Department of Informatics,

University of Leicester

CO4015 Computer Science Project

Dissertation

for

Stock Market Prediction using python

Author: Osato Osagie (1690 11647)

# Introduction

Artificial Intelligence (AI) is the ability of machines to behave like humans and learn autonomously. For instance, a machine might display learning and problem-solving abilities without the use of hard-coded software containing detailed instructions [1].

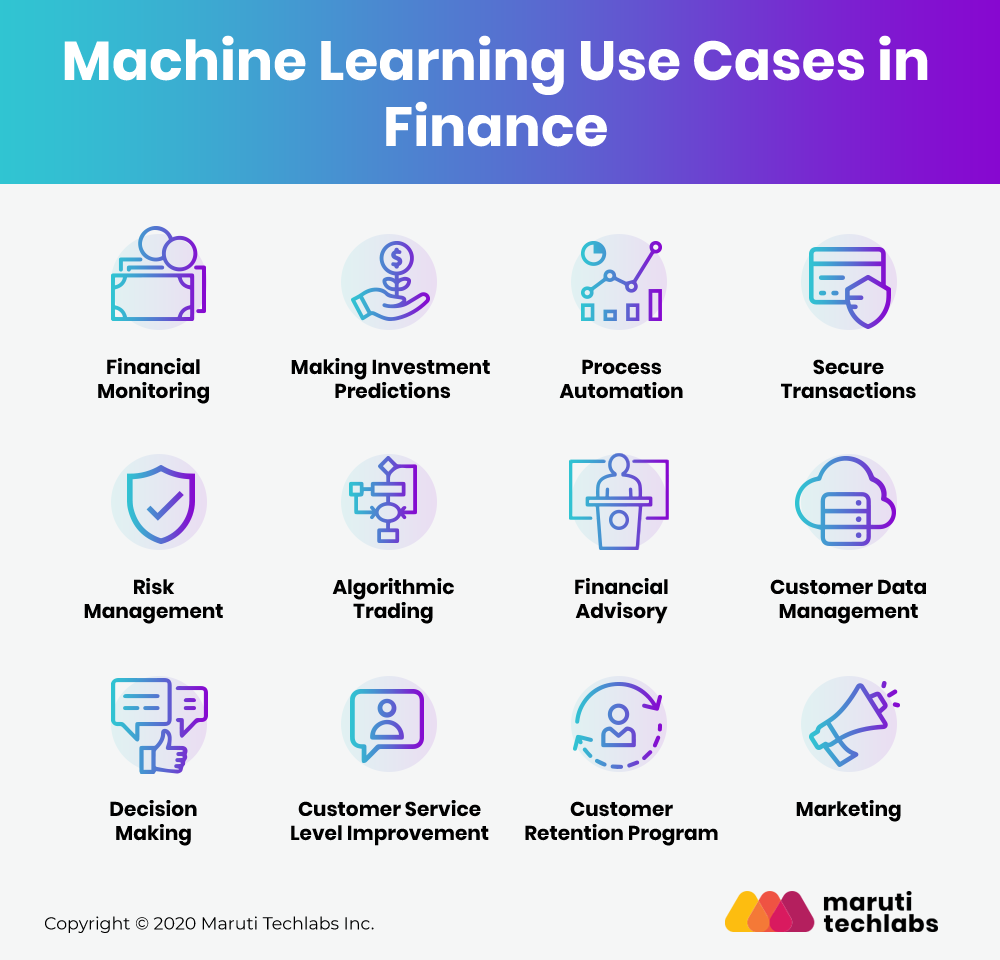


Figure 1- An image showing the main use cases of Machine Learning in Finance [2]

Machine Learning (ML) is a subfield of AI, that enables machines to learn from historic data or experiences without being explicitly programmed. Figure 1 shows the different use cases for Machine Learning in finance. The project I will be building focuses on using ML to make investment predictions. Using ML to make investment predictions is advantageous as it can lead to better predictions of stock prices, fewer errors, and greater efficiency for the investor. To do this, ML algorithms extract key insights from the dataset, learns from it then apply several techniques to accurately predict the result.

However, it is important to take into consideration the other factors that might affect the price of a company’s stock. The stock market is very volatile, thus meaning no system can accurately predict it.

## 1.1 Aims

Predicting markets has become an increasing priority for investors. The primary goal of an investor is to buy a stock when its value is low and sell when the value of the stock is high. However, this can be daunting for financial investors as they are unaware of the stocks that will return maximum profits. Using Machine learning to predict the long-term value of a stock makes this process somewhat easier. This project aims to predict the stock market price of a company using supervised machine learning algorithms.

## 

## 1.2 Objectives

The models will be implemented in python using the Alpha Vantage API for real-time and historical equity data. The models built in this project need to be able to predict the price of a company’s stock by using historical data. Also, the project must be able to produce visualizations that illustrate patterns in price movement.

In this proposed project, I will be implementing two prediction systems; one based on Random Forest and the latter using Support Vector Machines (SVM). Doing this will enable me to compare the forecasting ability of both systems. Each will be evaluated and tested with the same data to determine their prediction accuracy.

There have been several other approaches to this problem. For example, Lei in [3] exploited the Wavelet Neural Network (WNN) to predict stock price trends. However, Neural networks (NN) have several drawbacks. One drawback of using NN is that it has a slow conversion rate, thus meaning that it can be time-consuming to train.

NN also uses an optimization algorithm for finding local optima called Gradient descent. However, this often gets suck in the local maxima, thus making it a challenge to find global minima and maxima.

Another approach to this problem was Piramuthu’s [4]. A study was conducted which thoroughly evaluated the different methods of feature selection for data mining applications. However, this study can be criticized because it only used decision trees. This, therefore, makes it impossible to trust the reliability of the results, because we are unsure if we can replicate the result of this model on a more complex model.

My approach to this problem will be different as it uses a combination of SVM and Random Forest.

Decision Trees offer several benefits over using Neural Networks. For example, they perform faster than Neural Networks after training and provide a visual representation of the data. SVM also offers its benefits over using NN. Unlike NN it is guaranteed to find a global optimum and less memory is required to store the predictive model. By comparing the results of both models, I will also be able to increase the validity of my results.

# Survey of Literature/Information Sources

To get better results I have decided to use Technical Indicators. Technical indicators are mathematical calculations based on the price, volume, or open interest of a security [5]. Whilst these indicators are designed to analyze short-term price movements, they are also useful to long-term investors who want to identify entry and exit points.

According to Investopedia [5], the following are some of the most commonly used technical indicators used by traders:

1. **Relative Strength Index (RSI)-** This indicator measures the size of recent price changes, which allows us to evaluate overbought or oversold conditions in a stock’s price. RSI will be used to obtain signals about bullish and bearish price momentum. An asset is thought to be overbought when the RSI is about 70% and oversold when it is below 30%. RSI can be calculated using the following formula

Diagram

Description automatically generated with medium confidence

Figure - RSI formula.

1. **Exponential Moving Average (EMA)-** this technical indicator is similar to the SMA; however, it places more emphasis on the most recent data points, which means it will react more than a simple moving average to recent price changes. The formula for calculating EMA is shown in figure 3, where smoothing usually equals 2.

Diagram

Description automatically generated

Figure - EMA formula.

1. **Moving Average Convergence Divergence (MACD)-** MACD describes the relationship between the two aforementioned moving averages, it helps the investors to understand whether the price movement is strengthening or weakening.

The periods for the MACD indicator are presented as 26 and 12. The MACD is calculated by taking away the 26-day EMA from the 12-day EMA. a ‘signal line’ represents a 9-day EMA of the MACD. It acts as a trigger for buy and sell signals and is plotted on top of the MACD. The formula for MACD is shown in figure 4.

Text, logo

Description automatically generated

Figure MACD formula.

1. **Commodity Channel Index (CCI)**- CCI measures the change between the present price and the historical average price. CCI above zero illustrates that the present price is higher than the historical average price. It is referred to as a ‘unbound oscillator’, which means that it can fluctuate indefinitely. Therefore, we can determine overbought and oversold levels for each individual asset by taking into account it’s historical extreme CCI levels from which the price reversed. CCI can be calculated using the formula shown in figure 5.

Text

Description automatically generated

Figure 5 CCI formula.

1. **Price Rate of Change Indicator (ROC)**- ROC is an unbounded momentum indicator. This means it can fluctuate indefinitely. When a ROC is rising above zero, this confirms an uptrend and vice versa. When a price of an asset is consolidating it hoovers around zero. This indicator provides little insight except for confirming the consolidation. ROC is calculated using the formula shown in figure 6.

Graphical user interface, text, application

Description automatically generated

Figure ROC formula.

1. **Average Directional Index (ADX)**- ADX is used by Technical Analysts to make a judgement on the strength of a trend. It uses a positive (+DI) and negative (-D1) directional indicator as well as using the trendline. When the trend is considered strong when the ADX is above 25. A trend is weak when the ADX is below 20 or the price is trendless. If a price doesn’t have a trend, it doesn’t mean that the price isn’t moving. One explanation for this is that the trend change is too volatile for a clear direction to be present. ADX can be calculated using the formula shown in figure 7.

Table

Description automatically generated

Figure ADX formula

1. **Collateralized Mortgage Obligation (CMO)-** CMO are investment debt securities made up of bundled mortgages ordered according to how risky they are and sold as one investment.

## 2.1 Programming Language

For statistical computing, the two widely used programming languages used in the industry are Python and R. Between both, I have chosen to use python because it is faster and scalable. It is also one of the most supported languages nowadays, supported by many robust libraries which are constantly growing. Also, python has a diverse pack of visualization options available which makes it ideal for creating graphs and charts. To learn python and understand machine learning, I have taken a course on Udemy [6].

## 2.2 Jupyter Notebook

The programming environment I am using for this project is Jupyter notebook [7] is the programming environment I will be using in this project. It is a web application that allows users to create documents containing live code and visualizations.

To access Jupyter Notebook, I use the Jupyter Notebook App. This is a server-client application that allows editing and running of a notebook document through a web browser. This app has a dashboard that emulates the features of a file manager, for example, a user can rename and delete files. Also, it has a notebook kernel that acts as the computer engine. It executes the code stored in the notebook documents.

## 2.3 Alpha Vantage Stock API

Alpha Vantage Stock API [8] provides free real-time stock quotes, historical data, cryptocurrencies, technical indicators, FX rates, and more. This is the source I will be relying on to get my data to build the models. I will be collecting the last 2 years of a company’s stock data.

# Requirements

For this project I have split my requirements into two types:

1. Functional requirements- these types of requirements define the different components of the system.
2. Non- functional requirements- these requirements specify the quality attribute of the system. For example, how accurately a model predicts the price of a stock.

## 3.1 Functional Requirements

* The system must be able to Obtain 2 years of a company’s stock data from the Alpha Vantage Stock API.
* The system must be able to create functions that calculate the value of the aforementioned technical indicators.
* Split data into two parts, training and testing. 75% of the data will be allocated for training and 25% of the data will be used for testing.
* The system must be able to train the model using the selected training data
* The system must be able to test the model using testing data.
* The system must be able to create visualization of the market index.
* The system must be able to build a stock prediction model using SVM.
* The system must be able to build a stock prediction model using decision trees**.**

## 3.2 Non-functional Requirements

* **Accuracy & performance** 
  + The system must be correct when compared to reality.
  + The system must maintain the same performance on multiple datasets.
* **Fairness**
  + Sensitive features (e.g., race, gender) must be removed from the dataset to ensure fair results**.**
* **Transparency**
  + The steps taken to obtain results must be clearly outlined.