Universidad Popular Autónoma del Estado de Puebla

Vicerrectoría Académica

Decanato de Ingenierías

Modelo de predicción del comportamiento del mercado de valores usando Aprendizaje automático

Tesis para obtener el Grado de Maestro en Ciencia de Datos e Inteligencia de Negocios

Presentada por:
Ravi Chillanki

Directora de tesis

Dra. Maria del Rocio Guadalupe Morales Salgado

H. Puebla de Zaragoza, México. August de 2022

Universidad Popular Autónoma del Estado de Puebla

Vicerrectoría Académica

Decanato de Ingenierías

Maestría en Ciencia de Datos e Inteligencia de Negocios

Se aprueba la Tesis/Proyecto Práctico:

"Modelo de predicción del comportamiento del mercado de valores usando Aprendizaje automático"

Nombre del alumno
Ravi Chillanki

Comité Asesor

_____

Dra. Maria del Rocio Guadalupe Morales Salgado
Directora de Tesis

_____	_____
Dr. Gonzálo Jorge Urcid Serrano	Mtro. Roberto Salazar Marquez
Co-Director	Asesor
_____	_____
Dra. Alejandra Aldrette Malacara	Dr. Jorge Rafael Aguilar Cisneros
Asesora	Asesor

H. Puebla de Zaragoza, México August de 2022

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Abstract

Worldwide, number of retail investors entering stock market are increasing enormously day by day. The financial data of the companies are also available publicly on Yahoo finance. This represents fertile ground for all fields of data science that seek to build predictive models using such data. This thesis focuses on predicting the trend of a company stock in the Stock Exchange while taking into consideration that the stock market of a country is no more completely localized as global issues have been showing impact on Stock exchanges of different countries. I collected 8 years of financial data from Yahoo finance, daily news of Reliance Industries from Google news and top daily global news articles from The Guardian.com. This paper explores the different models that are used in the prediction modelling from traditional machine learning by conducting extensive tests on them with feature engineering and found that my proposed approach with XGBClassifier outperforms achieving approximately 84.2% accuracy. This paper adds to the stock analysis research community in both the financial and technological areas by providing extensive design and evaluation of prediction term lengths, feature engineering, and data pre-processing approaches. This work contributes to the normal public and also stock analysis research community with the prediction model.

Keywords

Machine learning, predictive models, Stock Market, Classification.

Contents

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

LIST OF FIGURES

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

LIST OF TABLES

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

CHAPTER I: PURPOSE AND ORGANIZATION

1.1. Introduction

For a long time, stock price changes were believed to be unpredictable. Evaluation of the well-known ‘Random Walk hypothesis’ (Mihir, 2019) and the ‘Efficient market hypothesis’ (Alexandra , 2015) which states that a market is efficient with respect to a set of current information it is impossible to obtain economic profits in this market, led to this belief. In other words, if it is impossible to outperform the market due to the randomness of stock prices, unless a different (often excessive) type of risk is considered, the economic benefits cannot increase. However, it is not clear how that risk would be measured. Therefore, there is no doubt that predicting stock market price trends is a challenging task. In contrast, the 'Wisdom of Crowds' hypothesis (Hamada, 2020), which stems from collaborative filtering theory, states that many individuals, each with limited information, can provide highly accurate assessments if their information is obtained appropriately. However, it is not known to be useful in predicting stock market returns; however, some individual and institutional investors can analyze the market to make a profit (Avery, 2016). The inefficiency of the forecast is accentuated due to the various uncertainties involved and due to the presence of multiple variables, all of which can potentially influence the market value on a particular day. Over time, a number of explanatory variables have been added to this huge literature, these include country-specific economic conditions, investor sentiment towards a company, political events, etc. Consequently, stock markets are susceptible to rapid changes, which can appear to be random fluctuations in stock prices.

Stock exchange data are generally dynamic, non-parametric, chaotic and noisy in nature, making investments inherently risky. Stock price movement is considered to be a random process with fluctuations, which are more prominent in the short term. However, some stocks tend to show reasonable difference in value over long time (What are Multibagger Stocks, 2022). This is a digital era where every small news circulates within seconds. With this type of facilities available at finger tips, now a days, many retail investors are doing intra-day trading, short term trading, swing trading, long term investments (Types of Stock trading, 2019). While selling and buying pressures play role in intra-day trading, technical indicators in the stock trend of a company is a factor in Swing trading and traders rely on fundamental analysis of

a company in the stock market, global news has also become a factor affecting the trends in stock exchange (Mohammad, Md, & Rosni, 2021). This is clearly seen during many incidents in different countries. Outbreak of Covid-19 is of those which we have seen in recent times, how that has shown a global impact over stock exchanges of number of countries throughout the world (Khan, et al., 2020). When a new covid-19 variant was detected in African countries, it showed an immediate effect on Indian Stock Exchange with an approximate decline of 2.2% in overall market making the market Bearish, 2.7% decline in Hong Kong, Seoul faced a decline of 1.5%, London's benchmark fell by an unusually wide margin of 3.3 and Tokyo had to face bear market with a fall of 2.5% (Global stocks sink after South Africa finds new Covid variant, 2021). Another incident which trembled economies of several countries was Afghanistan crisis when Taliban have taken over it (Marc, 2021). The Ukraine Russia conflict had also triggered turmoil in economies and exacerbated the uncertainties around the global economy's recovery (Kammer, Azour, Selassie, Goldfajn, & Rhee, 2022). So it is worth considering global information in predicting stock market prices or trends.

Every investors looks for profit in stock exchange investments. When the market is bullish, investors doesn't care much to play safe game, rather they book profits, do different types of trading, increase the value of the portfolios and happy with the investments and gains. It makes the investors to be more keen on risk-free trading when the stock market's downturn keeps us up at night. This is commonly seen when the market is volatile and bearish. As every investor now a days is accessing information and getting updates so rapidly, the impact of global news is also being reflected on many stock exchanges that quickly and shattering the general expectations in the market trend of the investors because of this rapid change in market sentiment.

It is necessary to understand the state of the stock market and the nature of the price movement of a company stock in the market. Websites like Moneycontrol.com which has been a platform for Indian Stock Exchange, until two years ago, used to display the current trend of a company stock, whether it is bullish or bearish (Reliance Industries, n.d.). This helped investors to further analyze based on its current trend. For trading with minimum risk, knowing this kind of information in advance would

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410
Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

benefit retail investors, when the market is very volatile. Global news have many times showed impact on portfolios of short term traders and long term traders, as the investors couldn't predict the trend of the stocks in their portfolio going to be bullish or bearish, they do inappropriate actions of holding, selling and buying the stocks in that situation. They may face losses when they hold or buy stocks which are happening to be bearish and sell stocks of a company which could turn out to be bullish because of the situation and sentiment in the market.

With the growing technology in this digital era, the scope of Machine learning has been increasing enormously. This field is continuing its support on various problems in the digital space around us, be it statistics, medical, social, literature and several sciences. Though there have been work arounds with Statistical methods to look at the game of money in the stock market, machine learning and deep learning techniques are also shedding light to the researchers to advance in getting even more from automatic analysis and forecasting of various aspects in the stock market business.

Several papers in the past straightforwardly tried to target the price of the stocks. With the agenda of predicting the stock prices in the exchange, different statistical algorithms were used to forecast the stock value. Conventional machine learning algorithms were also been used to achieve the same. There was always been advancements in this regard to forecast the value as better as possible. The stock value is a dynamic number that is defined by lots of factors every second. Because of its high volatility, there has never been a direct formula to find its value and get maximum out of it. Several researches also have been conducted on sentiment analysis in the stock market to predict the same. More than half of the participants are concerned about a market fall, and 81% believe market volatility will continue this year (invest in you, 2022). With the goal that an investor to have high confidence to make minimal risk investment in the stock exchange, it is necessary to accurately predict stock market price trends if it is bullish or bearish.

1.2. Research Objectives

1.2.1. General Objective

Development of a Stock Market Behavior Prediction Model Using Machine Learning

1.2.2. Specific Objectives

1. Extract the data from the sources and consolidate in to a structured form.
2. Carry out the data transformation, validate and clean the data which can further be used for feature engineering and ultimately as input to the model.
3. Perform feature engineering on the dataset by analyzing the data to find the relationships and dependencies between different features.
4. Prepare model training data to train the model and test with test data.
5. Technical analysis of different models to select the best model that fits with the data and solve the classification problem.
6. Evaluate the model.

1.3. Project Justification

Many common people need and should do the analysis before investing. The stock market is the common platform for millions of entrepreneurs, investors and ordinary people. But there is no tool to predict whether a stock will be 'bullish' or 'bearish'. There are some platforms for banks and private companies but there are no platforms for the common public although the stock market data is public.

1.4. Scope and Limitations

1.4.1. Scope

- Develop the system.
- The system is a prototype that can be developed in real time.
- This system is open to the public
- Comparative study of machine learning algorithms to identify the most suitable algorithm.

1.4.2. Limitations

- The project does not aim to forecast the value of stocks for n days.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

- It is not suitable for long-term investors.
- The model does not consider the internal decisions of a company that could affect its position in the stock market.
- The model does not deal with shares of private companies or IPOs.

1.5. Project Viability

There are parameters that have a significant impact on the viability of the project.

1. It is viable because it is developed with the technology of the latest computational tools.
2. The open data used by the model is a combination of historical data available on Yahoo finance and news headlines from The Guardian.
3. Ability to program effectively in technical languages.
4. Accessibility of computing resources.
5. Time availability.

1.6. Original Contribution

Previously, there were many studies and implementations of projects to predict the value of a company's stock that used only statistical historical data of that company's stock and various other variables But predicting the stock growth direction indicator based on historical statistical data and world news is a new approach that leads everyone to take a step forward to find a solution to the problem of predicting stock trend on the next business day stock market. I've motivated from several global issues like covid-19, Afghanistan crisis, Russia-Ukraine conflict etc. impacting economies, which gave me an impulse to study on this and create a system based on stock exchanges. This new approach of predicting the direction of stock growth not only improves the accuracy of the model, but also the confidence of users, as realistic news is used as model variables that can help users to rely on it and avoid losses by predicting if a stock is going to be bullish or bearish soon.

CHAPTER II: THEORY

2.1. Stock Exchange

Stock market is a platform where securities such as stocks, derivates and bonds are traded. This is run by Exchange board of the country. The participant i.e. investors, traders, brokers, companies, have to be already registered with a Demat account on stock exchange to be able to trade in stock market (Demat account, n.d.). On this platform on a business day, buyers and sellers come forward and trade to make profits.

How do Companies and investors use the market today

When a company decides to launch on the market, the company will advertise itself to big investors. If they think the company is a good idea, they get the first crack at investing and the sponsor the company's initial public offering or IPO (IPO Process, n.d.). This launches the company on to the official public market, were any company or individual who believes the business could be profitable might buy a stock. Buying stocks makes those investors partial owners in the business. Their investments help the company to grow and as it becomes more successful, more buyers may see potential and start buying stocks and start buying stocks. As demand for those stocks increases, so does their price, increasing the cost for prospective buyers, and raising the value of the company stocks people already own. For the company, this increased interest helps fund new initiatives, and also boosts its overall market value by showing how many people are willing to invest in their in their idea. However, if for some reason a company starts to seem less profitable, the reverse can also happen. If investors think their stock value going to decline they will sell their stock with the hopes of making profits before the company loses more value. As stocks are sold and demand for the stock goes down the stock price falls and with it the company's market value (Easley, M. Kiefer, & O'Haara, 1997). This can leave investors with big losses unless the company starts to look profitable. Again this see-saw of supply and demand is influenced by many factors. Companies are under the unavoidable influence of market forces such as the fluctuating price of materials, changes in

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

production technology and the shifting costs of labor (Petri & Banga, 2021). Investors may be worried about changes in leadership, bad publicity or larger factors like new laws and trade policies and of course plenty of investors are simply ready to sell valuable stocks and pursue personal interests. All these variables cause day to day noise in the market, which can make companies appear more or less successful and in the stock market appearing to lose value often leads to losing investors and in turn losing actual value. Human confidence in the market has the power to trigger everything from economic booms to financial crisis (McMillan, 2019). And this difficult to track variable is why most professional promote reliable long term investment over trying to make quick cash. However experts are constantly building tools in efforts to increase their chances of success in this highly unpredictable system. But the stock market is not just for the rich and powerful. With the dawn of the internet, everyday investors can buy stocks in many of the exact same ways a large investor would. And as more people educate themselves about this complex system they too can trade stocks, support the businesses they believe in, and pursue their financial goals (Rooij, Lusardi, & Alessie, 2011). The first step is getting invested.

.

2.1.1. Buy and Sell Stocks

A stock exchange is a public marketplace for shares and as with any market place there are buyers and sellers. When a buyer and a seller agree on a price, a deal is done. When someone wants to sell a stock, they submit the price at which they are willing to sell it on exchange. Buyer do similar; they submit the price at which they want to buy a particular stock. When the lowest selling price meets the highest buying price, a deal is done. This process is fully automated and could happen thousands of times per second for popular stocks as there are many investor who trade them. The quotes an investor see for a certain stock are the prices for an immediate deal. In order to access stock exchange, a participant need a stock broker who executes the deals for the participant on his behalf. A lots of such applications have been developed and available online to make almost all the operations in trading in stock market. Once the participant registers successfully and required balance is maintained to start trading, the user searches for the stock and start making transactions (Kotak securities, n.d.).

The main things to consider when the participant is buying shares are which company, what price and how many shares to buy. Deciding which one to buy and how many stocks to buy depends on the participant's strategy and risk appetite. Similar strategy with the intention to book profits, a participant can sell the holdings through the same platform.

2.1.2. Bull-Bear

This is a trader lingo in stock market that refers to as state that people casually use for, when they are active or inactive as bullish or bearish. So, 'Bullish' is basically associated with 'good', a price going up. If the 'Bulls' are in control of a stock that would imply that the price is going up. If the participant believe that the price is going up, they he says he is bullish on certain stock (Geier, 2021). So, if something is getting better in some aspect we use the term bullish. On the flipside, when it is said bulls are in control of that stock that implies that the price over time is just going lower and lower. So bears associate that with falling price or things not going so good. So, in overall, these refer to the trend of a stock performance in the exchange, which investors take into account before trading or investing in any stock with the idea that buying a bullish stock would generate wealth and getting rid of the stocks which are bullish would leave them without facing losses in the portfolio (Angel One, n.d.). It is shown in the *Figure 1*.

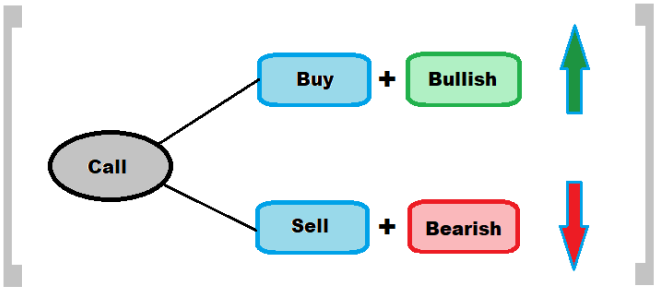


Figure 1. Stock trend based on buy and sell call operations. Source: Own creation.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

2.1.3. Strategies

Stock exchanges allow companies to raise capital and investors to make informed decisions using real-time price information. Exchanges can be a physical location or an electronic trading platform. Conventional investment strategists use data on management group composition, competitive advantages, market outlook, past trading performance, and market capitalization to decide the best trades. Several theories were found that address certain questions, such as whether it is possible that one company's methodology can influence another methodology's decisions, whether robotic predictions include conventional strategies by clearly resolving manual intuition into numbers.

2.1.3.1. Efficient Market Hypothesis

The Efficient Market Hypothesis, in markets that are efficient where information is volatile, essentially believes that forecasting stock prices for profit is not possible (Campanella, 2016). Be that as it may, if this were true, the field of investment in Finance would not exist. The EMH, to which many financial analysts adhere, assumes that stock markets are made up of rational investors who have considered all available information on the current market values of stocks, meaning that the markets are efficient and the stock price forecasting is not possible. to obtain benefits (Manahov, 2014). The Efficient Market Hypothesis states that stock prices can be forecast only by events that are not yet known, but as they happen they will be reflected in company values in terms of stock prices (Urquhart, 2013).

2.1.4.2. Random Walk Theory

Random walk theory states that past stock price movements cannot predict future stock prices because the stock market is volatile and stock prices are independent of other market factors (Shonkwiler, 2013). Furthermore, changes in the market and stock prices occur independently of the previous one, and they all have the same distribution. By aligning with EMH, RWT means that all share price changes are random and efficient markets reflect all publicly available information.

2.1.5. Stock Value Movement Prediction

The two most popular trading strategies, namely fundamental trading and technical trading, are contributed by the two EMH and RWT theories. From these trading strategies arises the need to predict the future price of a share in the market.

2.1.5.1. Fundamentals Trading Approach

Fundamentals Trading is a vigorous procedure that relies on data on the intrinsic value of fundamental trading rather than time series data on the cost of stocks. This strategy uses information from a company's balance sheet and forecast earnings released each quarter, the projects the company is managing, and new implementations and contracts. In this way, the fundamental trading methodology incorporates both historical and forecast trade execution information (Bohl, 2021). The same methodology can be applied to the Forex and Commodity markets. Traders study micro and macroeconomics as well as geopolitical data and analyses how all this information relates to each other. A growing gross domestic product(GDP) is a good economic indicator of strength. Inflation, interest rates, trade or currency account balance, credit, Gross domestic product(GDP), employment data, Industrial production, Retail sales, Consumer Price Index (CPI) and Monetary policies are key factors from a fundamental viewpoint.

2.1.5.2. Technical Trading Approach

Technical trading strategy looks at historical price data as a guide to the future and financial time series trading volumes to uncover current patterns. The patterns result in an estimate and an executed trading plan. The technical trading strategy accepts that the stock prices on the screen reflect all the data from real-time news and other known market weights. The specialized exchange generally relies on relapse and moving average plans to make forecasts (Murphy & Gebbie, 2019). Support, Resistance and Trends are some key factors in price action. Technical analysis can clue us in to sentiment and speculation. Technical analysis can also inform the timing for an investment

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

2.2. Machine Learning

Machie learning is a subset of Artificial Intelligence (AI) which provides machines the ability to learn automatically and improve from experience without being explicitly programmed. It involves continuously feeding data into a machine so that it can interpret this data, understand the useful insights, detect patterns and identify key features to solve problems, which is very similar to how our brain works (Expert System Team, 2022). Although machine learning is a field within computer science, it differs from traditional computational approaches. In traditional computing, algorithms are sets of explicitly programmed instructions that computers used to compute or solve problems. Instead, machine learning algorithms allow computers to train on data inputs and use statistical analysis to generate values that fall within a specified range. Because of this, machine learning makes it easy for computers to build models from sample data to automate decision-making processes based on data inputs (Duggal, 2021). There are multiple applications of machine learning and usually, the algorithms are categorized as supervised, unsupervised, and reinforced.

2.2.1. Supervised learning

Supervised means to oversee or direct a certain activity and make sure it is done correctly. In this type of learning, the machine learns under guidance. The goal of supervised machine learning algorithms is to predict a result value based on a set of input values (Trevor, Robert, & Jerome, 2001). The machine learning model is supervised by ‘teaching the model, that is we load the model with knowledge so that we can have it predict future instances. Generally speaking, the model is trained on a labelled dataset, so it can predict the outcome of out-of-sample data. A labelled dataset consists of features that act as input and the columns called features which include data that will be predicted as output. There are two types of supervised learning, classification and Regression.

2.2.1.1. Classification

Classification is a process of categorizing a given set of data into classes. It can be performed on both structured and unstructured data. The process starts with predicting the class of given data points. The classes are often referred to as target, label or categories. In this supervised learning, a classifier, an algorithm that is used to map

the input data to a specific category, is used to create a model called Classification model, which predicts or draws conclusion to the input data given for training and it will predict the class or category for the data. If the outcome has possibility of two different classes, it is called Binary Classification and if the outcome has possibility of more than two different types, it is called Multi-class classification (Brownlee, 2020). Logistic regression, Decision tree, K-Nearest Neighbor, Support Vector Machine, Naïve Bayes and Random Forest are some of the core algorithms that are used for classification problems in machine learning.

2.2.1.2. Regression

Regression analysis is a form of predictive modelling system which examines the relationship between a independent and dependent variable. A regression analysis involves graphing a line over a set of data points that most closely fits the overall shape of the data. A regression shows the changes in dependent variable on one axis to the changes in the explanatory variables in other axis. Regression analysis are used in determining the strength of predictors, forecasting an effect and trend forecasting (Ray, 2015). Simple regression considers one quantitative and independent variable to predict the other quantitative, but dependent variable. A straight line drawn to fit to the data. Where as in Multiple Regression type, it considers more than one quantitative and qualitative variable to predict a quantitative and dependent variable.

2.2.2. Unsupervised learning

Unsupervised learning means we do not supervise the model but we let the model work on its own to discover information that may not be visible to human eye. Unsupervised learning uses machine learning algorithms that draw conclusions on unlabeled data. Unsupervised learning has more difficult algorithms than supervised learning, since we know little to no information about the data, or the outcomes that are to be expected. With unsupervised learning, we are looking to find things such as groups or clusters, perform density estimation and dimensionality reduction. Unsupervised learning creates a less controllable environment, as the machine is creating outcomes for us.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

2.2.3. Reinforcement learning

Reinforcement means to establish or encourage a pattern of behavior. Reinforcement learning is a type of machine learning that learns to solve problems by trial and error (Expert System Team, 2022). It starts with an agent interacting with an environment. The agent is trying to achieve a multi-step goal within the environment. The environment has a state, which the agent can observe. The agent sense the state of the environment through what is sees, hears, feels or sense via other means. The agent has actions which can modify the state of the environment. The agent receives reward signals as it moves closer to its goal. The agent uses these rewards to determine which actions were successful and which actions were not. This state, action and reward loop is repeated over and over until the agent learns by trial and error how to operate effectively with in the environment. The object of the agent, is to learn how to always choose the right action, given any state of the environment that leads it closer to its goal (Hao, 2018).

2.2.4. Predictive Modeling

Predictive analysis, a branch of data analysis which is mainly used to predict future events or outcomes. It is solely based on data driven approaches and techniques to reach conclusions or solutions. The analysis makes use of analytical techniques and predictive modeling in order to find relevant patterns in large data sets. In turn these patterns can be used to make various opportunities in the businesses by identifying risk and benefit points etc. “Predictive analysis is an anticipatory technique for forward looking solutions and insights to assess any situation” (Editorial, 2021). The majority of predictive analytic methods use machine learning terms and algorithms for model creation, particularly to train the models. As a result, understanding how to select various predictive strategies for building the model is critical.

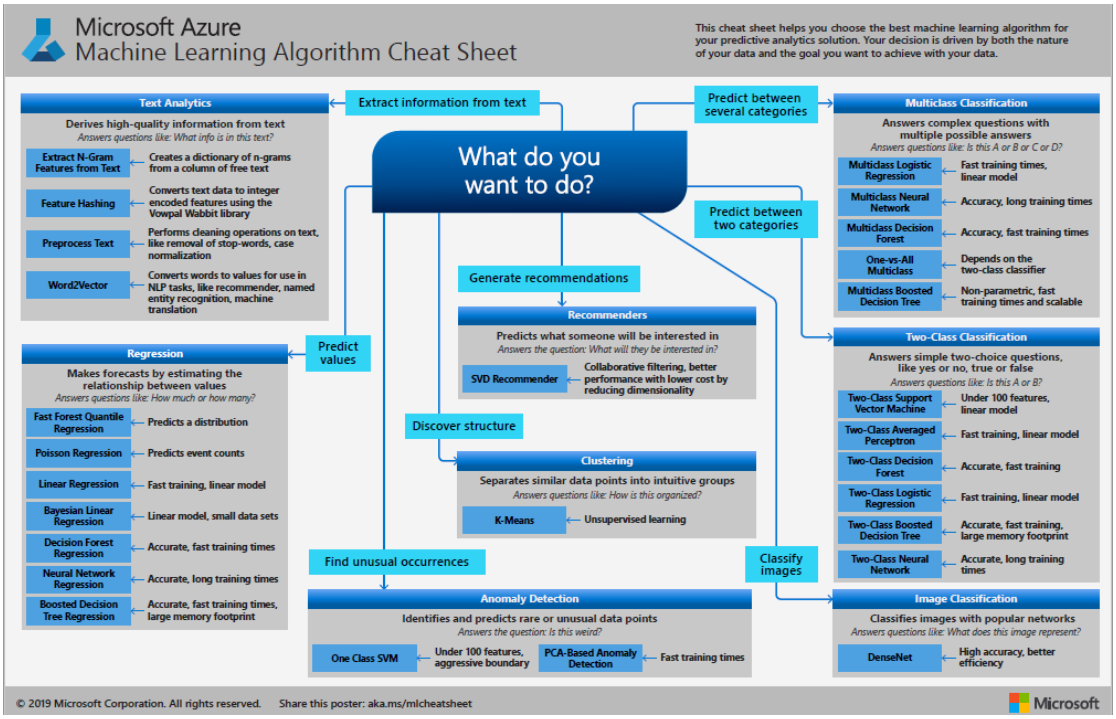


Figure 2. Algorithm Cheat Sheet (Microsoft Corporation, 2019)

The predictive modeling process begins once a set of current or historical data is collected for analysis. “Then data scientists or analysts create algorithms and statistical models, train them with subsets of the data and run them against the full data set to generate the predictive model” (Subramanian, Paul, Kim, & Srinivasan, 2021). In many circumstances, many models are utilized concurrently to provide a single forecast. “Though the terms predictive modeling and predive analytics are sometimes used interchangeably, modeling can be seen instead as the hands on part of analytics applications” (Subramanian, Paul, Kim, & Srinivasan, 2021). Few predictive analysis techniques are Regression, Classification, Clustering, Time series and Forecasting. Some consideration for effective predictive modeling include acquiring, sorting, cleansing and preparing enough data for analysis which is often said to take about 80% of the process. The selection of model depends upon the problem statement, target variable, linear separability of the data, size of data etc. *Figure 2* is a cheat sheet of machine learning algorithms provided by Microsoft; it gives a basic idea and an impulse to figure out what model can be used for a specific scenario.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

2.2.4.1. Logistic Regression

Logistic regression is a statistical analysis technique used for binary classification, where the output prediction can only take one of the two possible values. Logistic regression has evolved into an essential technique in the field of machine learning. “The approach allows an algorithm to be used in a machine learning application to classify incoming data based on historical data” (Lawton, 2022). The more relevant data you input, the better the algorithm should predict the rankings within the data sets. This model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables but linear relationship between the dependent and the independent variables is not necessary. It is one of the most widely used machine learning algorithms for binary classification problems, including predictions such as "this or that", "yes or no", and "A or B".

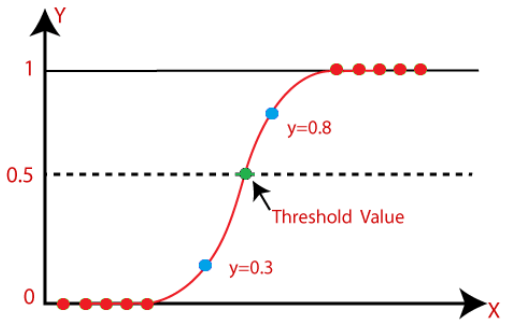


Figure 3. Example for Logistic Regression. Source: own creation

Logistic regression is based on logit or sigmoid function that will map any data in to the range of 0 to 1 (Wadmare, 2021). It forms a shape as shown in *Figure 3*. This is achieved from the mentioned functions applications. The odds ratio is defined as the probability P of success in comparison to the probability of failure.

$$\text{logit(p)} = \log(\text{odds}) = y = \log\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots\ldots + \beta_n X_n \quad (2.1)$$

$$P = \left(\frac{1}{1+e^{-y}}\right) \quad (2.2)$$

$$P = \left(\frac{1}{1+e^{-(\beta_0+\beta_1 X_1 +\beta_2 X_2+\ldots\ldots+\beta_n X_n)}} \right) \quad (2.3)$$

It is a logit function with y being the output variable and X make the independent variables with $\beta_0, \beta_1, \beta_2, \ldots, \beta_n$ used as unknown constant values and to be determined on the input training data. In this model, logit is changed by β_0 when single unit is multiplied by β_1. The change in probability depends upon the value multiplied. If β_1 is less than zero, P will decrease and if β_1 is greater than zero then P will increase.

2.2.4.2. XGBClassifier

Gradient boosting is a method where new models are created that compute the error in the old model and then the remainder is added to make the final prediction. The XGBoost, stands for eXtreme Gradient Boosting, is a boosting algorithm based on gradient boosted decision trees algorithm. XGBoost has built-in cross-validation function, a better regularization technique to reduce overfitting, and it is one of the differences from the gradient boosting (Pathak, 2019). 'XGBoost' is an open-source library which provides machine learning algorithms under the gradient boosting methods. XGBoost was designed to be used with large, complicated data sets. Being an ensemble algorithm, the core idea of bootstrap aggregation is Aggregating various samples of the original dataset to train different classifiers chosen randomly with replacement. And as part of 'bagging', it uses voting approach for Classification and calculating mean for regression (Hachcham, 2021).

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

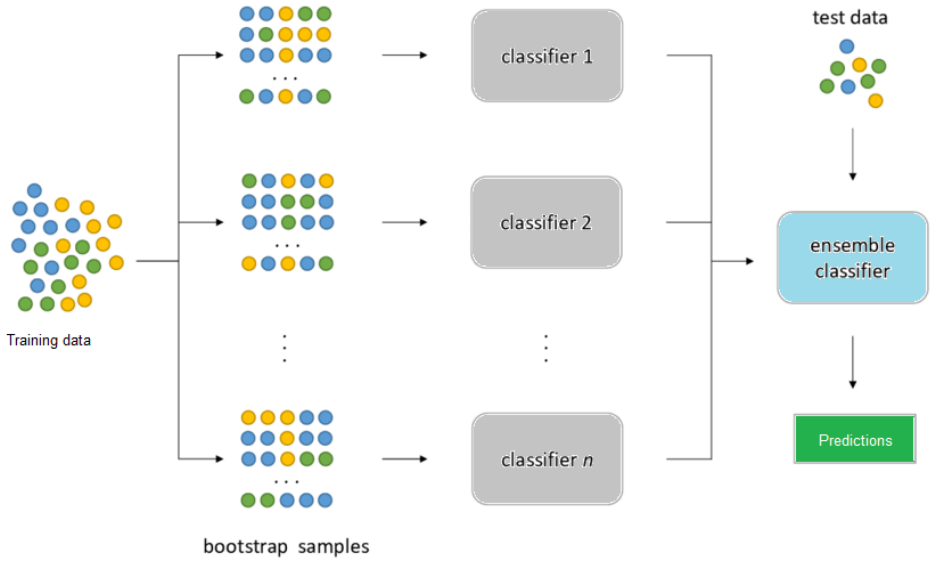


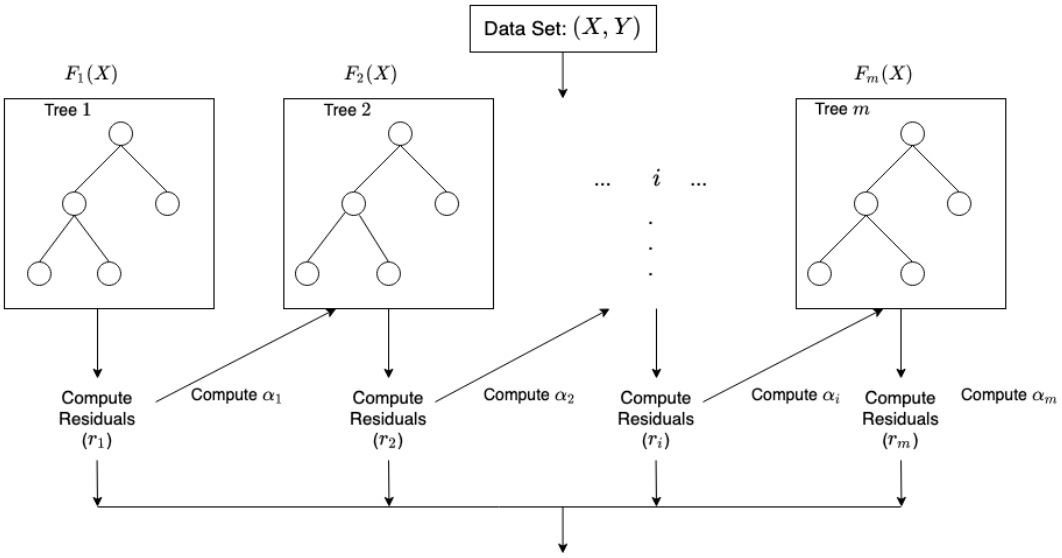
Figure 4. Conceptual diagram of xgboost algorithm. Source: own creation

The very first step in fitting XGBoost to the training data is to make an initial prediction. This prediction can be anything for example, the probability of observing an effective dosage it is 0.5, regardless of whether you are using XGBoost for Regression or Classification. The residuals, the differences between the observed and predicted values, shows how good the initial prediction is. XGBoost tree is fit to the residuals. 'ƛ' in the formula is the regularization parameter. *Figure 4* Shows a conceptual workflow of this algorithm.

$$\text{Similarity score} = \frac{(\sum Residual_i)^2}{\sum[Previous\ probability_i * (1 - Previous\ probability_i)] + ƛ} \tag{2.4}$$

$$\text{Gain} = left_{similarity} + right_{similarity} - root_{similarity} \tag{2.5}$$

Since this boosting algorithm learns in a sequential iteration fashion, where it inserts new trees that predicts the residuals of errors of previous trees which are later clubbed with the previous trees. So it adaptively minimizes loss while training continues iteratively



$$F_m(X) = F_{m-1}(X) + \alpha_m h_m(X, r_{m-1}) \tag{2.6}$$

Where α_i and r_i are the regularization parameters and residuals and residuals computed with the i^{th} tree respectfully and h_i is a function that is trained to predict residuals, r_i using X for the i^{th} tree. To compute α_i we use the residuals computed, r_i and compute the following:

$$\arg \min_{\alpha} = \sum_{i=1}^{m} L(Y_i, F_{i-1}(X_i) + \alpha h_i(X_i, r_{i-1})) \tag{2.7}$$

Where L(Y, F(X)) is a differentiable loss function.

The XGBoost algorithm works as follows:

An initial tree F_0 is obtained to predict the target variable 'y', the result is associated with a residual (y – F_0).

1. A new tree h1 is obtained that adjusts to the error of the previous step.
2. The results of F_0 and h_1 are combined to obtain the tree F_1, where the root mean square error of F_1 will be less than that of F_0: F_1 (x) < - F_0 (x) + h_1 (x)
3. This process is continued iteratively until the error is minimized as much as possible as follows: F_m (x) < - F_{m-1} (x) + h_m (x)

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

4. In every iteration, it is to be decided if it can be optimized clustering similar residuals if they are split into two groups. Once the tree with a leaf node on left and right are created with the root, Gain can be calculated from above formula (2.5).

As part of 'bagging', it uses voting approach for Classification and calculating mean for regression to get the final output of predictions.

2.2.4.3. Decision Tree

The decision tree algorithm is a member of the supervised learning algorithm family. It can be used to solve both regression and classification problems. Decision trees are really easy to read and understand. It belongs to one of the few models that are interpretable where you can understand exactly why the classifier has made that particular decision. "A decision tree is a graphical representation of all the possible solutions to a decision based on certain conditions" (Subramanian D. , 2019). It is called so as it starts with a root and then branches off to a number of solutions just like a normal tree. Its size depends upon number of conditions and decisions. There are two kinds of nodes decision nodes and leaf nodes. The form one contains a condition to split the data and the later one helps us to decide the class of a new data point. The model will choose the split that maximizes the information gain. While predicting the class of a data point, the way to quantify the impurity or uncertainty is to use Entropy. If entropy is high then it is very uncertain about the randomly picked point needs more information to describe the class.

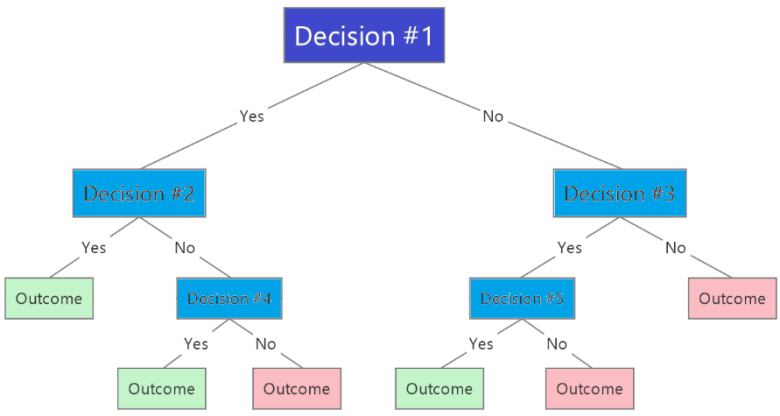


Figure 5. Conceptual diagram of Decision tree algorithm. Source: own creation

The data is split recursively using the binary tree until pure leaf nodes are reached as shown in *Figure 5*. Decision tree regressors work in the same fashion as classifier, but the value of new data point is found from the average output value of all the data points present in the leaf nodes. That's how a decision tree can be used to solve both classification and regression problems.

2.2.4.4. SVC

Support Vector Machine(SVM) is also a member of supervised learning algorithm family. It is a classification technique based on optimal margin. But SVM is useful in solving both classification and regression problems. SVM works in such a way that apart from creating the hyperplane like linear regression and logistic regression, it creates two margin lines which are distanced from the hyperplane so that it can easily be separable for both the classes . A margin line passes through one of the nearest neighbor data point of a class as shown in Figure 6. These margins lines are parallel to the hyperplane. The sum of the distances between optimal hyperplane line and both margin lines is called Margin or marginal distance. There can be multiple hyperplanes, to separate the two different classes of data points. But they hyperplane which gives maximum marginal distance value is considered. When the marginal distance around the optimal hyperplane is more, it means the classification has been more effective.

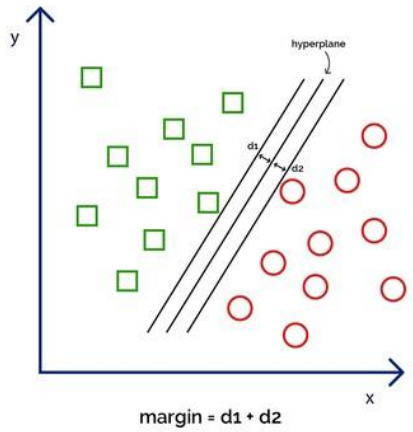


Figure 6. Conceptual diagram of Support Vector Machine classification algorithm. Source: (Jhawar, 2020)

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

In case of non-linearly separable data points, SVM uses a concepts like soft margin, whjch tries to find the hyperplane to separate but endure some misclassified data points with some degree of tolerance and SVM kernels, whos main purpose is to convert the 2D or low dimension to a high dimension to find a non-linear decision boundary. One of the parameters which contributes to the feature points to have more influence in finding the decision boundary is Gamma, as shown in Figure 7.

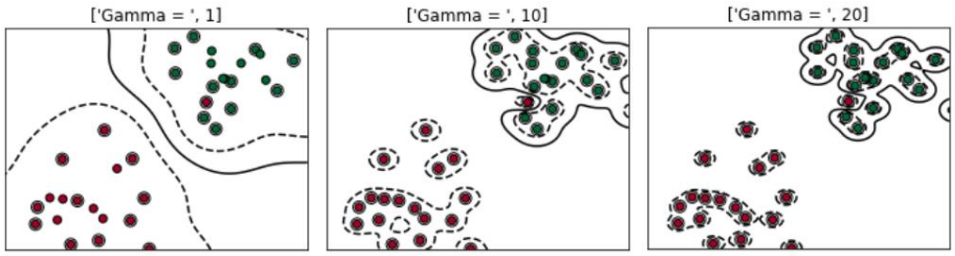


Figure 7. Different gamma hyperparameter values defining the decision boundaries with respect to non-linearly separable points. Source: **(LILLY, 2019)**

It also has been mentioned in his article by Grace Wahba how some statisticians employed to work on SVM with kernel trick issues like optimization problem, unequal misclassification costs, non-representative sample sizes, multiple classes and non-linear data distribution (Grace, 2006).

2.2.5. Model Evaluation parameters

2.2.5.1. Confusion matrix

A confusion matrix is used to evaluate a classification model. In a binary classification, for example, when the model is trained with known outcomes, we know if the model made correct predictions or not. Once we have the model predictions, we can assign one of the four labels to each prediction, as in *Figure 8*. The correct predictions are labelled as TP or TN. TP means True Positive is the count of 'trues' that the model correctly predicted. TN or True Negative is the count of 'falses' that the model correctly predicted. The incorrect predictions are labelled as FP or FN. FP means False Positive is known as Type-1 error, it is the count of 'falses' that the model predicted incorrectly; in other words, this is where the model predicted True where it should have predicted False and FN meaning False Negative refers to a case

in which the model predicts ‘false’ when the result is actually ‘true’, in this case the model would have predicted ‘false’ when it should have predicted ‘true’. This is known as a Type-2 error.

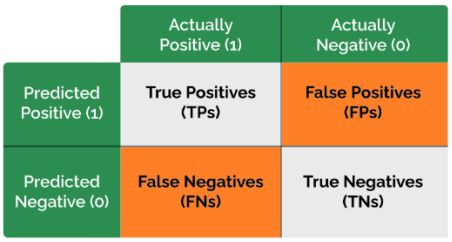


Figure 8. Confusion matrix. Source: own creation

Depending on the problem and use case it is to be decided if we are willing to accept higher number of Type-1 or Type-2 errors. For example, if we are classifying observations as either having cancer or not having cancer, we would be more concerned with high number of False Negatives; in other words, we would want to minimize the number of predictions where the model falsely predicts that a model’s test result does not indicate cancer. Similarly, if we were classifying the observations is either spam mail or not a spam mail, we would be more concerned with high numbers of False Positives, we would want to reduce the number of predictions where the model falsely predicts that a mail is spam.

If the goal is to minimize number of False Negatives or number of failed predictions that have an actual outcome of positive or true. It means we have decided that we are more willing to accept too many False Positives predicting that the mail which is actually spam would be normal email, then too many False Negatives. To minimize the number of False Negatives, we can try different threshold, which is a cut-off point between what gets classified as Positive or Negative. Reducing the default or previously set threshold would minimize number of False Negatives.

2.2.5.2. Accuracy

Accuracy is a simple metric for evaluation of classification models. It is the percentage of observations which are correctly predicted by the model. We calculate the accuracy by the below formula

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

$$\text{Accuracy} = \frac{TP}{TP+TN+FN+FP} \quad (2.6)$$

Accuracy seems simple but should be used with caution when the classes to predict are not balanced.

2.2.5.3. Precision

Precision is often used along with Recall. Precision is the proportion of positive results that were correctly classified. Precision can be compared to FPR (False Positive Rate) that it tells what proportion of false observations were incorrectly classified as true. If we had many more test observations that were false resulting in an imbalance of the test observations then we might want to use precision rather than FPR, this is because precision ignores the number of True Negatives, so if there is imbalance in the sample set, then precision is not effective.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2.7)$$

2.2.5.4. Recall

Recall is the percentage of correct positive classifications or True Positives from cases that are actually positive. Depending on the use case, one may need to check number of False Negatives. For example, if a model results in large number of False Negatives, it might indicate that the model failed to classify the observations correctly.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2.8)$$

2.2.6. Grid Search with Cross Validation

"Grid search is the process of performing hyper parameter tuning in order to determine the optimal values for a given model" (Krishni, 2019). This method allows us to scan through multiple free parameters with few lines of code. The grid search method is the simplest, easiest to implement and understand and crude method, but sadly not efficient when the number of parameters is large. This method is available on different platforms like Python-Sklearn library providing GridSearchCV

class which has a means of automatically iterating over these hyperparameters using cross validation. This method takes the model to be trained and different values of the hyperparameters that are allowed for the underlying algorithm. It then calculates the error for various hyperparameters values allowing to choose the best hyperparameter values.

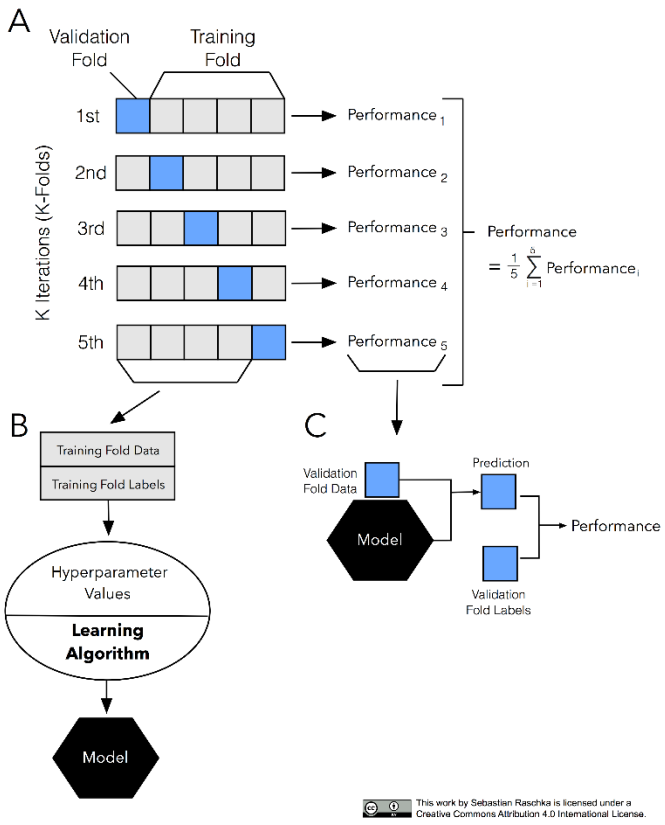


Figure 9. Process involved in Cross validation with 5-folds to evaluate a model. Source: (Sebastian, 2018)

2.2.6.1. Cross Validation

Cross validation is a technique used to estimate the model performance while controlling the model from overfitting especially when the data size is limited. In this technique, data can be partitioned into some number of partitions, also referred as folds, then each fold is analyzed before calculating the mean of overall error. KFold is a type of cross validation which is used in Gridsearch, in which a parameter to specify the number of folds is allowed, that corresponds to cross validation. So here number

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

of folds is specified i.e. if k = 5, data is partitioned into 5 folds. The input data split into 5 folds as shown in the Figure 9 classified as 4 training folds and 1 validation fold which is ultimately used to evaluate the performance. Then the mean of the performances over 5 iterations are calculated to give a mean value. This method is best approach as it results in less biased model and if input data is limited. However, the computation to make the evaluation is downside as it runs from k times.

2.3. Sentiment analysis

To a situation humans understand and respond with certain emotion which is termed as a sentiment. In the same way machines are trained how to understand the nature of data. Sentiment analysis is a field that tries to give machine software the ability to understand the emotions of the user. On an abstract view, it has two main components like Subjectivity and Polarity. Subjectivity expresses about personal feeling, views or beliefs. A subjective expression comes in many forms like opinion, allegation, speculation, beliefs and desires. A subjective sentence may not express any sentiment. For example, ‘Surya is writing a novel’, ‘Thomas wants a camera to capture photos’ etc. While data subjectivity refers to the information's subject, or the word executing an action, the term "objective" refers to traits that pertain to the subject of a phrase or a noun to which an action is performed (MasterClass staff, 2022). Polarity measure tells how positive or negative a statement or data is. In this regard, the perspective of Polarity is divided in to positive(+1) , negative(-1), neutral(0). For example, if there is a string data ‘The flower is beautiful’. Th first step is to remove the unnecessary words like ‘the’ and ‘is’, and identify the important ones. So the sentence becomes ‘flower beautiful’. Each word will be classified into one of the three categories depending on the train data. This will most likely result with a neutral for ‘flower’ and positive for ‘beautiful’. Summation of this will result with score of 1. This means the overall sentiment of the sentence is positive. Achieving this type of nature of data analysis will improve the user experience and it will also result in a better experience for users dealing with machines. This is being achieve with the technology we have in the field of Machine learning.

CHAPTER III: PROJECT/THESIS DEVELOPMENT

3.1 Associated Technologies/Tools

3.1.1 Python

Mark Lutz defined this language as "Python is a general-purpose, open source computer programming language" (Lutz, 2010). A UCAR scientist briefed concisely about advantages of Python as "I have used a combination of Perl, Fortran, NCL, Matlab, R and others for routine research, but found out this general- purpose language, Python, can handle almost all in an efficient way from requesting data from remote online sites to statistics, and graphics" (Python Training, 2020). Python strengths are that it is easy to understand code , means developers can focus more on what they are trying to accomplish, it supports for a wide library of plugins both let it support many kinds of projects and execute them fast. Python helps the developer by being easy to read and write as the code seems like general English. Libraries provided by Python as well as third party libraries available separately make it easier to do many high level tasks. The kinds of manual tasks that python can make easier include managing virtual machines and containers, making APIs for web services and modifying network settings. Because of its enormous features and ease of use Python has become a preferred language for simplifying many other complicated tasks like scientific and statistical work or machine learning applications .

3.1.2. Pandas

Pandas is a Python library for tabular data analysis. Pandas is based on two core Python libraries: matplotlib for data visualization and NumPy for math operations. Pandas acts as a wrapper over these libraries, allowing you to access many of the methods of matplotlib and NumPy with less code. Before pandas, most analysts used Python for data preparation and processing, then switched to a more domain-specific language like R for the rest of their workflow. Pandas introduced two new object types for storing data that make analytical tasks easier and eliminate the need to

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

switch tools: Series, which have a list-like structure, and DataFrames, which have a tabular structure.

3.1.3. Scikit-learn

In the official page Scikit-learn, it has been defined as “an open source machine learning library that supports supervised and unsupervised learning” (scikit-learn developers, n.d.). Its name arguably comes from SciPy Toolkit. It is based on some of the technologies one may already be familiar with, such as NumPy, pandas, and Matplotlib. Sklearn often prefers to work with arrays and it is very fast and efficient. In terms of Machine learning, Sklearn is one of the leading packages by providing its support with clustering, regression, classification, Support vector machines and dimensionality reduction. Sklearn lags supporting in deep learning.

3.1.4. Jupyter Notebook

Jupyter is an acronym for Julia, Python, and R, the three programming languages that Jupyter started with, although it currently supports a large number of languages (Python Training, 2020). Jupyter is a free, open source, interactive web tool known as a Computational Notebook, a single document, as shown in *Figure 10*, where you can run code, see the result, and also add explanations, formulas, charts, and make your work more transparent, understandable, and repeatable. divisible. Its uses include data cleaning and conversion, numerical simulation, statistical modeling, and machines. It will be the tool for carrying out the exercises by developers in the field of Artificial Intelligence, it will combine the theoretical, practical and technical knowledge of Artificial Intelligence with the necessary resources to define the requirements and the prior planning of its implementation, the identification of opportunities. and all its aspects. corporate impact.

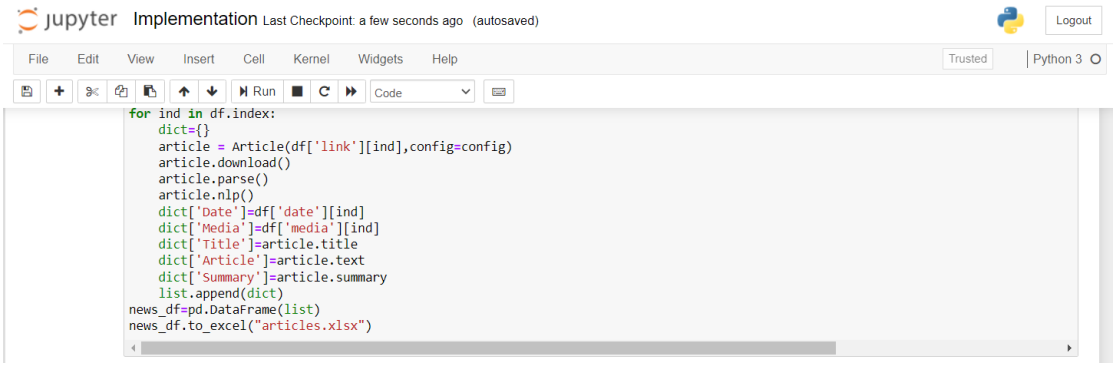


Figure 10. sample screen of jupyter notebook ide with python code. Source: cropped from own python code

3.1.5. R

Ross Ihaka and Robert Gentleman defined R language as "A language for data analysis and graphics" (Ross & Robert, 1996). Though main focus is on visualization and computation of statistical data, being a programming language, it facilitates providing a system for mathematical calculations, debugging and interfaces to other technical languages. Semantics of this language are of the functional programming language, syntax is similar to C and appearance is similar to S. Just like many programming languages, R also has support for user defined functions, parameterization, variables, datatypes. It is clear why the developers of this language highlighted it as a statistical computing language, because of the support it provides with simple pre-defined functions for statistical computation and outputs they return to understand and analyze easily. R being an open source software, any developer can access the source, modify and update it as needed. R community is another advantage for the users and developers with many incredible platforms like twitter, stackoverflow, r bloggers, rstudio community etc.

3.2. Methodology Process

Data mining is defined as "the non-trivial extraction of potentially useful information from a large volume of data, in which the information is implicit, where it is about interpreting large amounts of data and finding relationships or patterns, to achieve this, learning techniques, statistics, and databases will be required" (Molina, 2002)

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

A methodological data mining process has been developed to break down the implementation process of generating a stock market direction prediction model as presented in Figure 11 that consists of 6 phases. The **CR**oss **I**ndustry **S**tandard **P**rocess for **D**ata **M**ining (*CRISP-DM*) is the process model that serves as the base for this data science process. First of all, as understanding the business, problem is defined. In the next phase, necessary data will be collected from different sources namely Yahoo finance and The Guardian news by Web scraping and containers. This data will be processed with the transformation step (Jason, 2021) and ends with the process to extract the necessary attributes or features that combine the feature from the two data sets. Then the model is developed and evaluated, to deploy it and make it available for other applications in the final phase.

3.2.1. Business Understanding

When to buy and sell a stock is a critical problem when the stock market is volatile and effected by global news. Often some noise generates in the stock market because of the news which spreads across the world so fast in this digital era. Volumes change, buyer-seller pressures increase, general public fear about falling market etc. Safe investment with minimal risk becomes a concern for a common retail investor. A model that predicts if the stock will be bullish to buy or bearish to sell on a trading day must be built on the timeseries data of trading days with financial information and global news that are impacting the economy of a place. To develop that, it is also important to figure out what other artifacts contribute to this huge problem like what relevant data supposed to be used, where the data is available and once the relevant information is available it is to be analyze to figure out the problem and build such model.

FIGURE 11. CRISP-DM 1.0 METHODOLOGY. SOURCE: (MITHUN, 2018)

3.2.2. Data Understanding

The data understanding phase begins with initial data collection and continues with activities that allows to become aware of the data, detect data quality issues, determine first insights about the data, and/or spot interesting subsets to form hypotheses. about hidden information, determine the consistency of field values, the number and distribution of null values, find out-of-range values that may be noise for the process. The aim is to ensure the completeness and correctness of the data. The data for this work has been chosen keeping in this in mind. Among various factors that affect stock price movement, global news impacting the economies, statistical values of the company stock and news about the company itself corresponding to fundamental analysis are considered for this problem and worked on the integration of these datasets as depicted in Figure 12.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

3.2.3. Data Preparation

This phase covers all the activities as shown in Figure 12, to create the model required final dataset from the raw data. So, this include data transformation, data profiling, data structuring, sentiment analysis etc. that would prepare the dataset needed.

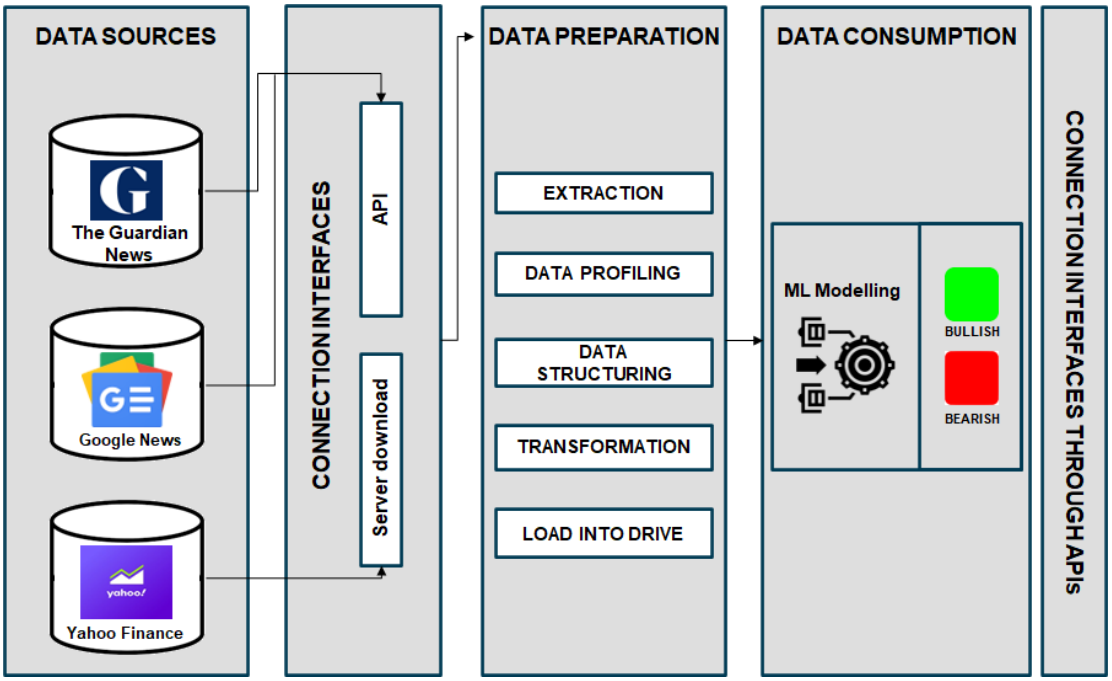


Figure 12. Data flow over several phases to output a prediction through machine learning model. Source: own creation based on the project specific dataset

3.2.4. Modeling

In this phase, experimentation of several models is performed. With the data ready for the models and hyperparameters of several models, modeling is done. This is the model which would ultimately give the output of this classification problem of finding if the stock is bullish or bearish on a trading day after the next phase of evaluation. This has been depicted in the Figure 12 where in the data flow process is shown and used in the modeling in the data consumption phase.

3.2.5. Evaluation

The models built with different techniques in the previous method will be evaluated based on the success criteria and in this work it is the accuracy. All the models created with hyperparameter tuning of every learning algorithm will be evaluated based on accuracy and then a best model will be selected and again used over the test data that is data after the train data since this a time series dataset. Hence the end result is the selection of the best model.

3.2.6. Deployment

Depending on the user requirements further after developing this model, the implementation phase may be to generate a report or implement repeatable data mining. This is where the knowledge obtained is transformed into actions within the business process, either by looking at the model and results, or by applying it to multiple data sets or as part of the process. For example, this can be used to tag to the company stock page with the label of bullish or bearish just like few years website moneycontrol.com (Money Control, s.f.) used to highlight the trend live.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

CHAPTER IV: SOLUTION AND IMPLEMENTATION

4.1. Data gathering and Preparation

Based on the problem statement, in this phase, data from the sources should be identified and collected which is defined as Extraction, followed by transformation process that adopts the steps of the preprocessing method starting from text cleanup, null value removal, redundancy, and any other form of necessary transformations.

4.1.1. Raw data

The sources of the data being open, are gathered from different sources which are available always online. With a preferred approach of collecting data, financial information of a company stocks in the market during the trading days and also corresponding days' global news information, making second dataset, and main news headlines of Reliance company on business days and recent holidays, making the third dataset, are mainly contributing to generate the input data for analysis and model building.

4.1.1.1. Financial data

This is one of the datasets that gives the information on the public stock statistics of any company. This data is collected from Yahoo finance website publicly. This data has 7 attributes (continuous data) such as: company stock values, date, opening value, closing value, maximum, minimum, adjusted closing value and volume levels.

Method for data collection

Date range : 28.02.2014 - 27.04.2022

Number of records : 2009

Source : Yahoo finance page of Reliance Industries page, Reliance Industries Limited (RELIANCE.NS) Stock Price, News, Quote & History - Yahoo Finance

Stock data of any publicly listed company on exchange is available on Yahoo finance website (Yahoo finance, n.d.). For this study in the stock details page of DJIA company, under the 'historical data' menu, almost 7 years of data is downloaded. The details of the variables available in the downloaded data file is described in the Table 1

TABLE 1. DATA VARIABLES AVAILABLE IN DOWNLOADED FINANCIAL DATA OF RELIANCE INDUSTRIES

Variable	Description
Date	Date of the trading day on which the statistics are being recorded.
Open	At the beginning of trading data, it is the price at which first or few first buyers and sellers get together at an agreed upon price.
Close	The price of the stock at which the trading day is closed.
High	The bottom of the bar chart. It is the highest price of which the transaction occurred throughout the day.
Low	The bottom of the bar chart. It is the lowest price of which the transaction occurred throughout the day.
Adj. Close	The stock price that adjusts its closing price to reflect the stock's worth after any corporate activities have been taken into account.
Volume	Volume is the quantity of the stocks that transactions between sellers and buyers at a point of time.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

	A	B	C	D	E	F	G
1	Date	Open	High	Low	Close	Volume	Adj Close
2	08-08-2012	11432.08984	11759.95996	11388.04004	11734.32031	212830000	11734.32031
3	08-11-2012	11729.66992	11867.11035	11675.53027	11782.34961	183190000	11782.34961
4	08-12-2012	11781.7002	11782.34961	11601.51953	11642.46973	173590000	11642.46973
5	13-08-2012	11632.80957	11633.78027	11453.33984	11532.95996	182550000	11532.95996
6	14-08-2012	11532.07031	11718.28027	11450.88965	11615.92969	159790000	11615.92969
7	15-08-2012	11611.20996	11709.88965	11599.73047	11659.90039	215040000	11659.90039
8	18-08-2012	11659.65039	11690.42969	11434.12012	11479.38965	156290000	11479.38965
9	19-08-2012	11478.08984	11478.16992	11318.5	11348.54981	171580000	11348.54981
10	20-08-2012	11345.94043	11454.15039	11290.58008	11417.42969	144880000	11417.42969
11	21-08-2012	11415.23047	11476.20996	11315.57031	11430.20996	130020000	11430.20996
12	22-08-2012	11426.79004	11632.12988	11426.79004	11628.05957	138790000	11628.05957
13	25-08-2012	11626.19043	11626.26953	11362.62988	11386.25	148610000	11386.25
14	26-08-2012	11383.55957	11436.24023	11340.41016	11412.87012	119800000	11412.87012
15	27-08-2012	11412.45996	11554.45996	11381.76953	11502.50977	120580000	11502.50977
16	28-08-2012	11499.87012	11715.17969	11499.79004	11715.17969	149150000	11715.17969
17	29-08-2012	11713.23047	11713.23047	11543.38965	11543.95996	166910000	11543.95996
18	09-02-2012	11545.62988	11790.16992	11471.90039	11516.91992	177090000	11516.91992
19	09-03-2012	11506.00977	11554.37988	11416.53027	11532.87988	174250000	11532.87988
20	09-04-2012	11532.48047	11532.48047	11176.01953	11188.23047	229200000	11188.23047
21	09-05-2012	11185.62988	11245.15039	11037.84961	11220.95996	198300000	11220.95996
22	09-08-2012	11224.87012	11570.66016	11224.79004	11510.74023	273000000	11510.74023
23	09-09-2012	11514.73047	11577.5	11230.73047	11230.73047	257300000	11230.73047
24	09-10-2012	11233.91016	11380.62988	11215.25977	11268.91992	214260000	11268.91992
25	09-11-2012	11264.44043	11445.67969	11098.66992	11433.70996	247820000	11433.70996

Figure 13. Some records of downloaded Financial data of Reliance Industries, Source: **(RELIANCE INDUSTRIES)**

4.1.1.2. Company specific news data

The second dataset is made of contemporary information of the company, in this case Reliance. The data source is well known Google news page. Google news encapsulates freely accessible and publicly disclosed information from various sources like, newspapers, channels, blogs, websites, conference coverages etc. Finding news about a company every day on news is not general. No company gets into headlines of news all the days. Even no public newspaper or website or channel shows information of a company all the days. Famous companies can make it out with slightly more probability but now always though. Keeping this in mind, source like Google fits into the requirement, which itself gets good information and highlights from different sources being a search engine. Maximum of 5 Reliance company related headlines are collected on every calendar day and a complete of 8 years data.

Method for data collection

APIs : googlenews, pandas

Date range : 28.02.2014 - 27.04.2022

Number of records : 2007

Language : Python

The google news is publicly available to those who access internet, the same data has been collected using the free python library to access its API and get 'Reliance' company specific data. On some business days no news on Reliance, on some weekends or holidays there might talks around Reliance. Such cases are also taken in to account while profiling the data. The news on non-business days are grouped with the following trading days news. Since it makes sense the company news shows impact on coming trading day indirectly through investors' opinions. So for a certain business day maximum of 5 company specific news headlines are saved. This is processed through the sentiment analysis function implemented in Python with necessary libraries Vadersentiment and Textblob. This would generate necessary numerical data that can be integrated with remaining features of the model.

4.1.1.3. Global news data

The third document is a contextual dataset with yet another contemporary information around the world. This document has nominal data of showing maximum of 25 daily news headlines from 'The Guardian' news website for every trading day. The Guardian news website has won 'The best news website' award in the last press awards. Collecting global news from such news website seemed reasonable. The collected news headlines are transformed to generate numerical data using sentiment analysis technique that reflects how important these variables in addressing the problem.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Method for data collection

API : newspaper, pandas

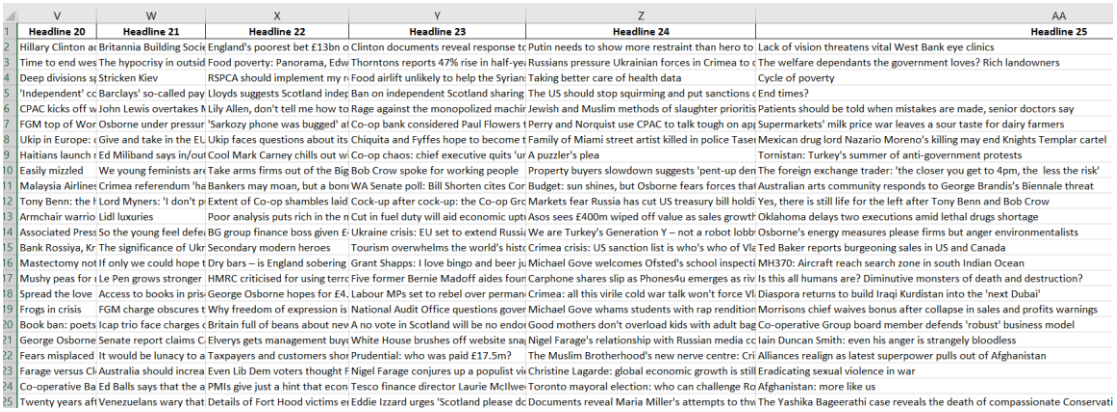
Date range : 28.02.2014 - 27.04.2022

Number of records : 2007

Language : Python

The Guardian news website is public accessible and available online. The page contains several news which can be read, saved and shared with others. This news information is collected by using the news api with Python. This data collection process is iterated for number of times that gets tracks data for 8 years of days and thus obtain 30 top headlines in scan of the news pages for every iteration. This generates a huge file with number of rows more than 2000 of which business days are subset. The dates in each document are mapped clearly. Similar approach of data profiling and processing is carried out on this dataset as well including Sentiment analysis. *Figure 14* shows an example of The Guardian news website page and *Figure 15* is the data collected through API of news data.

Figure 14. Cropped image from The Guardian news website published on 4th June 2022. Source: (The Guardian International, s.f.)



	V	W	X	Y	Z	AA
1	Headline 20	Headline 21	Headline 22	Headline 23	Headline 24	Headline 25
2	Hillary Clinton a	Britannia Building Soci	England's poorest bet £13bn o	Clinton documents reveal response t	Putin needs to show more restraint than hero to	Lack of vision threatens vital West Bank eye clinics
3	Time to end wes	The hypocrisy in outsid	Food poverty: Panorama, Edw	Thorntons reports 47% rise in half-ye	Russians pressure Ukrainian forces in Crimea to c	The welfare dependants the government loves? Rich landowners
4	Deep divisions s	Stricken Kiev	RSPCA should implement my r	Food airlift unlikely to help the Syrian	Taking better care of health data	Cycle of poverty
5	'Independent' cc	Barclays' so-called pay	Lloyds suggests Scotland inde	Ban on independent Scotland sharing	The US should stop squirming and put sanctions c	End times?
6	CPAC kicks off w	John Lewis overtakes M	Lily Allen, don't tell me how to	Rage against the monopolized machir	Jewish and Muslim methods of slaughter prioritis	Patients should be told when mistakes are made, senior doctors say
7	FGM top of Wor	Osborne under pressur	'Sarkozy phone was bugged' al	Co-op bank considered Paul Flowers t	Perry and Norquist use CPAC to talk tough on ap	Supermarkets' milk price war leaves a sour taste for dairy farmers
8	Ukip in Europe: (	Give and take in the EU	Ukip faces questions about its	Chiquita and Fyffes hope to become t	Family of Miami street artist killed in police Tase	Mexican drug lord Nazario Moreno's killing may end Knights Templar cartel
9	Haitians launch r	Ed Miliband says in/out	Cool Mark Carney chills out wi	Co-op chaos: chief executive quits 'ur	A puzzler's plea	Tornistan: Turkey's summer of anti-government protests
10	Easily mizzled	We young feminists are	Take arms firms out of the Big	Bob Crow spoke for working people	Property buyers slowdown suggests 'pent-up den	The foreign exchange trader: 'the closer you get to 4pm, the less the risk'
11	Malaysia Airline	Crimea referendum 'ha	Bankers may moan, but a bon	WA Senate poll: Bill Shorten cites Cor	Budget: sun shines, but Osborne fears forces that	Australian arts community responds to George Brandis's Biennale threat
12	Tony Benn: the I	Lord Myners: 'I don't p	Extent of Co-op shambles laid	Cock-up after cock-up: the Co-op Grc	Markets fear Russia has cut US treasury bill holdi	Yes, there is still life for the left after Tony Benn and Bob Crow
13	Armchair warrio	Lidl luxuries	Poor analysis puts rich in the n	Cut in fuel duty will aid economic upt	Asos sees £400m wiped off value as sales growth	Oklahoma delays two executions amid lethal drugs shortage
14	Associated Press	So the young feel defe	BG group finance boss given £	Ukraine crisis: EU set to extend Russi	We are Turkey's Generation Y – not a robot lobb	Osborne's energy measures please firms but anger environmentalists
15	Bank Rossiya, Kr	The significance of Ukr	Secondary modern heroes	Tourism overwhelms the world's histc	Crimea crisis: US sanction list is who's who of Vla	Ted Baker reports burgeoning sales in US and Canada
16	Mastectomy not	If only we could hope t	Dry bars – is England sobering	Grant Shapps: I love bingo and beer ju	Michael Gove welcomes Ofsted's school inspecti	MH370: Aircraft reach search zone in south Indian Ocean
17	Mushy peas for I	Le Pen grows stronger	HMRC criticised for using terrc	Five former Bernie Madoff aides foun	Carphone shares slip as Phones4u emerges as riv	Is this all humans are? Diminutive monsters of death and destruction?
18	Spread the love	Access to books in pris	George Osborne hopes for £4.	Labour MPs set to rebel over perman	Crimea: all this virile cold war talk won't force Vl	Diaspora returns to build Iraqi Kurdistan into the 'next Dubai'
19	Frogs in crisis	FGM charge obscures t	Why freedom of expression is	National Audit Office questions gover	Michael Gove whams students with rap rendition	Morrisons chief waives bonus after collapse in sales and profits warnings
20	Book ban: poets	Icap trio face charges c	Britain full of beans about nev	A no vote in Scotland will be no endo	Good mothers don't overload kids with adult bag	Co-operative Group board member defends 'robust' business model
21	George Osborne	Senate report claims C	Elverys gets management buy	White House brushes off website sna	Nigel Farage's relationship with Russian media cc	Iain Duncan Smith: even his anger is strangely bloodless
22	Fears misplaced	It would be lunacy to a	Taxpayers and customers shor	Prudential: who was paid £17.5m?	The Muslim Brotherhood's new nerve centre: Cri	Alliances realign as latest superpower pulls out of Afghanistan
23	Farage versus Cl	Australia should increa	Even Lib Dem voters thought F	Nigel Farage conjures up a populist vi	Christine Lagarde: global economic growth is still	Eradicating sexual violence in war
24	Co-operative Ba	Ed Balls says that the a	PMIs give just a hint that econ	Tesco finance director Laurie McIlwe	Toronto mayoral election: who can challenge Ro	Afghanistan: more like us
25	Twenty years aft	Venezuelans wary that	Details of Fort Hood victims e	Eddie Izzard urges 'Scotland please dc	Documents reveal Maria Miller's attempts to thv	The Yashika Bageerathi case reveals the death of compassionate Conservati

Figure 15. Some records showing the news headlines from The Guardian news website. Source: cropped image from own data file

4.1.2. Data cleaning

Based on the objective of the project and the data dictionary, attributes that were not considered relevant were eliminated, later the data cleaning was carried out using the Jupyter notebook with the Pandas and Python library where a code was written to detect null values, duplicates and data anomalies. Finally, a separate code was created to correct each deviation. There are some null values and duplicate rows in the combined dataset that need to be fixed to clean up the data.

Null values

The downloaded financial data from Yahoo finance website showed some null rows. These rows are of holiday days. No important information is missing for the trading days. So, these rows can be deleted from the dataset without any further analysis needed. This leaves the document clean and proper. Remaining two datasets are having proper data they were collected through efficient python code. However all these columns together can be cleaned if they have null rows like Yahoo finance data, by below code step, where 'df' is dataframe variable of pandas and dropna is a function to drop the rows with null values.

df.dropna()

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Duplicates

Rarely it will be found in the data set with a few duplicate records. These are redundant and can be removed from the data set. Redundant data rows can be removed using Python code. The python method/code used to achieve this is:

df.drop_duplicates(keep = False, inplace = True)

4.1.3. Data Profiling

Data profiling is "the process of analyzing and exploring data to understand how it's structured, what it contains, the relationships between data sets, and how it could potentially be used most effectively (data-profiling, n.d.). With three different data sources keying on the 'dates' of business days over past 8 years requires some profiling. The maximum number of news data on a day is limited to 25. So, it first gets maximum of 5 news headlines from google news which is about company specific news. These 5 news headlines are picked from the group of current and business day and the followed by continuous holidays. So now the news dataset comprises of 0-5 headlines. Applying the same approach on 'The Guardian' news data, 20-25 headlines are finalized. The sum of these two would get us an integrated dataset of 25 news headlines. So all the rows pertaining to non-business days are filtered out by this process and business days' financial data is now mappable to integrated news headlines dataset.

4.1.4. Data Transformation

Financial dataset has continuous values and News headlines datasets has nominal values. The latter datasets are processed through sentiment analysis to get corresponding numeric values out of it. All the 25 headlines corresponding to every trading day is carried out with sentiment analysis process and subjectivity, objectivity, positive, neutral and negative values are saved aside. *Figure 16*. shows the final newly generated dataset with sentiment analysis data over news dataset.

	A	B	C	D	E	F
1	Date	Subjectivity	Objectivity	Positive	Neutral	Negative
2	08-08-2012	75	25	18.75	25	56.25
3	08-11-2012	83.33333333	16.66666667	41.66666667	16.66666667	41.66666667
4	08-12-2012	56.25	43.75	18.75	43.75	37.5
5	13-08-2012	38.46153846	61.53846154	15.38461538	61.53846154	23.07692308
6	14-08-2012	45.45454545	54.54545455	36.36363636	54.54545455	9.090909091
7	15-08-2012	70	30	10	30	60
8	18-08-2012	100	0	0	0	100
9	19-08-2012	22.22222222	77.77777778	22.22222222	77.77777778	0
10	20-08-2012	70	30	10	30	60
11	21-08-2012	50	50	20	50	30
12	22-08-2012	50	50	0	50	50
13	25-08-2012	55.55555556	44.44444444	22.22222222	44.44444444	33.33333333
14	26-08-2012	66.66666667	33.33333333	0	33.33333333	66.66666667
15	27-08-2012	30.76923077	69.23076923	30.76923077	69.23076923	0
16	28-08-2012	33.33333333	66.66666667	0	66.66666667	33.33333333
17	29-08-2012	27.27272727	72.72727273	18.18181818	72.72727273	9.090909091
18	09-02-2012	46.66666667	53.33333333	0	53.33333333	46.66666667
19	09-03-2012	16.66666667	83.33333333	0	83.33333333	16.66666667
20	09-04-2012	40	60	20	60	20
21	09-05-2012	36.36363636	63.63636364	9.090909091	63.63636364	27.27272727
22	09-08-2012	23.07692308	76.92307692	0	76.92307692	23.07692308
23	09-09-2012	53.84615385	46.15384615	15.38461538	46.15384615	38.46153846
24	09-10-2012	33.33333333	66.66666667	33.33333333	66.66666667	0
25	09-11-2012	16.66666667	83.33333333	0	83.33333333	16.66666667

Figure 16. Part of the data after transformation using sentiment analysis. Source: own generation using python code

4.1.5. Data Labelling

Data labelling, also called data annotation, is the process of adding tags or labels or any other information to a set of data (Data annotation & Data labeling, 2022). The current dataset is a timeseries dataset with financial and contemporary news information. On every business day if the stock is bullish or bearish is the target that is going to be predicted. Hence, it is clear that labels of the dataset are binary. The stock values are listed out in the dataset for over 8 years. So, the label of a row is marked '1(bullish)' if the stock value is increased from previous day and if the stock value on a business day is decreased from previous trading day, the row is labelled as '0(bearish)'. This annotation can be done by a human manually or can be automated. Since it is a time-consuming task when done manually, it has been automated to generate the labels for the data with Python script. The generated labels are concatenated to the integrated dataset.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

4.2. Feature Extraction

Important features contributing in predicting the label, out of 11 total features are figured out. Firstly, as part of feature selection, 10 features excluding 'Close' feature, which has been used in feature labelling, are selected and feature extraction is performed on these preferred 10 features namely Open, High, Low, Volume, Adj Close, Subjectivity, Objectivity, Neutral, Negativity and Positivity. *Figure 17* shows the correlation between the variables before feature extraction technique is applied to analyze how they are correlated to each other and . From the figure it is theoretically makes sense as Subjectivity and Objectivity are contrasting features and so are Positivity, Neutral and Negativity features. There is no drastic correlation between any two variables as we see logically, no attribute has great dependency on other attribute in the context of stock market. However the financial data variables showed some great correlation among themselves and Neutral just like Objectivity are not valued worst in correlation with most of other variables. This are evident from the correlation statistics visible in the *Figure 17*.

	Open	High	Low	Adj Close	Volume	Subjectivity	Objectivity	Positive	Neutral	Negative
Open	1.00000000	0.99973462	0.99974975	0.99946005	0.18425466	-0.13387165	0.13387165	-0.09268598	0.23985815	-0.20676142
High	0.99973462	1.00000000	0.99968305	0.99979684	0.19388017	-0.13396049	0.13396049	-0.09305726	0.23998418	-0.20660827
Low	0.99974975	0.99968305	1.00000000	0.99976546	0.17809495	-0.13372599	0.13372599	-0.09221807	0.23963409	-0.20685191
Adj Close	0.99946005	0.99979684	0.99976546	1.00000000	0.18687901	-0.13445515	0.13445515	-0.09272598	0.23989919	-0.20675536
Volume	0.18425466	0.19388017	0.17809495	0.18687901	1.00000000	-0.05660859	0.05660859	-0.06319919	0.07317968	-0.03571536
Subjectivity	-0.13387165	-0.13396049	-0.13372599	-0.13445515	-0.05660859	1.00000000	-1.00000000	0.13054869	-0.15424355	0.07678400
Objectivity	0.13387165	0.13396049	0.13372599	0.13445515	0.05660859	-1.00000000	1.00000000	-0.13054869	0.15424355	-0.07678400
Positive	-0.09268598	-0.09305726	-0.09221807	-0.09272598	-0.06319919	0.13054869	-0.13054869	1.00000000	-0.53286824	-0.16538787
Neutral	0.23985815	0.23998418	0.23963409	0.23989919	0.07317968	-0.15424355	0.15424355	-0.53286824	1.00000000	-0.74633638
Negative	-0.20676142	-0.20660827	-0.20685191	-0.20675536	-0.03571536	0.07678400	-0.07678400	-0.16538787	-0.74633638	1.00000000

Figure 17. Correlation matrix of Features. Source: own creation based on R code

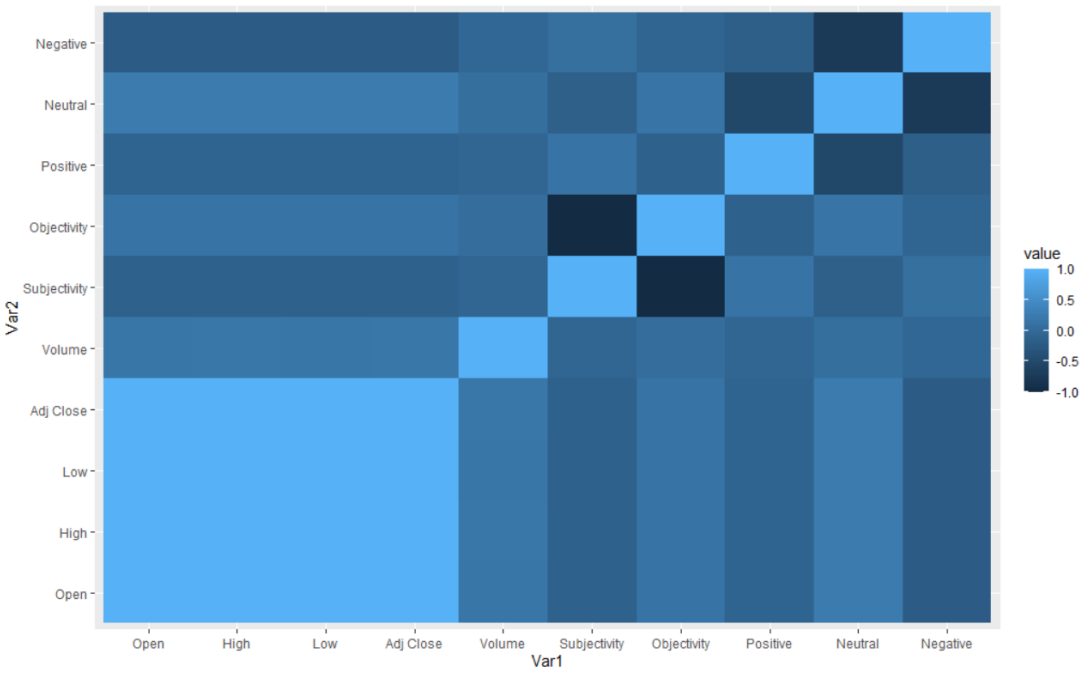


Figure 18. Correlation Heatmap. Source: created using R code on own dataset

4.2.1. Principal Component Analysis(PCA)

Principal component analysis (PCA) (Hotelling, 1933; Pearson, 1901) is a dimension reduction and decorrelation technique that transforms a correlated multivariate distribution into orthogonal linear combinations of the original variables (Barnett, 2017). PCA is centrally a dimension reduction technique in order to condense the features further down in order to capture as much variance as possible while at the same time minimize the amount of features PCA in particular going to provide us with a data driven hierarchical coordinate system based on data to represent the statistical variations in the dataset. It determines the direction of the linear relationship between two continuous variables.

PCA has been described by many researchers have from different perspectives since its invention and definition, in 1901, through approximating multivariate distributions by lines and planes, by Pearson (Alexander & Andrei, 2008; Pearson, 1901). Among these, in machine learning and pattern recognition, to define PCA, covariance matrix of training samples is well known and commonly used concept. So, PCA revolves around Covariance matrix. Considering k variables with n number of samples which

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

means for this dataset, 10 variables and 2007 number of samples. Before making out PCA, these variables are transformed to have a mean of 0 and variance of 1. The core of PCA is represented by Eigen values and Eigen vectors, where eigenvectors(principal components), also referred to as right vectors or column vectors, are unit vectors with a magnitude of one determine the magnitude and eigenvalues are coefficients applied to these vectors to give them their magnitude. "The eigenvector matrix may be thought of as a rotation matrix, providing a new basis where the correlated data are made orthogonal" (Barnett, 2017). The eigenvectors identified for the principal components generated for this stock dataset in R can been seen in *Figure 19*. Eigenvectors are fetched using ´loadings´ held by ´princomp´ function output and eigenvalues are calculated from the formula of squaring standard deviation ´sdev´ details held by ´princomp´ output as shown in *Figure 20*.

```
Loadings:
             Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9 Comp.10
Open          0.467  0.144                0.733                0.459
High          0.468  0.144                       0.668        -0.551
Low           0.467  0.144                      -0.742        -0.445
Adj Close     0.467  0.143                -0.679                0.537
Volume        0.117                0.192 -0.974
Subjectivity -0.135  0.618 -0.299  0.104                                     0.707
Objectivity   0.135 -0.618  0.299 -0.104                                     0.707
Positive             0.231  0.242 -0.803 -0.163               0.456
Neutral       0.202 -0.282 -0.645                             0.675
Negative     -0.165  0.146  0.562  0.538  0.101               0.579

               Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9 Comp.10
SS loadings       1.0    1.0    1.0    1.0    1.0    1.0    1.0    1.0    1.0     1.0
Proportion Var    0.1    0.1    0.1    0.1    0.1    0.1    0.1    0.1    0.1     0.1
Cumulative Var    0.1    0.2    0.3    0.4    0.5    0.6    0.7    0.8    0.9     1.0
```

Figure 19. Eigenvectors calculated using R code. Source: own calculation

```
      Comp.1       Comp.2       Comp.3       Comp.4       Comp.5       Comp.6
4.317253e+00 2.012965e+00 1.576000e+00 1.147024e+00 9.459253e-01 5.379133e-04
      Comp.7       Comp.8       Comp.9      Comp.10
1.886773e-04 6.124175e-05 4.564510e-05 0.000000e+00
```

Figure 20. Eigenvalues calculated using R code. Source: own calculation

Maximum number of principal components that could be created for a dataset with n number of features is equal to n. So, for this dataset maximum number of principal components possible to create are 10. From *Figure 21* that is evident, which have been created in R language with the function 'princomp' with the parameter 'cor = TRUE', which is basically a logical value representing whether the function should use the

covariance matrix or the correlation matrix; and in this case correlation matrix has been used during the creation of principal components.

```
Importance of components:
                          Comp.1    Comp.2    Comp.3    Comp.4     Comp.5       Comp.6       Comp.7       Comp.8
Standard deviation     2.0777999 1.4187900 1.2553883 1.0709920 0.97258691 2.319296e-02 1.373599e-02 7.825711e-03
Proportion of Variance 0.4317253 0.2012965 0.1576000 0.1147024 0.09459253 5.379133e-05 1.886773e-05 6.124175e-06
Cumulative Proportion  0.4317253 0.6330218 0.7906217 0.9053241 0.99991665 9.999704e-01 9.999893e-01 9.999954e-01
```

Figure 21. Summary of Principal component analysis showing calculated Standard deviation, Proportion of Variance and Cumulative proportion. Source: own calculation using R code.

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8
Open	0.971	0.204	0.108	0.003	0.059	0.017	0.001	0.000
High	0.972	0.204	0.108	0.005	0.049	-0.001	0.009	0.000
Low	0.971	0.204	0.108	0.002	0.065	-0.001	-0.010	0.000
Adj Close	0.971	0.203	0.108	0.004	0.056	-0.016	0.001	0.000
Volume	0.244	-0.005	0.033	0.206	-0.947	0.000	0.000	0.000
Subjectivity	-0.280	0.876	-0.375	0.111	-0.006	0.000	0.000	0.000
Objectivity	0.280	-0.876	0.375	-0.111	0.006	0.000	0.000	0.000
Positive	-0.185	0.328	0.304	-0.860	-0.159	0.000	0.000	0.004
Neutral	0.419	-0.400	-0.810	0.086	0.023	0.000	0.000	0.005
Negative	-0.343	0.207	0.706	0.576	0.098	0.000	0.000	0.005

Figure 22. Correlation matrix of Principal components and actual feature in the dataset. Source: own calculation based on R code

The Figure 22 is an output from a code in R, that shows the correlation matrix between the data available in the stock dataset and the principal components generated. This is incredibly useful in order to determine if there is any slight relevancy between the data held under the actual features and the newly generated component´s data. The higher it is to one, closer it is related to corresponding principal component. It is already seen from the cumulative proportion that first 5 components held more than 0.90; similarly many actual feature are correlated more than 0.5 with certain principal components like first principal component well correlated with statistical information providing variables of stock data, they are, Low, High, Adj Close and Open. Second component showed good correlation of 0.876 with Subjectivity feature. Similarly Principal components 3 and 4 with Negative feature with the scores of 0.706 and 0.576 respectively. The correlation of the components with the actual variables is seen decreasing from first principal component to the last just contradictory to the cumulative proportion accumulation.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Figure 23 has been generated in R which is graph that displays relation between the principal components and variances against them. Considering the elbow method (Wamika, 2021) to determine number of principal components that could be engaged. 5 principal components are explaining most of the data variation which as it is already seen from Figure 21 cumulative proportion of first 5 principal components was 0.99.

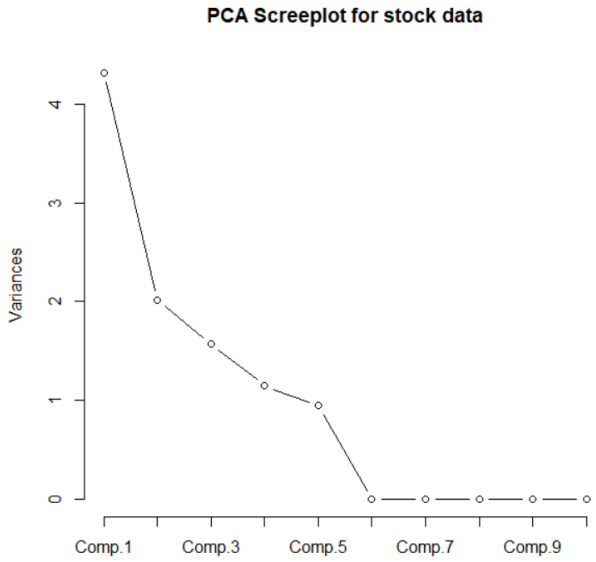


Figure 23. Principal component Analayis Screeplot for stock data

The components chosen from this screeplot is not a rule of thumb. It depends upon the data, problem statement and the analysis that a user wants to perform with the data. The objective of this work since not just descriptive analysis and a machine learning model that could resolve a problem, the number of components obtained from this screeplot analysis can be considered as a base and with the hyperparameter tuning of how the model responds in giving out the results this can be altered to get the best output.

Corresponds to the selection of a suitable and specific model; For this, techniques that is appropriate to the problem, have adequate data, meet the requirements of the problem, appropriate technique to obtain a model and full knowledge of the technique.

A plan must be generated to test the quality and validity of the built model, the selected technique is executed on the data prepared to generate one or more models.

All modeling techniques have a set of parameters that determine characteristics of the model to be generated, in the same way the models must be interpreted according to the knowledge of the domain and the pre-established success criteria.

This study will specifically apply various machine learning classification methods to identify the pattern and thus predict the direction in the form of classification. The study chooses 4 different machine learning algorithms that are generic classification algorithms namely Logistic Regression, Decision Trees, SVC, XGBClassifier. With the available parameters for each technique multiple models are created with hyperparameter tuning to find out best model in terms of accuracy as the success criteria during the evaluation. Figure 24 is an overview explaining the process how models are built and evaluated to finally propose a model and get the predictions on real time test data and make it a final model.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

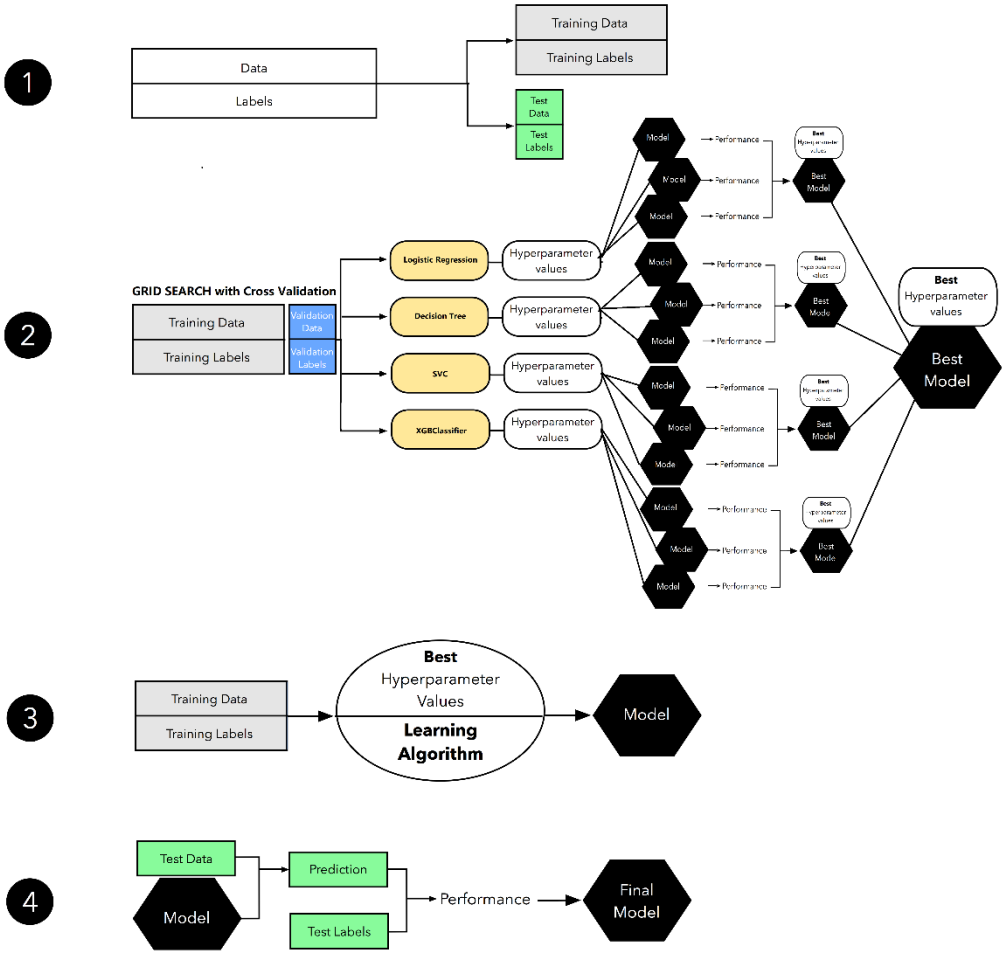


Figure 24. Process flow to find best model. 1.Data partitioning 2. Hyperparameter tuning with Gridsearch to build multiple models 3. Fitting the training data into best model identified 4. Predicting with test data. Source: own creation

4.3. Data partitioning

The dataset, model gets inputted with, will have the time series data with financial statistical data and in addition to that data transformed with sentiment analysis from The Guardian news headlines. This makes a dataset of 10 independent variables and one dependent variable that says the direction or trend of the stock during the trading day. Below are the variables in the dataset that will be inputted into the models.

- Date
- Open
- Volume
- High
- Low
- Adj. Close

- Subjectivity
- Objectivity
- Positive
- Neutral
- Negative
- Label (Dependent variable)

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Date	Subjectivity	Objectivity	Positive	Neutral	Negative	Open	High	Low	Close	Volume	Adj Close	Label
2	08-08-2012	75	25	18.75	25	56.25	11432.08984	11759.95996	11388.04004	11734.32031	212830000	11734.32031	0
3	08-11-2012	83.33333333	16.66666667	41.66666667	16.66666667	41.66666667	11729.66992	11867.11035	11675.53027	11782.34961	183190000	11782.34961	1
4	08-12-2012	56.25	43.75	18.75	43.75	37.5	11781.7002	11782.34961	11601.51953	11642.46973	173590000	11642.46973	0
5	13-08-2012	38.46153846	61.53846154	15.38461538	61.53846154	23.07692308	11632.80957	11633.78027	11453.33984	11532.95996	182550000	11532.95996	0
6	14-08-2012	45.45454545	54.54545455	36.36363636	54.54545455	9.090909091	11532.07031	11718.28027	11450.88965	11615.92969	159790000	11615.92969	1
7	15-08-2012	70	30	10	30	60	11611.20996	11709.88965	11599.73047	11659.90039	215040000	11659.90039	1
8	18-08-2012	100	0	0	0	100	11659.65039	11690.42969	11434.12012	11479.38965	156290000	11479.38965	0
9	19-08-2012	22.22222222	77.77777778	22.22222222	77.77777778	0	11478.08984	11478.16992	11318.5	11348.54981	171580000	11348.54981	0
10	20-08-2012	70	30	10	30	60	11345.94043	11454.15039	11290.58008	11417.42969	144880000	11417.42969	1
11	21-08-2012	50	50	20	50	30	11415.23047	11476.20996	11315.57031	11430.20996	130020000	11430.20996	1
12	22-08-2012	50	50	0	50	50	11426.79004	11632.12988	11426.79004	11628.05957	138790000	11628.05957	1
13	25-08-2012	55.55555556	44.44444444	22.22222222	44.44444444	33.33333333	11626.19043	11626.26953	11362.62988	11386.25	148610000	11386.25	0
14	26-08-2012	66.66666667	33.33333333	0	33.33333333	66.66666667	11383.55957	11436.24023	11340.41016	11412.87012	119800000	11412.87012	1
15	27-08-2012	30.76923077	69.23076923	30.76923077	69.23076923	0	11412.45996	11554.45996	11381.76953	11502.50977	120580000	11502.50977	1
16	28-08-2012	33.33333333	66.66666667	0	66.66666667	33.33333333	11499.87012	11715.17969	11499.79004	11715.17969	149150000	11715.17969	1
17	29-08-2012	27.27272727	72.72727273	18.18181818	72.72727273	9.090909091	11713.23047	11713.23047	11543.38965	11543.95996	166910000	11543.95996	0
18	09-02-2012	46.66666667	53.33333333	0	53.33333333	46.66666667	11545.62988	11790.16992	11471.90039	11516.91992	177090000	11516.91992	0
19	09-03-2012	16.66666667	83.33333333	0	83.33333333	16.66666667	11506.00977	11554.37988	11416.53027	11532.87988	174250000	11532.87988	1
20	09-04-2012	40	60	20	60	20	11532.48047	11532.48047	11176.01953	11188.23047	229200000	11188.23047	0
21	09-05-2012	36.36363636	63.63636364	9.090909091	63.63636364	27.27272727	11185.62988	11245.15039	11037.84961	11220.95996	198300000	11220.95996	1
22	09-08-2012	23.07692308	76.92307692	0	76.92307692	23.07692308	11224.87012	11570.66016	11224.79004	11510.74023	273000000	11510.74023	1
23	09-09-2012	53.84615385	46.15384615	15.38461538	46.15384615	38.46153846	11514.73047	11577.5	11230.73047	11230.73047	257300000	11230.73047	0
24	09-10-2012	33.33333333	66.66666667	33.33333333	66.66666667	0	11233.91016	11380.62988	11215.25977	11268.91992	214260000	11268.91992	1
25	09-11-2012	16.66666667	83.33333333	0	83.33333333	16.66666667	11264.44043	11445.67969	11098.66992	11433.70996	247820000	11433.70996	1

Figure 25. part of the data to be inputted into models. Source: own generation local file

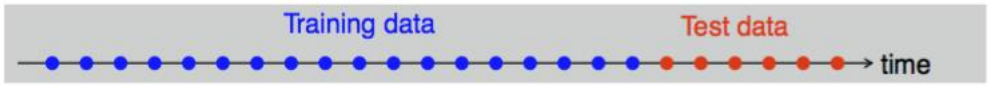


Figure 26. Representation of how timeseries data is split before inputting into models.

Having this dataset as source for the machine learning models that will be constructed, it will be split into training data and test data in the ratio of 80:20. So out of 2007 total business day corresponding records, 1605 records will be held in training data and rest for test dataset. Important point here to be considered is that the data shouldn't be randomized to balance or neutralize or with an intention of having fair distribution as it is done in most of the cases because this data is a time series data and it is supposed to be used in sequence to let the model learn as it is as represented in the Figure 26.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Figure 27. Data distribution. a. before splitting the data into train and test data. a1. after splitting, distribution of train data. a2. test data distribution after splitting. Source: own creation with matplotlib

4.4. Model Selection

This is a phase of experimentation with different learning algorithms and number of models generated through parameter tuning. This has been achieved by tuning the hyperparameters of every learning technique by using Grid Search, which is perhaps the simplest and crude method. This method is implemented with Python-Sklearn library providing GridSearchCV class. This method allowed to try all possible combinations of the given parameters, Cross-validate the models using k-fold cross-validation and thus evaluate the performance of each combination of hyperparameters. Gridsearch achieves this by splitting the data passed further into train and validation to tune the hyperparameters passed to it as explained in section 2.2.6.

4.4.1. Logistic Regression - Hyperparameter tuning

This class implements regularized logistic regression using the 'liblinear' library, 'newton-cg', 'sag', 'saga' and 'lbfgs' solvers (scikit-learn developers, s.f.). Hence the parameters C, which indicates the inverse of regularization strength, 'penalty' which specifies the type and norm of regularization for objective function coefficients and solver which demonstrates the type of optimization function. Different values are chosen for these parameters to make the search space, tabulated in **TABLE 2**, for this learning algorithm creating 36 models.

Table 2. Hyperparameters search space for Logistic Regression learning algorithm

Parameter	***Search space***
solver	newton-cg, lbfgs, liblinear, saga
penalty	l2, none
C	100, 10, 1.0, 0.1

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

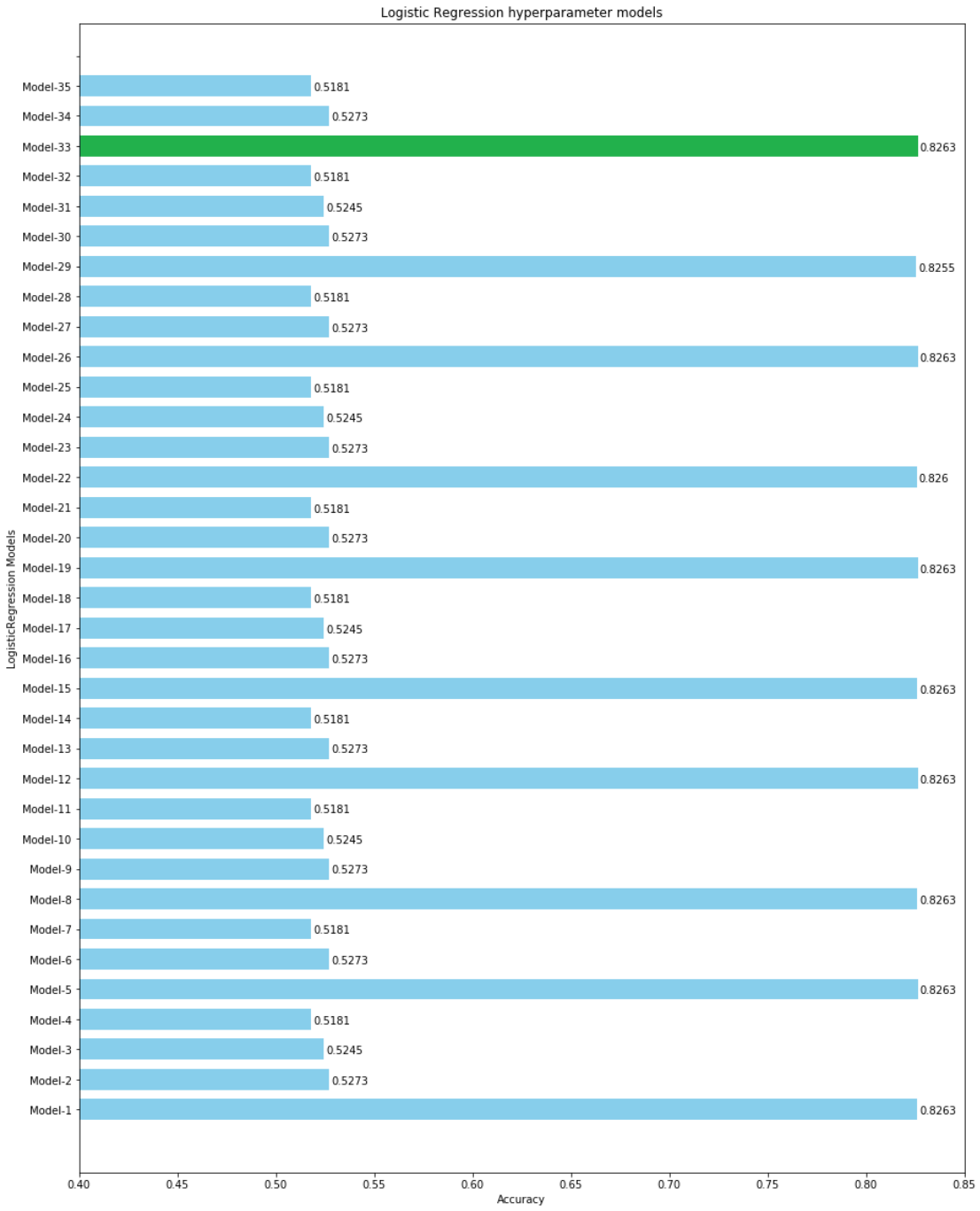


Figure 28. comparision of accuracies of Logistic regression hyperparameter models

Figure ***28*** is a bar chart plotted with Python in Jupyter notebook using matplotlib library that show mean_test_accuracy of each model created during GridSearch implementation. Besides the mean_test_accuracy several other scores are captured for all the models which are arranged in the **TABLE 3**. From the *Figure* ***28***, which is a comparison of accuracies, an evaluation metric criteria, it is found that Model-33 gave best accuracy and the hyperparameters which supported in generating this are when

Table 3. Evaluation metric scores of Logistic regression hyperparameter models.

INDEX	ACC.	StdDev	PRECISION	F1	RECALL	AUC	PARAMETERS
Model-1	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 100,'penalty': 'l2', 'solver': 'newton-cg'}
Model-2	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 100, 'penalty': 'l2', 'solver': 'lbfgs'}
Model-3	0.525	0.028	0.543	0.506	0.486	0.541	{'C': 100, 'penalty': 'l2', 'solver': 'liblinear'}
Model-4	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 100, 'penalty': 'l2', 'solver': 'saga'}
Model-5	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 100, 'penalty': 'none', 'solver': 'newton-cg'}
Model-6	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 100, 'penalty': 'none', 'solver': 'lbfgs'}
Model-7	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 100, 'penalty': 'none', 'solver': 'saga'}
Model-8	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 10, 'penalty': 'l2', 'solver': 'newton-cg'}
Model-9	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 10, 'penalty': 'l2', 'solver': 'lbfgs'}
Model-10	0.525	0.028	0.543	0.506	0.486	0.541	{'C': 10, 'penalty': 'l2', 'solver': 'liblinear'}
Model-11	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 10, 'penalty': 'l2', 'solver': 'saga'}
Model-12	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 10, 'penalty': 'none', 'solver': 'newton-cg'}
Model-13	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 10, 'penalty': 'none', 'solver': 'lbfgs'}

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Model-14	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 10, 'penalty': 'none', 'solver': 'saga'}
Model-15	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 1.0, 'penalty': 'l2', 'solver': 'newton-cg'}
Model-16	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 1.0, 'penalty': 'l2', 'solver': 'lbfgs'}
Model-17	0.525	0.028	0.543	0.506	0.486	0.541	{'C': 1.0, 'penalty': 'l2', 'solver': 'liblinear'}
Model-18	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 1.0, 'penalty': 'l2', 'solver': 'saga'}
Model-19	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 1.0, 'penalty': 'none', 'solver': 'newton-cg'}
Model-20	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 1.0, 'penalty': 'none', 'solver': 'lbfgs'}
Model-21	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 1.0, 'penalty': 'none', 'solver': 'saga'}
Model-22	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 0.1, 'penalty': 'l2', 'solver': 'newton-cg'}
Model-23	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 0.1, 'penalty': 'l2', 'solver': 'lbfgs'}
Model-24	0.525	0.028	0.543	0.506	0.486	0.541	{'C': 0.1, 'penalty': 'l2', 'solver': 'liblinear'}
Model-25	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 0.1, 'penalty': 'l2', 'solver': 'saga'}
Model-26	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 0.1, 'penalty': 'none', 'solver': 'newton-cg'}
Model-27	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 0.1, 'penalty': 'none', 'solver': 'lbfgs'}
Model-28	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 0.1, 'penalty': 'none', 'solver': 'saga'}
Model-29	0.826	0.018	0.830	0.831	0.832	0.913	{'C': 0.01, 'penalty': 'l2', 'solver': 'newton-cg'}

Model-30	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
Model-31	0.525	0.028	0.543	0.506	0.486	0.541	{'C': 0.01, 'penalty': 'l2', 'solver': 'liblinear'}
Model-32	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 0.01, 'penalty': 'l2', 'solver': 'saga'}
Model-33	0.826	0.018	0.831	0.831	0.832	0.914	{'C': 0.01, 'penalty': 'none', 'solver': 'newton-cg'}
Model-34	0.527	0.050	0.559	0.450	0.382	0.552	{'C': 0.01, 'penalty': 'none', 'solver': 'lbfgs'}
Model-35	0.518	0.026	0.552	0.420	0.340	0.542	{'C': 0.01, 'penalty': 'none', 'solver': 'saga'}

4.4.2. Decision Tree - Hyperparameter tuning

As with other classifiers, DecisionTreeClassifier takes as input two arrays: an array X, sparse or dense, of shape (number of samples, number of features) holding the training samples, and an array Y of integer values, shape (number of samples,), holding the class labels for the training samples (Decision trees, s.f.). This algorithm uses rather different parameter unlike the previous classifier mentioned. The parameters criterion, which indicates a function to calculate quality of the split, uses gini as default value. Another parameter that specifies maximum depth of the decision tree is max_depth that uses none as default. Parameter splitter demonstrates which method to choose the split at every node like random or best, which is the default. Different values are chosen for these parameters to make the search space, tabulated in **TABLE 4**, for this learning algorithm creating 36 models.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Table 4. Hyperparameter search space for Decision tree Learning algorithm

Parameter	***Search space***
criterion	gini, entropy
splitter	best, random
max-depth	5, 6, 7, 8, 9, 10

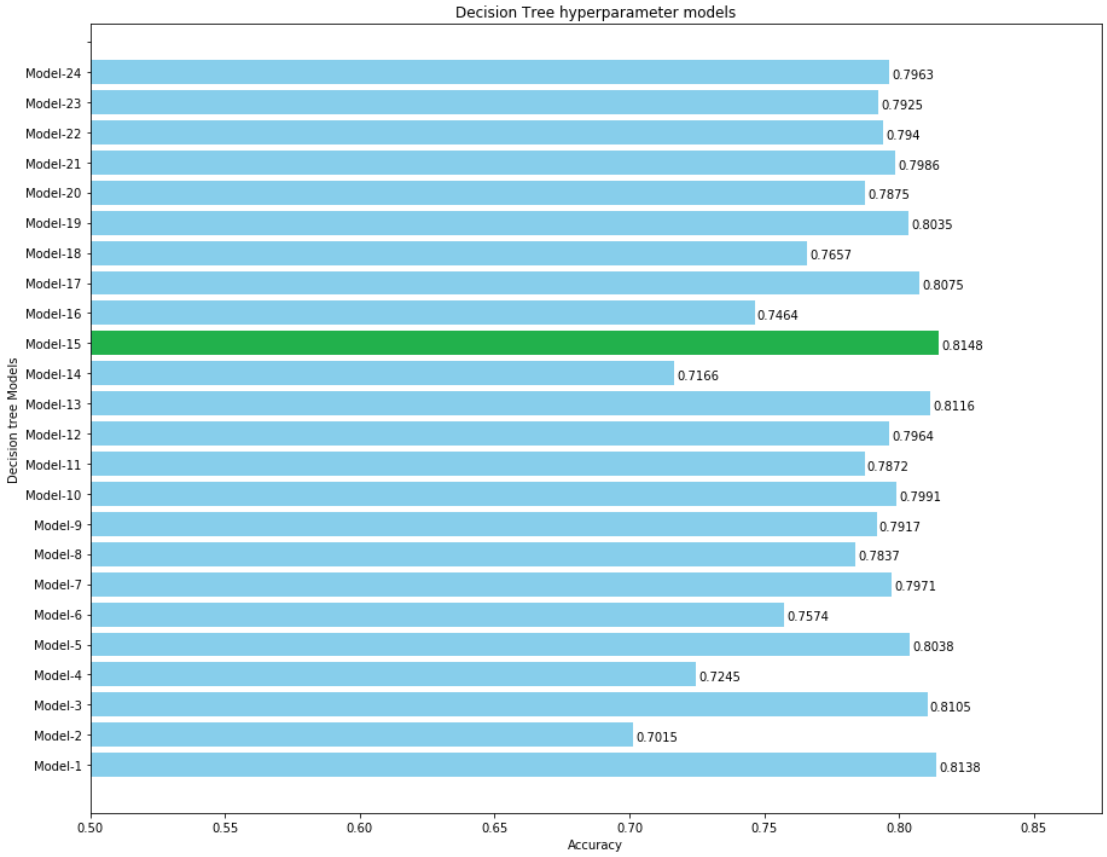


Figure 29. comparision of accuracies of Decision tree hyperparameter models

Like for LogisticRegression, *Figure **29*** is also a bar chart plotted with Python in Jupyter notebook using matplotlib library that show mean_test_accuracy of each model created during GridSearch implementation. Along with accuracy mean values of standard deviation, Precision, Recall, F1 and ROC AUC scores are also captured for all the models which are arranged in the **TABLE 5**. From the *Figure **29***, which is a comparison of accuracies, an evaluation metric criteria, it is found that Model-15 gave best accuracy and the parameters which supported in generating this are when

Table 5. Evaluation metric scores of Decision tree hyperparameter models.

INDEX	ACC.	StdDev	PRECISION	F1	RECALL	AUC	PARAMETERS
Model-1	0.814	0.019	0.827	0.817	0.809	0.881	{'criterion': 'gini', 'max_depth': 5, 'splitter': 'best'}
Model-2	0.702	0.097	0.730	0.719	0.782	0.745	{'criterion': 'gini', 'max_depth': 5, 'splitter': 'random'}
Model-3	0.810	0.018	0.822	0.814	0.808	0.858	{'criterion': 'gini', 'max_depth': 6, 'splitter': 'best'}
Model-4	0.725	0.089	0.779	0.716	0.726	0.785	{'criterion': 'gini', 'max_depth': 6, 'splitter': 'random'}
Model-5	0.804	0.017	0.818	0.807	0.798	0.832	{'criterion': 'gini', 'max_depth': 7, 'splitter': 'best'}
Model-6	0.757	0.067	0.805	0.753	0.748	0.821	{'criterion': 'gini', 'max_depth': 7, 'splitter': 'random'}
Model-7	0.797	0.020	0.808	0.802	0.797	0.813	{'criterion': 'gini', 'max_depth': 8, 'splitter': 'best'}
Model-8	0.784	0.052	0.807	0.788	0.788	0.851	{'criterion': 'gini', 'max_depth': 8, 'splitter': 'random'}
Model-9	0.792	0.023	0.802	0.796	0.793	0.801	{'criterion': 'gini', 'max_depth': 9, 'splitter': 'best'}
Model-10	0.799	0.031	0.803	0.807	0.818	0.861	{'criterion': 'gini', 'max_depth': 9, 'splitter': 'random'}

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Model-11	0.787	0.023	0.799	0.792	0.786	0.791	{'criterion':'gini', 'max_depth':10, 'splitter': 'best'}
Model-12	0.796	0.032	0.811	0.801	0.794	0.857	{'criterion': 'gini', 'max_depth': 10, 'splitter': 'random'}
Model-13	0.812	0.020	0.827	0.814	0.804	0.885	{'criterion': 'entropy', 'max_depth': 5, 'splitter': 'best'}
Model-14	0.717	0.097	0.772	0.716	0.757	0.762	{'criterion': 'entropy', 'max_depth': 5, 'splitter': 'random'}
Model-15	0.815	0.018	0.823	0.819	0.817	0.871	{'criterion': 'entropy', 'max_depth': 6, 'splitter': 'best'}
Model-16	0.746	0.076	0.773	0.755	0.788	0.805	{'criterion': 'entropy', 'max_depth': 6, 'splitter': 'random'}
Model-17	0.808	0.019	0.822	0.811	0.802	0.853	{'criterion': 'entropy', 'max_depth': 7, 'splitter': 'best'}
Model-18	0.766	0.067	0.785	0.774	0.791	0.837	{'criterion': 'entropy', 'max_depth': 7, 'splitter': 'random'}
Model-19	0.803	0.020	0.813	0.808	0.804	0.837	{'criterion': 'entropy', 'max_depth': 8, 'splitter': 'best'}
Model-20	0.787	0.046	0.820	0.787	0.772	0.855	{'criterion': 'entropy', 'max_depth': 8, 'splitter': 'random'}
Model-21	0.799	0.020	0.810	0.803	0.796	0.821	{'criterion': 'entropy', 'max_depth': 9, 'splitter': 'best'}
Model-22	0.794	0.041	0.811	0.798	0.796	0.863	{'criterion': 'entropy', 'max_depth': 9, 'splitter': 'random'}

Model-23	0.792	0.022	0.804	0.797	0.791	0.809	{'criterion': 'entropy', 'max_depth': 10, 'splitter': 'best'}
Model-24	0.796	0.023	0.805	0.802	0.805	0.861	{'criterion': 'entropy', 'max_depth': 10, 'splitter': 'random'}

4.4.3. SVC - Hyperparameter tuning

This class is implemented based on 'libsvm library. Three parameters are necessary to tune over this learning algorithm. Hence the parameters C, which indicates the penalty parameter of the error term with a default value of 1, gamma parameter that indicates coefficients of the kernel and its default value is auto, Basically the number of iteration can be with in solver, or -1 for no limit but this max_iter parameter holds a default value of -1. Different values are chosen for these parameters to make the search space, tabulated in **TABLE 6**, for this learning algorithm creating 36 models

Table 6. Hyperparameters search space for SVC learning algorithm

Parameter	***Search space***
max_iter	-1, 0, 1
gamma	1, 0.1, 0.01
C	100, 10, 1.0, 0.1

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Figure 30. comparision of accuracies of SVC hyperparameter models

*Figure **30*** is a bar chart plotted with Python in Jupyter notebook using matplotlib library that show mean_test_accuracy of each model created during GridSearch implementation. Besides the mean_test_accuracy several other scores are captured for all the models which are arranged in the **TABLE 7**. From the *Figure **30***, which is a comparison of accuracies, an evaluation metric criteria, it is found that Model-25 gave best accuracy and the hyperparameters which supported in generating this are when

Table 7. Evaluation metric scores for SVC hyperparameter models

INDEX	ACC	StdDEV	PRECISION	F1	RECALL	AUC	HYPERPARAMETERS
Model-1	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 100, 'gamma': 1, 'max_iter': -1}
Model-2	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 100, 'gamma': 1, 'max_iter': 0}
Model-3	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 100, 'gamma': 1, 'max_iter': 1}
Model-4	0.515	0.001	0.515	0.680	1.000	0.501	{'C': 100, 'gamma': 0.1, 'max_iter': -1}
Model-5	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 100, 'gamma': 0.1, 'max_iter': 0}
Model-6	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 100, 'gamma': 0.1, 'max_iter': 1}
Model-7	0.515	0.002	0.515	0.680	1.000	0.503	{'C': 100, 'gamma': 0.01, 'max_iter': -1}
Model-8	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 100, 'gamma': 0.01, 'max_iter': 0}
Model-9	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 100, 'gamma': 0.01, 'max_iter': 1}
Model-10	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 10, 'gamma': 1, 'max_iter': -1}
Model-11	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 10, 'gamma': 1, 'max_iter': 0}
Model-12	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 10, 'gamma': 1,

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

							'max_iter': 1}
Model-13	0.515	0.001	0.515	0.680	1.000	0.501	{'C': 10, 'gamma': 0.1, 'max_iter': -1}
Model-14	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 10, 'gamma': 0.1, 'max_iter': 0}
Model-15	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 10, 'gamma': 0.1, 'max_iter': 1}
Model-16	0.515	0.002	0.515	0.680	1.000	0.503	{'C': 10, 'gamma': 0.01, 'max_iter': -1}
Model-17	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 10, 'gamma': 0.01, 'max_iter': 0}
Model-18	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 10, 'gamma': 0.01, 'max_iter': 1}
Model-19	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 1.0, 'gamma': 1, 'max_iter': -1}
Model-20	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 1.0, 'gamma': 1, 'max_iter': 0}
Model-21	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 1.0, 'gamma': 1, 'max_iter': 1}
Model-22	0.515	0.001	0.515	0.680	1.000	0.501	{'C': 1.0, 'gamma': 0.1, 'max_iter': -1}
Model-23	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 1.0, 'gamma': 0.1, 'max_iter': 0}
Model-24	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 1.0, 'gamma': 0.1, 'max_iter': 1}
Model-25	0.515	0.002	0.515	0.680	1.000	0.503	{'C': 1.0, 'gamma': 0.01, 'max_iter': -1}
Model-26	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 1.0, 'gamma': 0.01, 'max_iter': 0}

Model-27	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 1.0, 'gamma': 0.01, 'max_iter': 1}
Model-28	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 0.1, 'gamma': 1, 'max_iter': -1}
Model-29	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 0.1, 'gamma': 1, 'max_iter': 0}
Model-30	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 0.1, 'gamma': 1, 'max_iter': 1}
Model-31	0.515	0.001	0.515	0.680	1.000	0.501	{'C': 0.1, 'gamma': 0.1, 'max_iter': -1}
Model-32	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 0.1, 'gamma': 0.1, 'max_iter': 0}
Model-33	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 0.1, 'gamma': 0.1, 'max_iter': 1}
Model-34	0.515	0.001	0.515	0.680	1.000	0.503	{'C': 0.1, 'gamma': 0.01, 'max_iter': -1}
Model-35	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 0.1, 'gamma': 0.01, 'max_iter': 0}
Model-36	0.515	0.001	0.515	0.680	1.000	0.500	{'C': 0.1, 'gamma': 0.01, 'max_iter': 1}

4.4.4. XGBClassifier - Hyperparameter tuning

This ensemble technique allows several parameters for the optimization. As this is a binary classification problem, only one regression tree is induced. Four parameters are necessary to be oprimized while using this classification technique. Firstly experimented with n_estimators parameter which is nothing but number of estimators

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

or trees. Another parameter used is objective which specifies the problem specific learning objective or the task of learning with this algorithm. For training models based on this algorithm, tree_method parameter is used for choosing algorithms. Gamma parameter is used to prevent the tendency of the model from overfitting. This is an unbounded parameter with a range of 0 to infinity. Different values are chosen for these 4 parameters to make the search space, as specified in **Table 8**, for this learning algorithm creating 36 models.

Table 8. Hyperparameters search space for XGBClassifier learning algorithm.

Parameter	***Search space***
objective	binary: logistic, binary: hinge
gamma	4, 10, 16
tree_method	auto, approx
n_estimators	150, 170, 190

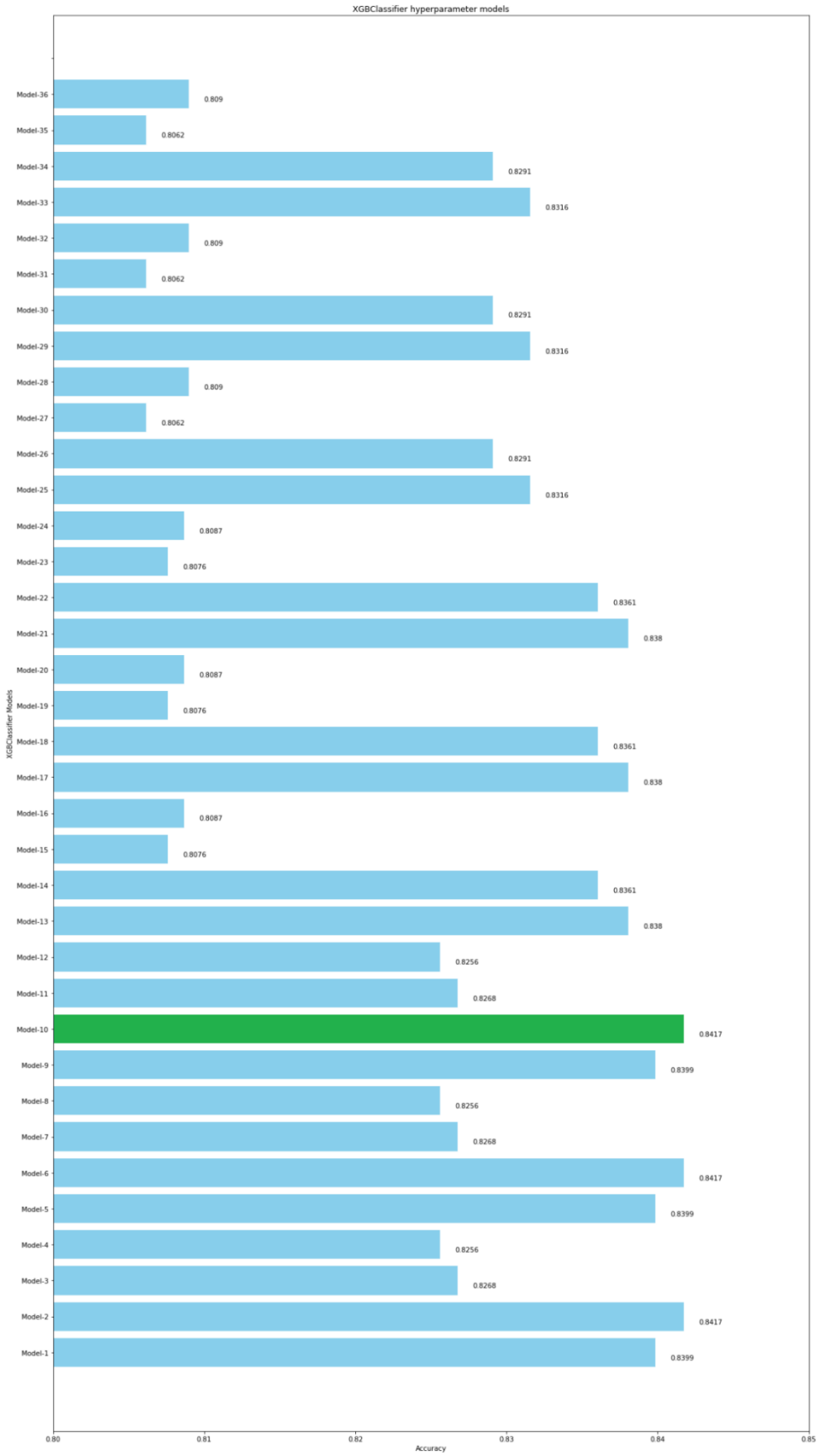


Figure 31. comparision of accuracies of XGBClassifier hyperparameter models

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Like for previously mentioned algorithms, *FIGURE 31* is depicts a bar chart plotted with Python in Jupyter notebook using matplotlib library that show mean_test_accuracy of each model created during GridSearch implementation. Along with accuracy mean values of standard deviation, Precision, Recall, F1 and ROC AUC scores are also captured for all the models which are arranged in the **TABLE 9**. From the *Figure **31***, which is a comparison of accuracies, an evaluation metric criteria, it is found that Model-10 gave best accuracy and the hyperparameters which supported in generating this are when

Table 9. Evaluation metric scores for XGBClassifier learning algorithm.

INDEX	ACC	StdDEV	PRECISION	F1	RECALL	AUC	HYPERPARAMETERS
Model-1	0.840	0.017	0.847	0.844	0.841	0.917	{'gamma': 4, 'n_estimators': 150, 'objective': 'binary:logistic', 'tree_method': 'auto'}
Model-2	0.842	0.015	0.847	0.843	0.840	0.918	{'gamma': 4, 'n_estimators': 150, 'objective': 'binary:logistic', 'tree_method': 'approx'}
Model-3	0.827	0.021	0.819	0.818	0.818	0.812	{'gamma': 4, 'n_estimators': 150, 'objective': 'binary:hinge', 'tree_method': 'auto'}
Model-4	0.826	0.020	0.820	0.817	0.816	0.812	{'gamma': 4, 'n_estimators': 150, 'objective': 'binary:hinge', 'tree_method': 'approx'}
Model-5	0.840	0.017	0.847	0.844	0.841	0.917	{'gamma': 4, 'n_estimators': 170, 'objective': 'binary:logistic',

							'tree_method': 'auto'}
Model-6	0.842	0.015	0.847	0.843	0.840	0.918	{'gamma': 4, 'n_estimators': 170, 'objective': 'binary:logistic', 'tree_method': 'approx'}
Model-7	0.827	0.022	0.820	0.817	0.816	0.812	{'gamma': 4, 'n_estimators': 170, 'objective': 'binary:hinge', 'tree_method': 'auto'}
Model-8	0.826	0.021	0.818	0.818	0.818	0.812	{'gamma': 4, 'n_estimators': 170, 'objective': 'binary:hinge', 'tree_method': 'approx'}
Model-9	0.840	0.017	0.847	0.844	0.841	0.917	{'gamma': 4, 'n_estimators': 190, 'objective': 'binary:logistic', 'tree_method': 'auto'}
Model-10	0.842	0.015	0.847	0.843	0.840	0.918	{'gamma': 4, 'n_estimators': 190, 'objective': 'binary:logistic', 'tree_method': 'approx'}
Model-11	0.827	0.020	0.820	0.818	0.817	0.812	{'gamma': 4, 'n_estimators': 190, 'objective': 'binary:hinge', 'tree_method': 'auto'}
Model-12	0.826	0.021	0.818	0.818	0.818	0.812	{'gamma': 4, 'n_estimators': 190, 'objective':

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

							'binary:hinge', 'tree_method': 'approx'}
Model-13	0.838	0.019	0.839	0.840	0.842	0.915	{'gamma': 10, 'n_estimators': 150, 'objective': 'binary:logistic', 'tree_method': 'auto'}
Model-14	0.836	0.016	0.845	0.841	0.838	0.916	{'gamma': 10, 'n_estimators': 150, 'objective': 'binary:logistic', 'tree_method': 'approx'}
Model-15	0.808	0.016	0.826	0.833	0.842	0.826	{'gamma': 10, 'n_estimators': 150, 'objective': 'binary:hinge', 'tree_method': 'auto'}
Model-16	0.809	0.015	0.832	0.838	0.845	0.832	{'gamma': 10, 'n_estimators': 150, 'objective': 'binary:hinge', 'tree_method': 'approx'}
Model-17	0.838	0.019	0.839	0.840	0.842	0.915	{'gamma': 10, 'n_estimators': 170, 'objective': 'binary:logistic', 'tree_method': 'auto'}
Model-18	0.836	0.016	0.845	0.841	0.838	0.916	{'gamma': 10, 'n_estimators': 170, 'objective': 'binary:logistic', 'tree_method': 'approx'}
Model-19	0.808	0.016	0.826	0.833	0.842	0.826	{'gamma': 10, 'n_estimators': 170, 'objective': 'binary:hinge',

							'tree_method': 'auto'}
Model-20	0.809	0.015	0.832	0.838	0.845	0.832	{'gamma': 10, 'n_estimators': 170, 'objective': 'binary:hinge', 'tree_method': 'approx'}
Model-21	0.838	0.019	0.839	0.840	0.842	0.915	{'gamma': 10, 'n_estimators': 190, 'objective': 'binary:logistic', 'tree_method': 'auto'}
Model-22	0.836	0.016	0.845	0.841	0.838	0.916	{'gamma': 10, 'n_estimators': 190, 'objective': 'binary:logistic', 'tree_method': 'approx'}
Model-23	0.808	0.015	0.825	0.833	0.841	0.826	{'gamma': 10, 'n_estimators': 190, 'objective': 'binary:hinge', 'tree_method': 'auto'}
Model-24	0.809	0.015	0.832	0.838	0.845	0.831	{'gamma': 10, 'n_estimators': 190, 'objective': 'binary:hinge', 'tree_method': 'approx'}
Model-25	0.832	0.019	0.840	0.833	0.828	0.912	{'gamma': 16, 'n_estimators': 150, 'objective': 'binary:logistic', 'tree_method': 'auto'}
Model-26	0.829	0.018	0.840	0.833	0.826	0.912	{'gamma': 16, 'n_estimators': 150, 'objective':

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

							'binary:logistic', 'tree_method': 'approx'}
Model-27	0.806	0.017	0.820	0.825	0.831	0.818	{'gamma': 16, 'n_estimators': 150, 'objective': 'binary:hinge', 'tree_method': 'auto'}
Model-28	0.809	0.020	0.824	0.832	0.841	0.825	{'gamma': 16, 'n_estimators': 150, 'objective': 'binary:hinge', 'tree_method': 'approx'}
Model-29	0.832	0.019	0.840	0.833	0.828	0.912	{'gamma': 16, 'n_estimators': 170, 'objective': 'binary:logistic', 'tree_method': 'auto'}
Model-30	0.829	0.018	0.840	0.833	0.826	0.912	{'gamma': 16, 'n_estimators': 170, 'objective': 'binary:logistic', 'tree_method': 'approx'}
Model-31	0.806	0.017	0.820	0.825	0.830	0.818	{'gamma': 16, 'n_estimators': 170, 'objective': 'binary:hinge', 'tree_method': 'auto'}
Model-32	0.809	0.020	0.824	0.832	0.841	0.825	{'gamma': 16, 'n_estimators': 170, 'objective': 'binary:hinge', 'tree_method': 'approx'}
Model-33	0.832	0.019	0.840	0.833	0.828	0.912	{'gamma': 16, 'n_estimators': 190, 'objective': 'binary:logistic',

							'tree_method': 'auto'}
Model-34	0.829	0.018	0.840	0.833	0.826	0.912	{'gamma': 16, 'n_estimators': 190, 'objective': 'binary:logistic', 'tree_method': 'approx'}
Model-35	0.806	0.017	0.819	0.825	0.831	0.818	{'gamma': 16, 'n_estimators': 190, 'objective': 'binary:hinge', 'tree_method': 'auto'}
Model-36	0.809	0.020	0.824	0.832	0.841	0.825	{'gamma': 16, 'n_estimators': 190, 'objective': 'binary:hinge', 'tree_method': 'approx'}

4.5. Results and Evaluation

This section is details about the results obtained from models of different classification algorithms which are built over pre-processed and feature extracted data and tuned parameters. The data used for this work is historical time series data. Before the data partitioning and even after splitting into train and test data, the distribution of the classes hasn't been imbalanced as seen in Figure 27. The models got trained with the data of the trading days with, data during trading days, when there was huge volatility in the economies because of severe impacting several global news in the world such Covid-19, Russia-Ukraine war, Afghanistan crisis etc. The models were trained and adopted to such dynamics in data. In total 4 different learning algorithms that resolve classification problem generated 131 models through parameter tuning with 5-fold cross validation during training of each learning algorithm according their

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

respective search space. The high number of models are due to their combination of parameters during the model building.

First of the fitting of the models is analyzed from the Calibration plot also referred to as Reliability curves. As shown in the Figure 32. Th reliability curve is appropriate for performance comparison of various predictive models also exploring which threshold value for deciding target label class is leading to model underfit or overfit. The 45 degree dotted line between the axes is a perfectly calibrated line which acts as a base and the data points below this line in any model line have underfitted and above are overfitted. While the points in SVM model line which are below the perfectly calibrated dashed line, it is underfitted. The remaining classifiers are considerable though Decision tree points are slightly away from the dashed line, Logistic regression line points and XGBoost line points made comparatively close to the dashed line. Particularly the point in the XGBoost model line are not very much deviated from this base line which tells that the point have neither much underfitted nor overfitted and always around the line of perfectly calibrated. This graph has been plotted in Jupyter with scikitplot library using python to depict the reliability curves of these models.

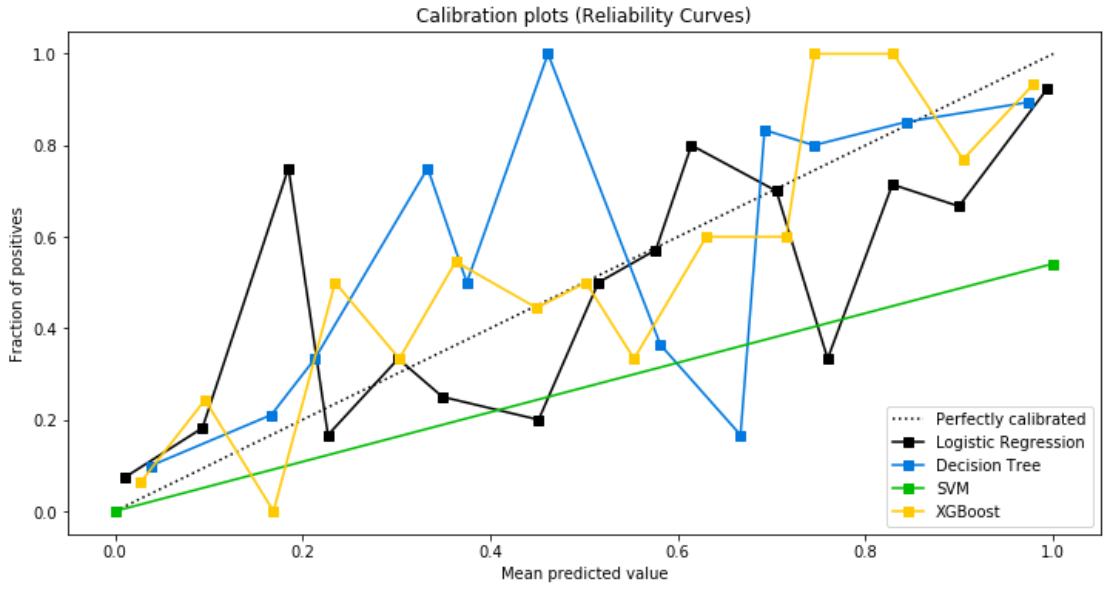


Figure 32. Calibration plots for different classification learning algorithms

Using the gridsearchcv features of hyperparameter tuning and cross-validation, for each learning algorithm several models are created. We have seen the results

produced by each model in the section 4.4 with evaluation metric scores in tables 3,5,7 and 9. In this regard, best performing model with their respective defining hyperparameters are picked. These models corresponding to the respective learning algorithms and their evaluation metric scores are tabulated and shown in Table 10.

TABLE 10. TOP PERFORMING HYPER TUNED MODELS OF EACH LEARNING ALGORITHM AND THEIR RESPECTIVE EVALUATION METRIC SCORES.

Model	Accuracy	Std. Deviation	Precision	F1	Recall	AUC
Logistic Regression	0.826	0.018	0.831	0.831	0.832	0.914
Decision Tree	0.815	0.018	0.823	0.819	0.817	0.871
SVC	0.515	0.002	0.515	0.680	1.000	0.503
XGBClassifier	**0.842**	0.015	0.847	0.843	0.840	0.918

Of all the learning algorithms, SVM based classifier has shown poor evaluation metric scores particularly accuracy which has been considered as the success criteria in the evaluation of models for this problem. It has given the lowest accuracy and precision of 51.5% for its best performing model among the models generated with parameter tuning. The area under roc curve score is also lowest recorded as 50.3%. It is already seen in Figure 32 of reliability curves, that this model is underfitting resulting in poor performance of the model which is now evident with these statistics. This model can certainly be not considered for the best model as it can clearly cannot generalize well to new data to predict the class label. The remaining three classification algorithm showed their efficiency in resolving the classification problem. Out of 24 models generated through parameter tuning based on Decision tree learning algorithm, the search space defined with {'C': 1.0, 'gamma': 0.01, 'max_iter': -1} emerged as best model with a standard deviation of 0.018, precision of 82.3% and the success criteria accuracy of 81.5%. Even the AUC score of 87.1% is also good if observed scores of all models. Though the scores displayed by best performing model of Decision tree learning algorithm, it falls behind XGBClassifier and Logistic regression algorithms generated models; as the best performing model of the latter produced 1% more

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

accuracy and 4% more AUC score i.e. 82.6% of accuracy and 91.4% of AUC score. XGBClassifier also given approximately equal AUC score. One thing which is observed equal with Decision tree best model and Logistic regression best model is the standard deviation of 0.018. However, overall considering the success criteria accuracy score and remaining evaluation metrics including 83.1% of Precision and F1 score, best model of Logistic Regression outperforms the models developed based on Decision tree.

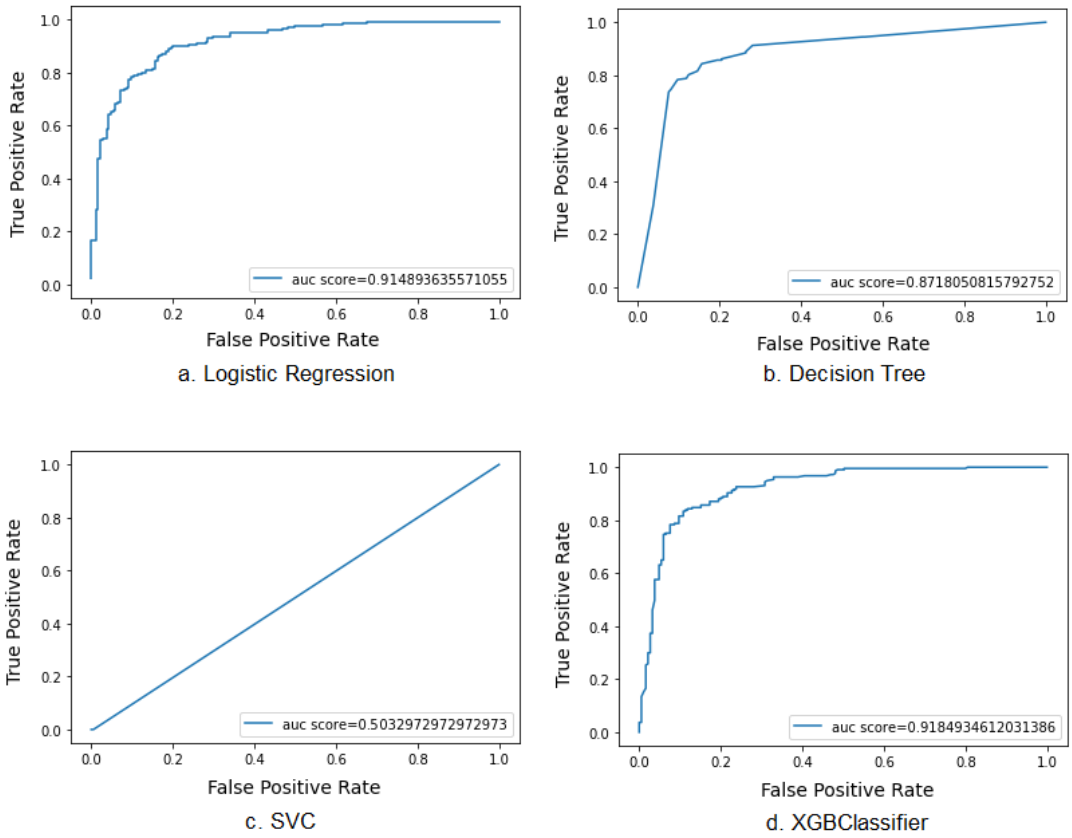


Figure 33. ROC AUC Curves of different models. a. ROC curve of Logistic Regression model. b. ROC Curve of Decision tree. c. ROC Curve of SVC model. d. ROC Curve of XGBClassifier. Source: own generation with Seaborn using python

The highest accuracy has been recorded by XGBClassifier with a value of 84.2%. The hyperparameters which defined this model are {'gamma': 4, 'n_estimators': 190, 'objective': 'binary:logistic', 'tree_method': 'approx'}. This best performing model of XGBClassifier not only given best success criteria of accuracy, it has recorded highest precision of 84.7%, which demonstrates low false-positive predictions, leading to more reliable prediction. Compared to best performing models of remaining algorithms, even the Recall score is highest with a value of 84% which results in a

substantial decrease in false-negative predictions. The standard deviation score registered by this best performing model 0.015 is also not more than that of Logistic regression and Decision Tree models. The AUC score though is approximately equal to that of Logistic Regression, score registered by XGBClassifier model is highest with a value of 91.8%, which is highest among all the models generated. This has been plotted as graph in *Figure **33*** with the AUC scores of all the best performing models of the corresponding classification algorithm .

4.5.1. Proposed Model

From the evaluation metric results of all the models seen in in the sections 4.4, it has been clear that the best performing model of XGBClassifier outperformed in terms of several evaluation metric besides the accuracy. All the evidences of the evaluation metrics cements the reliability of the best performing model of XGBClassifier in predicting the class label nothing but the direction of the stock price movement ultimately, which is being proposed with this work. It is worth looking into model performance through the principal concept of Confusion matrix.

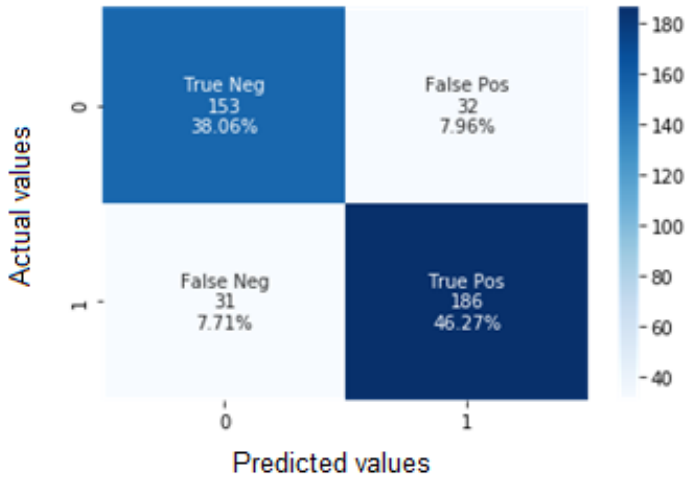


Figure 34. Confusion matrix for the best performing model of XGBClassifier algorithm.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Figure 34 demonstrates the confusion matrix, generated with seaborn library in Python, of XGBClassifier algorithm based model, that defines the relationship between predicted outcomes and actual values. As explained in section 2.2.5. confusion matrix not only just shows the statistics of the predictions made by the model, other evaluation metrics like accuracy, precision, recall etc. can also be calculated, which in this case are calculated directly from the metrics functions available in sklearn library. The XGBClassifier based model has been fit with the training data with the search space defined by the best hyperparameters already found. This trained model is inputted with the test data which was partitioned before training the model. Based on the outcome predictions, the confusion matrix, seen in Figure 34, has been generated. It can be seen the False Positives and False Negative predictions are less like 32 and 31 respectively whereas the True Positives i.e. class-1 are correctly predicted as class-1 and True Negatives i.e class-0 are correctly predicted as class-0, are also efficiently determined. These are pertaining to 186 and 153 values respectively.

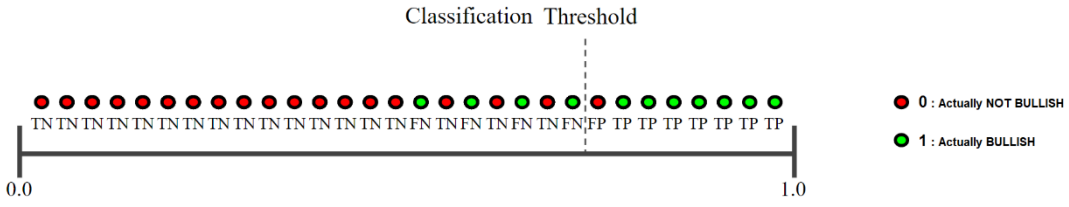


Figure 35. An example depicting the solution for type-1 error to mitigate false-positive predictions

In this classification problem, having an ultimate intention that the user wishes to reduce risk during volatile market, it is reasonable to categorize this as Type-1 error which is a False Positive conclusion. So it is better to consider the data point as '0' or bearish if the model cannot predict precisely that it is '1'. As explained in section 2.2.5.1. this Type-1 error can be addressed by decreasing the Recall value. So that False-Positives decreases, sacrificing the predictions of False-Negative. This can be achieved by increasing the threshold for classification, which is depicted in the Figure 35. And again this is basically our choice as per the requirement. As per the current study, though it is suggested to change to align with the problem statement, default threshold value has been used for this classification problem.

CHAPTER V: CONCLUSION AND FUTURE WORK

In finance sector organizations, companies, applications and workflows, data analytics and data mining are critical. These methods can be applied to obtain the knowledge that can be utilized to make effective and successful models that support in reducing risk, losing less and earn more. Stock market is more than just transactions of stocks in the finance area. It is the biggest live money game in a nation because of its unpredictive nature and volume of cash flow that happens every second and susceptible nature to global news, internal news, competitors' results and performance, political happenings, people's opinion etc. This paved a path to let it emerge as a complex system that requires lots of data and analysis of the same. As it is obvious goal of any data analysis is to make decision, Machine learning allows analysts to address the problem of predicting the trends in stock market. It is science, allowing updates and new developments and findings to emerge that improve lives of every user. While it is general wish of every human to make money, it is important to make better decisions while playing this huge money game of stock market; hence not just financial professionals, many stock market experts and analyzers, youtubers, economists, financial advisers and stock brokers are increasingly relying on data analytics. While none of these so called experts or analyzers can predict the stock market, predicting as close as possible is what the goal of everyone is. In this regard, a machine learning model can be considered reliable, based on the results, the reliability of the approach by which model was developed while withstanding the uncertainty causing trends in the data. Adapting to the bull market and bear market nature is always a concern for any market analysis particularly in machine learning model based decision making. Developing and improving the model feeding with data pertaining to abnormality in stock market trends is important. That not only makes the model robust but also trustworthy for the users.

This work aimed to scientifically improve from other researches on stock market prediction, focusing on the adaptability of model to bullish as well as bearish market nature, in predicting the direction of a company stock and to achieve improved accuracy and precision metrics.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

The dataset used for this thesis, has been derived from multiple sources for last 8 years. It is time series dataset with balanced proportion of classes to be predicted. However the distribution has been random covering multiple time instances with different market trends. There were phases with bullish and bearish trends, thus the potentiality of model will be high. Further, the global news data has been transformed through sentiment analysis and grouped with the financial data of the stock. 8 Strong features out of the further processing and transformation using Principal Component Analysis, that contribute well to the prediction of the target class, are selected. The prediction cannot be biased towards one class and the balanced distribution in the data handled that.

With the process machine readable data, following the appropriate methodology of CRISP-DM 1.0, this thesis analyzed various machine learning algorithms giving their best results with their corresponding hyperparameters. The evaluation of model's classification results while avoiding overfitting, was done using the K-fold cross validation on the input dataset. With this every machine learning algorithm has been experimented with Gridsearch that automatically iterated over the hyperparameters using cross validation. Parameter tuning has been crucial in exploring besting combination of parameters that boosted the classifiers to give best results and thus defined a best model that will resolve the classification problem.

While different algorithm based models performed differently but to their best, XGBClassifier algorithm outperformed with its results. Since this boosting algorithm learns in a sequential iteration fashion, where it inserts new trees that predicts the residuals of errors of previous trees which are later clubbed with the previous trees. So it adaptively minimizes loss while training continues iteratively. The model built over this algorithm achieved best scores of accuracy 84.2%, precision 84.7%, F1 score of 84.3%, a Recall score of 84.0% and a best AUC score of 91.8% compared to other models. Considering the evaluation metrics precision, recall, AUC along with the accuracy and XGBClassifier performing best on time series data with different trends during different phases with best results speaks about its robustness and efficiency in resolving this classification problem and ultimately the market professionals and retail investors to rely on prediction of direction of a company stock value in the market.

For future research, the mentioned concepts can be analyzed and expanded further. The scope of stock market is huge. The factors that show impact on stock market are

dynamic and numerous. Like the considered factors, financial data, global news data and company specific news data, the scope of adding other factors like internal information of a company and their new projects, they budget increment, quarterly results, political movements, competitors actions etc. Even relation between other nations stock exchange also shows impact on a nations stock exchange, like some experts of Indian Stock Exchange analyzes and predicts based on the previous day United States' stock market trend. This scope expansion would help the final machine learning model to emerge more robust and adaptive. More synthetic data covering several time periods with different market trends can be used for training the model by taking care of overfitting problem. Any new advancement in Machine learning can be experimented. Though perfect prediction of stock market is impossible, which is what the interesting aspect of this whole money game, the goal is to progress towards the best prediction of stock market as technology grows.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

APPENDIX

A.1. R Libraries and functions

TABLE 11. LIST OF USED R LIBRARIES AND FUNCTIONS

Library / Function	Description
autoplot :: function	A generic procedure creates a ggplot2 object suitable for graphing various data objects.
biplot :: function	Function to plot information of relationship between variables or distance between observations in a multidimensional dataset.
cor :: function	Calculates the correlation between two data vectors
geom_tile :: function	A graphing library function used to create visualizations like heatmap in ggplot2.
ggfortify :: library	Unified plotting tools interface to plot using ggplot2.
ggplot :: library	A comprehensive package to create visualization from the input data.
head :: function	Function that returns top rows data of a dataframe or matrix or vector or function that is passed as argument.
library :: function	Function to load the specified package.
lm :: function	Function to fit linear regression models to dataframe.
melt :: function	Function that enables to reshape, elongate dataframe in userdefined way, converting data from wide format to long format.
plot :: function	Function to create a plot based on the data passed in the form of matrix, dataframe, vectors or any object.
points :: function	To draw a group of points, at specified coordinates to an existing plot, of specified color, size and shapes.
princomp :: function	Function to perform Principal component Analysis using spectral decomposition approach. Returns many values besides eigenvectors and eigenvalues.
readxl :: library	To read data from excel file into R.
reshape2 :: library	An updated version of Reshape package to transform data between formats.
round :: function	Function to round off values to specified number of decimals.
screeplot :: function	Creates a visualization of variance against number of principal components to determine the proportions, deviation and other values.
summary :: function	The data, be it a dataframe or vector or a model, is summarized by this function and calculates values like quartiles.
text :: function	To draw the texts provided in the vector 'labels' at the x and y coordinates.

A.2. Python Libraries, Classes and functions

TABLE 12. LIST OF USED PYTHON LIBRARIES, CLASSES AND FUNCTIONS

Library / Class / Function	Description
accuracy_score :: library	computes the accuracy score for predicted labels against true labels.
append :: function	concatenates one or more items to existing list
astype :: function	function used to cast a pandas data object to a specified data type
auc :: function	computes the area under roc curve
barh :: function	used to graph the bar char on coordinates with the provided data
calibration_curve :: library	Used to evaluate how the probabilities of classes differ or how calibrated the classifier is.
classification_report :: function	To evaluate the quality of predictions from a classification algorithm and returns values like precision, recall, support and others.
confusion_matrix :: function	Creates confusion matrix for accuracy evaluation.
cross_val_score :: library	Evaluates the cross validation score.
DataFrame :: Class	Creates two dimensional tabular data.
datetime :: library	Module that provides classes for working with dates and times.
DecisionTreeClassifier :: Class	Implements decision tree classifier.
drop :: function	In the pandas dataframe, deletes specified labels from columns or rows.
enumerate :: function	Function provides a pivot or index of the current list item while iterating over list.
f1_score :: function	Calculates f-score.
fillna :: function	replaces null values in a dataframe with specified value.
fit :: function	To input the data into a learning algorithm to build a predictive model.
flatten :: function	Flattens the output.
GoogleNews :: library	API to search for articles from Google news.
GridSearchCV :: Class	Seach over all parameters space of the classifier.
groupby :: function	group the rows in dataframe by a specified value.
head :: function	To display starting rows of the dataframe.
heatmap :: function	A plot visualizing a tabular data as a color encoded matrix.
json :: library	Library to work with json data objects.
KFold :: Class	Splits the dataset into k fold for cross validation.

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

LogisticRegression :: Class	Implements logistic regression classifier.
matplotlib :: library	A data visualization, plotting library in Python.
newspaper :: library	A python library which uses webscraping algorithm to extract and curate news articles.
numpy :: library	Python library used for working on data in the form of arrays and matrices.
pandas :: library	A data manipulation and analysis library built on top of numpy.
PCA :: Class	Library to create principal components and used for dimensionality reduction.
plot_calibration_curve :: function	Used to plot the calibration curve.
plot_ks_statistic :: function	To plot the curve based on emperical distribution function KS.
precision_score :: function	Calculates the score of precision which is an evaluatin metric of predicition.
predict :: function	To predict the labels using the trained model with with certain data.
predict_proba :: function	Computes and returns the class probabilities for input data in the form of an array of lists.
re	Module that helps to work with special text strings or regular expressions.
read_csv :: function	To read the data from csv file
recall_score :: function	Calculates the score of recall which is an evaluatin metric of predicition.
RepeatedStratifiedKFold :: Class	Used to reduce the error in the model performance by repeating stratified K-fold specified number of times dynamically in each repitition.
requests :: library	Python library to make http requests and returns a response object.
roc_auc_score :: function	Calculates the area under the ROC from prediction scores
roc_curve :: function	Calculates receiver operating characteristic.
round :: function	Function to round off a number.
scikitplot :: library	A plotting library to visualize the data.
seaborn :: library	Plots statistical graphs.
SentimentIntensityAnalyzer :: library	Class implemented by VaderSentiment library to calculate sentiment intensity score to sentences.
sklearn.calibration :: library	From the inputs of classifier, graph is plotted with the predicted probabilities for each class.
sklearn.decomposition :: library	Library to decompose data vectors such as pca.
sklearn.linear_model :: Class	A class providing several functions to perform linear models in Machine learning.
sklearn.metrics :: library	Module that implements several scores to measure performance of classification.

sklearn.model_selection :: library	Library that provides functions to analyze and work on data like kfolds and splitting data.
sklearn.svm :: library	Used to implement SVM supervised learning methods.
SVC :: Class	Implemets Support vector classifier model.
textblob :: library	API to handle Natural Language Process(NLP) tasks and process textual data.
train_test_split :: function	Splits the dataset into a random train and test subsets.
transform :: function	Projects each row of the data learned by fitting, into vector space with created principal components.
vaderSentiment :: library	A Lexicon and rule based sentiment analysis tool and processes textual data
XGBClassifier :: Class	Implements extreme gradient boosting classifier.
xgboost :: library	Package used to Implements extreme gradient boosting model.

A.3 Code snippet

```
# Import libraries
from GoogleNews import GoogleNews
from newspaper import Article
import pandas as pd
from datetime import *

# Collecting data from GoogleNews
df = pd.DataFrame()
startdt = '02/17/14'
for i in range(2000):
    dt = (datetime.strptime(startdt,'%m/%d/%y')-
timedelta(days=i)).strftime("%m/%d/%Y")
    googlenews=GoogleNews(start=dt,end=dt)
    googlenews.search('RELIANCE')
    for j in range(1,2):
        googlenews.getpage(j)
        result=googlenews.result()
        if len(result)>0 :
            tempdf=pd.DataFrame({'Date':dt,'Title':[row['title'] for
row in result],'Description':[drow['desc'] for drow in result]})
            tempdf.drop_duplicates('Title',keep='first',inplace = Tru
e)
        else :
            break
```


UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

```
        if len(tempdf)>10:break
    df = df.append(tempdf.iloc[:25],ignore_index=True)


# Collecting The Gaurdian News website articles and grouping with
Yahoo data
yahoodf = pd.read_csv('/content/drive/MyDrive/RELIANCE.NS.csv')
yahoodf.dropna(inplace=True)
googledf = pd.read_excel('/content/drive/MyDrive/google.xlsx')
googledf.drop(googledf[(~googledf['Title'].str.contains("Reliance"))
& (~googledf['Unnamed: 2'].str.contains("Reliance"))].index, inplace
= True)
googledf.Date = pd.to_datetime(googledf.Date)
googledf.sort_values('Date',inplace=True)


API_ENDPOINT = 'http://content.guardianapis.com/search'
apikey='bd636a5e-6518-4e03-a4d9-16661af68960'
pivot = 0
datelist = list(set(yahoodf.Date))
lst = []
tmplist = [[], [],[], []]
holi = 0
df = pd.DataFrame(columns=['Date','Headline 1','Headline 2','Headline
 3','Headline 4','Headline 5','Headline 6','Headline 7','Headline 8',
'Headline 9','Headline 10','Headline 11','Headline 12','Headline 13',
'Headline 14','Headline 15','Headline 16','Headline 17','Headline 18'
,'Headline 19','Headline 20','Headline 21','Headline 22','Headline 23
','Headline 24','Headline 25','Open','High','Low ','Close','Adj Close
','Volume'])
for dtindex in googledf['Date'].map(lambda t: t.date()).unique():
  dt = dtindex.strftime('%Y-%m-%d')
  dt_DMY = dtindex.strftime('%d-%m-%Y')
  my_params = {
      'from-date': dt,
      'to-date': dt,
      'order-by': "newest",
      'show-fields': 'all',
      'page-size': 30,
      'lang': 'en',
      'productionOffice': 'US',
      'section': 'commentisfree|world|culture|business|politics|socie
ty',
      'api-key': apikey
  }
  resp = requests.get(API_ENDPOINT, my_params)
  data = resp.json()

  tmplist[holi]=list(googledf[googledf.Date==dt]['Title'].values)[:5]
```

```
  guardiannews = [res['fields']['headline'].replace("\n","") for res
in data['response']['results']]
  tmplist[holi].extend(guardiannews)
  tmplist[holi] = tmplist[holi][:25]
  if dt_DMY in datelist:
    headline = [dt_DMY]
    if holi == 0 :
      headline.extend(tmplist[0][:25])
    if holi == 1 :
      headline.extend(tmplist[0][:12])
      headline.extend(tmplist[1][:13])
    if holi == 2 :
      headline.extend(tmplist[0][:8])
      headline.extend(tmplist[1][:8])
      headline.extend(tmplist[2][:9])
    if holi == 3 :
      headline.extend(tmplist[0][:6])
      headline.extend(tmplist[1][:6])
      headline.extend(tmplist[2][:6])
      headline.extend(tmplist[3][:7])

    headline = headline[:26]
    headline.extend(yahoodf[yahoodf.Date==dt_DMY].values[0][1:])
    df.loc[len(df)] = headline
    pivot = pivot +1
    tmplist[0]=[]
    tmplist[1]=[]
    tmplist[2]=[]
    tmplist[3]=[]
    holi = 0

  else:
    holi = holi+1
    if holi > 3 : holi =3



# Import libraries
import pandas as pd
from textblob import TextBlob
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
import re
from datetime import datetime as dt
```


UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

```
# Performing sentiment analysis and Transforming data
df_news = df_news[df_news.columns.values[1:26]]
df = df_news.copy()
df['Para'] = df.agg('-'.join, axis=1)
df['Subjectivity'] = None
df['Objectivity'] = None
df['Positive'] = None
df['Neutral'] = None
df['Negative'] = None
for idx, row in df.iterrows():

    text = row['Para']
    blob = TextBlob(text)
    analyzer = SentimentIntensityAnalyzer()
    polarity_scores = analyzer.polarity_scores(text)

    df['Subjectivity'][idx] = (blob.sentiment.subjectivity)*100
    df['Objectivity'][idx] = (1 - blob.sentiment.subjectivity)*100
    df['Positive'][idx] = (polarity_scores['pos'])*100
    df['Neutral'][idx] = (polarity_scores['neu'])*100
    df['Negative'][idx] = (polarity_scores['neg'])*100
mask = ['Subjectivity', 'Objectivity', 'Positive', 'Neutral', 'Negati
ve']
df = df[mask]
x[mask]=df[mask]
df_news = df_news[df_news.columns.values[1:26]]
df = df_news.copy()
df['Para'] = df.agg('-'.join, axis=1)
df['Subjectivity'] = None
df['Objectivity'] = None
df['Positive'] = None
df['Neutral'] = None
df['Negative'] = None
for idx, row in df.iterrows():
    text = row['Para']
    blob = TextBlob(text)
    analyzer = SentimentIntensityAnalyzer()
    polarity_scores = analyzer.polarity_scores(text)
    df['Subjectivity'][idx] = (blob.sentiment.subjectivity)*100
    df['Objectivity'][idx] = (1 - blob.sentiment.subjectivity)*100
    df['Positive'][idx] = (polarity_scores['pos'])*100
    df['Neutral'][idx] = (polarity_scores['neu'])*100
    df['Negative'][idx] = (polarity_scores['neg'])*100
mask = ['Subjectivity', 'Objectivity', 'Positive', 'Neutral', 'Negati
ve']
findf[mask]=df[mask]


# Data labelling and consolidating with main dataframe
temp = findf.Close[0]
```

```
labels=[]
lst = findf.Close
for i,val in enumerate(lst):
  if val>lst[i+1]:
    labels.append(0)
  else :labels.append(1)
  temp = val
labels.append(0)
findf["Label"] = labels


# Import libraries
import pandas as pd
import seaborn as sns
import numpy as np
import warnings
from matplotlib import pyplot as plt
%matplotlib inline
from pandas import Series, datetime
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split, KFold, cross_va
l_score, GridSearchCV, TimeSeriesSplit
from sklearn.metrics import classification_report, confusion_matrix,
accuracy_score, mean_squared_error,precision_score,recall_score,f1_sc
ore
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
from sklearn.ensemble import AdaBoostClassifier, GradientBoostingClas
sifier, RandomForestClassifier, ExtraTreesClassifier
from sklearn.metrics import roc_curve, auc,roc_auc_score
import matplotlib.pyplot as plt
import random
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.arima_model import ARIMA
from xgboost import XGBClassifier
import seaborn as sns
from sklearn.decomposition import PCA
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.model_selection import GridSearchCV
import scikitplot as skplt
from sklearn.calibration import calibration_curve
pca = PCA(n_components=7)
```


UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

```
# Loading the dataset to a dataframe
sentence_file = "C:/Users/ravic/DATASET.xlsx"
df = pd.read_excel(sentence_file, parse_dates=[1])
grid = []
grid.append(dict(solver=['newton-
cg', 'lbfgs', 'liblinear','saga'],penalty=['l2','none'],C=[100, 10, 1
.0, 0.1, 0.01]))
grid.append(dict(criterion=['gini','entropy'],splitter=['best','rando
m'],max_depth=[5,6,7,8,9,10]))
grid.append(dict(max_iter = [-
1,0,1], gamma= [1, 0.1, 0.01],C=[100, 10, 1.0, 0.1]))
grid.append(dict(objective= ['binary:logistic','binary:hinge'],gamma
= [4,10,16],tree_method=['auto','approx'],n_estimators= [150,170,190]
))

# Change the NaN values to the mean value of that column
consideredcolumns = ['Subjectivity', 'Objectivity', 'Positive', 'Nega
tive', 'Neutral','Open','High','Low','Volume']
nan_list = ['Subjectivity', 'Objectivity', 'Positive', 'Negative', 'N
eutral']
for col in nan_list:
    df[col] = df[col].fillna(df[col].mean())

# Separate the dataframe for input(X) and output variables(y)
X = df.drop('Label',axis=1)
y = df.Label

# Split the initial 80% of the data as training set and the remaining
 20% data as the testing set
train_size = int(len(X.index) * 0.8)
X_train, X_test = X.loc[0:train_size, :], X.loc[train_size: len(X.ind
ex), :]
y_train, y_test = y[0:train_size+1], y.loc[train_size: len(X.index)]
models = []
models.append(('Logistic Regression' , LogisticRegression()))
models.append(('Decision Tree' , DecisionTreeClassifier()))
models.append(('SVM' , SVC()))
models.append(('XGBoost', XGBClassifier(n_estimators=170, max_depth=N
one,eval_metric='error')))
X = df.loc[:,'Open':'Negative']
X = X.drop(['Close'],axis=1)
pca.fit(X)
transformed = pca.transform(X)
pca_df = pd.DataFrame(transformed)
warnings.filterwarnings("ignore")
X_train_pca, X_test_pca = pca_df.loc[0:train_size, :], pca_df.loc[tra
in_size: len(X.index), :]

# Learning algorithms Hyperparameter tuning models generation
```

```
i = 0
cv = RepeatedStratifiedKFold(n_splits=5)
df = pd.DataFrame(columns = ['Model','Accuracy','StdDev','Precision',
'F1','Recall', 'ROC_AUC','Parameters'])
for name, model in models:
    row = [name]
    grid_search = GridSearchCV(refit='accuracy',estimator=model, para
m_grid=grid[i], n_jobs=-
1, cv=cv, scoring=['accuracy', 'f1', 'precision', 'recall','roc_auc']
,error_score=0)
    grid_result = grid_search.fit(X_train_pca, y_train)
    means = grid_result.cv_results_['mean_test_accuracy']
    stds = grid_result.cv_results_['std_test_accuracy']
    precisions = grid_result.cv_results_['mean_test_precision']
    f1s = grid_result.cv_results_['mean_test_f1']
    recalls = grid_result.cv_results_['mean_test_recall']
    rocs = grid_result.cv_results_['mean_test_roc_auc']
    params = grid_result.cv_results_['params']
    for mean, stdev, precision, f1, recall,roc_auc, param in zip(mean
s, stds, precisions, f1s, recalls, rocs, params):
        if mean>0:
            df.loc[len(df)]=row + [mean, stdev, precision, f1, recall
, roc_auc, param]

# Visualization of Logistic regression Hyperparameter tuning models
tempdf = ndf.groupby(['Model']).get_group('Logistic Regression')
fig, ax = plt.subplots(figsize=(15,20))
models = tempdf.Index
bin_width=1.75
positions = np.arange(1,len(models)+2,1)
plt.yticks(positions*2.5,models)
_=ax.set(title='Logistic Regression hyperparameter models',ylabel='Lo
gisticRegression Models', xlabel='Accuracy')
ax.set(xlim=[0.4,0.85])
for i,clas in enumerate(models) :
    try :
        tx_a =  tempdf.loc[tempdf.Index==clas]['Accuracy']
        bin=ax.barh(2.5*(i+1),tx_a,height=bin_width,color='skyblue')
        ax.text(bin[0].get_width()+1,2.5*(i+1), tx_a)

    except ValueError :
        pass

for i in ax.patches:
    plt.text(i.get_width()+0.001, i.get_y()+0.5,
             str(round((i.get_width()), 4)),
             fontsize = 10,
```


UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

```
            color ='black')

# Visualization of DecisionTree Hyperparameter tuning models
tempdf = ndf.groupby(['Model']).get_group('Decision Tree')
fig, ax = plt.subplots(figsize=(15,12))
models = tempdf.Index
bin_width=2
positions = np.arange(1,len(models)+2,1)
plt.yticks(positions*2.5,models)
_=ax.set(title='Decision Tree hyperparameter models',ylabel='Decision
 tree Models', xlabel='Accuracy')
ax.set(xlim=[0.50,0.875])
for i,clas in enumerate(models) :
    try :
        tx_a =  tempdf.loc[tempdf.Index==clas]['Accuracy']
        bin=ax.barh(2.5*(i+1),tx_a,height=bin_width,color='skyblue')
        ax.text(bin[0].get_width()+1,2.5*(i+1), tx_a)

    except ValueError :
        pass
for i in ax.patches:
    plt.text(i.get_width()+0.001, i.get_y()+0.5,
             str(round((i.get_width()), 4)),
             fontsize = 10,
             color ='black')

# Visualization of SVC Hyperparameter tuning models
tempdf = ndf.groupby(['Model']).get_group('SVM')
fig, ax = plt.subplots(figsize=(10,15))
models = tempdf.Index
bin_width=2
positions = np.arange(1,len(models)+2,1)
plt.yticks(positions*2.5,models)
_=ax.set(title='SVC hyperparameter models',ylabel='SVC Models', xlabe
l='Accuracy')
ax.set(xlim=[0.510,0.52])
for i,clas in enumerate(models) :
    try :
        tx_a =  tempdf.loc[tempdf.Index==clas]['Accuracy']
        bin=ax.barh(2.5*(i+1),tx_a,height=bin_width,color='skyblue')
        ax.text(bin[0].get_width()+1,2.5*(i+1), tx_a)
    except ValueError :
        pass
for i in ax.patches:
    plt.text(i.get_width()+0.0001, i.get_y()+0.5,
             str(round((i.get_width()), 4)),
             fontsize = 10,
             color ='black')

# Visualization of XGBClassifier Hyperparameter tuning models
```

```
tempdf = ndf.groupby(['Model']).get_group('XGBoost')
fig, ax = plt.subplots(figsize=(20,40))
models = tempdf.Index
bin_width=2
positions = np.arange(1,len(models)+2,1)
plt.yticks(positions*2.5,models)
_=ax.set(title='XGBClassifier hyperparameter models',ylabel='XGBClass
ifier Models', xlabel='Accuracy')
ax.set(xlim=[0.8,0.85])
for i,clas in enumerate(models) :
    try :
        tx_a =  tempdf.loc[tempdf.Index==clas]['Accuracy']
        bin=ax.barh(2.5*(i+1),tx_a,height=bin_width,color='skyblue')
        ax.text(bin[0].get_width()+1,2.5*(i+1), tx_a)

    except ValueError :
        pass
for i in ax.patches:
    plt.text(i.get_width()+0.001, i.get_y()+0.5,
             str(round((i.get_width()), 4)),
             fontsize = 10,
             color ='black')
# Generating Confusion matrix for best model
    clf = m[3]
    clf.fit(X_train_pca, y_train)
    y_pred_pca = clf.predict(X_test_pca)
    accu_score = accuracy_score(y_test, y_pred_pca)
    print("Exactitud de", name + ": %.2f%% " %(100*accu_score))

    print("\nConfusion matrix de",name,'\n')
    print(confusion_matrix(y_test, y_pred_pca))
    pca_report = classification_report(y_test, y_pred_pca)
    print("\nClassification report de ",name," \n\n" + str(pca_report
))
#Visualization of Confusion matrix
cf_matrix = confusion_matrix(y_test, y_pred_pca)
group_names = ['True Neg','False Pos','False Neg','True Pos']
group_counts = ['{0:0.0f}'.format(value) for value in
                cf_matrix.flatten()]
group_percentages = ['{0:.2%}'.format(value) for value in
                     cf_matrix.flatten()/np.sum(cf_matrix)]
labels = [f'{v1}\n{v2}\n{v3}' for v1, v2, v3 in
          zip(group_names,group_counts,group_percentages)]
labels = np.asarray(labels).reshape(2,2)

sns.heatmap(cf_matrix, annot=labels, fmt='', cmap='Blues')
```


UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

GLOSSARY

Bearish: Refers to a weak market, where the overall trend or direction of price, is moving lower. *p. 10,11,12,13,16,37,39,40,48,83,84,85*

Bullish: Refers to a strong market where the overall trend or direction of price, is moving higher. *p. 10,11,12,13,16,37,39,40,48,84,85*

Buying pressure: During a trading session, There are more buyers than there are ready sellers for a stock putting upward pressure on its price. *p. 9,37*

Capital gain: Profit from stock sale, usually over one year. *p. 17*

Close: The price of the stock at the closure of the stock exchange on a trading day. *p. 42, 49*

Day trading : The act of buying an asset with the intention of selling asset on the same day. *p. 9*

High: High level of Security. *p. 42, 49*

Institutional Investors: Organizations that trade and invest large amount of other people´s money. *p. 9*

Investor: Who think of shares of sock as the ownership interest in a company that they are. *p. 9,10,11,12,13,14,15,16,17,37,44,85*

IPO: Process where a previously unlisted company sells new or existing securities and offers them to the public for the first time (IPO Process, s.f.). *p. 13,14*

Low: Low level of Security. *p. 15,16,42,49,52*

Market sentiment: the overall attitude of participants and outsiders toward a particular security or financial market. *p. 10*

Opening: The price of the stock at the beginning of the trading day. *p. 41, 42,49,52*

Portfolio: A set of investments held by an investor. *p. 10,11,16*

Retail Investors: Individuals or common public who are investing their own money rather than giving money to an Institution to invest it for them. *p. 3,9,11,37,85*

Selling pressure: During a trading session, There are more sellers than there are ready buyer for a stock putting downward pressure on its price. *p. 9,37*

Share: Represents an ownership claim on a business being a shareholder of business´s stock. *p. 13,15,16,17,18*

Stock exchange: A platform where trading or transaction takes place between buyer and sellers of stocks of various companies. *p. 3,9,10,11,13,14,15,17,86*

Trade: Transaction of selling and buying on financial market. *p. 14,15,17,18*

Volatility: the frequency and magnitude of movements in a stock price or several listings in market, up or down. The more frequent and bigger the value fluctuations, the more volatile the market is said to be. *p. 10,11,17,37,83*

Volume: Number of stocks active with buyers and sellers in transactions during a trading session. *p. 18,37,41,42,49,55*

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

REFERENCES

Alexander, G. N., & Andrei, Z. Y. (2008). Principal graphs and manifolds. *abs/0809.0490*. doi:https://doi.org/10.48550/arXiv.0809.0490

Alexandra , G. (2015). The Efficient Market Hypothesis: Review of Specialized Literature and Empirical Research. *Procedia Economics and Finance, 32*. doi:https://doi.org/10.1016/S2212-5671(15)01416-1

Angel One. (n.d.). *Introduction to Bullish Option Strategies*. Retrieved from AngelOne: https://www.angelone.in/knowledge-center/derivatives/bullish-option-strategies

Avery, J. (2016). Just one more hand: Life in the casino economy. *Contemporary Sociology, 45*(5), 640-642. doi:https://doi.org/10.1177/0094306116664524mm

Barnett, R. M. (2017). Principal Component Analysis. *Geostatistics Lessons*. In J. L. Deutsch (Ed.). Retrieved from http://geostatisticslessons.com/lessons/principalcomponentanalysis

Bohl, L. (Ed.). (2021, August 12). *Trading*. Retrieved from schwab.com: https://www.schwab.com/learn/story/how-to-pick-stocks-using-fundamental-and-technical-analysis

Brownlee, J. (2020, August 19). *Python Machine Learning*. Retrieved from Machine learning mastery: https://machinelearningmastery.com/types-of-classification-in-machine-learning/

Campanella, F. M. (2016). Efficient Market Hypothesis and Fundamental Analysis: An Empirical Test in the European Securities Market. *Review of Economics & Finance, 6*, 27-42. Retrieved from https://EconPapers.repec.org/RePEc:bap:journl:160103

Data annotation & Data labeling. (2022). Retrieved from shaip: https://www.shaip.com/blog/the-a-to-z-of-data-annotation/

data-profiling. (n.d.). Retrieved from Datameer: https://www.datameer.com/data-profiling/

Decision trees. (n.d.). Retrieved from scikit learn: https://scikit-learn.org/stable/modules/tree.html#tree

Demat account. (n.d.). Retrieved from Angelone.in: https://www.angelone.in/knowledge-center/demat-account/what-is-demat-account

Duggal, N. (2021, November 12). *Introduction to Machine Learning - A Step by Step Guide*. Retrieved from simplilearn.com: https://www.simplilearn.com/tutorials/machine-learning-tutorial/introduction-to-machine-learning

Easley, D., M. Kiefer, N., & O'Haara, M. (1997, June). The information content of the trading process. *Journal of Empirical Finance, 4*(2), 159-186. doi:https://doi.org/10.1016/S0927-5398(97)00005-4

Editorial, S. (Ed.). (2021, September 10). *How to Select a Predictive Modeling Technique?* Retrieved from Silicon Valley Cloud IT: https://www.siliconvalleycloudit.com/how-to-select-a-predictive-modeling-technique/

Expert System Team. (2022, March 14). *Machine Learning definition*. Retrieved from experti.ai: https://expertsystem.com/machine-learning-definition/

Geier, B. (2021, September 1). *What's a Bullish Stock and When Should You Buy?* Retrieved from Yahoo Finance: https://finance.yahoo.com/news/bullish-stock-buy-230458361.html

Global stocks sink after South Africa finds new Covid variant. (2021, November 26). Retrieved from Business-Standard: https://www.business-standard.com/article/markets/global-stocks-sink-after-south-africa-finds-new-covid-variant-121112600711_1.html

Grace, W. (2006, Aug). Comment: [Support Vector Machines with Applications]. *Statistical Science, 2*(3), 347-351. Retrieved from https://www.jstor.org/stable/27645768

Hachcham, A. (2021, August 12). *XGBoost: Everything You Need to Know*. Retrieved from Neptuneblog: https://neptune.ai/blog/xgboost-everything-you-need-to-know

Hamada, D. N. (2020). Wisdom of crowds and collective decision-making in a survival situation with complex information integration. *Cognitive Research: Principles and Implications, 5*(1), 48. Retrieved from https://doi.org/10.1186/s41235-020-00248-z

Hao, K. (2018, November 17). *What is machine learning*. Retrieved from technologyreview: https://www.technologyreview.com/2018/11/17/103781/what-is-machine-learning-we-drew-you-another-flowchart/

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* New York City, USA: Springer.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology, 24*, 417-441. Retrieved from https://doi.org/10.1037/h0071325

invest in you. (2022, May 10). Retrieved from cnbc.com: https://www.cnbc.com/2022/05/10/43percent-of-investors-say-theyre-too-nervous-to-get-into-markets-right-now.html

IPO Process. (n.d.). Retrieved from corporatefinanceinstitute.com: https://corporatefinanceinstitute.com/resources/knowledge/finance/ipo-process/

Jhawar, A. (2020, May 1). *Support Vector Machine*. Retrieved from Medium: https://medium.com/analytics-vidhya/support-vector-machine-bf7dfa64b893

Kammer, A., Azour, J., Selassie, A. A., Goldfajn, I., & Rhee, C. (2022, March 15). *How War in Ukraine Is Reverberating Across World's Regions*. Retrieved from blogs.imf.org: https://blogs.imf.org/2022/03/15/how-war-in-ukraine-is-reverberating-across-worlds-regions/

Khan, K., Zhao, H., Zhang, H., Yang, H., Muhammad, H. S., & Jaganger, A. (2020, July 30). The Impact of COVID-19 Pandemic on Stock Markets: An Empirical Analysis of World Major Stock Indices. *The Journal of Asian Finance, Economics and Business, 7*(7), 463-474. doi:https://doi.org/10.13106/jafeb.2020.vol7.no7.463

Kotak securities. (n.d.). *Open a trading account*. Retrieved from Kotak Securities: https://www.kotaksecurities.com/ksweb/Help/Customer-Queries/Open-a-Trading-Account

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Krishni. (2019, Jan 5). *An Introduction to Grid Search*. Retrieved from medium: https://medium.datadriveninvestor.com/an-introduction-to-grid-search-ff57adcc0998

Lawton, G. (2022, January). *Logistic Regression*. (E. Burns, Editor) Retrieved from TechTarget: https://www.techtarget.com/searchbusinessanalytics/definition/logistic-regression

Lilly, C. (2019, Jan 7). *Support Vector Machine — Simply Explained*. Retrieved from Towards Data Science: https://towardsdatascience.com/support-vector-machine-simply-explained-fee28eba5496

Lutz, M. (2010). *Programming Python: Powerful Object-Oriented Programming.* O'Reilly Media. Retrieved from https://books.google.com.mx/books?id=q8W3WQbNWmkC

Manahov, V. H. (2014). A note on the relationship between market efficiency and adaptability – New evidence from artificial stock markets. *Expert Systems with Applications*(41), 7436-7454. Retrieved from https://doi.org/10.1016/j.eswa.2014.06.004

Marc, J. (2021, August 14). *Analysis: Taliban gains give investors cause for concern beyond Afghanistan*. Retrieved from Reuters: https://www.reuters.com/business/taliban-gains-give-investors-cause-concern-beyond-afghanistan-2021-08-14/

MasterClass staff. (2022, February 25). *The Difference Between Subjective and Objective Information*. Retrieved from MasterClass: https://www.masterclass.com/articles/subjective-vs-objective-information-explained#what-is-objective-information

McMillan, B. (2019, June 27). *Markets*. Retrieved from Forbes: https://www.forbes.com/sites/bradmcmillan/2019/06/27/consumer-confidence-drops-why-does-it-matter

Microsoft Corporation. (2019, Noviembre). *Machine Learning Algorithm Cheat Sheet.* Retrieved from https://download.microsoft.com/download/3/5/b/35bb997f-a8c7-485d-8c56-19444dafd757/azure-machine-learning-algorithm-cheat-sheet-nov2019.pdf?WT.mc_id=docs-article-lazzeri

Mihir, D. (2019). Testing the Random Walk Hypothesis in the Indian Stock Market Using ARIMA Modelling. *Journal of Applied Management and Investments, 8*(2). Retrieved from https://EconPapers.repec.org/RePEc:ods:journl:v:8:y:2019:i:2:p:71-77

Mithun, S. (2018, September 25). *CRISP-DM*. Retrieved from Think Insights: https://thinkinsights.net/digital/crisp-dm/

Mohammad, A., Md, A. I., & Rosni, B. (2021, May 03). Factors affecting share prices: A literature revisit. *AIP Conference Proceedings, 2339*(1). doi:https://doi.org/10.1063/5.0045110

Molina, L. C. (2002). *Data mining: torturando a los datos hasta que confiesen.* Retrieved March 25, 2020, from http://www.uoc.edu/molina1102/esp/art/molina1102/molina1102.html

Money Control. (n.d.). Retrieved from Money Control: https://www.moneycontrol.com/

Murphy, N., & Gebbie, T. (2019). Learning the dynamics of technical trading strategies. *Quantitative Finance, 21*, 1325-1349. doi:https://doi.org/10.1080/14697688.2020.1869292

Norman, D. (2004). *Emotional Design: Why We Love (Or Hate) Everyday Things.* Perseus Books. doi:ISBN 0-465-05135-9

Norman, D. (2005, Jul). Human-centered design considered harmful. *12*(4). Retrieved from https://doi.org/10.1145/1070960.1070976

ORACLE. (2020). *Data Mining Concepts*. Retrieved March 25, 2020, from What Is Data Mining?: https://docs.oracle.com/cd/B28359_01/datamine.111/b28129/process.htm#CHDFJEJI

Pathak, M. (2019, November 7). *Python*. Retrieved from Datacamp: https://www.datacamp.com/tutorial/xgboost-in-python

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine, 2*(11), 559–572. Retrieved from https://doi.org/10.1080/14786440109462720

Petri, P. A., & Banga, M. (2021). The economic consequences of globalisation in the United States. *Globalisation and its Economic Consequences*, 192 - 213. doi:https://doi.org/10.4324/9781003138501-8

Python Training. (2020). Retrieved from Unidata: https://unidata.github.io/python-training/python/introduction/

Ray, S. (2015, August 14). *Comprehensive guide regression*. Retrieved from Analytics Vidhya: https://www.analyticsvidhya.com/blog/2015/08/comprehensive-guide-regression

Reliance Industries. (n.d.). Retrieved from Moneycontrol: https://www.moneycontrol.com/india/stockpricequote/refineries/relianceindustries/RI

Reliance Industries. (n.d.). Retrieved from Yahoo Finance : https://finance.yahoo.com/quote/RELIANCE.NS

Rooij, M. V., Lusardi, A., & Alessie, R. (2011, August). Financial literacy and stock market participation. *Journal of Financial Economics, 101*(2), 449-472. doi:https://doi.org/10.1016/j.jfineco.2011.03.006

Ross, I., & Robert, G. (1996). R: A Language for Data Analysis and Graphics. *Journal of Computational and Graphical Statistics, 5*. Retrieved from http://www.jstor.org/stable/1390807

scikit-learn developers. (n.d.). *Getting started*. Retrieved from Scikit learn: https://scikit-learn.org/stable/getting_started.html

scikit-learn developers. (n.d.). *logistic regression*. Retrieved from scikit learn: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Sebastian, R. (2018, Dec 03). Model Evaluation, Model Selection, and Algorithm Selection in Machine. *CoRR, abs/1811.12808*. Retrieved from http://arxiv.org/abs/1811.12808

Shonkwiler, R. W. (2013). Finance with Monte Carlo. *New York: Springer*. Retrieved from https://doi.org/10.1007/978-1-4614-8511-7

Subramanian, B., Paul, A., Kim, J., & Srinivasan, M. K. (2021). Predictive Modeling and Mobility Pattern Analysis. 1-4. doi:10.1109/ICOT54518.2021.9680660

UPAEP
POSGRADOS
17 SUR 901
Barrio de Santiago
Puebla, Pue. México
C.P. 72410

Tel.: 01 (222) 229 9400
Fax: 01 (222) 2325251
01 800 224 2200
www.upaep.mx

Subramanian, D. (2019, July 7). *Decision Tree in Layman's Terms*. Retrieved from towardsdatascience: https://towardsdatascience.com/decision-tree-in-laymans-terms-part-1-76e1f1a6b672

The Guardian International. (n.d.). Retrieved from The Guardian: https://www.theguardian.com/international

Trevor, H., Robert, T., & Jerome, F. (2001). *The Elements of Statistical Learning: Data Mining, Inference and Prediction.* Springer. Retrieved from https://link.springer.com/book/10.1007/978-0-387-84858-7

Types of Stock trading. (2019, March 26). Retrieved from Kotak Securities®: https://www.kotaksecurities.com/ksweb/articles/types-of-stock-trading

Urquhart, A. H. (2013). Efficient or adaptive markets? Evidence from major stock markets using very long run historic data. *International Review of Financial Analysis, 28*, 130-142. Retrieved from https://doi.org/10.1016/j.irfa.2013.03.005

Wadmare, K. (2021, May 19). *Logistic Regression in Machine Learning*. Retrieved from Medium: https://medium.com/analytics-vidhya/logistic-regression-in-machine-learning-f3a90c13bb41

Wamika, J. (2021, February 19). *Implementation of Principal Component Analysis(PCA) in K Means Clustering*. Retrieved from medium: https://medium.com/analytics-vidhya/implementation-of-principal-component-analysis-pca-in-k-means-clustering-b4bc0aa79cb6

What are Multibagger Stocks. (2022). Retrieved from 5paisa: https://www.5paisa.com/stock-market-guide/stock-share-market/what-are-multibagger-stocks

Yahoo finance. (n.d.). Retrieved from Yahoo finance: https://finance.yahoo.com/