PROJECT REPORT

STOCK MARKET PRICE PREDICTION & TREND ANALYSIS

IE-7275 Data Mining in Engineering

Prof. Srinivasan (Sri) Radhakrishnan

Group 12

Bhavya Batra

Shruti Telang

Table of Contents

1) PROBLEM SETTING

The stock market attracts thousands of investors from around the globe. It allows companies to issue and sell their shares to the common public, thus, raising necessary capital from the investors. On the other hand, Investors get these company shares which they can expect to hold for their preferred duration. The prediction of a stock market trend acts as an early financial distress warning system for the investors as well as shareholders. Nowadays stock market data is stored digitally and is easily accessible. The availability of this large amount of data allows us to analyze the trends and predict stock prices. It will facilitate both appropriate stock selection and significant profit on trading.

2) PROBLEM DEFINITION

The Stock Market does not follow any fixed model and is affected by several factors. So, an investor or trader needs to be watchful of the market behavior before investing in any stock. Financial traders and investors often face this problem as they don't understand which stocks to buy or sell to get optimal profits. The stock market data is available everywhere but analyzing all this information physically is not easy. The objective of our project is to investigate the trends and factors affecting the stock prices and then predict stock prices using data mining techniques. The purpose is to comparatively analyze the effectiveness of prediction algorithms on stock market data and get general insight on this data through visualization to predict future stock behavior and value at risk for each stock.

3) DATA SOURCES

A stock market index is a statistical measure that signifies the direction of price movement. There are multiple indexes available in the stock market. Each index has its methodology for calculating the stock parameters. The Dataset for this project has been obtained from NASDAQ's official website. It stands for "National Association of Securities Dealers Automated Quotations." NASDAQ is one index that highly attracts growth-oriented companies. It is one of the largest stock exchanges based on market capitalization. The Dataset is of 1 year for 12 different companies. Below is the link to the dataset:

https://www.nasdaq.com/market-activity/stocks/nflx/historical (1)

4) DATA DESCRIPTION

Stock market data allows investors to know the latest price and observe other historical trends in the market. Real-time and historical parameters are equally important when it comes to studying the stock market. The data consists of both real-time and historical parameters derived from NASDAQ. The dataset contains a total of 3025 data entries and 7 attributes. The parameters are as follows:

Attribute	Datatype	Description
Date	object	Date of stock sale. It is from 11th August'20 to 11th August'21 for 12 companies.
Stock	object	The Stock variable represents the common stock name for 12 companies which are Netflix, Tesla, JP Morgan, Starbucks, Apple, CVS, FedEx, AT&T, United Airlines, Amazon, Pfizer, Walmart.
Volume	float	It represents the number of shares traded on a particular day.
Open	object	It represents the price at which a stock of a particular company started trading when the market opens.
High	object	Highest price of the stock for a given date
Low	object	Lowest price of the stock for a given date
Close/Last	object	Closing price generally refers to the last price at which a stock trades during a regular trading session.

Table 1

5) DATA MINING TASKS

Some data mining tasks were performed on the dataset before implementing Prediction Models. These majorly include Data Preprocessing, Data Transformation, and Exploratory Data Analysis.

a. DATA PREPROCESSING

i. Since the data was obtained from the official website of NASDAQ, there were no missing values in the dataset.
ii. Some special characters were removed from the dataset.
iii. Data description shows the need for type conversion of the features used. The date was converted String to Datetime format. Open, High, Low, and Close were converted from String to Float.
iv. Stock values were encoded using LabelEncoder because some models cannot take categorical variables as predictors.
v. Since Stock market is based on time series so feature extraction technique Catch22 was used on Closing price (response variable).

b. EXPLORATORY DATA ANALYSIS

Some visualizations were created to analyze the trends of the stock market and to gain some general insights.

i. Correlation Matrix: The matrix shows that variables Open, High, Low and Close are highly correlated.

	Volume	Open	High	Low	Close/Last
Volume	1.000000	-0.189322	-0.188509	-0.190230	-0.189190
Open	-0.189322	1.000000	0.999925	0.999913	0.999849
High	-0.188509	0.999925	1.000000	0.999885	0.999913
Low	-0.190230	0.999913	0.999885	1.000000	0.999932
Close/Last	-0.189190	0.999849	0.999913	0.999932	1.000000

Figure 1

ii. Scatter Plot: Below is the Scatter plot between the Closing Price and Volume of 12 company's stocks. The plot shoes that volume and closing price are negatively correlated. If the price of a particular company's stock increases the number of shares traded in the period decreases.

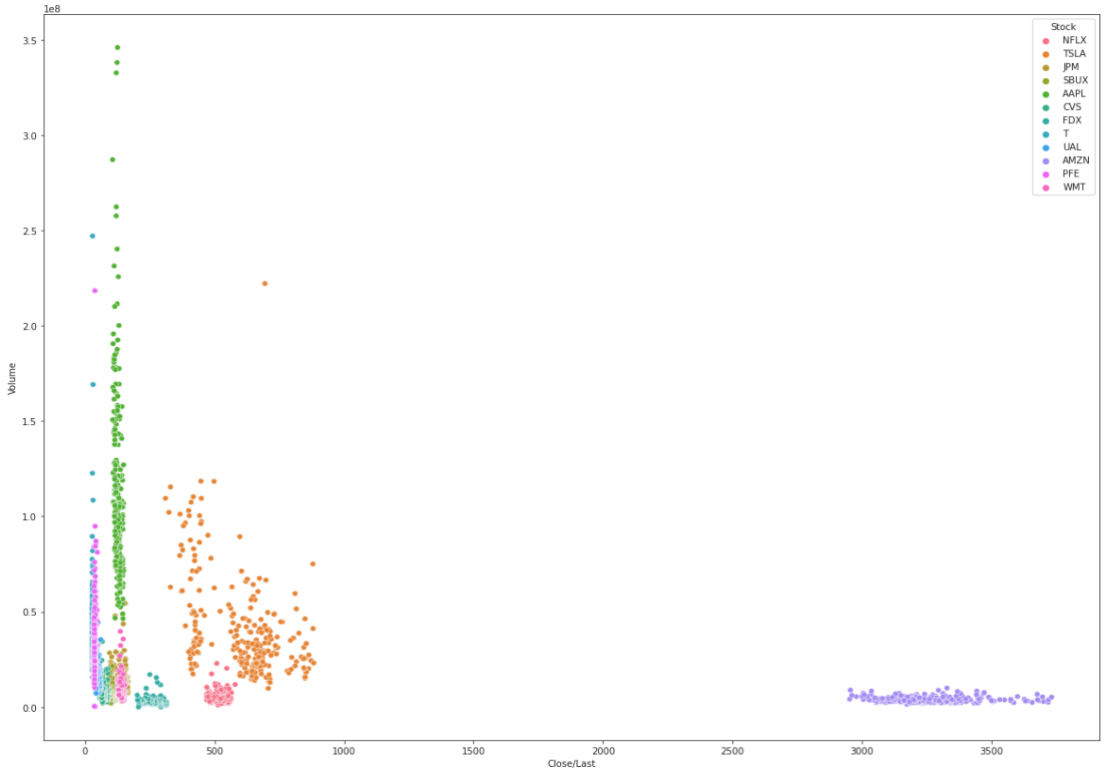

Figure 2

iii. Time Series Chart: The time series chart shows the trend of all the companies for 1 year. It shows that Tesla has the highest closing price followed by Netflix. Companies like AT&T, Pfizer, United Airlines have consistent closing prices throughout the year.

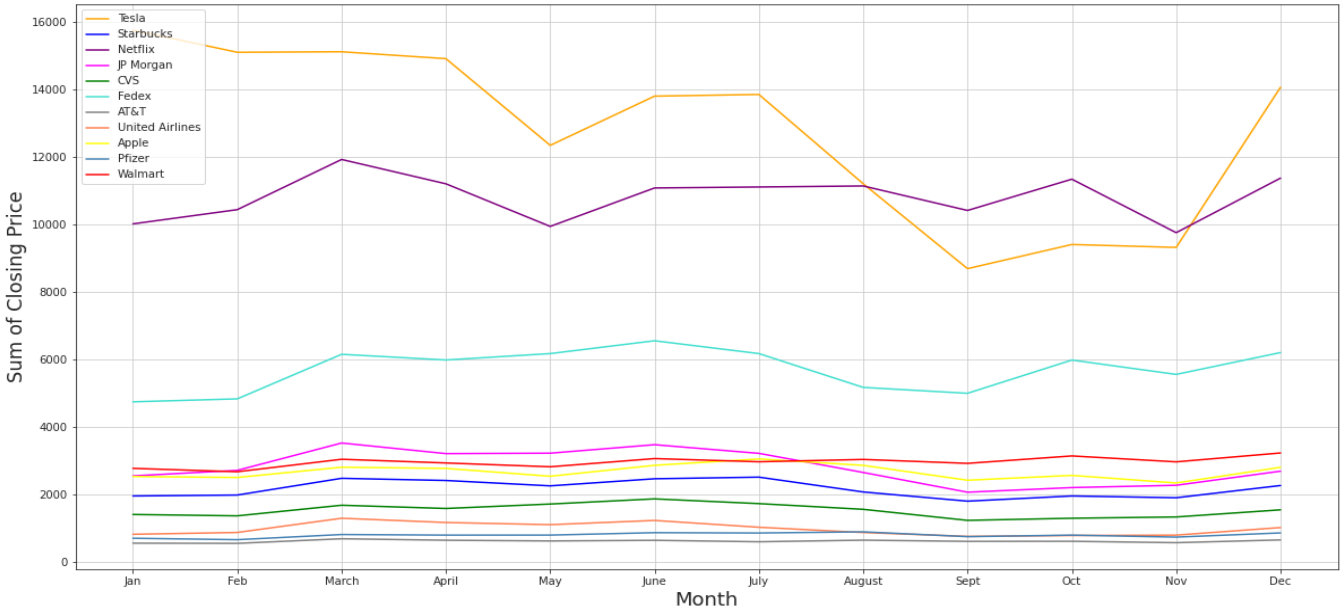

Figure 3

iv. OHLC Chart: OHLC (Open High Low Close) chart is a type of plot showing open, high, low, and close values. It depicts the fluctuation in price for a given stock in a time range. Since the data being used here is for the current day, the OHLC plot below has been created for the group of sectors. The tip of the lines in the plot represents high and low prices. The horizontal segments represent close and open prices. Increasing stock values are in green, decreasing are in red. There is a slider at the bottom of this plot to zoom in and visualize the price for each day. Placing the cursor on a single line on the chart gives us the OHLC values for that company.

Figure 4

c. **CATCH22 AND VARIABLE SELECTION:** Catch22 captures a diverse and interpretable signature of time series in terms of their properties, including linear and non-linear autocorrelation, successive differences, value distributions and outliers, and fluctuation scaling properties.

CAnonical Time-series CHaracteristics

- Collection of 22 time series features
- 22 reduced features from a set of 4791 in hctsa (Highly Comparative Time-series Analysis) package.
- Exhibits strong classification/regression results if used.
- Selecting relevant features for obtaining maximum accuracy possible.
- Usage: correlation matrix for detecting features possessing high multicollinearity.

- Dropping either of these feature pairs; using the rest for model training.

6) DATA MINING MODELS

This section involves training Machine Learning models with the mentioned features for predicting almost accurate closing prices of stocks. Multiple models were trained for this purpose and a comparative analysis was done to choose the best performing model. Since this was a regression problem, R-squared, Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) were considered as the evaluation metrics.

R-squared is a statistical measure of how well a model fits the training data. RMSE and MAE on the other hand helps one know the significant error between the actual and predicted values of the response variable.

Following are the models considered in this project with their respective training and testing performances:

A. LINEAR REGRESSION: This is a linear model that calculates the value of the response variable(y) from a linear combination of the predictors. Predictors are nothing but input variables(x). Linear Regression fits a given data in a linear equation. Each input feature has a weight assigned to it and, the response variable is calculated by simply summing up the products of these input features and their respective coefficients.

The Linear Regression model gave good results and did not require any feature selection. It worked well for this dataset since most features were highly correlated with each other. The other term for this condition is multicollinearity which is very common when it comes to studying the stock market. It is always advisable to remove multicollinearity but, the stock market is an exception.

PERFORMANCE:

Metric	Value
R-squared	0.999906663147768
RMSE	8.584406568993206
MAE	2.816913696017794

Table 2

The figure above shows how well the Linear Regression model fitted training data. The model gave an error of $8.5 (RMSE) and $2.8 (MAE) which is pretty good when one wants to predict the closing price of a stock.

B. K NEAREST NEIGHBOURS: k Nearest Neighbors or kNN is a supervised machine learning algorithm. It is a non-parametric method used for both classification and regression problems. kNN, while predicting, calculates the value of the response variable based upon how similar it is to a set of training features. The similarity in mathematical terms is nothing but the distance between response and the training features. The set of training features is defined by the value of k. The hyperparameters for kNN considered while feature training is as follows:

- k: Number of neighbors
- Distance metric: This is the distance metric used for calculating similarly. Value of the response variable will be the same as the datapoint lying closest to it. Distance metric most commonly holds the following values {'manhattan', 'euclidian', 'cosine', 'jaccard'}
- Weights: Derived using the weight function. This is required for prediction tasks. Weights can take up the values {'uniform', 'distance'

PERFORMANCE:

The best result is obtained for k = 3 and manhattan as the distance metric. k takes up values from 1 to 21. A subset of all these possible combinations can be seen below.

Best performing model

	#neighbors	distance	weight	rmse	R2 square
11	3	manhattan	distance	15.8352	0.999655

Table 3

All results

	#neighbors	distance	weight	rmse	R2 square
0	1	euclidean	uniform	17.5302	0.999578
1	1	euclidean	distance	17.5302	0.999578
2	1	manhattan	uniform	15.9652	0.99965
3	1	manhattan	distance	15.9652	0.99965
4	2	euclidean	uniform	17.326	0.999587
5	2	euclidean	distance	16.3157	0.999634
6	2	manhattan	uniform	17.1543	0.999596
7	2	manhattan	distance	15.9545	0.99965
8	3	euclidean	uniform	18.9884	0.999504
9	3	euclidean	distance	16.8468	0.99961
10	3	manhattan	uniform	17.2794	0.99959
11	3	manhattan	distance	15.8352	0.999655
12	4	euclidean	uniform	20.0246	0.999449
13	4	euclidean	distance	17.1279	0.999597
14	4	manhattan	uniform	17.7886	0.999565
15	4	manhattan	distance	16.1888	0.99964
16	5	euclidean	uniform	27.5543	0.998956
17	5	euclidean	distance	21.8132	0.999346
18	5	manhattan	uniform	18.3977	0.999535

Table 4

The figure below is a line plot of neighbors versus the rmse values.

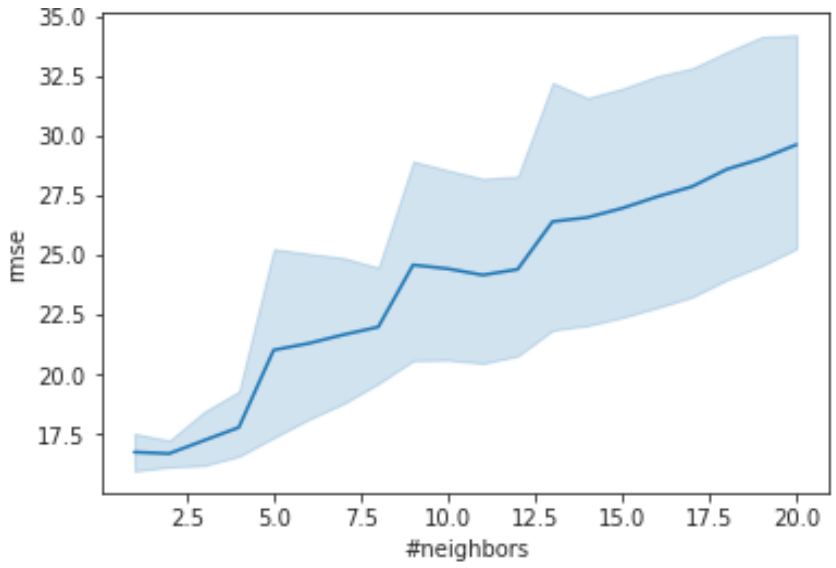

Figure 5

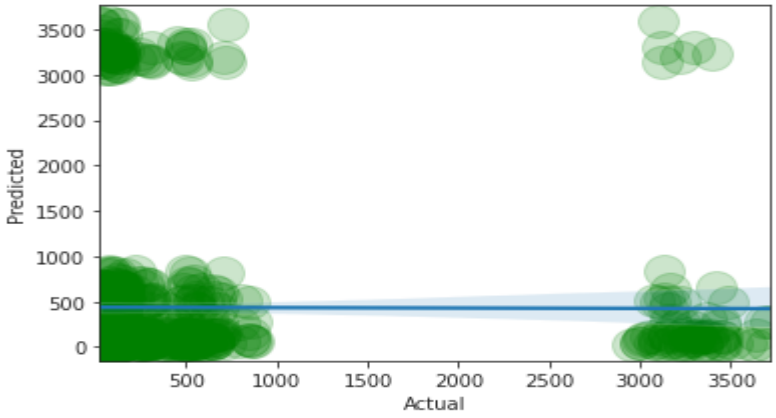

Figure 6

C. RANDOM FOREST: It is another supervised machine learning algorithm. It uses ensemble learning methods for prediction tasks. The algorithm considers all possible feature combinations to predict the respective outcomes and then averages them out to calculate the final value of the response variable. The hyperparameters considered while training features in this project are as follows:

Max_Depth : Maximum depth of the tree
Criterion : Measuring criteria for split
n_estimators : Number of trees in the forest

PERFORMANCE:

Best performing model

	depth	Puritymethod	Rmse	R2 Squared
29	15	mae	8.25106	0.999905

Table 5

All results

	depth	Puritymethod	Rmse	R2 Squared
1	1	mae	206.336	0.940785
2	1	mse	193.022	0.94818
3	2	mae	82.104	0.990624
4	2	mse	81.5665	0.990746
5	3	mae	50.0613	0.996514
6	3	mse	45.4062	0.997132
7	4	mae	19.7721	0.999456
8	4	mse	21.2069	0.999374
9	5	mae	12.4537	0.999784
10	5	mse	12.4241	0.999785
11	6	mae	9.28066	0.99988
12	6	mse	9.44986	0.999876
13	7	mae	9.67319	0.99987
14	7	mse	9.19455	0.999882
15	8	mae	9.39195	0.999877
16	8	mse	8.59731	0.999897
17	9	mae	9.27351	0.99988
18	9	mse	8.94395	0.999889

Table 6

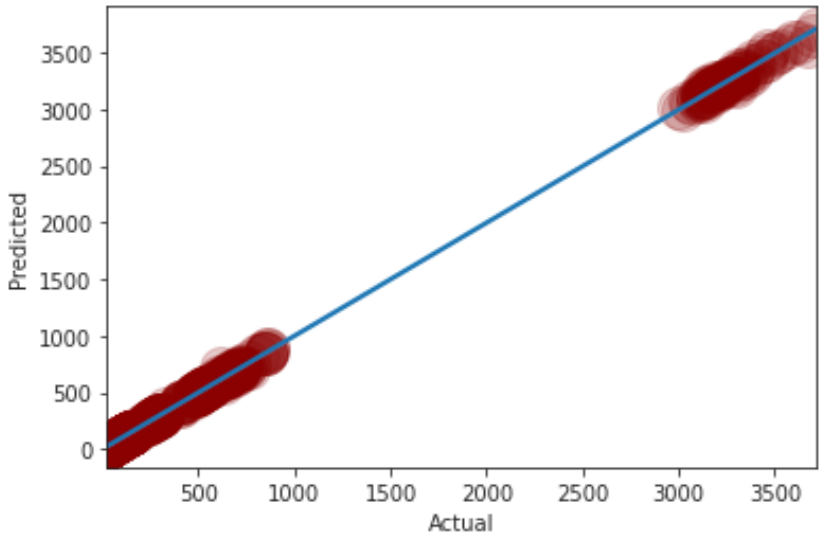

Figure 7

D. NEURAL NETWORK: It was implemented with the help of the Multi-Layered Perceptron (MLP) regressor function in the sci-kit-learn package in python. MLP consists of perceptrons in several layers that are interconnected with each other, forming a network. This network begins with an input layer, followed by one or more hidden layers and a final output layer. The figure below depicts a typical neural network.

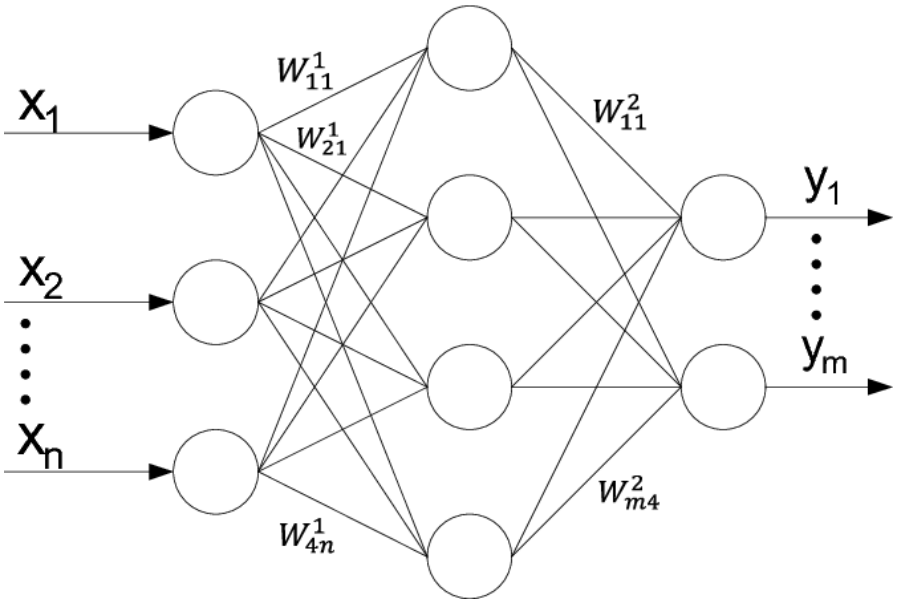

Figure 8

Firstly, the input node values are multiplied with the edge weights and passed through the hidden layer. Each hidden layer node consists of an activation function (also called threshold function). If the value from the input node is above the decided threshold, the value is passed on to the next layer. This procedure is carried out till the output layer is reached. The hidden layer is also responsible for weight tuning. Tuning is performed till a minimal error is achieved. This is the purpose of backpropagation that is performed on achieving the output. True value of the output is compared with the predicted value and weight adjustments are performed iteratively until the model can predict values almost accurately.

Hyperparameters of an MLP regressor considered are as follows:

- Learning rate: Rate of weight adjustment
- Transfer function/activation: Activation or Threshold function for the hidden layer
- Hidden layer sizes: Number of neurons in the given number of hidden layers

PERFORMANCE:

	Learning Rate	Transfer Function	Hidden_Layer	R-squared	RMSE	MAE
22	0.01	relu	100	0.999811	12.108047	6.949956

Table 7

7) PERFORMANCE EVALUATION

Model	Performance Evaluation Metrics	
	R-squared	RMSE
Linear Regression	0.999906	8.5844
k-Nearest Neighbours	0.999695	15.8352
Random Forest	0.999905	8.25106
Neural Network	0.999811	12.1080

Table 8

8) PROJECT RESULTS

The figure below compares the individual performance of the four models executed in this project. Legend depicts the R-squared values exhibited by each model. RMSE values on the other hand are different for each model except for Linear Regression and Random Forest. If one closely observes, Random Forest gives the least error of $8.25106 (refer to the table in the previous section).

We can hence conclude that Random Forest with a depth of 15 and ‘mae’ as the purity method is the best model for predicting stock prices.

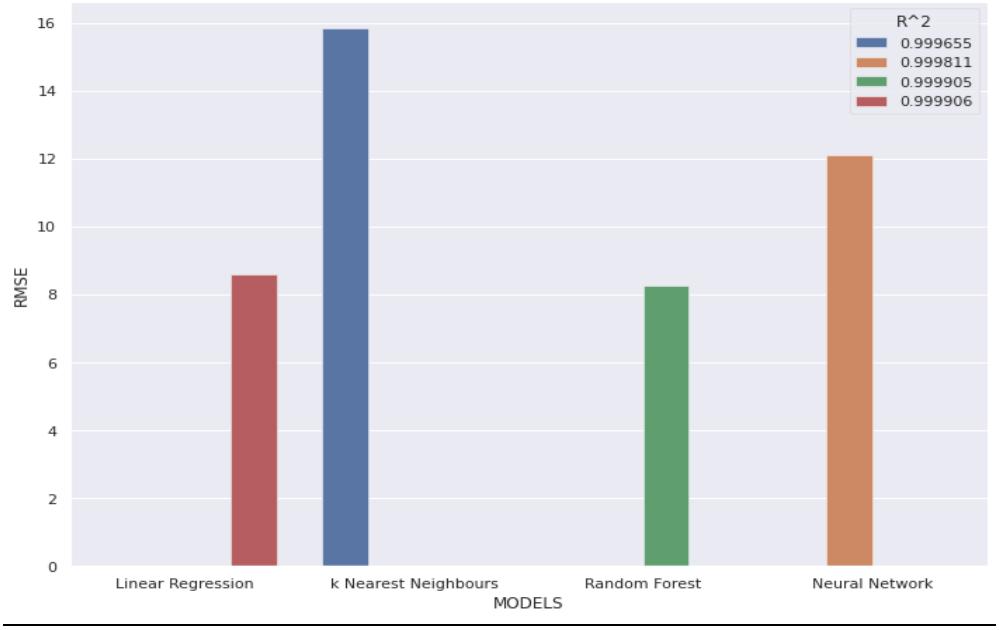

Figure 9

9) IMPACT OF THE PROJECT OUTCOMES

- This model can help financial trader, or an investor understand which stocks to buy or sell to get optimal profits.
- This model will help to know future closing price of stocks of 12 companies.

10) REFERENCES

[1] https://www.nasdaq.com/market-activity/stocks/nflx/historical

[2] https://www.investopedia.com/terms/o/ohlcchart.asp

[3] https://arxiv.org/abs/1901.10200

[4] https://scikit-learn.org/stable/supervised_learning.html#supervised-learning

[5] https://blog.minitab.com/en/adventures-in-statistics-2/regression-analysis-how-do interpret-r-squared-and-assess-the-goodness-of-fit

COVER SHEET

Project Title: Stock Market Price Prediction & Trend Analysis

Project Milestone: Project Report

Student Name: Bhavya Batra
Shruti Telang

Email: batra.bh@northeastern.edu
telang.sh@northeastern.edu

Contribution: Bhavya Batra- 50%
Shruti Telang- 50%

Submission Date: 20th August'21