RV Educational Institutions ®
RV College of Engineering ®
Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi
Approved by AICTE, New Delhi

Go, change the world®

"Stock movement prediction using Machine Learning "

PROJECT REPORT on
ARTIFICIAL INTELLIGENCE & MACHINE LEARNING
Lab(18CS62)
VI SEMESTER

2021-2022

Submitted by

<Khetan Rishabh Purushotam- 1RV19CS071>
<Mohamed Moin Irfan- 1RV19CS089>

Under the Guidance of

Prof. Revathi S A, Assistant Professor
Department of CSE, RVCE
Bengaluru-560059.

RV COLLEGE OF ENGINEERING®, BENGALURU - 560059
(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the Lab Project report work titled "**Stock movement prediction using Machine Learning**" has been carried out by <**Khetan Rishabh Purushotam(1RV9CS0710 and Mohamed Moin Irfan(1RV19CS089)>,** bonafide students of RV College of Engineering, Bengaluru, have submitted in partial fulfillment for the **Assessment of Course: Artificial Intelligence & Machine Learning (18CS62) – Lab Component** during the year 2021-2022. It is certified that all corrections/suggestions indicated for the internal assessment have been incorporated in the report.

Faculty Incharge
Department of CSE,
RVCE., Bengaluru –59

Head of Department
Department of CSE,
RVCE, Bengaluru–59

RV COLLEGE OF ENGINEERING, BENGALURU® - 560059
(Autonomous Institution Affiliated to VTU)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, **<Khetan Rishabh Purushotam(1RV9CS071) and Mohamed Moin Irfan(1RV19CS089) >** the students of 6th Semester B.E., Department of Computer Science and Engineering, R.V. College of Engineering, Bengaluru hereby declare that the Lab -project titled **"Stock movement prediction using Machine Learning"** has been carried out by us and submitted in partial fulfillment for the **Assessment of Course: Artificial Intelligence & Machine Learning (18CS62) lab component** during the year 2021-2022.

Place: Bengaluru **Signature**

Date:

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped me in carrying out this project work. I would like to take this opportunity to thank them all.

I deeply express my sincere gratitude to my guide **Internal Guide, Designation,** Department of CSE, RVCE, Bengaluru, for his able guidance, regular source of encouragement and assistance throughout this project

I would like to thank **Dr.Ramakanth Kumar P**, Head of Department, Computer Science & Engineering, R.V.C.E, Bengaluru, for his valuable suggestions and expert advice.

First and foremost I would like to thank **Dr. Subramanya K N**, Principal, R.V.C.E, Bengaluru, for his moral support towards completing my project work.

I thank my Parents, and all the Faculty members of the Department of Computer Science & Engineering for their constant support and encouragement.

Last, but not the least, I would like to thank my peers and friends who provided me with valuable suggestions to improve my project.

ABSTRACT

The stock market is a dynamic and volatile platform which provides an environment and opportunity for the traders to invest and trade in stocks of particular companies. The price of a stock is dependent on numerous static and dynamic features. Predicting the trend in future price movement of a particular company's stock can be extremely beneficial for investors and traders. In this project, we predict the direction in which a stock will be moving by studying the previous trends and using the concept of false positives and training them using the Random Forest classifier. We have seen a 71% accuracy in our predictions using the given model. The dataset will be imported through kaggle easily. For training the model would just need prices of the required stocks over certain periods of time. The parameters that the data will be trained on are OHLC (open, high, low, close) and some additional parameters(indicators) to achieve even higher accuracy that will comprise the innovative component of the proposed system. We will be using a Random Forest Classification algorithm as the dataset that we train is completely discrete and we will be using several indicators to calculate the data on which the training will be performed.

This project focuses on machine learning techniques such as random forest classifier and Decision Tree. The result of implementation shows almost 71% accuracy using random forest classifier The implementation is based on JP MORGAN, IBM, COSTCO, ARROWHEAD PHARMACEUTICALS and HOME DEPOT. The reason for choosing these company stocks is because they belong to all different industries.

This model can be used to predict the future movements in the prices of stocks. This will help us in gaining an edge over the traditional investors and can lead to monetary advantages.

Table of Contents

3. Design Details

3.1 Architecture

3.2 Methodology

3.3 Data set details

3.4 ML/DL techniques used

3..5 Hardware and Software requirements

4. Implementation details of the Project

4.1 Language/Tools/APIs used

4.2 Data preprocessing

4.3 Validation methodology

5. Results and Analysis

(Detailed discussion on results obtained along with interpretations, evaluation metrics used, performance analysis with similar works)

6. Conclusion and Future Enhancement

6.1. **Novelty in the proposed solution**

6.2 Limitations of the Project

6.3. Future Enhancements

6.4. Summary

7. References in IEEE format

Appendices

Appendix A: Screenshots

Appendix B: Printout of the base paper used for implementation of this project

1) Introduction

1.1 Project Domain and Problem addressing

The financial market is a dynamic and composite system where people can buy and sell currencies, stocks, equities and derivatives over virtual platforms supported by brokers. Stock markets are affected by many factors causing the uncertainty and high volatility in the market. Although humans can take orders and submit them to the market, automated trading systems (ATS) that are operated by the implementation of computer programs can perform better and with higher momentum in submitting orders than any human. Since most of the dealings in the markets are done by automated systems, it has now been well established that training the past data can help us in finding patterns in the movement of the markets which can be used to predict the future prices. If implemented successfully with a higher accuracy than existing systems, it could turn into a financial support system with minimal amount of risk.

1.2 Issues and Challenges

With the amount of volumes involved in the market nowadays, it has become increasingly difficult to predict stock movement. But using AI and ML systems, we can forecast assumptions based on previous patterns if we use the right agents to train our program.

1.3 Need for AI based solutions

AI has a huge potential in the prediction of stock prices. Taking the past performance and behavior of any stock and training the data available using neural networks and machine learning models can help in understanding how a stock might behave in the future. Industrially talking, the system would have huge relevance. It can be used by traders to gain an edge over others and can also be used by financial institutions for quant-vol trading.

1.4 Problem Statement

With the innovation in technology and their application in the stock market, the system has become increasingly complex and volatile which in turn has made human predictions highly inaccurate, but using Machine Learning to find out the patterns in the system using historical data can help us predict the future prices more accurately.

1.5 Project Objectives

We want to maximize our true positives - days when the algorithm predicts that the price will go up, and it actually goes up. Therefore, we'll be using precision as our error metric for our algorithm, which is true positives / (false positives + true positives). This will ensure that we minimize how much money we lose with false positives (days when we buy the stock, but the price actually goes down). This means that we will have to accept a lot of false negatives - days when we predict that the price will go down, but it actually goes up. This is okay, since we'd rather minimize our potential losses than maximize our potential gains.

1.6 Summary

It is now evidently clear that AI and ML can have huge significance in topics of prediction and using these systems in financial markets can be a huge bonus if applied correctly and carefully. AI systems can predict the movements using knowledge of complex mathematical functions on the basis of which the stocks move and by training them could be able to predict how it would move ahead.

2) Literature Survey

2.1 Introduction:

The work on the use of artificial intelligence and especially machine learning to predict the prices of any type of equity and commodity has been going on since a long time. With the increase in the technological developments in the field of Machine learning, it has started becoming clearer that historical patterns can be used in multiple ways to predict what can happen in the future relating to the prices of any type of equity or commodity. With this development, people have started creating more novel models to predict the movements in prices more accurately. Since these markets are a huge arena for making financial profits, all the giant financial institutions started conducting even more research in this field to gain an economic advantage over their competitors and this forced the work on such models to full force.

2.2 Related Work:

SL. NO	Publications	IMPLEMENTATIONS	CONS
1.	Saurav Agrawal, Dev Thakkar, Dhruvil Soni, Krunal Bhimani, Dr. Chirag Patel, “Stock Market Prediction using Machine Learning Techniques”.	Artificial neural network with backpropagation algorithm	Neither growth nor pruning methods were attempted for the selection of network architecture.

2.	K. Hiba Sadia, Aditya Sharma, Adarsh Paul, SarmisthaPadhi, Saurav Sanyal, “Stock Market Prediction Using Machine Learning Algorithms”.	Random forest Algorithms, support vector machine	Previous years dataset is considered. No real-time data are used for predicting stocks.
3.	Murtaza Roondiwala, Harshal Patel, Shraddha Varma, “Predicting Stock Prices Using LSTM”. International Journal of Science and Research 2017. ISSN: 2319-7064	Root Mean Square Error (RMSE), the difference between the target value and the obtained output value is reduced by using RMSE value. Recurrent Neural Network, Long Short-Term Memory	Doesn't focus on events in the environment, like news or social media. It exploits only one data source, thus it is highly biased.
4.	S Abdulsalam Sulaiman Olaniyi, Adewole, Kayode S, Jimoh, R. G, “Stock Trend Prediction Using Regression Analysis – A Data Mining Approach”. ARPN Journal of Systems and Software, Volume 1,	Linear regression, moving average	Used for limited company stocks More amount of data is not considered for prediction

	Issue 4, 2011. ISSN: 2222-9833		

5.	Gareja Pradip, Chitrak Bari, J. Shiva Nandhini, “Stock market prediction using machine learning”.	Artificial neural network, multiple linear regression, Bayesian Algorithm	using Bayes theorem bias is found. Predicted price is fluctuating they are not constant
6.	Vivek Kanade, Bhausaheb Devikar, Sayali Phadatare, Pranali Munde, “Stock market prediction: Using historic data analysis”. International journal of advanced research in computer science and software engineering, volume 7, issue 1, 2017. ISSN: 2277 128X. DOI: 10.23956/ijarcsse/V711/0112.	SVM, ANN SVM (Support vector Machine)	Only sentiment data are used from various news and Twitter resources no historical data are considered for predictions.

7.	Analyzing the Trend of Stock Marketand Evaluate the performance of Market Prediction using Machine Learning Approach by Mathanprasad L and Gunasekaran M	Random Forest, Support Vector Machine and Neural Network	This computational approach solves the long outstanding stock market problems and find out the solutions in stock market prediction research.
8.	Stock Market Analysis using Supervised Machine Learning by Kunal Pahwa and Neha Agarwal	Support vector Machine	This paper is limited to only supervised machine learning, and tries to explain only the fundamentals of this complex process.
9.	Stock Market Prediction based on Social Sentiments using Machine Learning by Tejas Mankar , Tushar Hotchandani , Manish Madhwani , Akshay Chidrawar, Lifna C.	Sentimental analysis in the market	Large amount of storage required for analyzing such sentiments in real time.

10.	Stock Market Prediction Using Machine Learning Techniques by Mehak Usmani, Syed Hasan Adil, Kamran Raza and Syed Saad Azhar Ali	The old statistical techniques including Simple Moving Average (SMA) and Autoregressive Integrated Moving Average (ARIMA) are also used as input. The machine learning techniques including Single Layer Perceptron (SLP), Multi-Layer Perceptron (MLP), Radial Basis Function (RBF) and Support Vector Machine (SVM) are compared.	The Multi-Layer Perceptron algorithm of machine learning predicted 57% correct market performance.

Table 1: related works

2.3 summary

The existing system on stock price prediction consists of basic LSTM models and recurrent neural networks. ANNs use adaptive weights to forecast stock prices. Y. Bing proposed an ANN to predict the index of the Shanghai Stock Exchange. The authors studied the market between March 17, 2010 to April 28, 2010. They considered 5 features of the market, open, high, close, low and volume. The neural network constructed was successful in predicting the daily lowest, highest, and closing value of the Shanghai Stock Exchange. M. Jia proposed a framework which made use of the bidirectional long-short term memory (BLSTM) neural network for predicting the future price of a stock. The authors used the historical data of the GREE stock. They collected data for 568 days from January 1, 2017 to May 14, 2019. The data consisted of 14 features such as open, high, close, volume etc. The data was normalized and pre-processed.The close value was used as the benchmark for the prediction. K. A. Althelaya proposed a Bidirectional LSTM for Short- and Long-Term Stock Market Prediction. The authors had made use of the Standard and Poor 500 Index (S&P500) historical data for their proposed work.

3) Design Details:

3.1 Architecture

Data is imported to the system using python command. Once the data is obtained, it has to be cleaned or preprocessed before feeding it to the system to remove any unwanted spikes which could jitter the results of the prediction. Once the data has been processed, we calculate the indicators on the basis of which we are going to design our system and train our model. Once all the requirements have been set we train and test the data by splitting them. Upon testing we can plot the predicted information through various mathematical tools for better understanding of the results.

3.2 Methodology

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

3.3 Dataset Details

Dataset is imported as csv files to the system. It is basically the [open close high low] of particular stocks over a period of 10 years or more. The file can be downloaded through official exchange websites or websites like yahoo finance and kaggle.

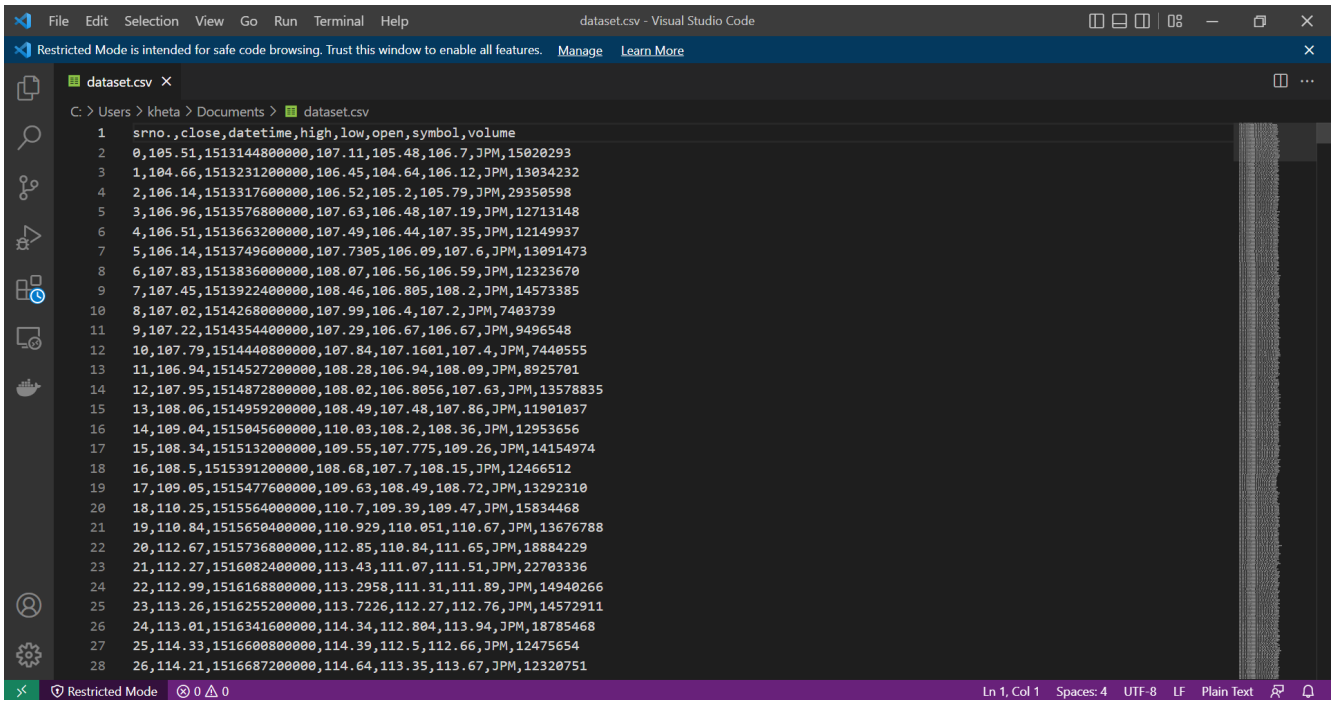

Fig 1 : Dataset Details

3.4 ML/DL Techniques

Random Forests is a supervised machine learning algorithm that uses multiple decision trees in aggregate to help make more stable and accurate predictions. Decision Trees are the fundamental building blocks of Random Forest. In essence, Decision Trees is flowlike chart structure where each node of the tree is used to test a particular attribute of the object. For example, imagine I have a person which will represent our object. We then test certain attributes of this person object. For example, one test would be whether they are male or female. The test will represent a "Decision Node" in our tree, and each of the possible outcomes "Male" or "Female" will represent a leaf node.The first "Decision Node" in our Decision Tree will be our "Root Node"

- **Root Node:** Represents the entire population or sample and this further gets divided into two or more homogeneous sets. Our starting point.
- **Splitting:** The process of dividing a node into two or more sub-nodes, for example we split on gender.
- **Decision Node:** When a sub-node splits into further sub-nodes, then it is called decision node.

- **Leaf/Terminal Node:** Nodes that do not split are called Leaf or Terminal nodes.
- **Pruning:** When we remove sub-nodes of a decision node, this process is called pruning. You can say the opposite process of splitting.
- **Branch/Subtree:** A subsection of the entire tree is called branch or sub-tree.
- **Parent and Child Node:** A node, which is divided into sub-nodes is called parent node of sub-nodes whereas sub-nodes are the child of parent node.

In machine learning, we have two categories of learning. Supervised learning and unsupervised learning. With unsupervised learning, we don't supervise the model and instead allow it to discover information on its own. We do this by providing an "UNLABELED" data set that doesn't tell the model what category or value is the "correct" answer. With supervised learning, we provide the model with a "LABELED" data set which tells the model what the "correct" value it should be. Random Forest, is an example of a supervised learning algorithm because we provide the model with a labeled data set.

Reasons for Using random forest method:

1. Instability: Even small changes to the input data can have dramatic changes to the overall structure of the decision tree.
2. They are often relatively inaccurate. Many other predictors perform better with similar data.
3. For data including categorical variables with different numbers of levels, information gain in decision trees is biased in favor of those attributes with more levels.
4. Calculations can get very complex, particularly if many values are uncertain and/or if many outcomes are linked.

These are some of the reasons it's preferable to use Random Forest because we will see that it helps overcome some of the weaknesses of Decision Trees.

3.5 Hardware and Software Requirements

Hardware requirements define what sort of hardware specifications we will be working with, and what will be needed to replicate in some other scenario.

CPU	2 GHz or faster
RAM	4 GB or higher
Disk Space	500 GB SSD or larger
Architecture	32-bit or 64-bit

The specifications given in table x.x is just an estimation. It can vary based on the kind of model used and the size of the dataset chosen.

Software requirements define what software is being used. It includes major stuff like what kind of operating system, what databases are being used. The projects' software requirements are given in table x.x.

Operating System	Windows 10 or newer
Database	Obtained through Kaggle or Yahoo Finance
Programming	Python 3.10.0 (Jupyter notebook)

This project is specifically built in jupyter notebook using python wherein all the dataset collection (imported through csv file), agents, training and testing of models and the results that the prediction produces are all implemented using various python libraries like pandas, numpy, scikit learn etc.

4) Implementation details of the Project

4.1 Language/Tools/APIs used

Project is built in Jupyter Notebook using Python 3.10.0. The tools and APIs that were used during the implementation were python libraries like numpy, pandas, scikit learn, matplotlib.

In [1]:
```
# Import libraries
import os
import sys
import requests

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.model_selection import RandomizedSearchCV
from sklearn.metrics import plot_roc_curve
from sklearn.metrics import accuracy_score, classification_report

from config import ACCOUNT_NUMBER, ACCOUNT_PASSWORD, CONSUMER_ID, REDIRECT_URI
```

Fig 2: Tools and libraries used

4.2 Data preprocessing

Print the price_data data frame to verify the data was loaded.

In [60]:
```
if os.path.exists('price_data.csv'):

    # Load the data
    price_data = pd.read_csv('price_data.csv')

else:

    # Grab the data and store it.
    grab_price_data()

    # Load the data
    price_data = pd.read_csv('price_data.csv')

# Display the head before moving on.
price_data.head()
```

Out[60]:

	close	datetime	high	low	open	symbol	volume
0	105.51	1513144800000	107.11	105.48	106.70	JPM	15020293
1	104.66	1513231200000	106.45	104.64	106.12	JPM	13034232
2	106.14	1513317600000	106.52	105.20	105.79	JPM	29350598
3	106.96	1513576800000	107.63	106.48	107.19	JPM	12713148
4	106.51	1513663200000	107.49	106.44	107.35	JPM	12149937

Fig 3 data loading

With the data now loaded, we can transform so we can calculate some of our trading indicators. The first thing we need to do is sort the data because we have multiple ticker symbols inside of our data frame. Take the data frame and call the sort_values method and specify the columns you wish to sort by using the by argument. In our case, we will be using a list of column names to sort by. The first sort is by the symbol column, and the second sort is by the datetime column.

Once we've sorted the data, we need to calculate the change in price from one period to the next. To do this, we will use the diff() method. Grab the close column and call the diff() method. The diff() method will calculate the difference from one row to the next.

In [61]:
```
# I Just need the Close
price_data = price_data[['symbol','datetime','close','high','low','open','volume']]


'''
    First, for average investors, the return of an asset is a complete and scale-free
    summary of the investment opportunity. Second, return series are easier to
    handle than prices series as they have more attractive statistical properties
'''


# sort the values by symbol and then date
price_data.sort_values(by = ['symbol','datetime'], inplace = True)

# calculate the change in price
price_data['change_in_price'] = price_data['close'].diff()
```

Fig 4 change in price() calculation

Technically, each row where the ticker symbol changes are incorrect because it's using the price from a different ticker. That means we need to have the first row of each ticker symbol be Nan for the change_in_price column. To do this, we need to break out into steps.

Step 1: Identify the rows where the ticker symbol changes. If we use the shift() method and shift every row down by one, the rows where the unshifted column DOES NOT EQUAL the shifted column is where the ticker changed. We will store these values in a variable called mask.

Step 2: Change those rows to NaN values. We can use the numpy.where() method to test our series. The test is simple, wherever the mask variable equals True, in other words, wherever the ticker symbol is different, set the change_in_price column to np.nan.

In [62]:

```
# identify rows where the symbol changes
mask = price_data['symbol'] != price_data['symbol'].shift(1)

# For those rows, let's make the value null
price_data['change_in_price'] = np.where(mask == True, np.nan, price_data['change_in_price'])

# print the rows that have a null value, should only be 5
price_data[price_data.isna().any(axis = 1)]
```

Out[62]:

	symbol	datetime	close	high	low	open	volume	change_in_price
2016	ARWR	1513144800000	3.40	3.41	3.01	3.30	2037395	NaN
504	COST	1513144800000	188.28	190.12	187.80	188.53	3062856	NaN
1512	HD	1513144800000	183.03	183.67	182.00	182.01	5177363	NaN
1008	IBM	1513144800000	153.91	156.73	153.89	156.60	5661618	NaN
0	JPM	1513144800000	105.51	107.11	105.48	106.70	15020293	NaN

Fig 5 NaN Values

4.5 Validation methodology:

A total of 6 Indicators are used in the calculation:

1) Relative Strength Index (RSI) : RSI is a popular momentum indicator that determines whether the stock is overbought or oversold. A stock is said to be overbought when the demand unjustifiably pushes the price upwards. This condition is generally interpreted as a sign that the stock is overvalued, and the price is likely to go down. A stock is said to be oversold when the price goes down sharply to a level below its true value. This is a result caused due to panic selling. RSI ranges from 0 to 100, and generally, when RSI is above 70, it may indicate that the stock is overbought and when RSI is below 30, it may indicate the stock is oversold.
2) Stochastic Oscillator : Stochastic Oscillator follows the speed or the momentum of the price. As a rule, momentum changes before the price changes. It measures the level of the closing price relative to the low-high range over a period of time.
3) Williams %R : Williams %R ranges from -100 to 0. When its value is above -20, it indicates a sell signal and when its value is below -80, it indicates a buy signal.
4) Moving Average Convergence Divergence (MACD) : EMA stands for Exponential Moving Average. When the MACD goes below the SingalLine, it indicates a sell signal. When it goes above the SignalLine, it indicates a buy signal.
5) Price Rate Of Change : It measures the most recent change in price with respect to the price in n days ago.

6) On Balance Volume : On balance volume (OBV) utilizes changes in volume to estimate changes in stock prices. This technical indicator is used to do buying and selling trends of a stock, by considering the cumulative volume: it cumulatively adds the volumes on days when the prices group, and subtracts the volume on the days when prices go down, compared to the prices of the previous day.

5) Result and Analysis

To get a more detailed overview of how the model performed, we can build a classification report that will compute the F1_Score, the Precision, the Recall, and the Support. Now, I'm assuming you don't know what these metrics are, so let's take some time to go over them.

1) Accuracy:

   Accuracy measures the portion of all testing samples classified correctly.

2) Recall

   Recall (also known as sensitivity) measures the ability of a classifier to correctly identify positive labels. The recall is intuitively the ability of the classifier to find all the positive samples. The best value is 1, and the worst value is 0.

3) Specificity

   Specificity measures the classifier’s ability to correctly identify negative labels.

4) Precision

   Precision measures the proportion of all correctly identified samples in a population of samples which are classified as positive labels. The precision is intuitively the ability of the classifier not to label as positive a sample that is negative. The best value is 1, and the worst value is 0.

In [94]:
```
# Define the traget names
target_names = ['Down Day', 'Up Day']

# Build a classifcation report
report = classification_report(y_true = y_test, y_pred = y_pred, target_names = target_names, output_dict = True)

# Add it to a data frame, transpose it for readability.
report_df = pd.DataFrame(report).transpose()
report_df
```

Out[94]:

	precision	recall	f1-score	support
Down Day	0.630508	0.699248	0.663102	266.000000
Up Day	0.749216	0.686782	0.716642	348.000000
accuracy	0.692182	0.692182	0.692182	0.692182
macro avg	0.689862	0.693015	0.689872	614.000000
weighted avg	0.697789	0.692182	0.693447	614.000000

Fig 6 Classification Report

Plotting of the Confusion matrix:

]:
```
from sklearn.metrics import confusion_matrix, plot_confusion_matrix

rf_matrix = confusion_matrix(y_test, y_pred)

true_negatives = rf_matrix[0][0]
false_negatives = rf_matrix[1][0]
true_positives = rf_matrix[1][1]
false_positives = rf_matrix[0][1]

accuracy = (true_negatives + true_positives) / (true_negatives + true_positives + false_negatives + false_positives)
percision = true_positives / (true_positives + false_positives)
recall = true_positives / (true_positives + false_negatives)
specificity = true_negatives / (true_negatives + false_positives)

print('Accuracy: {}'.format(float(accuracy)))
print('Percision: {}'.format(float(percision)))
print('Recall: {}'.format(float(recall)))
print('Specificity: {}'.format(float(specificity)))

disp = plot_confusion_matrix(rand_frst_clf, X_test, y_test, display_labels = ['Down Day', 'Up Day'], normalize = 'true', cmap=plt.cm.Blues)
disp.ax_.set_title('Confusion Matrix - Normalized')
plt.show()
```

```
Accuracy: 0.6921824104234527
Percision: 0.7492163009404389
Recall: 0.6867816091954023
Specificity: 0.6992481203007519
```

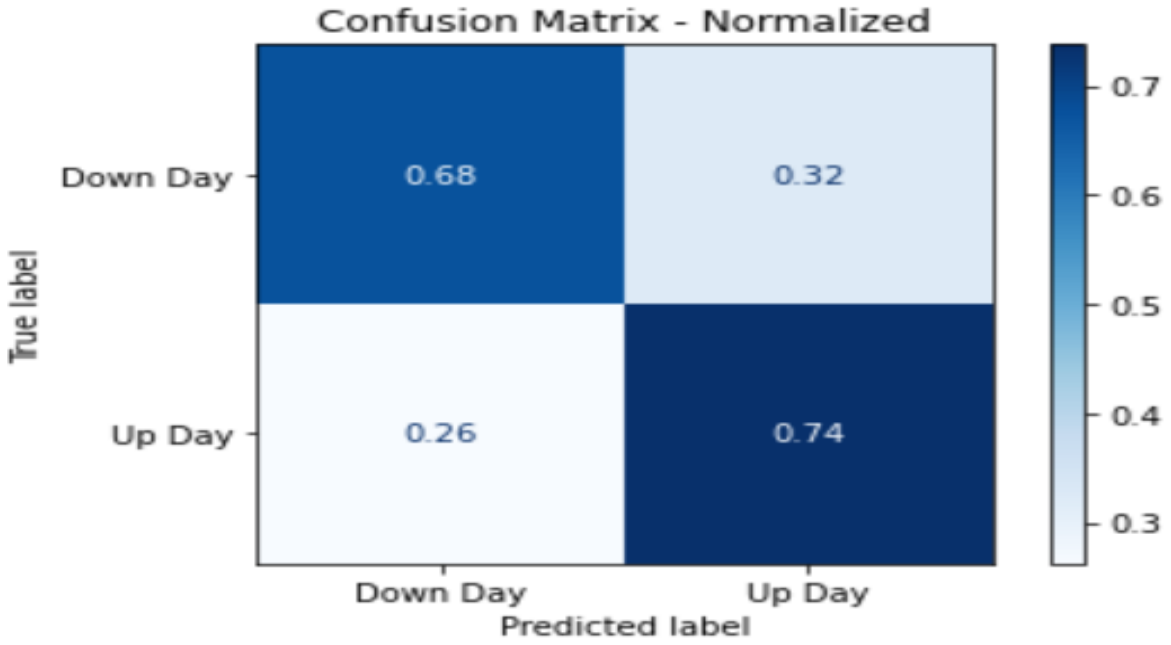

Fig 7 Confusion matrix

Plotting of Feature Significance Curve:

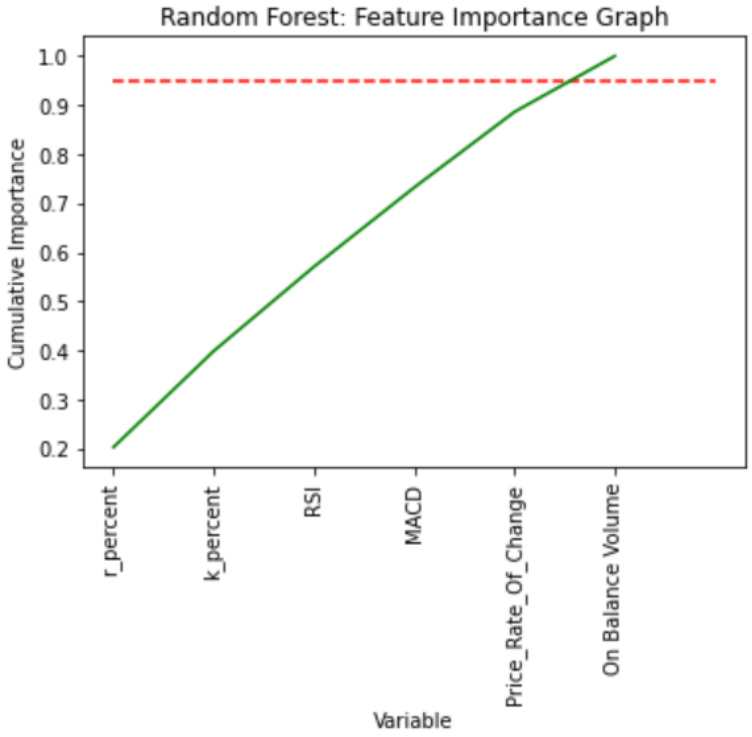

Fig 8 Feature importance Graph

Plotting of ROC Curve:

```
In [67]:  # Create an ROC Curve plot.
          rfc_disp = plot_roc_curve(rand_frst_clf, X_test, y_test)
          plt.show()
```

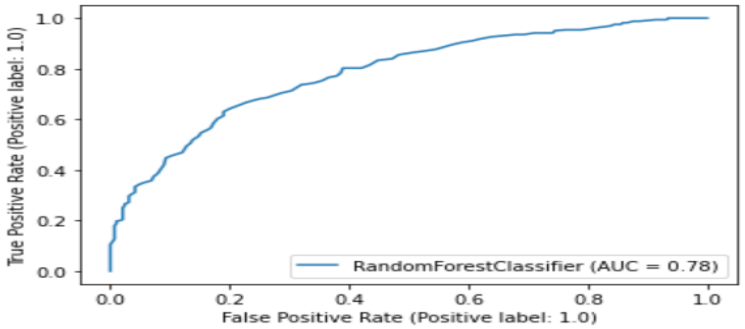

Fig 9 ROC Curve

6 Conclusion:

6.1. Novelty in the proposed solution

The novelty in the proposed solution is the use of indicators and not just raw data to train the ML model. Usage of such indicators help in understanding patterns better as they show market statistics with a better understanding. We have used a total of six indicators to determine the agents and all have been explained.

6.2 Limitations of the Project

One major limitation of the project is that it only predicts the direction of the stock and not the actual prices. If the model would be able to predict prices, then it would be a huge advantage to us as better trades could be made.

6.3. Future Enhancements

The project has a lot of scope in future and the topic and related research will always be in demand because the application of this has the power to control the flow of money which will keep the interest boosted. Hybrid models could be developed to get better accuracy and eliminate flaws that a single model produced. Apart from that, we can use NLP to view things from a sentimental analysis point of view.

6.4. Summary

Authors opine that application of machine learning techniques in stock price forecasting needs to be a well thought process and demands painstakingly detailed execution. The proposed approach is a paradigm shift in this class of problems by reformulating a traditional forecasting model as a classification problem. Moreover, knowledge discovery from the analysis should create new frontiers or applications such as a trading strategy based on the strengths of the classification accuracy, investigating the behavior of certain classes of stocks.

REFERENCES:

[1] B. Chhimwal and V. Bapat, “Impact of foreign and domestic investment in stock market volatility: Empirical evidence from India,” Cogent Economics & Finance, vol. 8, no. 1, Apr. 2020.

[2] S. Sridhar, S. Mootha, and S. Subramanian, “Decentralized Stock Exchange Implementation using Ethereum,” in 2020 International Seminar on Intelligent Technology and Its Applications (ISITIA), pp. 234- 241, 2020.

[3] C. Pop et al., “Decentralizing the Stock Exchange using Blockchain And Ethereum-based implementation of the Bucharest Stock Exchange,” in 2018 IEEE 14th International Conference on Intelligent Computer Communication and Processing (ICCP), pp. 459-466, 2018.

[4] N. Sakthivel and A. Saravanakumar, “Investors’ Satisfaction on Online Share Trading and Technical Problems Faced by the Investors: A Study in Coimbatore District of Tamilnadu,” International Journal of Management Studies, vol. V, no. 3(9), p. 71, Jul. 2018.

[5] D. Shah, H. Isah, and F. Zulkernine, “Stock Market Analysis: A Review and Taxonomy of Prediction Techniques,” International Journal of Financial Studies, vol. 7, no. 2, p. 26, May 2019.

[6] U. Hathi, “Indian Companies Act 2013 Highlights and Review,” SSRN Electronic Journal, 2014.

[7] S. V. Shenoy and K. Srinivasan, “Relationship of IPO Issue Price and Listing Day Returns with IPO Pricing Parameters,” International Journal of Management Studies, vol. V, no. 4(1), p. 11, Oct. 2018.

[8] G. Tanty and P. K. Patjoshi, “A Study on Stock Market Volatility Pattern of BSE and NSE in India,” Asian Journal of Management, vol. 7, no. 3, p. 193, 2016.

[9] S. Sridhar, S. Mootha, and S. Subramanian, “Detection of Market Manipulation using Ensemble Neural Networks,” in 2020 International Conference on Intelligent Systems and Computer Vision (ISCV), pp. 1–8, 2020

[10] S. Zavadzki, M. Kleina, F. Drozda, and M. Marques, “Computational Intelligence Techniques Used for Stock Market Prediction: A Systematic Review,” IEEE Latin America Transactions, vol. 18, no. 04, pp. 744–755, Apr. 2020.

[11] M. C. Joshi, “Factors Affecting Indian Stock Market,” SSRN Electronic Journal, 2013.

[12] S. Alhazbi, A. B. Said and A. Al-Maadid, "Using Deep Learning to Predict Stock Movements Direction in Emerging Markets: The Case of Qatar Stock Exchange," 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), Doha, Qatar, pp. 440-444, 2020.

[13] D. Wei, “Prediction of Stock Price Based on LSTM Neural Network,” in 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM), pp. 544-547, 2019.

[14] Y. Keneshloo, T. Shi, N. Ramakrishnan, and C. K. Reddy, “Deep Reinforcement Learning for Sequence-to-Sequence Models,” IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 7, pp. 2469–2489, 2019.

[15] K. Palasundram, N. Mohd Sharef, N. A. Nasharuddin, K. A. Kasmiran, and A. Azman, “Sequence to Sequence Model Performance for Education Chatbot,” International Journal of Emerging Technologies in Learning (iJET), vol. 14, no. 24, p. 56, Dec. 2019.

[16] H. Jain and G. Harit, "An Unsupervised Sequence-to-Sequence Autoencoder Based Human Action Scoring Model," 2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Ottawa, ON, Canada, pp. 1-5, 2019.

[17] K. A. Althelaya, E.-S. M. El-Alfy, and S. Mohammed, “Evaluation of bidirectional LSTM for short-and long-term stock market prediction,” in 2018 9th International Conference on Information and Communication Systems (ICICS), pp. 151-156, 2018.

[18] J. Chou and T. Nguyen, "Forward Forecast of Stock Price Using SlidingWindo Metaheuristic-Optimized Machine-Learning Regression," in IEEE Transactions on Industrial Informatics, vol. 14, no. 7, pp. 3132- 3142, July 2018.

[19] Y. Zhang and Q. Yang, “An overview of multi-task learning,” National Science Review, vol. 5, no. 1, pp. 30–43, 2018.

THANK YOU