**CSI 5335 Project**

**Who Wants to be a Millionaire?**

**Report**

**Abstract**

This document gives a brief report on the data analysis, the set of data used, and the criteria generated for the project.

**Methodology**

**Data Preprocessing.**

Data for 100 different companies is used for training the model. The data covers the stock transaction of all the companies for the years 2017 and 2018. There are multiple variables in the dataset for each company and they are the date, close, volume, open, high and low.

The columns Open and Close represent the starting and final price at which the stock is traded on a day. High, Low and represent the maximum, minimum of the share for the day.

Volume is the number of shares bought or sold in the day.

The data for 2017 is used for training the model, it is divided into two, 70% is used for training and 30% is used for testing.

All the data for 2018 is used for conducting transactions on the second part of the project.

**Steps in extracting data:**

* The data is first read from an hdfs file system into spark rdd and then converted to dataframe in scala.
* Some data is in double quotes and are removed.
* Some of the data is separated with commas whiles others are separated with spaces, the needed data is extracted.
* Data with spaces in between rows are also handled.
* The names of companies are extracted from the files. (It is assumed in this project that the respective file names are the names of the companies)
* Some of the data have spaces in between rows and this throws an exception especially in scala so spaces in between rows are removed.
* The date is also formatted in the same way.

**Program 1**

**Model Used and Parameters:**

The model used for predicting is random forest, this model is used because it consists of multiple decision trees and do well compared to a single decision tree.

The random forest used is set up with a tree depth of 10, number of decision trees used is 5, these are the only hyperparameters set for this model. The default values were used for all the remaining parameters.

**Features Extracted:**

To get a model that generalizes well irrespective of the company, the average values of the close, open, volume, high and low for every company is taken and used as the value for that day. In other words, each day has a single close, open, high, low and volume based on the mean of all the companies (I group the stocks by date and find the average over all the companies).

After this, features are extracted from these mean values, to train the model. These features are the rolling mean of the previous close, open, volume, high and low over a 3 days window period.

For the first days, the average of the respective columns is used. For the second day, the previous values (actual values in the previous day is used), for the third day the average of the first and second days is used and vice versa.

From several experiments run, it was observed that using the components of the dates, as features also some significant improvement in the model and hence the day, month and year was extracted and used as training features.

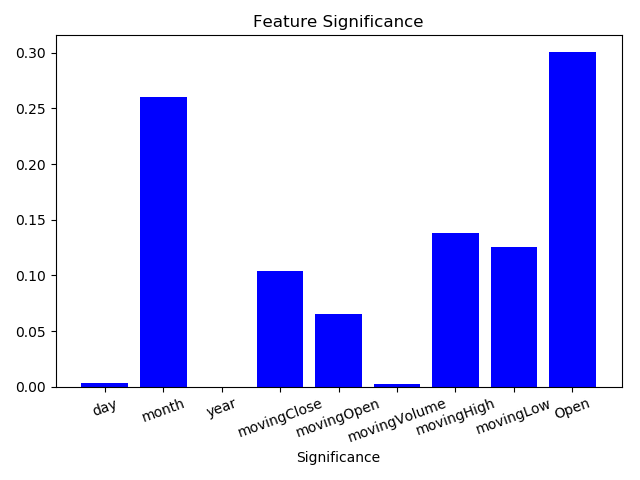


Fig 1 A bar chart showing the contribution of each feature to the model’s decision making.

Fig 1 shows a graph of the features generated and their contribution to the model in making decisions. To explain further, the day, month and year represent the date for the stock. The moving close, moving open, moving volume, moving high and moving low values are the average of the close, open, volume, high and low over the last 3 days period. The open is the open stock price.

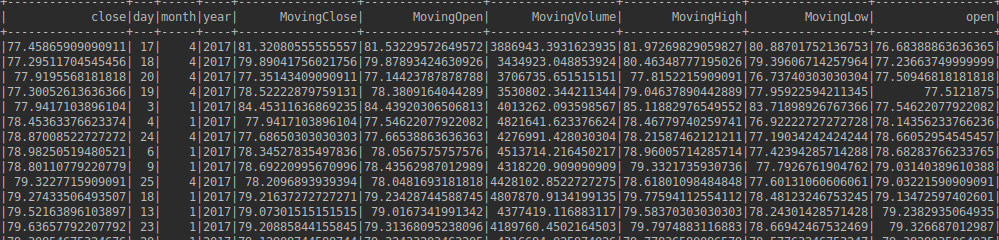
From figure 1, the opening stock price is the criteria, used the most in making decisions, followed by the month. This is not surprising since by skimming through the data, the closing stock price is mostly close to the opening stock and they have some correlation.

The month is the feature with the next highest impact, and this loosely speaking, this could mean that the stock prices varies significantly for different month periods.

The year makes no contribution to the model, and I believe this is particularly because the model was trained on a data with the same year values, Ideally, this feature should be removed from the training since it makes no contribution to the model selection and even if the data is trained over different years, it will only have a good impact if the prediction is for same year. Though it takes space and increases training time, it is kept in there just for analysis purposes.

The final model generated is saved and later loaded for program 2.

**Results**



**Fig 2.** picture **of the features used for training the model.**

The figure above shows the features used in training the model, the company itself is not used as a feature in other to generate a model which generalizes well on all data.

**Program 2**

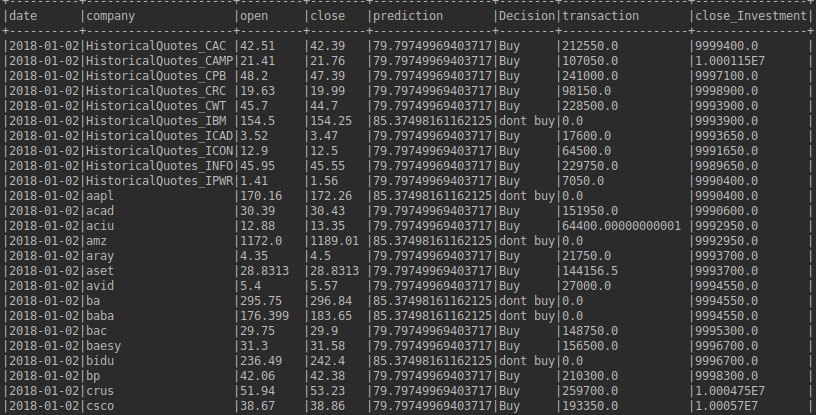
In the second program, the model generated from program 1 is loaded and the features are extracted in the same way as program 1 except that the average for each day is not taken. Each company’s stock is treated separately because we must determine if we will buy that company’s stock or not. The average over each Colum is taken over the 3 days period just like in program 1. The date components are extracted as well as the opening stock is also taken.

These features are fed into the saved model and the model returns a prediction of the closing stock. These predictions are compared to the opening stock and if they are higher than the opening stock, the stock is bought if there is enough money to buy.

1000 amount of the stock is purchased for every stock, at the close of each day, we evaluate all purchased stocks by comparing the actual closing stocks to see if we made a profit or loss and add the amount to the investment amount remaining(can be negative or positive depending on whether we made a profit or loss).

All stocks which are predicted as buy by the criteria are bought if there is enough fund, and a fixed amount 1000 is always bought.

**Results and Analysis**



**Fig 3 top 20 transactions conducted based on the model and criteria generated.**

Fig 3 shows the first 20 transactions made with the guidance of the model generated. The decision column is the suggestion, based on the prediction of the model. The transaction column is the amount of money spent on buying the stock and the close investment is the amount of money left at the end of the day when the stock purchased sold.

**Notes on how to set up the project is in “notes\_for\_project” file in the folder.**