ASSIGNMENT SUBMISSION FORM

This will be the first page of your assignment

Course Name	: **Machine Learning – Supervised Learning 1**
Assignment Title	: **Group Project**
Submitted by	: **Group 6**

Student Name	PG ID
Unnati Khinvasara	12120097
Harsimar Singh Arora	12120011
Rohini Purnima	12120027
Rohit Thakur	12120040
Mohit Kothari	12120035

ISB Honour Code

- I will represent myself in a truthful manner.
- I will not fabricate or plagiarise any information with regard to the curriculum.
- I will not seek, receive or obtain an unfair advantage over other students.
- I will not be a party to any violation of the ISB Honour Code.
- I will personally uphold and abide, in theory and practice, the values, purpose and rules of the ISB Honour Code.
- I will report all violations of the ISB Honour Code by members of the ISB community.
- I will respect the rights and property of all in the ISB community.
- I will abide by all the rules and regulations that are prescribed by ISB.

Note: Lack of awareness of the ISB Honour Code is never an excuse for a violation. Please go through the Honour Code in the student handbook, understand it completely. Please also pay attention to the following points:

- Please do not share your assignment with your fellow students under any circumstances if the Honour Code scheme prohibits it. The HCC considers both parties to be guilty of an Honour Code violation in such circumstances.
- If the assignment allows you to refer to external sources, please make sure that you cite all your sources. Any material that is taken verbatim from an external source (website, news article etc.) must be in quotations. A much better practice is to paraphrase the source material (it still must be cited).

(Please start writing your assignment below)

A. Problem Statement

An investment firm is looking for opportunities to invest on equities with a horizon of 3 months. The firm is interested to invest in companies in specific industry sector when the companies declare their quarterly results. They would like to understand if the investment would bring them a minimum expected return or not. So, they are planning to form a team of data scientists to analyse and build ML models to estimate the quarterly return on investment in a company based on the company performance indicators and the external industry and economic factors.

B. Data Description

For the purpose of this Project, we have decided to undertake analysis on the ***Automobile Sector***.
A list of companies for which the data set is build is given below -

Sr No	Company Name	Sub-Sector
1	Atul Auto	2 & 3 Wheelers
2	Bajaj Auto	2 & 3 Wheelers
3	Hero Motocorp	2 & 3 Wheelers
4	Scooters India	2 & 3 Wheelers
5	TVS Motors	2 & 3 Wheelers
6	Mahindra & Mahindra	Auto Truck
7	Eicher	LCV & HCV
8	Tata Motors	LCV & HCV
9	Hind Motors	Passenger Cars
10	Maruti Suzuki	Passenger Cars
11	Escorts Kubota	Tractors
12	HMT	Tractors
13	Ashok Leylands	Trucks LCVs
14	SML Isuzu	Trucks LCVs

We have collected data of 24 Quarters (from 2014 to 2019) for building the model dataset.
The Parameters which are considered for the input dataset are as given below -

1. ***Quarterly Share Price***
The Share Price of the Company is an important parameter to understand the trend and direction of the company price for prediction. Hence, we have taken the quarterly opening price of the respective share as a parameter for model training.

2. ***Revenue***
Gross Manufacturing Revenue of the Company is also being taken as a company specific parameter for model training since quarterly sales of the company is directly associated with company performance and affects the share price.

3. ***Earning Per Share (EPS)***
Quarterly EPS is being taken as a company specific parameter since it represents profitability of the company and directly influences the share price.

4. ***Currency Exchange Rate***
We have considered currency exchange rate of USD as a market specific parameter since Automobile Industry in India has heavy exports of products as well as import of raw material which is directly dependent on fluctuating currency rates.

5. ***Metal Price***
We have considered metal price of aluminum as an economic indicator since it is a significant raw material consumed in manufacturing process for automobile industries.

6. **Rubber Price**
Automotive industry is a major consumer of Rubber and hence we have considered quarterly rubber price as an industry parameter.

7. **Fuel Price**
The automobile industry has a direct relationship with fuel prices. Increase in the fuel prices have negative impact on automobile industry in terms of sale of product as well as usage. Hence we have considered petrol as an important industry parameter for the purpose of model building.

8. **Consumer Price Index (CPI)**
We have considered Consumer Price Index (Transport & Communication) as a macro economic factor while building the model since inflation in the sector would directly impact the sale of vehicles thereby influencing company share prices.

9. **Index of Industrial Production (IIP) : Motor Vehicles**
IIP is an index which shows the growth rates in the industry within a stipulated period of time. It represents industry-wise growth in production and hence is an important industry-wise parameter.

10. **Volatility of daily returns**
We have considered % volatility of average daily returns for the automobile industry as an important industry factor to gauge the working of the automobile industry at large.

11. **Excess total returns over NIFTY**
We have considered this parameter as a check to understand the comparison of the automobile industry over NIFTY Index and its impact on the share prices.

12. **Industry Production Metric**
We have considered the % quarterly growth rate of change of the Automobile Industry Production as an important parameter to understand the entire Industry's working and its effect on share prices.

13. **Exports (Q-o-Q% change)**
Quarterly change in Export growth of the Industry is a crucial parameter to understand the health and growth of Industry and hence we have considered the same as an industry parameter in the Model.

C. Model Details

- For purpose of Problem formulation, we have derived the following two metrics as the output variable for the Regression Model & the Classification Model respectively –

- Return Variable is % of the quarterly return from last quarter,
  = Stock Price $_{\text{Quarter(n+1)}}$ – Stock Price $_{\text{Quarter n}}$) / Stock Price $_{\text{Quarter n}}$

- Classification Criteria is whether above Return >=4%

- Since we have time series data for 24 Quarters, the data split for the purpose of model building is as follows –

  Training Set - 20 Quarters from March 2014 to Dec 2018
  Test Set - 4 Quarters from March 2019 to Dec 2019

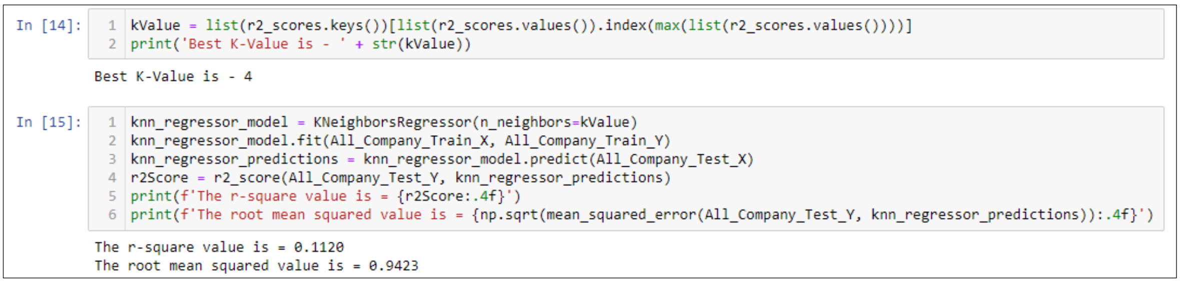
- Further, for the purpose of the model database, we have adopted One Hot Encoding to be able to use the Companies as categorical variables and hence created dummies of all companies for the model.

- It is to be noted that the model has been built in Jupyter Notebook (.ipynb file) and comments have been mentioned everywhere alongwith the codes to explain the modelling process. Request you to refer the same for modelling steps.

D. Regression Modelling

- **Find the best model between linear regression, KNN.**

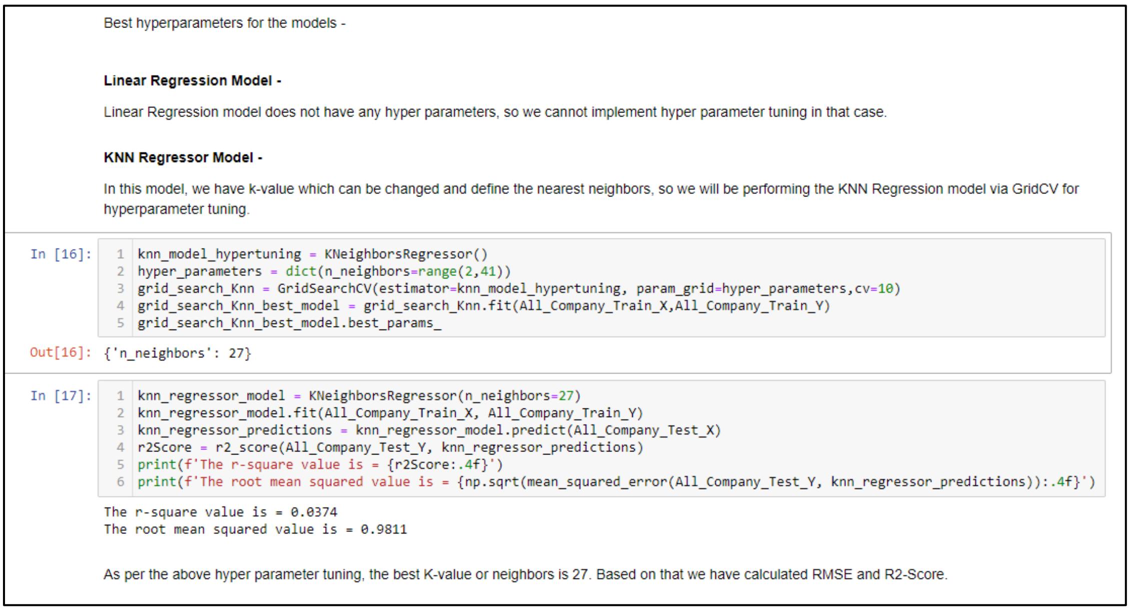
  KNN Regressor model seems to be a better model between the two, since we are getting better R-Squared value as compared with Linear Regression model (wherein we are getting a negative R-Squared value). This means that Linear Regression is not a good model to work with for this use case.



```
In [14]: kValue = list(r2_scores.keys())[list(r2_scores.values()).index(max(list(r2_scores.values())))]
         print('Best K-Value is - ' + str(kValue))

         Best K-Value is - 4

In [15]: knn_regressor_model = KNeighborsRegressor(n_neighbors=kValue)
         knn_regressor_model.fit(All_Company_Train_X, All_Company_Train_Y)
         knn_regressor_predictions = knn_regressor_model.predict(All_Company_Test_X)
         r2Score = r2_score(All_Company_Test_Y, knn_regressor_predictions)
         print(f'The r-square value is = {r2Score:.4f}')
         print(f'The root mean squared value is = {np.sqrt(mean_squared_error(All_Company_Test_Y, knn_regressor_predictions)):.4f}')

         The r-square value is = 0.1120
         The root mean squared value is = 0.9423
```

- **Find the best hyper parameters for the models.**

  Kindly refer the code file for the hyper parameter tuning. A screenshot of the same is reproduced below for reference -



Best hyperparameters for the models -

Linear Regression Model -

Linear Regression model does not have any hyper parameters, so we cannot implement hyper parameter tuning in that case.

KNN Regressor Model -

In this model, we have k-value which can be changed and define the nearest neighbors, so we will be performing the KNN Regression model via GridCV for hyperparameter tuning.

```
In [16]: knn_model_hypertuning = KNeighborsRegressor()
         hyper_parameters = dict(n_neighbors=range(2,41))
         grid_search_Knn = GridSearchCV(estimator=knn_model_hypertuning, param_grid=hyper_parameters,cv=10)
         grid_search_Knn_best_model = grid_search_Knn.fit(All_Company_Train_X,All_Company_Train_Y)
         grid_search_Knn_best_model.best_params_

Out[16]: {'n_neighbors': 27}

In [17]: knn_regressor_model = KNeighborsRegressor(n_neighbors=27)
         knn_regressor_model.fit(All_Company_Train_X, All_Company_Train_Y)
         knn_regressor_predictions = knn_regressor_model.predict(All_Company_Test_X)
         r2Score = r2_score(All_Company_Test_Y, knn_regressor_predictions)
         print(f'The r-square value is = {r2Score:.4f}')
         print(f'The root mean squared value is = {np.sqrt(mean_squared_error(All_Company_Test_Y, knn_regressor_predictions)):.4f}')

         The r-square value is = 0.0374
         The root mean squared value is = 0.9811
```

As per the above hyper parameter tuning, the best K-value or neighbors is 27. Based on that we have calculated RMSE and R2-Score.

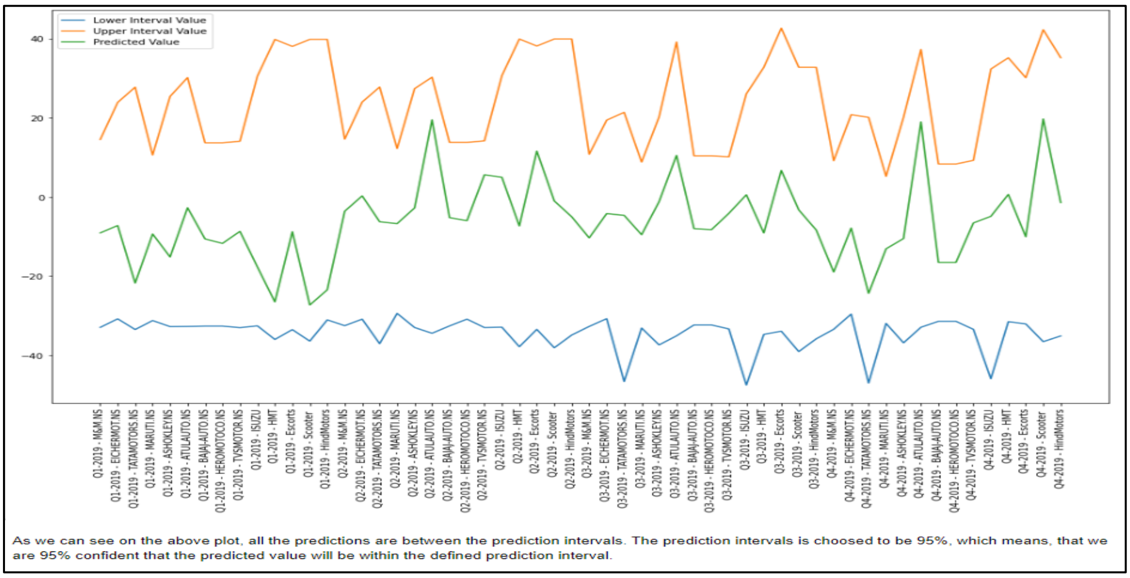
- **Calculate the accuracy of the best model in terms of R^2 and RMSE.**

  The accuracy for the KNN Regressor Model is as follows -

  R-squared value for the model = 0.1120
  Root Mean square error value for the model = 0.9423

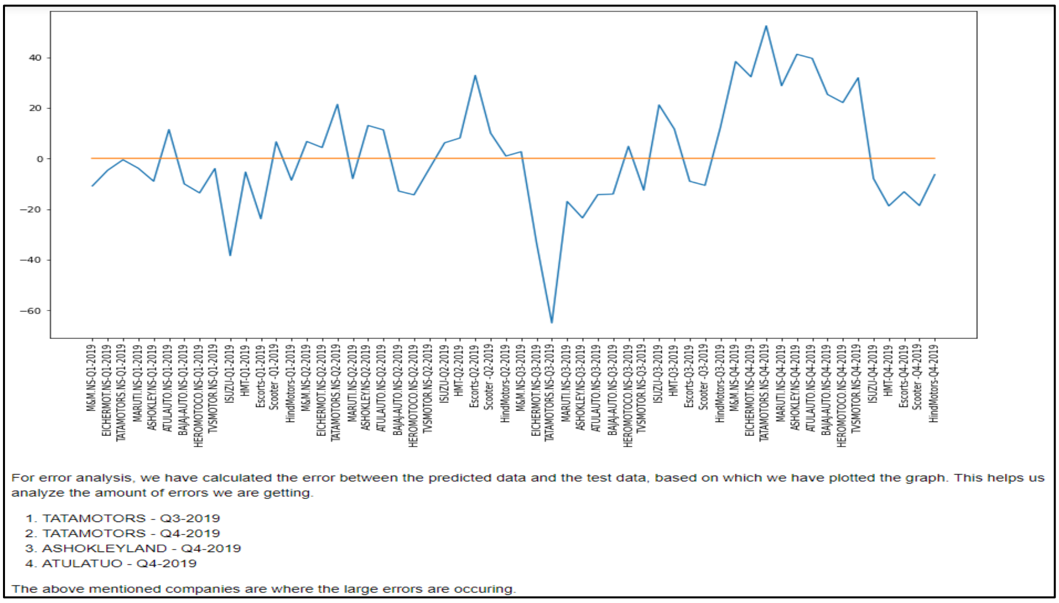
- **Calculate the prediction interval.**

  We have mapped out the prediction interval (refer .ipynp file for calculations) in the below graph wherein the green line is the predicted value and the others are the upper and lower bounds at 95% confidence. As we can see the prediction values are within the range of both the intervals.



As we can see on the above plot, all the predictions are between the prediction intervals. The prediction intervals is choosed to be 95%, which means, that we are 95% confident that the predicted value will be within the defined prediction interval.

- **Do an error analysis to find out where the large errors are occurring.**



For error analysis, we have calculated the error between the predicted data and the test data, based on which we have plotted the graph. This helps us analyze the amount of errors we are getting.

1. TATAMOTORS - Q3-2019
2. TATAMOTORS - Q4-2019
3. ASHOKLEYLAND - Q4-2019
4. ATULATUO - Q4-2019

The above mentioned companies are where the large errors are occuring.

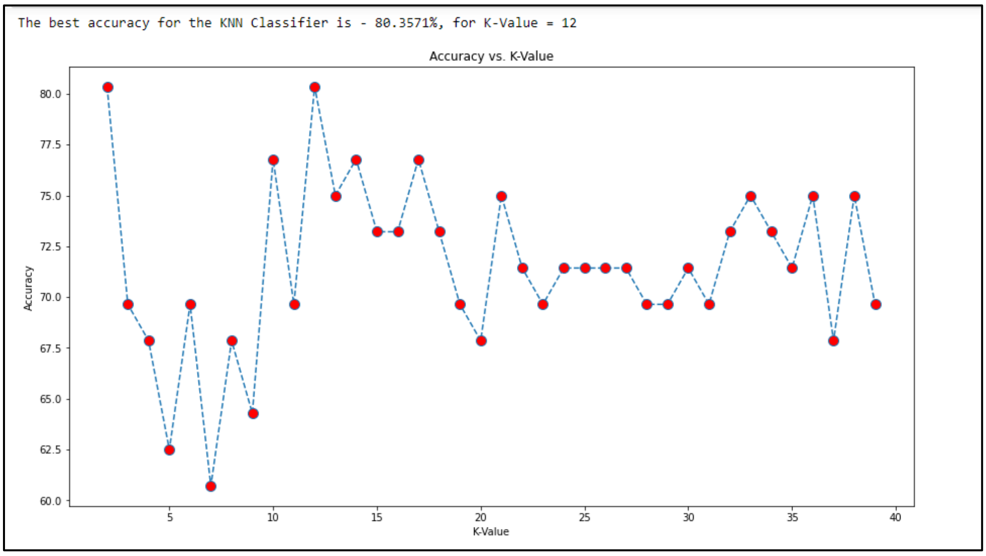
- **Save the model for future deployment.**

  The model has been dumped to the .pkl file, which can further be used for the predictions.

E. Classification Modelling

- **Find the best model between logistic regression, KNN and Decision Tree**.

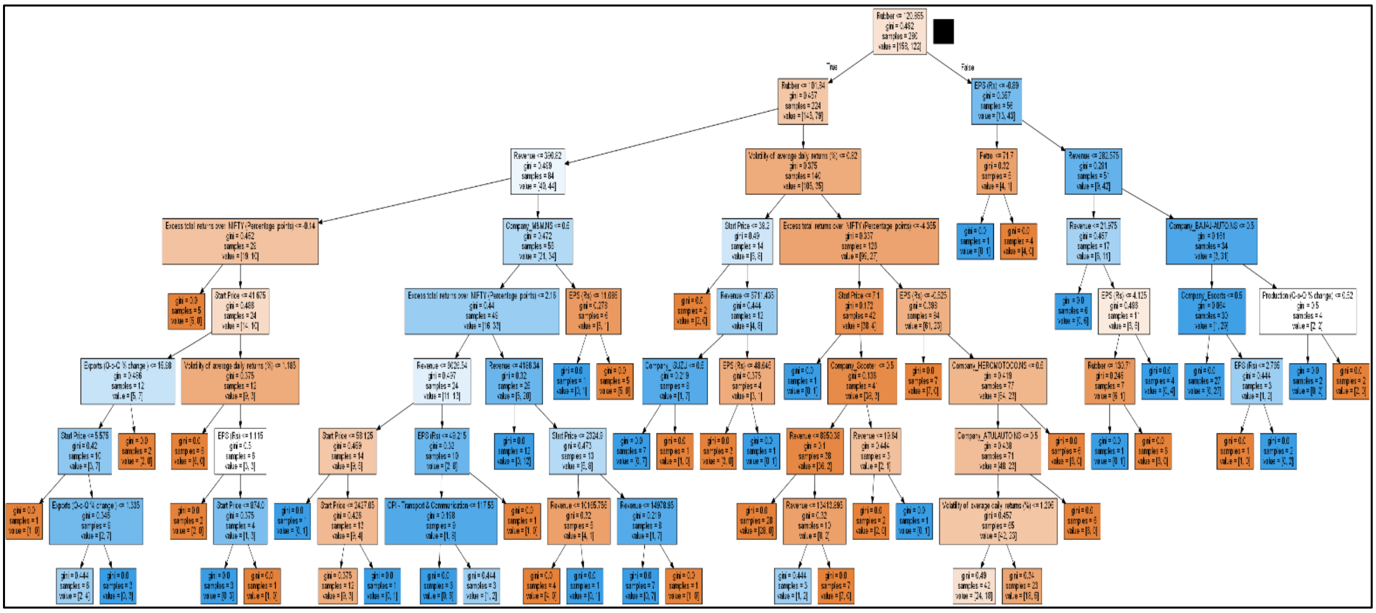
The best fit model between the 3 models above is the KNN Classifier Model with an accuracy of ~80%.



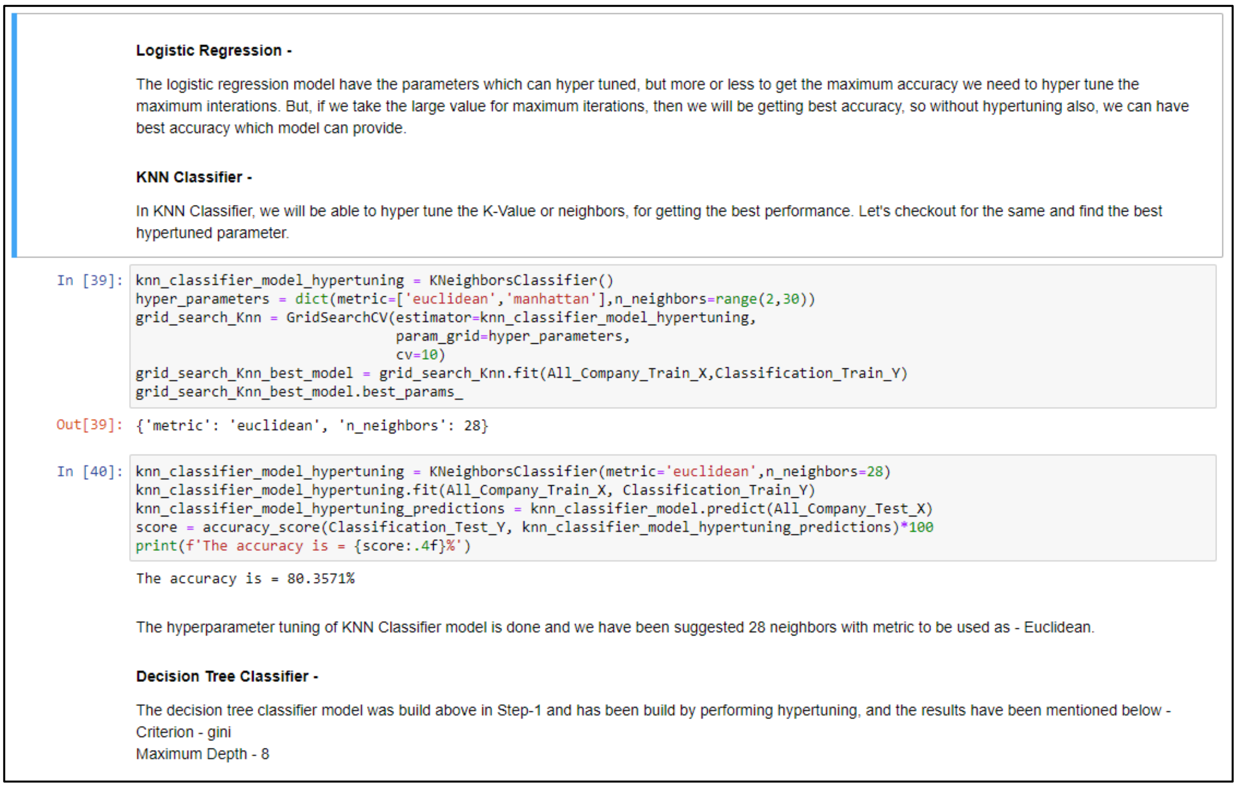
```
The best accuracy for the KNN Classifier is - 80.3571%, for K-Value = 12
```

Further, the other models of Logistic Regression and Decision Tree Classifier Model have given an accuracy of 69% and 51% respectively.

Screenshot of the Decision Tree has been replicated below for reference –



- **Find the best hyper parameters to choose the model.**



Logistic Regression -

The logistic regression model have the parameters which can hyper tuned, but more or less to get the maximum accuracy we need to hyper tune the maximum interations. But, if we take the large value for maximum iterations, then we will be getting best accuracy, so without hypertuning also, we can have best accuracy which model can provide.

KNN Classifier -

In KNN Classifier, we will be able to hyper tune the K-Value or neighbors, for getting the best performance. Let's checkout for the same and find the best hypertuned parameter.

```
In [39]: knn_classifier_model_hypertuning = KNeighborsClassifier()
         hyper_parameters = dict(metric=['euclidean','manhattan'],n_neighbors=range(2,30))
         grid_search_Knn = GridSearchCV(estimator=knn_classifier_model_hypertuning,
                                        param_grid=hyper_parameters,
                                        cv=10)
         grid_search_Knn_best_model = grid_search_Knn.fit(All_Company_Train_X,Classification_Train_Y)
         grid_search_Knn_best_model.best_params_

Out[39]: {'metric': 'euclidean', 'n_neighbors': 28}
```

```
In [40]: knn_classifier_model_hypertuning = KNeighborsClassifier(metric='euclidean',n_neighbors=28)
         knn_classifier_model_hypertuning.fit(All_Company_Train_X, Classification_Train_Y)
         knn_classifier_model_hypertuning_predictions = knn_classifier_model.predict(All_Company_Test_X)
         score = accuracy_score(Classification_Test_Y, knn_classifier_model_hypertuning_predictions)*100
         print(f'The accuracy is = {score:.4f}%')

         The accuracy is = 80.3571%
```

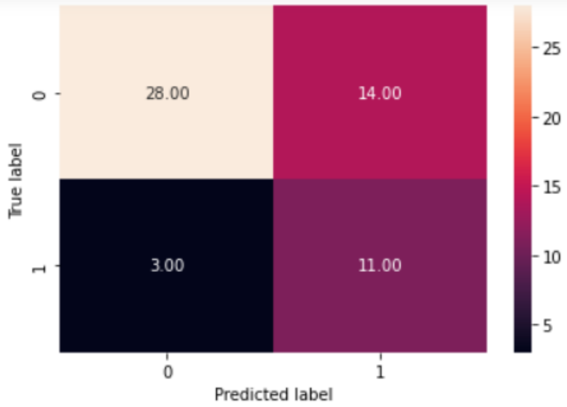
The hyperparameter tuning of KNN Classifier model is done and we have been suggested 28 neighbors with metric to be used as - Euclidean.

Decision Tree Classifier -

The decision tree classifier model was build above in Step-1 and has been build by performing hypertuning, and the results have been mentioned below -
Criterion - gini
Maximum Depth - 8

- **Build a confusion matrix and explain the cost of False negatives and false positives. (Only explain which has higher cost. No need to assign actual cost.)**

Logistic Regression Confusion Matrix –



False Negatives - 3 :-

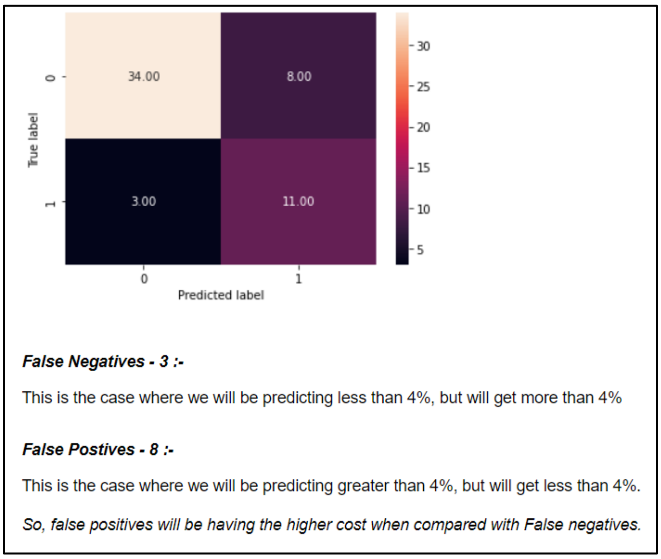
This is the case where we will be predicting less than 4%, but will get more than 4%

False Postives - 14 :-

This is the case where we will be predicting greater than 4%, but will get less than 4%.

So, false positives will be having the higher cost when compared with False negatives.

KNN Classifier Confusion Matrix –



False Negatives - 3 :-

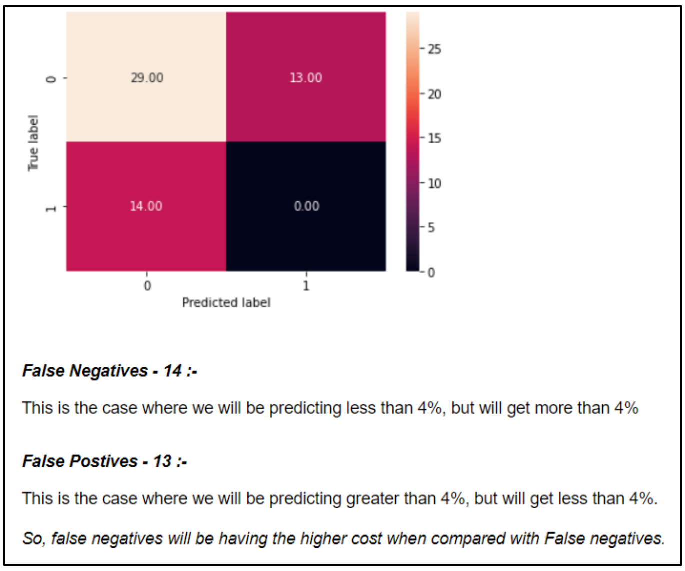
This is the case where we will be predicting less than 4%, but will get more than 4%

False Postives - 8 :-

This is the case where we will be predicting greater than 4%, but will get less than 4%.

So, false positives will be having the higher cost when compared with False negatives.

Decision Tree Confusion Matrix –



False Negatives - 14 :-

This is the case where we will be predicting less than 4%, but will get more than 4%

False Postives - 13 :-

This is the case where we will be predicting greater than 4%, but will get less than 4%.

So, false negatives will be having the higher cost when compared with False negatives.

- **Find out which accuracy metrics will be applicable for this use case and evaluate the model. And provide justification for the accuracy metrics chosen.**

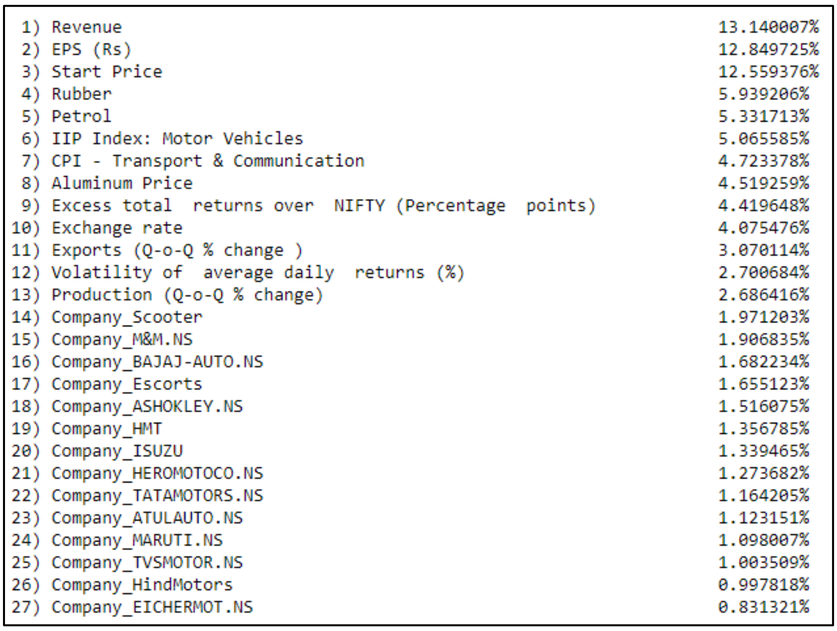
  We have chosen "Accuracy Score" of the model as the appropriate accuracy metric for choosing the model.

  Reasoning -
  If the predictions are for 0 or if the predictions are for 1, both are of equal importance to us. The usecase which we are trying to solve is of stock price change. This can go negative and positive as well, in both the cases we either need to take out the money or invest more in the market, which will help us have the maximum profits.

  Thus, Recall or Precision would not be accurate since, it would either be focusing on one of the use cases (i.e when to invest more or when to take out the money).

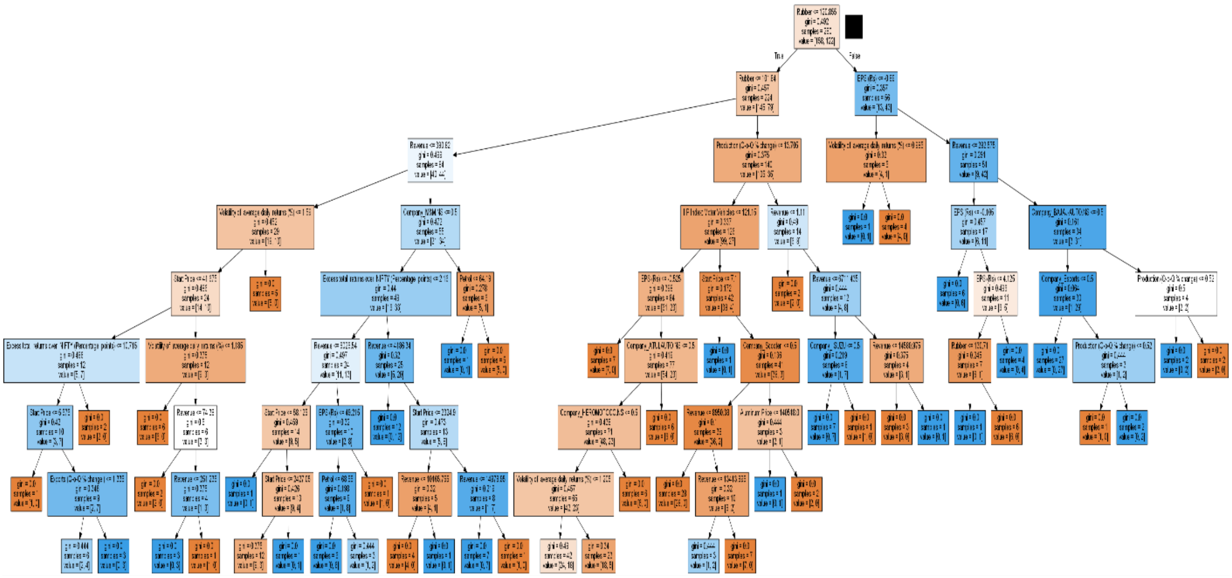
  In this scenario, "Accuracy Score" will help us to have both the things, hence we have chosen accuracy score.

- **Find the feature importance from the Random Forest model.**



```
 1) Revenue                                                 13.140007%
 2) EPS (Rs)                                                12.849725%
 3) Start Price                                             12.559376%
 4) Rubber                                                  5.939206%
 5) Petrol                                                  5.331713%
 6) IIP Index: Motor Vehicles                               5.065585%
 7) CPI - Transport & Communication                         4.723378%
 8) Aluminum Price                                          4.519259%
 9) Excess total  returns over  NIFTY (Percentage  points)  4.419648%
10) Exchange rate                                           4.075476%
11) Exports (Q-o-Q % change )                               3.070114%
12) Volatility of  average daily  returns (%)               2.700684%
13) Production (Q-o-Q % change)                             2.686416%
14) Company_Scooter                                         1.971203%
15) Company_M&M.NS                                          1.906835%
16) Company_BAJAJ-AUTO.NS                                   1.682234%
17) Company_Escorts                                         1.655123%
18) Company_ASHOKLEY.NS                                     1.516075%
19) Company_HMT                                             1.356785%
20) Company_ISUZU                                           1.339465%
21) Company_HEROMOTOCO.NS                                   1.273682%
22) Company_TATAMOTORS.NS                                   1.164205%
23) Company_ATULAUTO.NS                                     1.123151%
24) Company_MARUTI.NS                                       1.098007%
25) Company_TVSMOTOR.NS                                     1.003509%
26) Company_HindMotors                                      0.997818%
27) Company_EICHERMOT.NS                                    0.831321%
```

- **Build a final decision tree with features that have cumulative importance of at least 95% and visualize the decision tree and explain the rules.**



The explanation of the Decision Tree Rules is as mentioned below -

Cases when there is high chance of +ve return

- Company is not Ashok Leyland & M&M & Exports < 1.335 & Revenue < 390 & Rubber < 101
- Company is not escorts & Bajaj Auto & revenue < 282 & eps < -0.89 & Rubber <120

Case when there is high chance of -ve return:

- Company is not M&M & Start price < 7.1 & IIP motor index < 121 & Production <13 & Revenue < 390 & Rubber < 101

F. Conclusion

Summary of Accomplishments

1. Highest accuracy of 80.36% observed in KNN Method for the Classification Model
2. Prediction for the test data were observed to be within the Prediction intervals

Key Takeaways & Lessons

1. Importance of Parameters / Indicators – During the building of the Model, we realized that there are multiple factors affecting the prices of an industry / company. While some of them are internal related to respective company workings (like Revenue, Profit, etc), there are some larger macro-economic factors too which affect the share prices (like CPI, IIP, etc)
2. During the model building we concluded that Linear Regression is not a good model for the purpose of prediction in this specific use case, while KNN seems to be a better fit.
3. For the purpose of this model, we concluded that Accuracy is a better metric to rely on than Precision or Recall score.

G. Deliverables

i) Jupyter Notebook (.ipynb file) as the primary code file of the model
ii) Saved Model (.pkl file)
iii) Excel File (.xls) for the input dataset
iv) PDF Document (.pdf) as the Project Report

H. References

- CMIE Industrial Outlook Database for Industry-wide Economic Parameters
  (https://industryoutlook.cmie.com)
- Yahoo Finance for Historical Share Prices
  (https://finance.yahoo.com)
- Business Standard for Company specific Financial Data
  (https://www.business-standard.com/category/companies-results-1010301.htm)
- Index Mundi for Commodity Prices (Metal & Fuel)
  (https://www.indexmundi.com/india/)