RStudio.

After the data cleaning process, the total number of surveys we work with is 11,859 of which 2,073 end up converting and enrolling in a postgraduate degree, that is a conversion rate of 17,48%.

The values taken by each of the selected variables are described below:

- **Age (A)** takes values from 19 to 79 years old, the average age being 35.12.
- **Average grade (AG)** values added in 3 groups. Between 5 and 6.4, between 6.5 and 8.4 and more than 8.5.
- **Working (W)** dichotomist variable reporting YES or NO.
- **Job (J)**: eight different values are used (middle manager, administrative, other company levels, consultant, specialist analytics, upper management and two categories of public workers, A-B and C-D).
- **Professional Experience (PE)** values added in 4 groups. From 1 to 3 years, from 3 to 5 years, more than 5 years or without professional experience.
- **Company size (CS)** values added in 3 groups. Large (more than 250 employees), medium (between 50 and 250 employees) or small (less than 50 employees).
- **Budget (B)** values added in 3 groups. Less than 4500, between 4500 and 5500 or more than 5500.
- **Survey Time (ST)** takes values between 22.75 seconds and 980 seconds with a time average of 146.95.
- **Country (C)**: we find students from 54 different countries, 6 being the most represented (Ecuador, Colombia, Spain, Mexico, Guatemala and Peru).

In addition to these variables, we also incorporate the binary variable **Matriculate (M)** that defines whether the student has enrolled in a program.

The characteristics of those customers who filled in the knowledge survey asking for information about a postgraduate program offered by UNIR are the following:

They are between 26 and 30 years old. Mostly they are from Ecuador (27.83%) but also from Colombia (18.55%), Spain (16.02%), Mexico (15.18%) and Guatemala (7.60%). In the labour market, they cover different strata: middle manager (26.14%), administrative (22.77%), other company levels (16.02%), consultant (11.81%), specialist analytics (10.96%), upper management (9.27%) and public workers (2.52%). 48.06% of them work in big companies (with more than 250 employees), 27.83% in small ones (less than 50 workers) and 24.45% in medium companies. They have more than 5 years in-work experience

(62.40%) and only the 0.84% have no experience. Finally, 51.43% report a high grade in previous studies (more than 8.5 points) and 44.70% a score between 6.5 and 8.5 points. Fig. 1 represents each of the 9 variables described above.

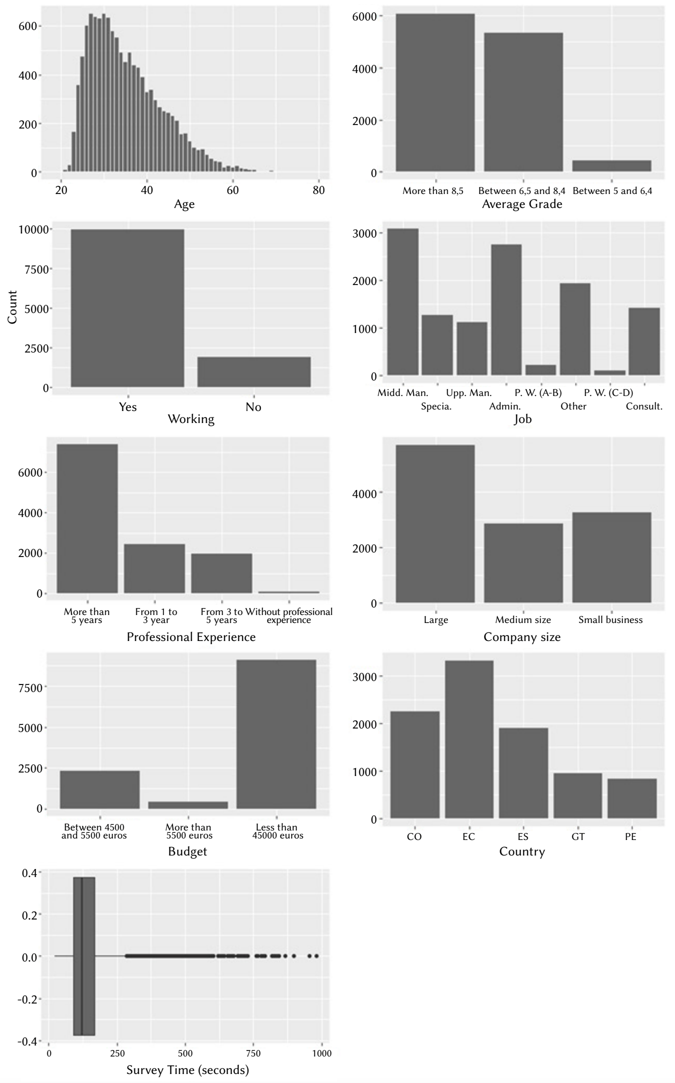


Fig. 1. Representation of the variables used.

C. Clustering Students

In marketing, segmentation is used to identify groups of customers that share homogeneous characteristics. For them, similar products and marketing strategies should be generated [16]-[18]. Among the objectives of segmentation are also the identification of target customers, the definition of the specific characteristics of customer groups, commonly known as the buyer persona [19]-[20].

Machine learning is used to segment students interested in pursuing a program at the university under study. These unsupervised learning techniques are used to gain insights into customers and can create clusters of students with a similar profile. Therefore, we used this approach to segment the potential students at this university into clusters.

Clustering techniques are tools that help to understand the different subgroups that exist within a data set. Specifically, these techniques aim to group the elements that are close enough to each other and far enough from other elements [21]. However, the choice of how to measure the distance between two elements is something that is not agreed upon in the literature and there are many subjective alternatives. Distance is a numerical measure of how far two individuals are apart, in other words, it measures the proximity or similarity between individuals [22].

The most common way to measure distance is the Euclidean distance, although there are other alternatives such as the Manhattan distance used for particular types of problems. However, our data contains mixed data types (numerical and categorical) where these distances are not applicable, therefore traditional clustering algorithms such as K-means or hierarchical clustering are not valid.

Therefore, for our case study, we use the Gower distance, which is a measure of distance that can be calculated for two individuals whose attributes are mixed. The Gower distance is computed as the average of the dissimilarities between individuals. Each Gower distance lies between [0 1].

$$d(i,j) = \frac{1}{p}\sum_{i=1}^{p} d_{if}^{(f)}$$

The partial dissimilarity $d_{(i,j)}^{(f)}$ depends on the type of variable that we are measuring. In the case of numerical variables, partial dissimilarity is the ratio between the absolute differences between observations and the maximum observed range of all individuals. In the case of categorical variables, the partial dissimilarity is 1 if the observations are different and 0 if not.

The selected clustering algorithm should fit well with the Gower distance. To do this, we select the k-medoids algorithm. The k-medoid algorithm, Partitioning Around Medoids (PAM) is a classic partitioning method similar to the well-known k-means method but, instead of iterating over the centroids, it iterates over the medoids, that is, it tries to find the most representative object for each cluster [21]. The algorithm clusters the objects in a total of k clusters where k must be given a priori. The selection of the optimal number of clusters (k) must be made considering statistical information obtained in the data, although if there is any reasoned justification a priori, the number of clusters may vary for different reasons. To select the optimal number of clusters in which to divide our data we use the silhouette width. The silhouette width is one of the most commonly used options for measuring the similarity between each point in a cluster and compares this similarity with the closest point of the neighbouring cluster. This metric lies between [-1 1] where higher values mean greater similarities [23].

Fig. 2 shows the result of the measurement for values of k between 2 and 10 where it can be observed that segmenting the students into 2 or 3 groups maximizes the similarity within the clusters and the dissimilarity between clusters. We have divided the sample into 2 and 3 groups following the results found in the silhouette analysis, however, using k=3 produces useful results for marketing and academic strategies as shown in the following sections.

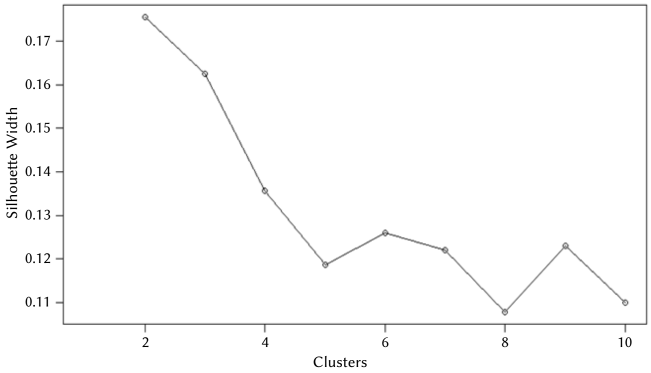


Fig. 2. Optimal number of clusters. Silhouette Width.

III. Results

This section presents the results of the k-medoids clustering technique using the Gower distance. As defined in the Methodology section, the k-medoids machine-learning algorithm for clustering data defines the most representative object of each cluster, that, in our case of study can be defined as the student profile in each cluster. Table I shows the characteristic that defines the profile of each cluster using k-medoids (K=2). Furthermore, along with these characteristics, we present the conversion rate (percentage of respondents who finally enrol in a postgraduate program).

TABLE I. Characteristics of the k-medoids (K=2)

Variable	Student profile	
	Cluster 1	Cluster 2
A	34.35	36.07
AG	>8.5	6.5< x <8.5
W	Yes	Yes
J	Middle-Manager	Clerical
PE	>5 years	>5 years
CS	> 250 empl.	> 250 empl.
B	<4500€	<4500€
ST	148.47	145.07
C	Ecuador	Spain
Conversion Rate[1]	17.52%	17.43%

[1] After performing a statistical test for equality of proportions, no differences were observed in terms of the conversion tare.

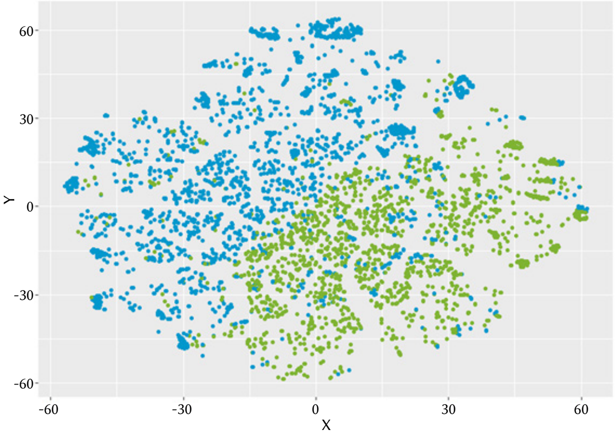


Fig. 3. Cluster plot from the "we want to know more about you" survey identifying customer personal characteristics (k=2).

To graph the division of students with k=2, Fig. 3 uses the t-Distributed Stochastic Neighbour Embedding technique [21], that helped us to visualize our multi-dimensional data into a two-dimensional plot.

Dividing the sample into two groups does not produce very useful results in terms of developing marketing or academic strategies. As can be seen in Table I, although there are differences between both clusters, the conversion rate is the same. This result, together with the small dissimilarity loss as observed in Fig. 3, leads us to perform the clustering with k = 3.

Table II shows the characteristic that defines the profile of each cluster using k-medoids (k=3) and the conversion rate for each cluster. Along with this, Fig. 4 shows in two dimensions the division into three groups of the students surveyed.

TABLE II. Characteristics of the K-Medoids (K=3)

Variable	Student profile		
	Cluster 1	Cluster 2	Cluster 3
A	38.35	28.66	36.14
AG	>8.5	>8.5	6.5< x <8.5
W	Yes	Yes	Yes
J	Middle-Manager	Clerical	Middle-Manager
PE	>5 years	<3 years	>5 years
CS	> 250 empl.	<50 empl.	> 250 empl.
B	<4500€	<4500€	<4500€
ST	152.33	143.58	144.08
C	Ecuador	Ecuador	Spain
Conversion Rate[1]	16.24%	16.46%	19.25%

[1] After carrying out a statistical test of equality of proportions, differences are observed in terms of the conversion tare.

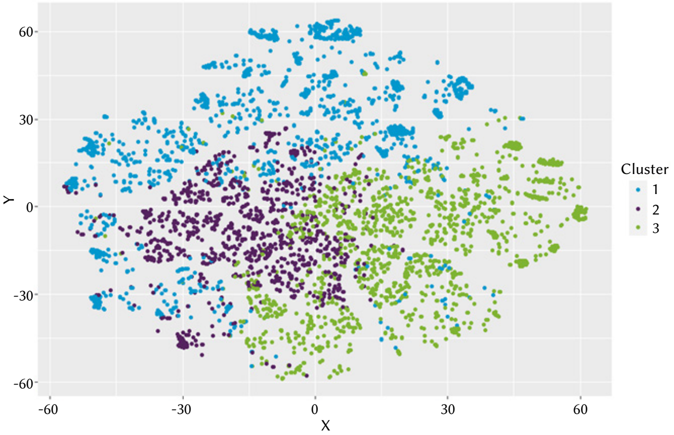


Fig. 4. Cluster plot from the "we want to know more about you" survey identifying customer personal characteristics (k=3).

For a better visualization of our results, we represent in Fig. 5, the variables in which we find differences between clusters that can be considered as the main discriminatory variables between the 3 clusters.

Cluster 1 present the lowest conversion rate. It includes people mostly from Ecuador (43.37%) and they are under 40 years old and working in middle management with more than 5 years' experience (88.41%). The conversion rate for this cluster is 16.24%. The previous mark register is the highest (up to 8 points) for the 95.40% of the customers included in this cluster. Customers included in this cluster can be considered as consolidated in the labour market and their professional career.

Cluster 2 contains the youngest customers with quite a high conversion rate (16.46%), not statistically significantly different from the previous cluster. Customers inside this cluster are from Ecuador (39.86%) and from Mexico (20.14%). They are working in administrative roles with low experience (63.47% have between 1 and 3 years' labour experience).

Cluster 3 contains people from Spain (31.14%) with an average age of 36.14. The conversion rate of this cluster is the highest (19.25%) and following a statistical test to compare its value with values of cluster 1 and 2, we find that it is statistically higher than the proportion registered by previous clusters. They work in big companies as middle managers with more than 5 years' experience (88.41%). The mark registered in previous studies is not high; 94.10% registers a mark between 6.5 and 8.5 points.

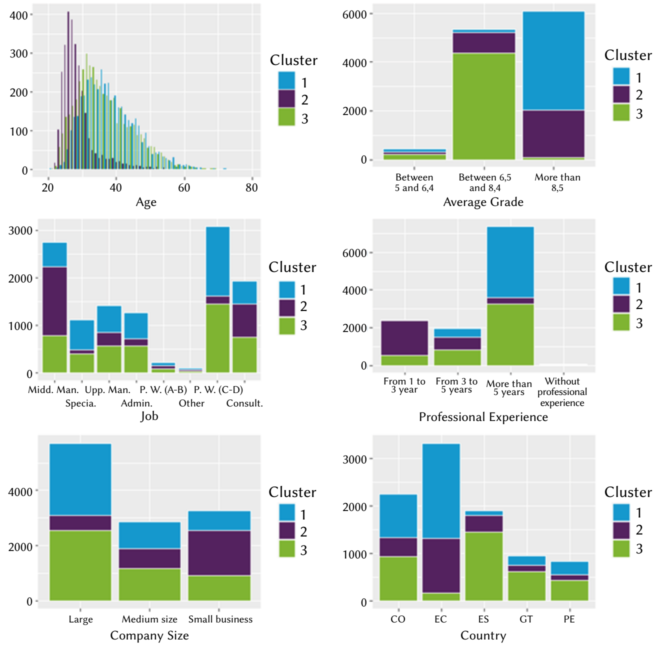


Fig. 5. Graphic representation of the composition of each cluster.

In brief, our set of results found three different segments one of which had a significantly higher conversion rate. These results can be helpful in adopting product design strategies and marketing strategies.

Using the segmentation provided for the cluster analysis we compute (Table III) the proportion of enrolment in each program offered. We observe that for all offered programs the conversion rate of cluster 3 is significantly higher for programs 1 to 7 and 10 to 12. However, other programs have a higher propensity to receive students included in cluster 1 (program 8, 9, 13, 15 and 16) and cluster 2 (program 14, 17 y 19).

The fact that we are able to identify those postgraduate programs that convert better for each target is valuable information for many marketing, academic and business purposes.

TABLE III. Proportion of Enrollment in Each Program Offered

Program	Student profile		
	Cluster 1	Cluster 2	Cluster 3
Program 1	19,57%	19,57%	**60,87%**
Program 2	28,57%	20,41%	**51,02%**
Program 3	18,64%	31,64%	**49,72%**
Program 4	24,10%	27,18%	**48,72%**
Program 5	30,77%	23,08%	**46,15%**
Program 6	37,93%	17,24%	**44,83%**
Program 7	36,31%	19,27%	**44,41%**
Program 8	**50,00%**	6,25%	43,75%
Program 9	**43,01%**	17,20%	39,78%
Program 10	27,78%	33,33%	**38,89%**
Program 11	36,54%	25,00%	**38,46%**
Program 12	30,77%	30,77%	**38,46%**
Program 13	**38,33%**	25,83%	35,83%
Program 14	25,00%	**40,91%**	34,09%
Program 15	**50,00%**	25,00%	25,00%
Program 16	**54,35%**	21,74%	23,91%
Program 17	29,41%	**47,06%**	23,53%
Program 18	**55,56%**	26,67%	17,78%
Program 19	31,58%	**52,63%**	15,79%

IV. Discussion

In the previous sections of this paper, the application of cluster analysis allowed us to identify similar student profiles at a private university and also to identify those postgraduate programs which each cluster are more likely to enrol in.

Now, we discuss the implications of our analysis from a marketing, business and academic perspective.

From a marketing point of view, it is important to mention the high cost per contact for the marketing and the admission department. Clustering potential students will increase the probability of a successful conversion for each contact thereby allowing these departments to focus on those customers with more probability of enrolling. In short, a direct increase in efficiency.

As indicated in the results section, we see that postgraduate programs can be advertised by targeting three client clusters. This analysis indicates that there is a group of students with a greater propensity for enrolment. Therefore, this first result makes it possible to prioritise the resources available to rank information requests from potential students. Along these lines, it seems reasonable to give response priority to those students with a greater propensity to convert.

When the proportion of students from each cluster enrolled is detailed (Table III), it can be seen how there are clear preferences of some student clusters for some courses. Thus, while the students in cluster 3 seem to be interested in, for example, Program 1 (60.83% of the enrolled students belong to cluster 3), the students in clusters 1 and 2 seem to consider the programs 18 and 19 of interest. Therefore, the promotion of courses must be directed to those profiles that principally have historically enrolled in them.

Moreover, the fact that groups of customers with specific profiles have been identified as more likely to convert will allow departments, with this improved knowledge, to personalise messages and better choose the channels used to reach them. Each contact could be guided towards a specific profile since we also know those programs in which contacts in their cluster have a higher conversion rate.

Companies have strategic objectives that shape the future. In this sense, the results seem to indicate that the university should think about a future in which the student profile of cluster 3 will increase its presence, given the greater propensity to enrol observed.

The results obtained could be improved if more data, for instance, geographic information such as the city of residence, were requested in the "we want to know more about you" surveys. Knowing the city of residence would give us approximate information on the income level of that person, knowing the average salaries of that geographical area. This fact would also allow us to link the message to the pricing strategies, further increasing the success of each contact.

The results achieved in this paper seem to recommend the integration of this cluster analysis in the balanced scorecard used for strategic decision-making. Also, the integration of this algorithm in the Customer Relationship Management (CRM) of the university would help to execute what is stated in this section and improve the efficiency of the marketing department.

Regarding the academic perspective, the results will allow a personalization of programs while they will contribute to the accurate perception of the programs by potential clients. Better customer knowledge will allow the personalization of the service which may lead to greater satisfaction and, as a consequence, to an increase in future enrolment.

The University performs innovation processes to improve their programs, content and methodology. These processes should take into account the information referring to those clusters most likely to choose each program-type in order to focus the improvements on the needs of each program's specific cluster. For example, if we want to create or to improve a program like program 1, academics should consider the specific characteristics of cluster 3 containing under 40-year-old Spanish students who do not at a high professional status. Meanwhile, if changes are made in Program 18, then the cluster 1 profile should be considered (professional students mostly from Ecuador).

V. Conclusion

The promotion of postgraduate programs represents an important part in the business management of private universities. Given the volatility of the environment and the fact that some graduate programs are ephemeral, attracting and enrolling new students is a priority in university management. In this context, it is essential to know the preferences of potential students to be able to communicate successfully with them.

The presence of universities on the Internet through web pages and social networks, facilitate communication between universities and their potential clients. Thus, the interactions of potential students with the university through the Internet helps to provide data that reflects preferences according to the profile of each potential student. Consequently, it is possible to observe the characteristics that drive the decision-making of students when enrolling in a particular program.

The detection of common characteristics shared by the students who enrol in the programs facilitates sales optimisation and increasing business efficiency due to the possibility of reallocating resources to profiles with a greater propensity to buy the program on offer. In other words, promotion strategies that incorporate the cluster analysis presented in this paper are more efficient due to the greater precision in the attention to the potential student. In addition, using this tool eliminates unnecessary information that can confuse future consumers. In an ideal scenario, we can generate specific marketing by including this algorithm in the university's CRM. By doing so, each program may be promoted based on the characteristics of those students who most frequently convert to said program. In this article, cluster analysis has been used to group potential students and design business strategies based on the characteristics of each group of those potential students. Specifically, we propose the development of marketing strategies for groups of students considering their propensity to enrol, both globally and at the program level.

Regarding the groups detected, we observe that, although there is a group with a greater propensity to enrol at the global level, the analysis at the program level reveals heterogeneous behaviour. Thus, it is observed how clusters 1, 2 and 3 have more presence in certain programs. This is because, since they are groups made up of homogeneous students, each group has a specific need. Thus, for example, while young profiles have more technological skills and are interested in programs with a high presence of technology, other profiles are consolidated at the work level and need to acquire soft skills, for example.

Although data-driven business management offers itself as a solution to improve the design of promotional strategies, we also observed some problems in our analysis case. These problems are related to data collection. For example, data collection ignores variables related to the potential student's disposable income. In the case of Spanish students, if the postal code were known, it would be possible to obtain the average income of the area in which the potential student resides. Thus, by comparing this data with the price of the program offered, an affordability index could be calculated. It should not be forgotten that in a global context, affordability is a quantity that can better explain conversion than just price. In other words, the degree is not as affordable for a potential student residing in a region with a low per capita income as one residing in a wealthy region. The strength of the data used is that the registration process is monitored, which makes it easier to follow the student who is interested in a certain program to predict if they will enrol or not. In this sense, this paper is an important tool for data-based business decision-making that generates efficiency in the sale of programs and personalization of the service provided. Innovations or changes in the methodology, pedagogy or contents of a degree should take into account the profile of the cluster that is more likely to be enrolled in each particular program.

References

[1] R. M. C. Croda, D. E. G. Romero, F. R. C. Villar, "The promotion of graduate programs through clustering prospective students," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 5, no. 6, pp. 23-32, 2019, doi: 10.9781/ijimai.2019.07.001.

[2] C. Dimartino, and S. B. Jessen, "Selling school: The marketing of public education," Teachers College Press, 2018.

[3] M. Coccoli, A. Guercio, P. Maresca, and L Stanganelli, "Smarter universities: A vision for the fast changing digital era," *Journal of Visual Languages & Computing*, vol. 25, no. 6, pp. 1003-1011, 2014.

[4] D-E. Gibaja-Romero, "Interacciones en economías de plataformas," in Perspectivas de la Industria 4.0, AlfaOmega, 2019.

[5] A. S. S. Tota, and M. C. U. Aguirre, "Marketing digital en universidades privadas en el estado Zulia," *Poliantea* vol. 13, no. 24, pp. 5-26, 2017.

[6] M. Yadav, "Social media as a marketing tool: Opportunities and challenges," *Indian Journal of Marketing*, vol. 47, no. 3, pp. 16-28, 2017.

[7] D. Vergidis, and C. Panagiotakopoulos, "Student Dropout at the Hellenic Open University: Evaluation of the Graduate Program, Studies in Education," *The International Review of Research in Open and Distributed Learning*, vol. 3, no. 2, 2002.

[8] D. Airey, and D. P. Stergiou, "Returning to education: A stressful experience?," in *Lifelong Learning for Tourism*, Routledge, pp. 51-69, 2017.

[9] A. Yasmin, S. Tasneem, and K. Fatema, "Effectiveness of digital marketing in the challenging age: An empirical study," *International Journal of Management Science and Business Administration*, vol. 1, no. 5, pp. 69- 80, 2015.

[10] J. Avorn, "Academic detailing: "marketing" the best evidence to clinicians," *Jama*, vol. 317, no. 4, pp. 361-362, 2017.

[11] F. Pucciarelli and A. Kaplan, "Competition and strategy in higher education: Managing complexity and uncertainty," *Business Horizons*, vol. 59, no. 3, pp. 311-320, 2016.

[12] M. E. M. Rosay, "La deserción en el posgrado: estudio comparativo entre maestristas de una universidad pública y privada," in Congresos CLABES, 2017.

[13] G. R. Johnson, C. Jubenville, and B. Goss, "Using institutional selection factors to develop recruiting profiles: Marketing small, private colleges and universities to prospective student athletes," *Journal of Marketing for Higher Education*, vol. 19, no. 1, pp. 1-25, 2009.

[14] R. Pizarro-Milian, "What's for sale at Canadian Universities? A mixed-methods analysis of promotional strategies," *Higher Education Quarterly*, vol. 71, no. 1, pp. 53-74, 2017.

[15] C. Lubienski, "Marketing schools: Consumer goods and competitive incentives for consumer information," *Education and Urban Society*, vol. 40, no. 1, pp. 118-141, 2007.

[16] K.K. Tsiptsis and A. Chorianopoulos, "Data mining techniques in CRM: inside customer segmentation" in John Wiley & Sons. 2011.

[17] C. Marcus, "A practical yet meaningful approach to customer segmentation", *Journal of consumer marketing*, 2008.

[18] R. S. Wu and P. H. Chou, "Customer segmentation of multiple category data in e-commerce using a soft-clustering approach," Electronic Commerce Research and Applications, vol. 10, no. 3, pp. 331-341, 2011.

[19] R. Burkholz, "Entwicklung einer Buyer Persona," in Marketing and Sales Automation (pp. 49-58). Springer Gabler, Wiesbaden, 2017.

[20] A. Revella, "Buyer personas: how to gain insight into your customer's expectations, align your marketing strategies, and win more business," In John Wiley & Sons, 2015.

[21] L. Rokach and O. Maimon, "Clustering methods" in Data mining and knowledge discovery handbook (pp. 321-352). Springer, Boston, MA, 2005.

[22] A. Bhat, "K-medoids clustering using partitioning around medoids for performing face recognition" *International Journal of Soft Computing, Mathematics and Control*, vol. 3, no. 3, pp. 1-12, 2014.

[23] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE" *Journal of machine learning research*, vol. 9, no. 11, 2008.

Eva Asensio Del Arco

Degree and PhD in Economic and Business Sciences from the Complutense University of Madrid (UCM). She has completed a Master in Humanities at the Francisco De Vitoria University (UFV). She taught at the UCM, at the UFV and at the Open University of Catalonia. Furthermore, she has worked for more than 20 years as a consultant in outstanding innovation projects in companies, foundations and public and private organizations. She has extensive teaching experience in public and private universities in the Business area.

Alejandro Almeida Márquez

Graduated in Economics and Business Administration, University of Extremadura and PhD in Economics, International University of Andalusia and the University of Huelva. He developed his thesis on spatial econometric models and has been a researcher at the Spanish Entrepreneurship Research Group of the University of Huelva. He is currently an undergraduate and graduate lecturer and member of the Data Analysis Applied to Economics and Business research group at UNIR.

Aida Galiano Martínez

PhD in Economics with mention of "European Doctor", University of Alicante, 2009. Affiliate research student, University College of London, UK (2007). Master's in quantitative economics from the International Quantitative Economic Doctorate (QeD) program, University of Alicante, 2005. Bachelor of Economics, University of Alicante. 2001. Experience in economic consulting in conducting economic sectorial and impact studies, Economic Strategies and Initiatives, Spin-off of the University of Zaragoza, 2008 – 2012. Associate Professor in Quantitative Analysis in Economics and Business at International University of La Rioja (UNIR). Accredited as Contratado doctor y professor de Universidad privada by ANECA (2018).

Juan Manuel Martín Álvarez

Phd in Economics with training in quantitative analysis for decision making. He has extensive experience as a teacher in public and private universities in the areas of Accounting, Finance, Statistics and Econometrics. He has worked in the development of business intelligence solutions for vending. Associate Professor in Quantitative Analysis in Economics and Business at International University of La Rioja (UNIR). Academic Coordinator of the Master in Business Intelligence at UNIR.

Research Collaboration Influence Analysis Using Dynamic Co-authorship and Citation Networks

Sidra Razzaq[1], Ahmad Kamran Malik[1]*, Basit Raza[1], Hasan Ali Khattak[2]*, Giomar W. Moscoso Zegarra[3], Yvan Díaz Zelada[3]

[1] Department of Computer Science, COMSATS University Islamabad, Islamabad (Pakistan)
[2] National University of Sciences and Technology (NUST), Islamabad (Pakistan)
[3] Escuela de Posgrado Newman, Tacna (Peru)

Received 20 April 2021 | Accepted 20 February 2022 | Published 1 March 2022

unir
LA UNIVERSIDAD
EN INTERNET

Abstract

Collaborative research is increasing in terms of publications, skills, and formal interactions, which certainly makes it the hotspot in both academia and the industrial sector. Knowing the factors and behavior of dynamic collaboration network provides insights that helps in improving the researcher's profile and coordinator's productivity of research. Despite rapid developments in the research collaboration process with various outcomes, its validity is still difficult to address. Existing approaches have used bibliometric network analysis with different aspects to understand collaboration patterns that measure the quality of their corresponding relationships. At this point in time, we would like to investigate an efficient method to outline the credibility of findings in publication—author relations. In this research, we propose a new collaboration method to analyze the structure of research articles using four types of graphs for discerning authors' influence. We apply different combinations of network relationships and bibliometric analysis on the G-index parameter to disclose their interrelated differences. Our model is designed to find the dynamic indicators of co-authored collaboration with an influence on the author's behavior in terms of change in research area/interest. In the research we investigate the dynamic relations in an academic field using metadata of openly available articles and collaborating international authors in interrelated areas/domains. Based on filtered evidence of relationship networks and their statistical results, the research shows an increment in productivity and better influence over time.

Keywords

Collaborative Research Analysis, Influence Analysis, Network Centrality, Quality Of Research, Statistical Analysis.

DOI: 10.9781/ijimai.2022.03.001

I. Introduction

Interaction among scientists is vital for novelty and productivity in their research areas. Research collaboration (RC) is seen as a primary indicator of certain effective means of cooperation among states. Besides, by being actively involved, professionals can quickly grasp the essentials of a large research area [1]. The dramatic growth in scientific collaborations provides further insight into the evolution of social sciences research collaboration. Sharing of knowledge increases the chances of spreading the benefits of protocols and ideas from one region to another [2]. The collaborators are using this approach to study the network research work both for their own progress and to have a better understanding of networked research. This approach is required because scholars are providing transparent transformation information about careers, research areas, interests, and their key factors of productivity: their publications record and influences [13].

Researchers somehow go beyond the study of network research to favor the effects of different types of relations: co-authors, productivity, and their influence. While the fact that most of the recent studies have been geared around quality appraisal is an unavoidable management feature at all levels. It encourages progress in growth analysis, quality, and performance of researchers for spreading interests and knowledge [5]. Such a professional assessment, which should be focused on the findings of the output of the scientist, is important not only for performance evaluation but also for gaining a high reputation in the research community [9]. To evaluate a scholar's quality of work, several studies suggest quantifying the writing practices as a good measure of a scholar's efficiency. The general idea is that if the author publishes and these articles are quoted, a scientist will receive good evaluation in the research community. In addition, the citations count characterizes the number of publications [14], consisting of various statistical methods such as co-citation, co-authors, etc. that evaluate data in the scientific corpus to provide a quantitative understanding of the growing literature and the flow of knowledge in an individual area. This allows investigators to overcome problems that hinder the achievement of progress in their profiles. Many measures including centrality have been used in the literature [18], [19], [38] to achieve these goals.

Analysis of scientific articles and citation collaboration has an extensive history of exploring explicitly the scientific outcomes; however, interests of collaborative researchers between authors and their publications and research areas are less explored. Such alliances also originated from social networks and are actively promoted

* Corresponding author.

E-mail addresses: ahmad.kamran@comsats.edu.pk (MA. Kamran), hasan.alikhattak@seecs.edu.pk (HA. Khattak).

across them because they transcend global, organizational, and administrative boundaries. The quality of community-based research is examined through the published articles and their references over years and areas [4] [3].

There has been a substantial rise in the number of partnerships among scientists in the science environment. They are exhibiting their information exchange practices by collectively publishing papers, which is an indicator of knowledge formation. Authors in [7] noticed that the development of scientific knowledge, including new theories, research problems, and research ideas are an important result of scientific partnerships. Despite all traditional analysis of researchers' communication through citation patterns, collaboration involves a rational breach to accelerate the efficiency and time needed for challenging discoveries. As a consequence, it was reported that the growing awareness of collaboration in research has contributed to a strong focus on the issue of collaboration [2].

Collaborative qualitative research is being judged for lacking transparency in scientific procedures and analysis. In fact; is it necessary to investigate the influence of how co-authorship/collaboration has developed in the social sciences? There is a demonstrable increase in the individual disciplines in which the task of scientific impact prediction is formulated as using standard methods for predicting performance. [12] [10]. However, few researchers examine or compare the factors across disciplines. Either, they focus on the development in a few fields or they investigate the main branch of research: the physical life, social sciences, and humanities which means that developments in the individual field are not visible [6].

Recent studies have also faced challenges in evaluation of correlation analysis over time [15]–[17]. The challenges involved are: increase in visibility of citation rates as research output, publications in other research areas due to less citations, and publications, impact on scholar performance. Moreover, few authors have examined or compared the factors across disciplines [39], [40].

On the contrary, it is beneficial to suggest that collaboration can be improved further by creating effective network relations in different domains. The influence of cooperation established in social sciences with the shift in research areas must be explored. Moreover, by considering the previous works, we are looking into how different research relations support each other in network language, finding their influence by considering issues faced to date. If the network responds is great; productivity and influence increase and there will be more opportunities to publish.

We aim to provide a new collaboration model to analyze the characteristics of co-authored research articles. We propose new performance metrics (active area, self and average citations, paper score and authors score) for inspecting the quantifiable analysis collaboration. In this study we measure the combinations of performance metrics to explore four types of relationship networks: publication to author, author to author, publication to publication, and publication to research areas, to find the relationship between centrality factors and g-index metrics as the key proxy in our collaboration model by considering the shortcomings of previous works.

Our main points of investigation, in this research, are the relevancy effect on scholars'articles in their research area (which was ignored by previous researchers), their affected quality of research, and the impact on scholars'articles' performance in terms of citations in their area.

The remaining sections of this work are structured as follows. Section II discusses the studies conducted previously. In section III the dataset details and mathematical formulation along with the proposed methodology are presented. Section IV demonstrates the results and their analysis and offers a discussion. Finally, the findings of this work along with future directions are presented in section V.

II. Literature Review

Complex research communication networks are being looked at as a new framework that is used in multiple studies and practical applications of social network analysis. Collaboration has been studied by researchers of various fields, with the aim of performing comprehensive analysis of three decades of co-authorship network data [20]. Similarly, authors have also proposed the inverted U-shaped collaboration network by considering the citations at individual level.

With the combination of cognitive and relational dimensions of social capital, a positive effect of relationships strengthens the ties of networks [8]. The higher the relationship value, the lower the biasedness, although this value makes for a mixture of strong or weak ties. Many researchers did not take notice of the ambiguity of researcher's names at the individual level, indirect ties, global networks, and other aspects of knowledge creation networks [21]. The ambiguity of author's names has been resolved with the selection of the preprocessing bibliometric method, which enables a better representation of the co-authorship collaboration network using digital bibliography and library project data sources [33], [43].

The bibliometric cooperation in social sciences has gained accolades at both domestic and international levels. Here researchers focus on two types of bibliometric methods, namely: the parametric [22]–[24] and descriptive models [25]–[28], in which they use research articles, citations, and their collaboration networks. Challenges such as measurement error, performance, scaling, dimensions, normalization, and quality were also addressed. Similarly, new strong correlation methods are proposed with slight modifications in social network algorithms [29] to influence the network and to evaluate their importance in the research community through centrality measures.

The normalized centrality measures and average ties strength have a strong effect on scientific academic performance, in terms of h-index and g-index. However, the results showed that researchers at national level perform better than those at international level in the network. As the node gains the central position in the network, it determines the opportunity to collaborate and share knowledge in terms of betweenness centrality but not of improving its performance. Usually the authors with high betweenness centrality are implied to have more importance. However, the weighted number of citations shows the influence of only significant papers and not the authors in the community [11], [30], [31]. Furthermore, studies of Italian academia [37] show that the data sources have strong influence on network analysis to test the scientific performance of the researchers. It has been observed that small-world structure, both at national and international level, characterizes the networks with three popular data sources: current index to statistics (CIS) [32], web of science, and national funded projects. However, in general, CIS is more widely used in international research topics [33].

In [34], the authors conducted social network analysis (SNA) to show that emerging scientific topics receive less attention from researchers compared to the subtopics derived from the main scientific topics. The authors discussed the co-authorship network of forest entrepreneurship, which is a new and emerging field of study. They concluded that the topic of forest entrepreneurship is understudied compared to subtopics such as innovation forest, forest industry, and the policy of forest entrepreneurship. Normally, the best method to determine the performance of a researcher is measuring centrality or community detection. However, scholars mostly work on different projects and they have different roles in each one. In [35], the authors concentrate on the overlapping of scholars in projects and the roles they play therein. Based on the comparison, in this study, it need to be analyzed whether the performance of the researchers correlates with their contribution patterns in the projects or not.

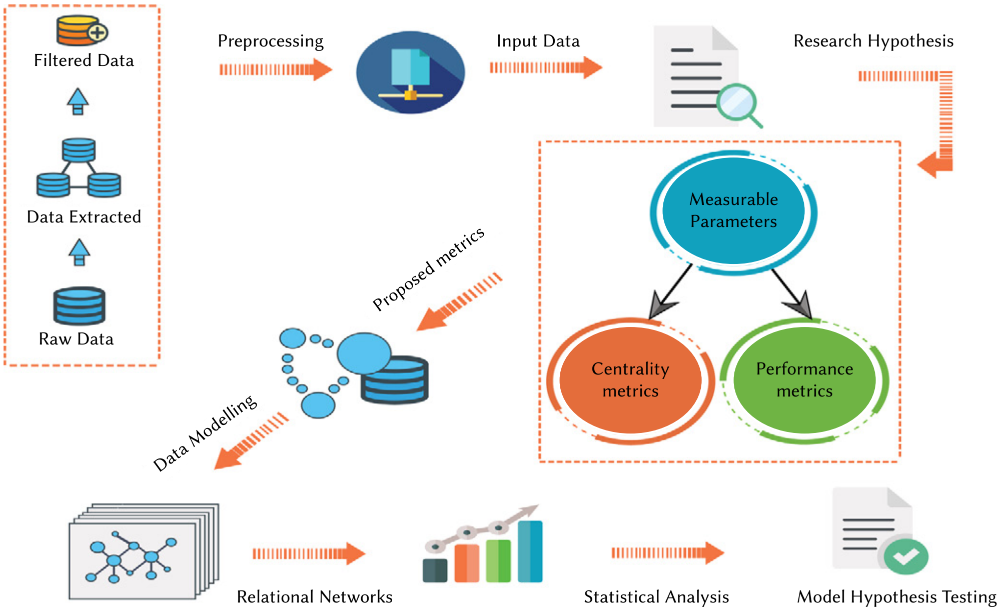


Fig. 1. Proposed collaboration model.

Psychometric rasch model has been used by authors to evaluate the researcher's performance at an individual level by considering the control variable errors of bibliometric research [36]. They achieved promising performance for percentage, fractional counting, and normalization, while ignoring the total citations, multiple co-authorship, and distribution assumptions. Moreover, in article [42], the authors proposed a method for finding a domain level influence. These researchers focus on author citations through the information linkage process. They use an assumption that author-nodes are reflected as more influential if they have a great self-influence. The authors in [44] study the benefits of collaboration in scientific studies. They show, how collaboration can affect the position of authors in a co-authorship network and increases productivity as well as the influence of the authors. The authors state that researchers have recently become more collaborative. They conclude that collaboration in scientific studies improves research as well as stabilizes the relationship between researchers. The impact of network size and correctness on researchers' productivity and influence has been studied in [45]. Assuming the number of publications as productivity and the number of citations as an influence, the researcher's analyze the times when the scholars were productive and influential.

III. Research Collaboration Influence Analysis

In this section, we address in detail the proposed methodology of research collaboration influence analysis.

Collaboration/joint collections of networks used in different research fields are considered to be a structural network where all objects are considered to be linked and their degree value is important. For instance, authors being tied to each other result in non-redundancy of some critical information flows in their ties because of some structural holes. This may enable scholars to enhance the research to accrue potential opportunities for controlling the information flows among them. As described earlier in the introduction part, the numbers of citations and publications are positively linked, which points to quality of research work.

To address the collaborative research problems, the proposed scheme represented in Fig. 1 is used. The key role of this collaboration model is to overcome the issues of researchers' growing professional relationships and the scientific influence of their collaboration networks in a research community. Many years ago, Garfield [41]; acknowledged that scientific enterprise has increased and become more complex. If the complexity and collaboration change, the quality and content of citations should also evolve. This paper explores the characteristics of international research articles and contributes to the enhancement of correlation analysis.

For collaborative research analysis, we extracted data via Arnet-Miner to perform the analysis of centrality measures, g-index, and introduced metrics: active area, self-citations, paper score, and authors score. Furthermore, we propose a new collaborative approach to find the better influence with performance metrics. Finally, we draw a correlation analysis to show the effectiveness of our model. We create four relationship graphs the publication—author, co—author, publication—publication and author—research area, to give leverage to the relationships of quantifiable research. In this regard, Python language, Network X, and SPSS tools are used for visualization and metrics calculation.

A. Data Pre-processing

The dataset used in this research is mostly taken from https://www.aminer.org/lab-datasets/soinf/ provided by [42], consisting of publications, co-authors, citations, and research area. We used this raw data to extract the required graphs from Arnet-Miner to bring into form that can be used for our research analysis. The raw data contains information on 2555 publications. The information includes publication title, year of publication, conference/journal, and authors' names. The publications are from 10 different research areas of computer science. The following list shows areas of publications with the codes:

- Area 75: Information retrieval
- Area 131: Bayesian networks / Belief function
- Area 107: Web services
- Area 199: Natural language system / Statistical machine translation
- Area 16: Data mining / Association rules
- Area 24: Database systems / XML data
- Area 145: Semantic web / Description logics
- Area 144: Web mining / Information fusion
- Area 162: Machine learning
- Area 182: Pattern recognition / Image analysis

In addition, the raw data include a citation relationship among the publications. The citation relationship shows the papers that have referenced a specific paper. The dataset provides 6101 citation relationships. Furthermore, we generated our own graphs to show the relationship between publications and research areas. We have performed the preprocessing whereby we removed duplicate records, blank spaces, and other unnecessary data and organized the data according to our requirements.

B. Measurable Parameters

There are many network properties that describe how nodes are connected to each other on a network. The SNA has many measurement types to systematically characterize nodes in the networks. A few measures used in our research are described below to find which node is important in the network and to find an influence on the other nodes. We determine the importance of nodes by calculating their score and then finding their position within a network. Here, importance means any effective authors and papers with respect to citations.

1. Degree Centrality

Degree centrality is considered as the simplest and most common way of finding important nodes in a network. For example, if a vertex has five edges, then we say that it has degree 5. Furthermore, in directed graphs, there are two kinds of degrees: in-degree and out-degree. The indegree shows the number of edges coming from one vertex to another, and out-degree is the edges originating from the vertex going outwards. Equation 1 is defined to find the degree of nodes. We are only considering in-degree. The formula for evaluating the normalized degree centrality is as follows; where d(G) represents the degree of a node (like number of papers that cited this node/paper) and N represents total nodes in the network (like total number of papers).

$$C_D(G) = \frac{d(G)}{N-1} \tag{1}$$

2. Eigenvector Centrality

Eigenvector centrality is a kind of extension of degree centrality. It is particularly focused on two things: the node itself and its neighbor's. Here, we calculate values with 0 and 1 only. The closer the value to 1, the higher the centrality. We use this centrality to find which nodes take information to other nodes quickly. Equation 2 is used for finding the centrality of this measure. Here $A = (a_{v,t})$ presents the adjacency matrix, i.e. $a_{v,t} = 1$ if vertex v is linked directly to vertex t and is 0 if otherwise.

$$x_t = \frac{1}{(\lambda)} \sum_{t \epsilon G} a_{v,t} x_t \tag{2}$$

3. Closeness Centrality

Closeness centrality indicates how close a node is to the other nodes in the entire network. It is highly effective for calculating the shortest possible paths among all nodes before assigning each node a score based on its sum of shortest paths. In equation 3, D (y, x) are used to find distance between y and x.

$$C(x) = \frac{1}{\sum_{x \neq y} D(y,x)} \tag{3}$$

4. Harmonic Centrality

Harmonic centrality is a variant of closeness centrality. Instead of summing the distances of a node to all other nodes, the harmonic centrality algorithm sums the inverse of those distances. It can display interesting results, especially for the top nodes of the graphs. Simply referring to equation 4, if the value is equal to zero then there is no path between x and y.

$$H(x) = \sum_{y \neq x} \frac{1}{D(y,x)} \tag{4}$$

5. Betweenness Centrality

Betweenness centrality indicates the extent to which a node lies on the shortest path among other nodes in the entire network. It indicates nodes that can primarily act as bridges between nodes and can be used for finding the individuals who influence the flow around a system. In equation 5, σ_{st} represents a pair of vertices used to compute the shortest path from node s to node t. V is used to define the vertex in a network.

$$C_b(v) = \sum_{s \neq v \neq t \epsilon v} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{5}$$

6. Clustering Co-efficient

The clustering coefficient is a measure of how likely it is that two nodes that are connected are part of a larger highly connected group of nodes. The global version gives an overall indication of the clustering in the whole network and can be applied to both undirected and directed networks, whereas the local version gives an indication of the embeddedness of single nodes. In equation 6, clustering coefficient C_i is shown for all vertices n.

$$C = \frac{1}{n} \sum_{i=1}^{n} C_i \tag{6}$$

C. Data Modeling (Research Collaboration)

For decades, academic science research used collaboration networks as a proxy. Apparently, the usage of bibliometric data analysis and accessibility makes the collaboration more interesting for the future research. Moreover, many scientists have been actively involved in co-authorship, for a strong relationship among researchers in their academic careers. When we use a network of researchers, they are always connected with solid connections between two or more persons, if they have a strong relation. Relational networks provide with an insight about which individual or specific node has a great influence is strongly connected. Furthermore, many activities have been performed by researchers with the assumption of receiving positive effects of profiles; the co-authorship or collaboration suggests more relevant or accurate influence and authors getting publications in many fields of related research.

If this is the case then it can be stated that they might be having different interests or maybe just trying to add co-authors as a matter of getting increased publications. What needs to be understood is whether they have any accurate citations of research conducted so far in different domains. It would definitely be worthwhile to know about researchers' performance until higher ranks, based primarily on their academic activities, research interests, and especially research area changes with the passage of time. This literature gap motivates us to study the scholar's in specific field. It also makes it feasible for readers to evaluate the credibility of publications and authors.

For this purpose, we also need to understand the influence of citations through mathematical modeling. In the research we investigate the dynamic relations in an academic field using metadata of openly available articles in inter-related areas/domains. We develop network relationship graphs based on co-authorship, publications through their number of citations (self and original), and research areas in which they are actively involved. We use centrality measures (helping us to identify the real influencer's) on introduced performance metrics to find a correlation between evaluated results and g-index. In addition, based on the lack of collaboration analysis in existing studies, this work provides research collaboration development across other scientific fields that can be effective in future research.

D. Constructed Parameters

1. Average Citations

In our filtered data, each article has one or more authors. To get better citation information, we calculated an average citation score by equation 7 for each author. Here, variable denoted by *ACS* represents "average citation score" of an author, *CPA* represents "citations of each paper authored", and *NP* represents "number of papers" published by the author. Average depends on total number of papers published by an author, so it is interesting to compare with total citations to know whether an author's citations are from few paper or many papers.

$$CS_i = \frac{\sum_{j=1}^{NP}(CPA_j)}{NP_i} \tag{7}$$

Algorithm 1: Creating Average Citations
Input: Publications Authors Bipartite Graph
Output: Average Citations Values for each publication by an author

```
    find citing papers
    add all citations
Divide total citations by total number of papers return average citations
```

2. Active Area of Author

The active area of an author is the research domain in which a researcher has been recently and most frequently performing research. For this purpose, we examined the publications of each researcher to determine their active areas.

Algorithm 2: Finding Active Area
Input: Average Citations Values
Output: Authors Active Areas

```
create an array of paper names
Paper names exist in publication area nodes
    get paper and year from selected nodes
    subtract current year from publication year
    area score = dividing constant factor by year
area in publication area
    find index of selected areas
    add score value to index area
get maximum area score
find index of active score area
return authors active areas list
```

Now, owing to the change of trends, and changing areas with respect to time, the researcher's productivity could be compromised. The trending area and publication time matters in research. It could be possible that an author published papers in one area and later in another area, because of lost of interest in the first area. This can create a great variance in results when comparing productivity of authors in general and in specific areas. For this purpose, we need to take into account the score of each paper depending on its publication time using equation 8. The variable "Age" shows how old the paper is. Age represents a difference of value between the publication year and the current year. We have given K = 0.2 a constant value, which is taken into consideration to reduce the variations between the score of papers published in two consecutive years. This is used to avoid the counter values, that produces 20% difference value for a max of two years.

$$Score_i = \frac{1}{Age} \times K \tag{8}$$

To find the active area of a researcher, we formulated the equation 9. Here NPA represents "number of publications in the area" for author j, AA represents "active area" i, and Score represents the sum of scores of all papers of the author j in the area i. The area i having max value is selected as the authors active area.

$$AA_i = \max(NPA_j + Score_j) \tag{9}$$

3. Citations

In research, there are two types of citations. The first is original citations by a random researcher. The second type is self-citations quoted by one of the authors of a paper. We categorized the total citations of a publication into original and self-citations and analyzed the influence or connection between network nodes. It also helped us in reranking (paper score) the publications based on citations in addition to research area.

Algorithm 3: Calculating Citations
Input: Authors Active Areas
Output: Original and Self Citation Values

```
create an array of paper edges from active areas
    get all neighbor papers that cited an original paper
    get total citations by adding 1 for each neighbor (citing) paper
finding original papers from neighbors
    get original citations that do not include authors
Self citations = total citations – Original citations
    return original and self citations list
```

For example, an author from the database "Fuad.M.Alkoot" published a total of one paper that got five citations from which only three papers originally cited his paper and two citations were made by self-citations. To calculate the self-citations, we used equation 10, where Symbols SC_i, T_c, and O_c represent self-citations, total citations, and original citations.

$$SC_i = T_c - O_c \tag{10}$$

4. Paper/author Ranking Score

We rank or rate publications through citations and active area of the author. This helps in finding author ranking score. This ranking will help students and readers select and evaluate a publication based on its ranking and also to evaluate a researcher in a specific research area.

Algorithm 4: Creating Paper Score
Input: Original and Self Citation, active area Values
Output: Paper Ranking List

```
finding paper rank
    assign area score 1 or 0.7 to paper
    get paper score against original, self citations, and
  area score using Equation 12
return paper ranking list
```

The technique used in equation 11 is giving a value of 1 in the case of author's active area and paper belonging to the same area and 0.7 in the case of different areas. Here S_a represents the score of each author against publications. It was considered, a weighted value if there were four authors on a paper, and if they were interested in the same area then we assigned 1 to all.

$$S_a = \begin{cases} 1 & \text{if authors belongs to same area} \\ 0.7 & \text{otherwise} \end{cases} \tag{11}$$

We used the categorized citations of the publication while ranking it. In equation 12, C_s represents self citations, C_o represents

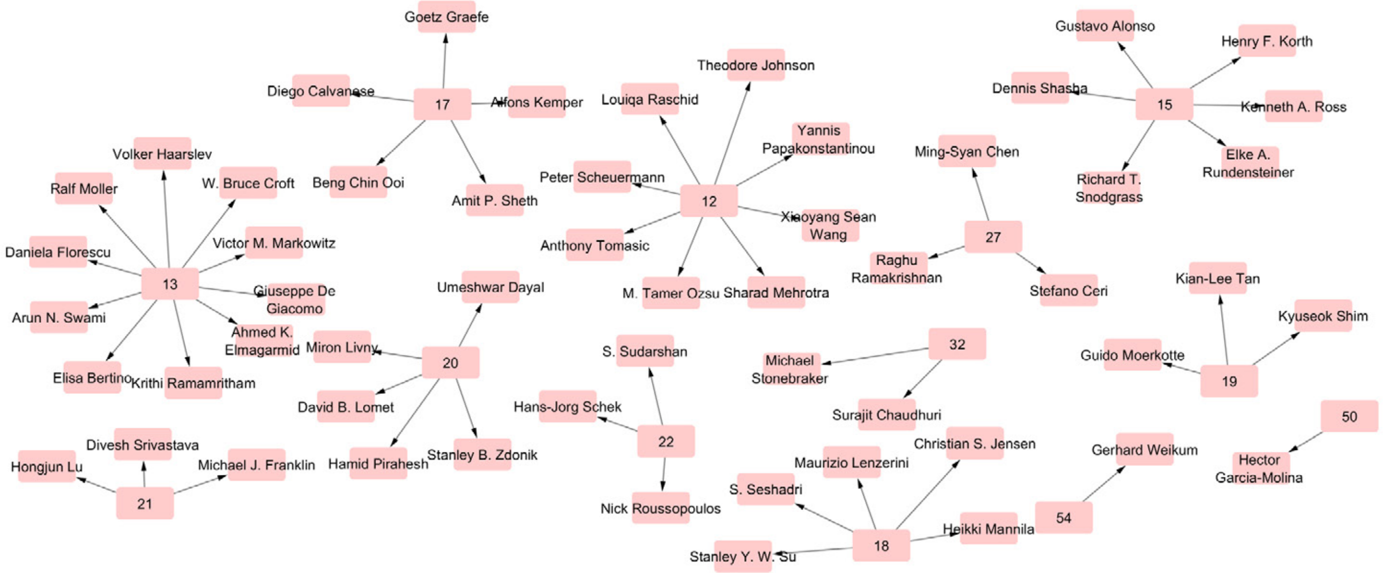


Fig. 2. Publications to authors network.

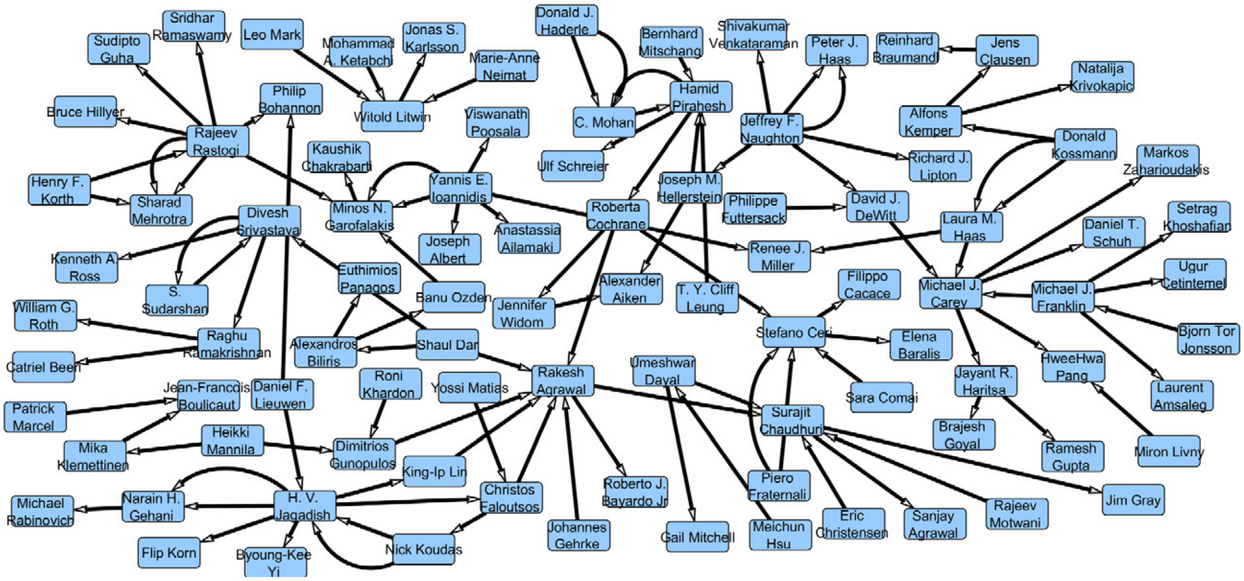


Fig. 3. Co-authorship network.

original citations, and S_a represents area score. In this equation, the hyper-parameters have been assigned to the model to categorize the citations. The value a is related to the original citations through which a paper would have achieved higher influence in the domain, so its value is high. On the other hand, b is related to the self-citations whose weightage is normally small. For experiments we used, 0.6 and 0.4 values for a and b respectively.

$$PS = (a \times C_o + b \times C_s + \frac{\sum(S_a)}{Number of S_a}) \qquad (12)$$

IV. Results and Discussion

To show the validity and productivity of our proposed methodology, we performed simulations to show an influence analysis of authors with respect to publications, research area, and citations. In this section, we discuss the evaluated data and generated graphs based on evaluated performance metrics. For data analysis and implementation, we used Python language and NetworkX tool in the Spyder anaconda application. Four types of network graphs are generated which include publication-author, co-author, publication-citation, and publication-research area that are presented here.

A. Relationship Networks

Fig. 2 represents the directed network relationship, which presents the relations between two-character nodes: the author's and publications. It shows the relationship of publications with their authors in a network. A certain type of small cluster is shown for each publication using its publication id and author nodes are connected with it. As the graph represents, there are several small clusters in which two or more nodes are involved. Each article id is linked with the corresponding authors and co-authors. We can say that many authors are involved in collaborative research publications in multiple areas.

Fig. 3 represents the author's relationships to each other, which is called co-authorship between them. This relationship shows, which author is connected to others, for example writing a paper together as a co-author or having any kind of contribution. This is how authors connect and form relationships by appearing as co-authors. All co-

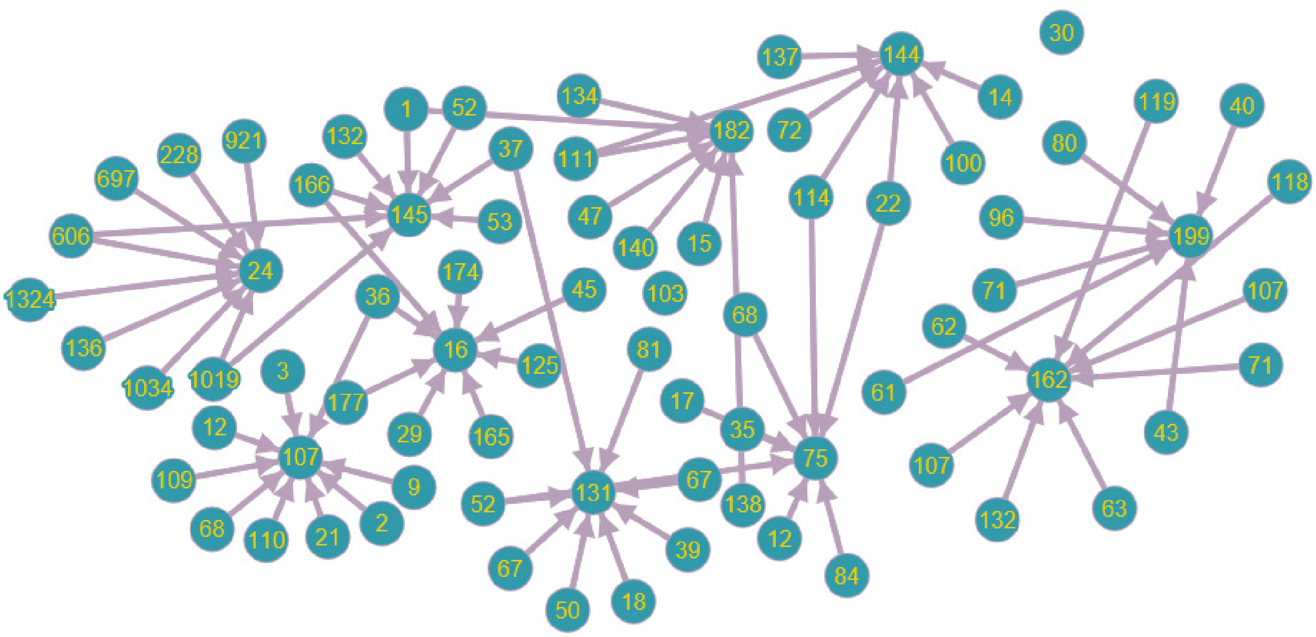


Fig. 4. Publications to area-id network.

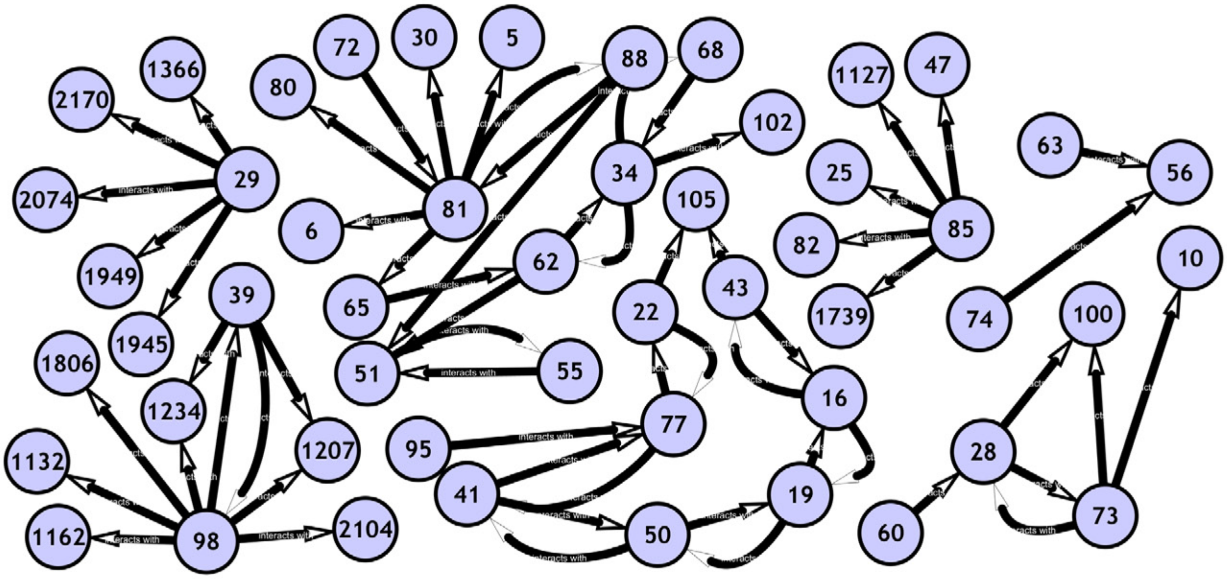


Fig. 5. Publications to publications network.

authors who are directly interacting in an article in different domains can be found in the graph. This is effective for all co-authors in calculating collaborative work scores, for example, if any author of the article has original citations it directly gives benefits to all co-authors.

There is another directed relationship graph which is shown in Fig. 4, presenting the relationship between the authors and their interest area in research publications, which is called the publication to active area relation network. It shows all publications related to similar area collectively. It creates a research area node to which all related nodes are connected. This graph is used to find research area of the publications.

Another directed network relationship graph shows relations with publications cited by another publication which is also called co-citation network. It represents citations that each paper has in other papers. Each publication node generates a directed edge to the corresponding node cited in that publication. Fig. 5, represents the graph; in which each vertex represents publication id and its in-degree can be used to find which publication has what number of citations.

B. Data Analysis

In this section, we present the results evaluated based on our proposed approach. As mentioned in the research methodology, our main goal was to find the paper score and author influence by comparing the network relationships in which the number of papers, authors, research area, and citations play a major role.

We are focused on creating the ranking score for authors and papers. In our data, each publication has one or more authors from different countries; we performed an analysis to know their progress in different research domains. It is observed that researchers change their research interest and with the passage of time they start publishing in another related research area. After this observation, we found which area the researcher had adopted or changed up to now. The active area of an author is the research area in which the researcher has been most recently and most frequently performing research. Because of rapid changes in research trends, researchers willingly change their minds, make collaborations, and publish in trending topics in their academic career to gain advanced knowledge of growing technology and information; however, this may affect the researcher's performance and profile ranking in specific research areas. With the consideration of these facts, we evaluated the active area of authors. Specifically, we depend on data collected from co-authored articles of international collaborators to evaluate the aforementioned factors. We found the research area in which each author has a special interest, as shown in table I, and in which they have recently been active and are making progress.

TABLE I. Selected Researchers Publishing in Specific Research Areas

Authors	Active
Byron Dom	16
Jonas S. Karlsson	24
Lyman Do	24
Younkyung Cha Kang	24
Fegaras	24
Regis Sabbadin	131
Sara Comai	144
Klaus Schild	145
Andrea Schaerf	145
Alessandro Artale	145
Quentin Elhaik	145
Pavel Paclik	182
C. Fairhurst	182
Miles Osborne	199
H. Gregory Silber	199

Because authors tend to work in different areas over a period, the quality of their research in one area may differ from that in another area. The quality of a publication depends on the area of expertise of its authors. If a researcher is not involved greatly in a certain research area, it is possible that his/her interest might have changed and also become focused to work on a few articles instead of many to meet the standards and reputation of an academic field.

Moreover, in the case of citations of an author, Google Scholar shows the sum of citations of all the publications. However, to have a better idea, we calculated different types of citation scores in addition to author's active area and used them for calculating paper score. It is difficult to evaluate a researcher only by the citations, as some publications might have high citations while some may have very low citations, including self and original citations. To, build a researcher's profile, we need to calculate the average citations as well. The Fig. 6 shows the visual representation of researchers performance in terms of their total citations, average citations and number of published papers.

Researchers having many papers and many citations will have lower average citations than the one having small number of papers and many citations. Furthermore, we evaluated authors who received a maximum number of papers but fewer citations as having less influence. Besides, researchers with fewer publications and many citations show highly influential results. This shows, the effectiveness and quality of researchers' publications. An increased number of papers does not help in increasing profile productivity. Instead, a high original citations value boosts one's profile ranking. Furthermore, we have used the active areas and evaluated citation results for the evaluation of researchers' profiles and the quality of their research.

In collaborative research, the concept of types of citations (self/original) becomes influenced in two ways. We consider citations with respect to number and also with respect to research areas. In this research, we focused particularly on two types of citations for finding the effects of citations rate on paper quality. First, the genuine citations by a random researcher, and second self-citations are those cited by one of the authors of papers. This method also helped us in ranking the publications. Because we have created the profiles of researchers based on their active areas, the approach helps other researchers, students, and readers to select and evaluate authors and their publications in selected research areas of computer science.

With the help of collected data and generated graphs, we evaluated the originality score of each paper. Table II, shows the originality score of some publications based on research areas and all citations factors. An author's score can be calculated by summing up all his publication scores. The rank that an author achieves through publications presents the influence of articles on the researcher's profile. The paper score is mainly based on citations and active research area of the author. The higher the number of original citation the higher the paper score. Also, if the paper is from active research area of an author, it increases the paper score. Moreover, researchers having most publications from active area with many citations, but with less self-citations, will get high profile ranking.

As mentioned in the earlier discussion, more self-citations affect the quality of a paper. Fig. 7 shows, the breakdown of citations into self-citations and original citations from total-citations of published papers along with their paper score. With the help of more citations, there is a chance of getting a high-ranking score. However, in many cases the reason behind a lower-than-average score is that self-citation directly affects the quality of articles. If a research is not quoted by other researchers, it will lose its worth and self-citations simply increase

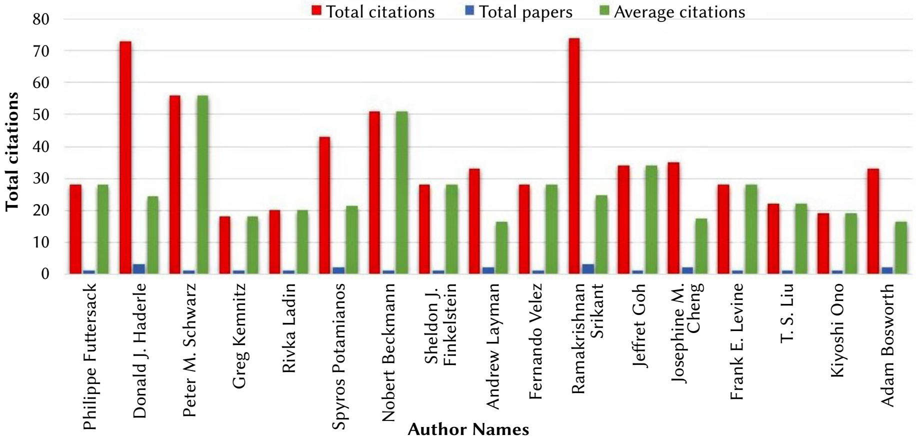


Fig. 6. A comparison of authors total and average citations depending on number of publications.

TABLE II. Comparison of Different Types of Citations and Calculated Paper Score

Paper Title	Total Citations	Original Citations	Self Citations	Paper Score
The R*-Tree: An Efficient and Robust Access Method for Points and Rectangles	51	37	14	28.8
Fast Algorithms for Mining Association Rules in Large Databases	55	53	2	33.6
The hB-Tree: A Multiattribute Indexing Method with Good Guaranteed Performance	15	10	5	9.5
ARIES: A Transaction Recovery Method Supporting Fine-Granularity Locking and Partial Rollbacks Using Write-Ahead Logging	56	36	20	30.6
ARIES/KVL: A Key-Value Locking Method for Concurrency Control of Multiaction Transactions Operating on B-Tree Indexes	26	12	14	8
Mining Association Rules between Sets of Items in Large Databases	32	25	7	18.8
An Approximate Analysis of the LRU and FIFO Buffer Replacement Schemes	6	0	6	3.4
An Effective Hash Based Algorithm for Mining Association Rules	23	20	3	14.2
Efficient and Effective Clustering Methods for Spatial Data Mining	30	23	7	17.6
Recovery and Coherency-Control Protocols for Fast Intersystem Page Transfer and Fine-Granularity Locking in a Shared Disks Transaction Environment	23	14	9	13

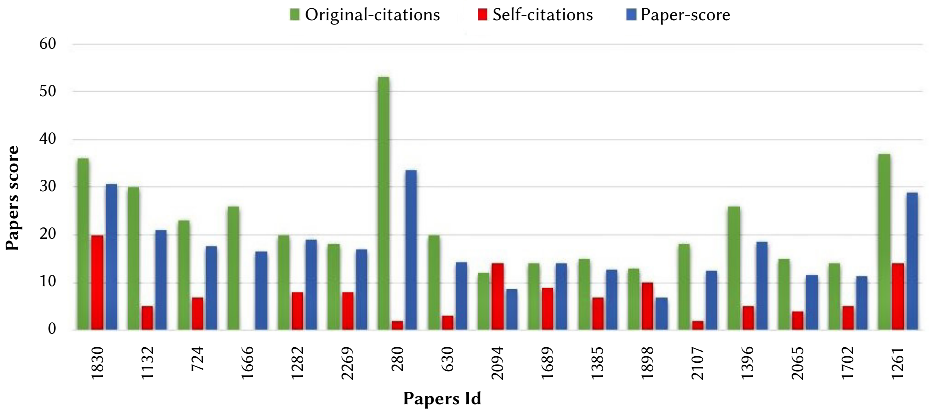


Fig. 7. Comparison of Original Citations, Self Citations, and Paper Score.

the citations rate, which indirectly affects the researcher's profile and paper score. From this representation, we conclude that most of the publications having a high original citations score get more paper score if they are not out of the active research area of authors. The paper-id 280 has a total of 53 original citations with two self-citations getting a high paper score of 33.6. However, paper-id 2094 has more self-citations (14) than original citations (12), which results in a low paper score of 13.8. Some papers having self-citations get a reasonable paper score beause they also have either more original citations or the paper is from active research area of authors. We can conclude that researchers try to raise their profile ranking by making more self-citations, directly affecting their performance and academic career because it is not effective to achieve the higher rank by self-citations.

Author's research area is an important factor to be considered especially in the era of Internet as many researchers are doing collaborative research and some of them change their active research area to get high research rank through another popular research area. In Fig. 8, the circle are created based on area-id and the numbers inside each circle represents the author's name and paper score. Size of the circles represent author score which is collective score of authors' papers ranking, which represents their performance score in the interest area over time. Through all results, we can conclude that citations and change of area plays a major role in scholar profile performance and research productivity throughout their academic career. If the authors remain focused on incrementing their citations, then it would eventually affect their paper quality and their profile.

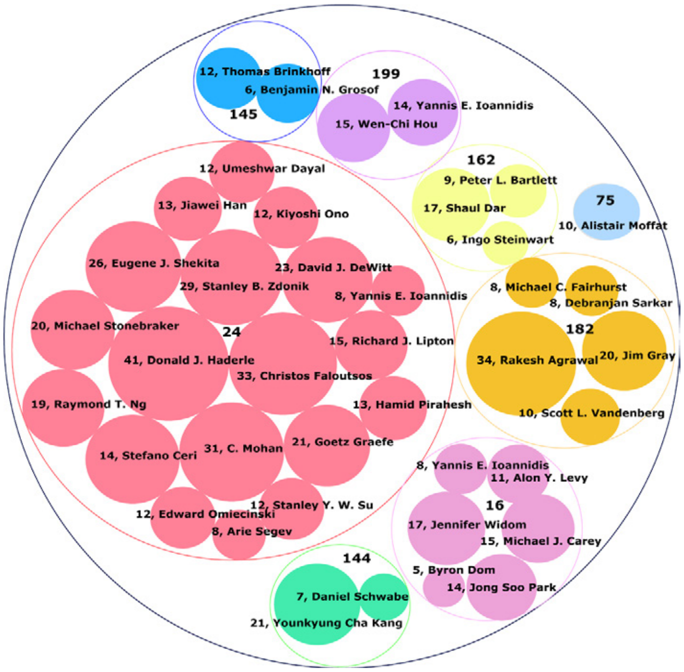


Fig. 8. Circles represent research areas in which inner bubbles are labeled with authors name, paper score, and the size of different inner bubbles represent author's score.

C. Results Analysis

To test and analyze the data, we used paired difference sample statistical tests, which evaluate the difference among groups of means. We evaluated our networks through paired difference analysis with g-index using IBM SPSS. As shown in Fig. 9, it calculates the mean difference value of independent ratio data by the standard error of each value, the freedom degree (df), mean difference, and 95% confidence interval between specified significant value and significant p-value. According to the mentioned analysis, co-authorship relation shows that except for the clustering co-efficient, all other parameters are statistically significant. The reason for not evaluating the value is that results have zero square difference error. This happened because we assigned the weighted values in which the correlation of clustering coefficients had negative results against that relationship. Although the results could be quite reasonable if the values of nodes were assigned with relatively similar information. Other factors mentioned in the figure are strongly correlated with this relationship.

Moreover, in Fig. 10, only one factor is not supporting this relationship with a significant value that is between author-g-index. Scientifically, many collaborations are performed with different scholars or showing strong connections with co-authors, which leads to the high significant value. Thus, in this context it shows negative results. As in the publication area relation graph, the authors were not directly involved, which naturally negatively influences g-index. Furthermore, in Fig. 11 publications to author relation, clustering coefficients and eigenvector centrality are factors that are not effective to support the network relation. The same case occurred, which shows the extent to which the clustering coefficient drives the behavior of instances and results in negative influence. It investigates the structure of scientific collaboration in a whole network, which is why it is scored against the global network graph. Apparently, closeness, harmonic, and clustering co-efficient are strong measures that show positive influence as represented in Fig. 12 except others. This relational network has directed edges and integer values in a global network which makes factors not suitable.

Based on the aforementioned results, it is suggested that the clustering coefficient is the major factor that does not fit in for finding the influence of aforementioned relational networks. The reason behind is that clusters need their neighbor bodies strong for all actors in a network. Two nodes attain the same probability if they

		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference		t	df	Sig. (2-tailed)
					Lower	Upper			
Pair 1	**Closeness Centrality - G - index**	-26.9388350	.0469028	.0046903	-26.9481416	-26.9295285	-5743.541	99	.000
Pair 2	**Degree Centrality - G - index**	-26.9850833	.0286307	.0028631	-26.9907643	-26.9794024	-9425.225	99	.000
Pair 3	**Betweenness Centrality - G - index**	-26.9990826	.0063113	.0006311	-27.0003349	-26.9978303	-42779.274	99	.000
Pair 4	**Harmonic Centrality - G - index**	-19.450	6.147	.615	-20.670	-18.230	-31.642	99	.000
Pair 5	**Eigenvector Centrality - G - index**	-26.9593602	.0880870	.0088087	-26.9768385	-26.9418818	-3060.538	99	.000
Pair 7	**Edge Centrality - G - index**	-26.9980579	.001561	.0001561	-26.9983675	-26.9977482	-172994.051	99	.000

Fig. 9. Paired difference analysis of co-authorship network.

		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference		t	df	Sig. (2-tailed)
					Lower	Upper			
Pair 1	**Authors - g - index**	15.667	49.810	7.189	1.203	30.130	2.179	47	.034
Pair 2	**Degree Centrality - g - index**	-26.8909575	.1425336	.0205729	-26.9323449	-26.8495700	-1307.103	47	.000
Pair 3	**Betweenness Centrality - g - index**	-26.9762951	.1136740	.0164074	-27.0093026	-26.9432876	-1644.151	47	.000
Pair 4	**Harmonic Centrality - g - index**	-2.5138750	3.5720863	.5155863	-3.5511008	-1.4766492	-4.876	47	.000
Pair 5	**Eigenvector Centrality - g - index**	-26.8855014	.0888129	.0128190	-26.9112900	-26.8597128	-2097.311	47	.000
Pair 6	**Clustering Coefficient - g - index**	-26.9267231	.1415273	.0204277	-26.9678183	-26.8856278	-1318.147	47	.000
Pair 7	**Edge Centrality - g - index**	-26.9782408	.0169456	.0024459	-26.9831613	-26.9733203	-11030.033	47	.000

Fig. 10. Paired difference analysis of publication to area network.

		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference		t	df	Sig. (2-tailed)
					Lower	Upper			
Pair 1	**Closeness Centrality - g - index**	-26.8744917	.0732103	.0163703	-26.9087551	-26.8402282	-1641.660	19	.000
Pair 2	**Degree Centrality - g - index**	-26.9407408	.0773169	.0172886	-26.9769262	-26.9045554	-1558.297	19	.000
Pair 3	**Betweenness Centrality - g - index**	-26.9927351	.0287211	.0064222	-27.0061769	-26.9792932	-4203.013	19	.000
Pair 4	**Harmonic Centrality - g - index**	-23.9750	2.4946	.5578	-25.1425	-22.8075	-42.981	19	.000
Pair 6	**Edge Centrality - g - index**	-26.9838625	.0109549	.0024496	-26.9889895	-26.9787354	-11015.713	19	.000

Fig. 11. Paired difference analysis of publication to author network.

		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference		t	df	Sig. (2-tailed)
					Lower	Upper			
Pair 1	**Closeness Centrality - g - index**	-26.989	.104	.011	-27.011	-26.968	-2483.000	91	.000
Pair 4	**Harmonic Centrality - g - index**	8.000	8.557	.892	6.228	9.772	8.967	91	.000
Pair 6	**Edge Centrality - g - index**	-26.870	.339	.035	-26.940	-26.940	-761.086	91	.000

Fig. 12. Paired difference analysis of publication to publications network.

are adjacent to each other. At the same time, it shows a significant influence on the publication citation network. The negative (greater) result of the eigenvector and other parameters not mentioned in the publication citation network are somehow counter-intuitive. Their relationship with g-index could be expected to be better. Another variable could be affected as a result of excessive data values of the comparison variable. It is also not possible for all publications to be connected because of self-citations. We can say that all other measures actively affect researchers' performance with respect to g-index. However, the betweenness variable does not differ largely as compared to other variables. So, with respect to other measures except for the betweenness centrality, it is stated that represented parameters are more effective in the context of all relationships.

Furthermore, we have performed correlation analysis t-test between SNA measures and g-index using the Python language NetworkX tool. To remove the biasedness, all data values were taken randomly for calculation and analysis. These results have slightly different values. In Fig. 13 we calculate the relationship between co-authors. In this analysis, all variables show a strong influence against performance measure except the relations between betweenness centrality and g-index. Based on this, we can say; that in this context, the mentioned significant variables are more effective with respect to the researcher's performance.

Furthermore, in Fig. 14, the relation between publications that are citations of researchers shows that the betweenness centrality, harmonic centrality, eigenvector centrality, edge centrality, and clustering coefficient are significant values. On the other hand, only the degree of centrality is not significant in this scenario. All significant variables exert a positive impact on researchers' performance with g-index. Likewise, we performed the test on publication —area and publication —author relationships as shown in Figs. 15 and 16. Here, all variables are statistically significant except degree and betweenness centrality and not effective to use in both scenarios. We could say that all variables have a positive impact on researchers' performance and profiles with a 95% confidence interval.

Based on all analysis performed, we can conclude that betweenness centrality is the only variable that is not significant in two scenarios, publication to area and publication to author and does not show a positive impact on researcher's performance with respect to g-index. Betweenness centrality is increased by 0.01, which also increases the g-index value. But if we add more authors, it may change centrality measures values. In addition, the reason of betweenness not showing the significant value is the dominancy of high values of researchers. For this reason, smaller node values around it take higher betweenness value and give other nodes the lower values.

Co-authorship (T-test results)

	0	1	2	3	4	5	6	7
0	Closeness Centrality		0.00	0.00	0.00	0.00	0.00	0.00
1	Degree Centrality			0.00	0.00	0.05	0.00	0.00
2	Betweenness Centrality				0.00	0.00	0.13	0.02
3	Harmonic Centrality					0.00	0.00	0.00
4	Eigon Centrality						0.00	0.00
5	Clustering Centrality							0.00
6	Edge Centrality							0.00
7	G-index	Degree	Betweenness	Harmonic	Eigon	Clustering	Edge	G-index

P-value < 0.05

Fig. 13. Correlation analysis for co-authorship relation.

Publications to Citations (T-test results)

	0	1	2	3	4	5	6	7
0	Closeness Centrality		0.00	0.00	0.00	0.00	0.00	0.00
1	Degree Centrality			0.00	0.00	0.98	0.00	0.00
2	Betweenness Centrality				0.00	0.00	0.00	0.00
3	Harmonic Centrality					0.00	0.00	0.00
4	Eigon Centrality						0.00	0.00
5	Clustering Centrality							0.00
6	Edge Centrality							0.00
7	G-index	Degree	Betweenness	Harmonic	Eigon	Clustering	Edge	G-index

P-value < 0.05

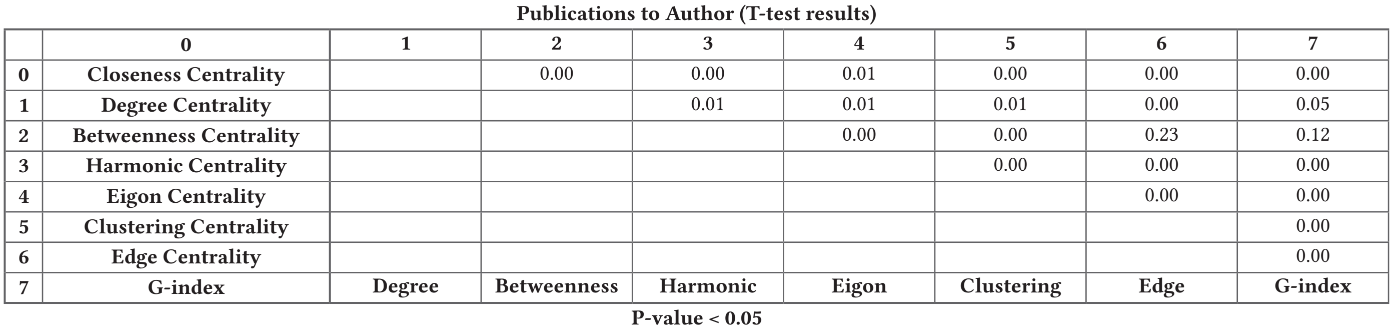
Fig. 14. Correlation analysis of publications to publications network.

Publications to Area (T-test results)

	0	1	2	3	4	5	6	7
0	Closeness Centrality		0.00	0.00	0.00	0.00	0.00	0.00
1	Degree Centrality			0.00	0.00	0.82	0.22	0.00
2	Betweenness Centrality				0.00	0.00	0.06	0.52
3	Harmonic Centrality					0.00	0.00	0.00
4	Eigon Centrality						0.09	0.00
5	Clustering Centrality							0.00
6	Edge Centrality							0.00
7	G-index	Degree	Betweenness	Harmonic	Eigon	Clustering	Edge	G-index

P-value < 0.05

Fig. 15. Correlation analysis of publications to area network.



Publications to Author (T-test results)

	0	1	2	3	4	5	6	7
0	Closeness Centrality		0.00	0.00	0.01	0.00	0.00	0.00
1	Degree Centrality			0.01	0.01	0.01	0.00	0.05
2	Betweenness Centrality				0.00	0.00	0.23	0.12
3	Harmonic Centrality					0.00	0.00	0.00
4	Eigon Centrality						0.00	0.00
5	Clustering Centrality							0.00
6	Edge Centrality							0.00
7	G-index	Degree	Betweenness	Harmonic	Eigon	Clustering	Edge	G-index

P-value < 0.05

Fig. 16. Correlation analysis of publications to author network.

V. Conclusion and Future Work

The study of research cultural growth has demonstrated an increased interest in publications and citations rate. The previous literature has involved the use of SNA techniques, significant testing strategies, and randomization to learn and find the influence of network models under some hypothesis with respect to different countries, institutions, and disciplines. Although these models are appropriate according to the network structure assumptions, they may not be the most informative for analysis according to the research area of interest. Presently, no general structural methodology exists to evaluate the network relationship graph or research area over particular measures.

In our collaboration method, we analyzed the citation scores of publications, author's progressive path of an interesting area/domain, author's score, and paper score in the computer science academic field. Fortunately, the study has shown promising results and productivity on the mentioned metrics. We created the following four collaboration networks and used them for our analysis.

- publications to authors
- authors to authors
- publications to publications
- publications to research areas.

The outcomes exhibit the productivity of authors regarding their academic progress in the selected field (computer science) and portrays it as an energetic field with the performed experiments that validate our research work. The significance of this approach is that it provides a template for future perspectives to measure the importance of each researcher in terms of different relations. This paper provides analysis on specific domains of computer science. In future, this can be extended to various research areas and can be analyzed with different datasets. Such future works can give better insights in the influence of publication and citations in various research domains.

References

[1] Iglic, H., Doreian, P., Kronegger, L., & Ferligoj, A. (2017). "With whom do researchers collaborate and why?," *Scientometrics*, 112(1), 153-174.

[2] Abramo, G., D'Angelo, A. C., & Murgia, G. (2017). "The relationship among research productivity, research collaboration, and their determinants," *Journal of Informetrics*, 11(4), 1016-1030.

[3] de Moya-Anegon, F., Guerrero-Bote, V. P., Lopez- Illescas, C., & Moed, H. F. (2018). "Statistical relationships between corresponding authorship, international co-authorship and citation impact of national research systems," *Journal of Informetrics*, 12(4), 1251-1262.

[4] Katz, J. S., & Martin, B. R. (1997). "What is research collaboration?," *Research policy*, 26(1), 1-18.

[5] Adams, J. (2012). "Collaborations: The rise of research networks," *Nature*, 490(7420), 335.

[6] Anwaar, F., Iltaf, N., Afzal, H. and Nawaz, R., 2018. "HRS-CE: A hybrid framework to integrate content embeddings in recommender systems for cold start items," *Journal of computational science*, 29, pp.9-18.

[7] Ductor, L. (2015). "Does co-authorship lead to higher academic productivity?," *Oxford Bulletin of Economics and Statistics*, 77(3), 385-407.

[8] Waheed, Hajra, Saeed-Ul Hassan, Naif Radi Aljohani, Julie Hardman, Salem Alelyani, and Raheel Nawaz. "Predicting academic performance of students from VLE big data using deep learning models," *Computers in Human Behavior 104* (2020): 106189.

[9] Stvilia, Besiki, et al. "Toward collaborator selection and determination of data ownership and publication authorship in research collaborations," *Library & Information Science Research* 39.2 (2017): 85-97.

[10] Kwiek, Marek. "International research collaboration and international research orientation: Comparative findings about European academics," *Journal of Studies in International Education* 22.2 (2018): 136-160.

[11] Thompson, P., Nawaz, R., Korkontzelos, I., Black, W., McNaught, J. and Ananiadou, S., 2013, October. News search using discourse analytics. In 2013 Digital Heritage International Congress (DigitalHeritage) (Vol. 1, pp. 597- 604). IEEE.

[12] Chen, Kaihua, Yi Zhang, and Xiaolan Fu. "International research collaboration: An emerging domain of innovation studies?," *Research Policy* 48.1 (2019): 149-168.

[13] Adams, J., Gurney, K., Hook, D., & Leydesdorff, L. (2014). "International collaboration clusters in Africa," *Scientometrics*, 98(1), 547-556.

[14] Pritchard, A. (1969). "Statistical bibliography or bibliometrics," *Journal of documentation*, 25(4), 348-349.

[15] Leung, X. Y., Sun, J., & Bai, B. (2017). "Bibliometrics of social media research: a co-citation and coword analysis," *International Journal of Hospitality Management*, 66, 35-45.

[16] Bordons, M., Aparicio, J., González-Albo, B., & Díaz-Faes, A. A. (2015). "The relationship between the research performance of scientists and their position in co-authorship networks in three fields," *Journal of Informetrics*, 9(1), 135-144.

[17] Csomós, G., & Lengyel, B. (2018). "Mapping the efficiency of international scientific collaboration between cities worldwide," *arXiv preprint* arXiv:1808.03730.

[18] Wai-Chan, S. (2017). International research collaboration creates higher impact.

[19] Freshwater, D., Sherwood, G., & Drury, V. (2006). "International research collaboration: Issues, benefits and challenges of the global network," *Journal of Research in Nursing*, 11(4), 295-303

[20] Türker, Ilker, and Abdullah Çavuşoğlu. "Detailing the co-authorship networks in degree coupling, edge weight and academic age perspective," *Chaos, Solitons & Fractals* 91 (2016): 386-392.

[21] Wang, Jian. "Knowledge creation in collaboration networks: Effects of tie configuration," *Research Policy* 45.1 (2016): 68-80.

[22] Morris, Steven A., and Michel L. Goldstein. "Manifestation of research teams in journal literature: A growth model of papers, authors, collaboration, coauthorship, weak ties, and Lotka's law," *Journal of the American Society for Information Science and Technology* 58.12 (2007): 1764-1782.

[23] Kundra, Ramesh, et al. "Studies in Co-authorship Pairs Distribution: Part-2: Co-author pairs' frequencies distribution in journals of gender studies," *COLLNET Journal of Scientometrics and Information Management* 2.1 (2008): 63-71.

[24] Egghe, Leo. "A model for the size-frequency function of coauthor pairs," *Journal of the American Society for Information Science and Technology* 59.13 (2008): 2133- 2137.

[25] Naldi, Fulvio, et al. "Scientific and technological performance by gender," *Handbook of quantitative science and technology research*. Springer, Dordrecht, 2004. 299-314.

[26] Carr, Phyllis L., et al. "Collaboration in academic medicine: reflections on gender and advancement," *Academic Medicine* 84.10 (2009): 1447-1453.

[27] Pepe, Alberto, and Marko Rodriguez. "Collaboration in sensor network research: an in-depth longitudinal analysis of assortative mixing patterns," *Scientometrics* 84.3 (2009): 687-701.

[28] Kretschmer, Hildrun, Bulent Ozel, and Theo Kretschmer. "Who is collaborating with whom? Part I. Mathematical model and methods for empirical testing," *Journal of Informetrics* 9.2 (2015): 359-372.

[29] Liu, Jie, et al. "A new method to construct coauthor networks," *Physica A: Statistical Mechanics and its Applications* 419 (2015): 29-39.

[30] Abbasi, Alireza, Jörn Altmann, and Liaquat Hossain. "Identifying the effects of co-authorship networks on the performance of scholars: A correlation and regression analysis of performance measures and social network analysis measures," *Journal of Informetrics* (2011): 594- 607.

[31] Bordons, María, et al. "The relationship between the research performance of scientists and their position in co-authorship networks in three fields," *Journal of Informetrics* 9.1 (2015): 135-144.

[32] Maggioni, Mario A., and Teodora Erika Uberti. "Networks and geography in the economics of knowledge flows," *Quality & quantity* 45.5 (2011): 1031- 1051.

[33] De Stefano, Domenico, et al. "The use of different data sources in the analysis of co-authorship networks and scientific performance," *Social Networks* 35.3 (2013): 370-381.

[34] Mourao, Paulo Reis, and Vítor Domingues Martinho. "Forest entrepreneurship: A bibliometric analysis and a discussion about the co-authorship networks of an emerging scientific field," *Journal of Cleaner Production* 256 (2020): 120413.

[35] Jeon, H. J., O. J. Lee, and J. J. Jung. "Is Performance of Scholars Correlated to Their Research Collaboration Patterns," *Front. Big Data* 2: 39. (2019).

[36] Mutz, Rüdiger, and Hans-Dieter Daniel. "The bibliometric quotient (BQ), or how to mea- sure a researcher's performance capacity: A Bayesian Poisson Rasch model," *Journal of Informetrics* 12.4 (2018): 1282- 1295.

[37] Abramo, Giovanni, Ciriaco Andrea D'Angelo, and Flavia Di Costa. "The effect of multidisciplinary collaborations on research diversification," *Scientometrics* 116.1 (2018): 423- 433.

[38] Clemente-Gallardo, J., Ferrer, A., Íñiguez, D., Rivero, A., Ruiz, G., & Tarancón, A. (2019). "Do researchers collaborate in a similar way to publish and to develop projects?," *Journal of Informetrics*, 13(1), 64-77.

[39] Kwiek, Marek. "International research collaboration and international research orienta- tion: Comparative findings about European academics," *Journal of Studies in International Education* 22.2 (2018): 136-160.

[40] Kwiek, M. (2018). "High research productivity in vertically undifferentiated higher education systems: Who are the top performers?," *Scientometrics*, 115(1), 415-462.

[41] Garfield, Eugene. "Is citation analysis a legitimate evaluation tool?," *Scientometrics* 1.4 (1979): 359-375.

[42] Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. Social Influence Analysis in Large-scale Networks. In Proceedings of the Fifteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'2009).

[43] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. ArnetMiner: Extraction and Mining of Academic Social Networks. In Proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'2008).

[44] Kong, Xiangjie, et al. "How does collaboration affect researchers' positions in co-authorship networks?," *Journal of Informetrics* 13.3 (2019): 887-900.

[45] Hayat, Tsahi, Dimitrina Dimitrova, and Barry Wellman. "The differential impact of network connectedness and size on researchers' productivity and influence," *Information, Communication & Society* 23.5 (2020): 701-718.

Sidra Razzaq

Sidra Razzaq has recently compeleted her Masters degree in Computer Science from COMSATS University at Islamabad (CUI), Islamabad, Pakistan. Her Research Interests include data science, social network analysis, and access control.

Ahmad Kamran Malik

Ahmad Kamran Malik received the Ph.D. degree from the Vienna University of Technology (TU-Wien), Vienna, Austria. He is currently an Assistant Professor with COMSATS University at Islamabad (CUI), Islamabad, Pakistan. He has published a book and many research papers in reputed international journals and conferences. His current research interests include data science, social network analysis, and access control.

Basit Raza

Basit Raza received the master's degree in computer science from the University of Central Punjab, Lahore, Pakistan, and the Ph.D. degree in computer science from International Islamic University Islamabad and the University of Technology Malaysia, in 2014. He is currently an Assistant Professor with the Department of Computer Science, COMSATS University at Islamabad (CUI), Islamabad, Pakistan. He has authored several articles in refereed journals. His research interests include database management systems, data mining, data warehousing, machine learning, deep learning, and artificial intelligence. He has been serving as a Reviewer for prestigious journals, such as Applied Soft Computing, Swarm and Evolutionary Computation, Swarm Intelligence, Applied Intelligence, IEEE Access, and Future Generation Computer Systems.

Hasan Ali Khattak

Hasan Ali Khattak received his Ph.D. in Electrical and Computer Engineering degree from Politecnico di Bari, Bari, Italy on April 2015, Master's degree in Information Engineering from Politecnico di Torino, Torino, Italy, in 2011, and B.CS. degree in Computer Science from the University of Peshawar, Peshawar, Pakistan in 2006. He is currently serving as Associate Professor - School of Electrical Engineering and Computer Science, National University of Sciences and Technology - Pakistan since October 2020. His current research interests focus on Future Internet Architectures such as the Web of Things and leveraging Data Sciences and Social Engineering for Future Smart Cities. Along with publishing in good research venues and completing successful funded National and International funded projects, he is also serving as a reviewer in reputed venues such as IEEE Access, IEEE Network Magazine, IEEE Consumer Electronics, Hindawi, SAI, IET, and a few national publishers. He is currently involved in several funded research projects in various domains such as Healthcare Information Management, Semantic Web of Things, and Fog Computing while exploring Ontologies and other Semantic Web Technologies. He has worked on Contiki OS, NS 2/3, and Omnet++ frameworks. His perspective research areas are the application of Machine Learning and Data Sciences for improving and enhancing Quality of life in Smart Urban Spaces especially in the Healthcare and Transportation domain through Knowledge management, predictive analysis, and visualization. He is an active Senior Member of IEEE, a professional member of ACM, and a member of societies such as IEEE ComSoc, IEEE VTS, and Internet Society.

Giomar W. Moscoso Zegarra

Prorrector and associate professor at Escuela de Posgrado Newman. RENACYT researcher qualified by CONCYTEC. Candidate for a Doctorate in Accounting and Finance, Master of Science with a mention in Auditing, Master of Business Administration, Commercial Engineer from the Tarapacá University of Chile, Certified Public Accountant, Bachelor of Administration from the Tarapacá University of Chile. Experience in financial, research and academic work at management level. Head of licensing teams and processes related to SUNEDU. Former University Defender at Escuela de Posgrado Newman and General Director of the scientific journal Iberoamerican Business Journal.

Yvan Díaz Zelada

Consultant with experience in business organization and management, market research, formulation and evaluation of projects and design of business models. He is a researcher recognized by CONCYTEC in the María Rostworowski II category of RENACYT. Doctoral candidate in Administration, graduated with maximum distinction as an MBA from the Master in Business Administration and Management from the University of Tarapacá (Chile), Master in Business Administration from Escuela de Posgrado Newman (Peru) and graduated from the Banking Administration Program by Catholic Center (Peru). Studies in commercial engineering at the Tarapacá University of Chile, and collegiate systems engineer (CIP 112273). He is currently Academic Director of Escuela de Posgrado Newman. He served as director of the e-Learning center for Escuela de Posgrado Newman. He has been a business officer for Banco de Credito del Peru (BCP), Mi Banco and Banco Solventa. He teaches at Escuela de Posgrado Newman, at Universidad San Pablo (Arequipa) and Universidad Continental (Arequipa). He has taught at different Peruvian universities in the south of Peru (Tacna and Arequipa) in the lines of business, innovation, administration and technology.

unir
LA UNIVERSIDAD
EN INTERNET

Rectorado
Avenida de la Paz, 137
26006 Logroño (La Rioja)
t (+34) 941 21 02 11

www.unir.net