Project Report

IE582 – Statistical Learning for Data Mining

Authors:

Anıl Turgut
Oya Hoban

15 January 2024

1.) INTRODUCTION

Predicting stock prices can help companies, investors, and decision-makers to plan and take action in a profitable way (1). Stock prices are volatile, dynamic, and unpredictable, and it can be a challenge to forecast their dynamic in the future (2). Machine learning models are used for decision making in finance, for various areas; predicting stock prices is one of them. Machine learning models can capture stock prices' dynamics by taking into account factors effecting them, and help with predictions (3). The aim of this project is to predict future prices of 30 different stocks in Borsa Istanbul by implementing data analysis and Machine Learning models on the hourly price data of stocks. In other words, a study will be carried out to predict the hourly data of 30 stocks with different ML algorithms by performing preprocessing, descriptive and predictive analyzes on the historical data given for the stock price prediction problem. The first phase of the project is building the prediction model, and the second is a competition where participants will be making a 10-hour stock price prediction for the stocks included in the first phase, every day starting from 25th December 2023 until 12th January 2024.

2.) APPROACH

To build forecasting models using R, on 30 different stocks in Borsa Istanbul, the operations done are as followed:

- Past hourly stock price data of each 30 different stocks in Borsa Istanbul are obtained as .csv files. It will be an automated system since the data is renewed after a day.
- Data is analyzed to fully understand which operations are required to make it ready for building models. Different visualizations and approaches such as clustering is executed to understand the underlying relations of stocks.
- Feature engineering is performed on the data to expand the features to increase the accuracy of the models. Also using an R library named 'Quantmod', relevant features that can help capturing the data structure are added to the existing data.
- Data is split into training and test samples. Multiple machine learning models are fitted to the train data. Parameter tuning is applied to the Machine Learning models to obtain the best performing version of the models on the data.
- The Machine Learning model with the highest performance on the test data is selected as the final model to use as prediction model.

- After selecting the ML model, an automated system is constructed to predict day t+1 after training the model with data up until day t.
- Once the competition phase of the project has started, hourly stock price prediction for each stock for the upcoming day has been submitted, starting from 25 December 2023 until 12 January 2024.

2.1) Explaining the Data

Hourly average prices of selected stocks from Borsa Istanbul, encompassing about 4 years, were used as raw data. The raw data consists of three columns: '*timestamp,'* '*price'* and '*short name'*.

Timestamp feature stores the combination of date and hour information, the date spans from 02-01-2018 to 22-12-2023; and the hour spans from 9:00 AM to 18:00 PM incrementing hourly, for each date.

Short name feature is for the abbreviation for each selected stock.

Price feature stores the average price for the corresponding stock at the specific time point, which is stored in *timestamp*. Price is the target that this study aims to build forecast models on.

2.2) Feature Engineering & Yahoo Finance Library

Including all available data and adding new features can be helpful for implementing stronger forecasting models. This can be done by using existing information in the raw data or using external data that is relevant to the targeted outcome. The year, month, day, and time features are extracted from the *timestamp* feature that is available in the raw data.

Moving average with 6 records of each stock price is calculated and added as a new feature called 'forecast_ma_6'. This feature captures the trend in the price data. If there are no previous 6 records in the current record, then its forecast will be set as its original price.

Related data about the stock prices can be searched for and added to the analysis, to have an improved model. Quantmod library (4) is useful for this purpose. It is an R library that provides a framework for quantitative financial modeling and trading. It can be used to extract related and meaningful information about stocks, such as daily opening & closing price, highest & lowest price, volume. The library is used in this study to extract daily volume information for each stock. Different than the volume information of stocks, trend and expectation visualizations are demonstrated to find an insight about the stock's behavior.

2.3) Descriptive Analysis of the Data: Visualization & Clustering

Before building any model, the data needs to be understood. For this purpose, descriptive analysis and visualizations are conducted on the final data obtained after feature engineering.

The final data has 438,858 rows and 9 columns. The variable names in the data are timestamp, short_name, year, month, day, hour, volume, forecast_ma_6 and price.

The price distribution for each stock is visualized in **Figure 1.** It can be observed that 'FROTO' and 'PGSUS' has high change throughout years compared to the rest of the stocks.

Average price of all stock prices in each month is calculated and visualized in **Figure 2**, to observe the trend throughout the months. It can be observed that there is an increasing trend throughout time.

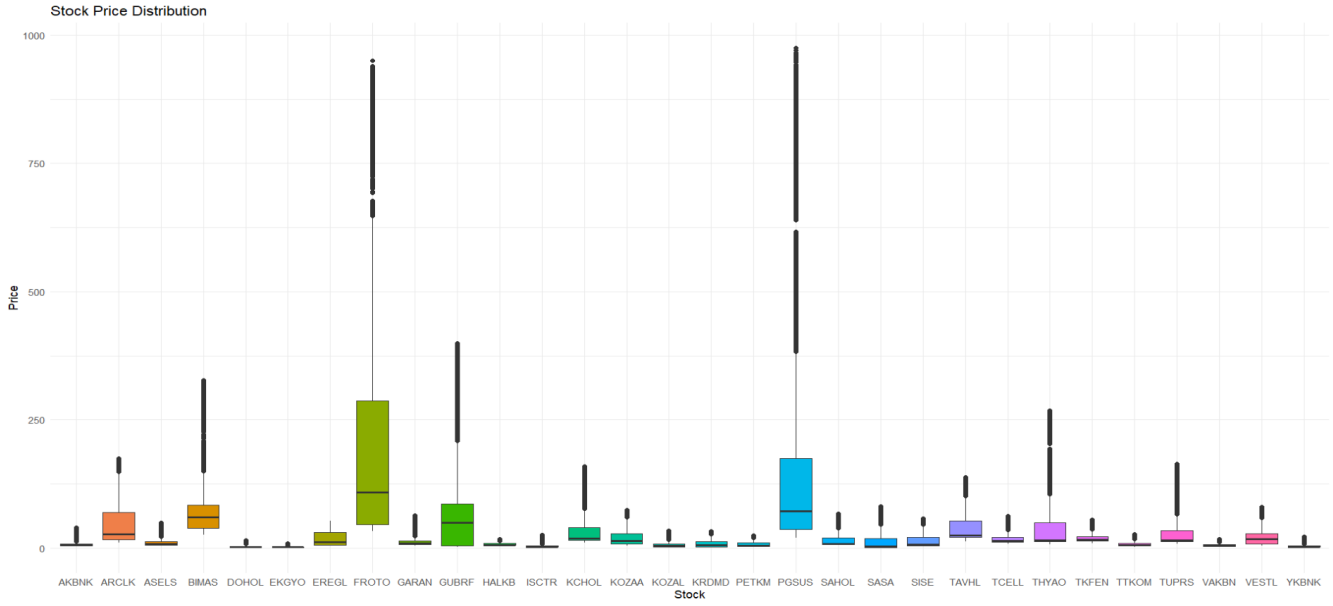

Figure 1. Stock Price Distribution for Each Stock

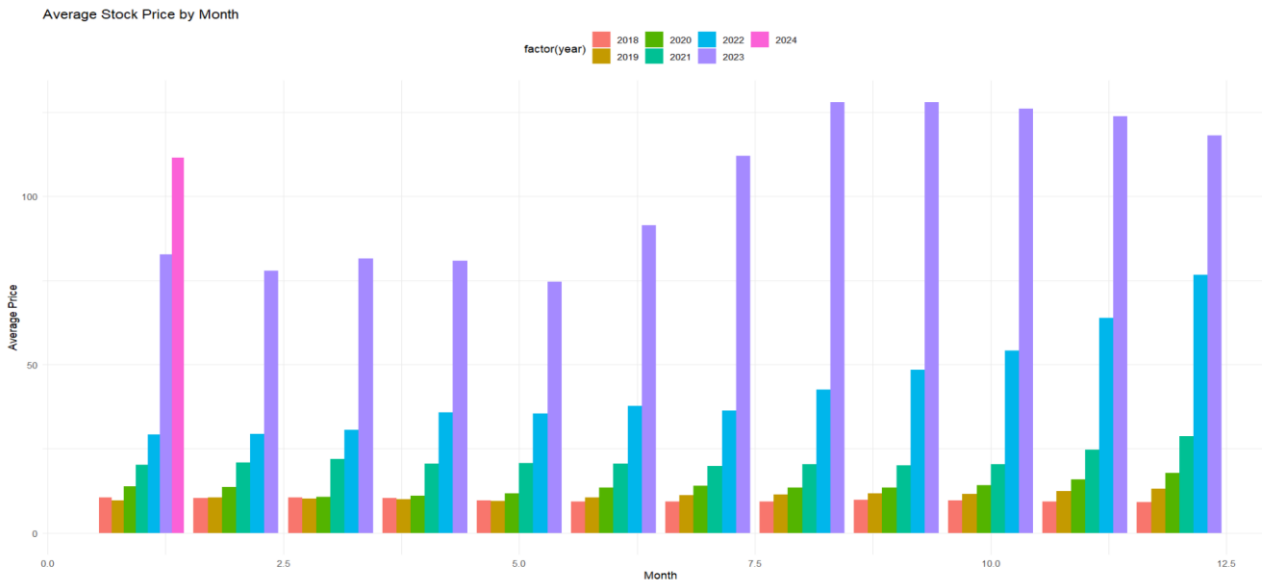

Figure 2. Average Stock Price by Month

Anomaly detection has been carried out for each stock. It has been assumed that an anomaly is detected if the price value in each date don't fall in the confidence interval, which is obtained by using the following formula:

$$\textit{The mean of the price data} \pm 3 \times \textit{Standard deviation of the price data} \quad (1)$$

It has been inferenced that none of the stock price data has any anomaly value and the stock prices from data provided is demonstrated as following:

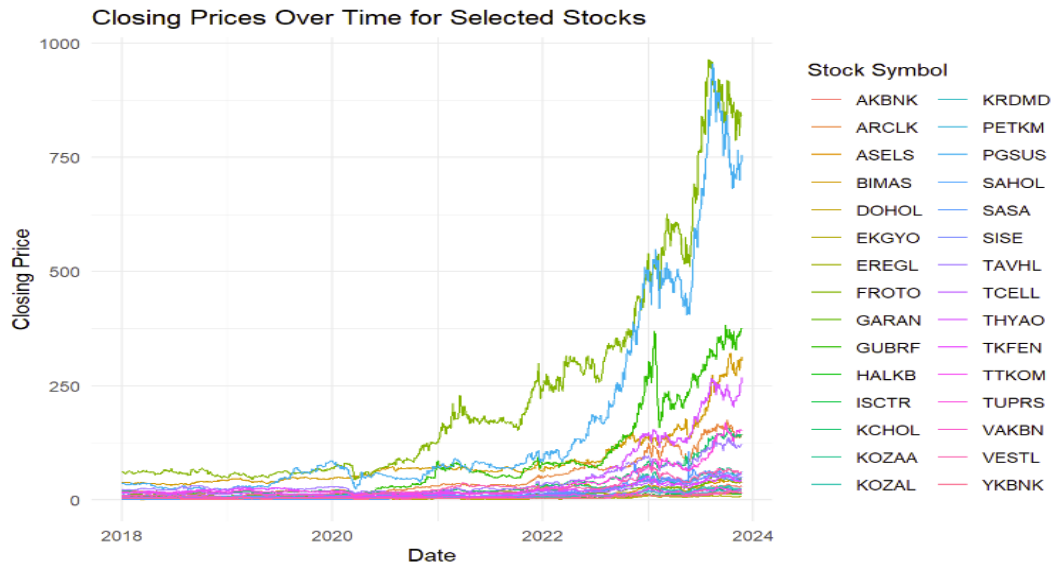

Figure 3. Closing Prices Over Time for Stocks

To observe the relationship between the volume variable and the price variable, the variables movement have been visualized for each date, using Quantmod library's functionalities. 'THYAO' and 'SISE' are selected as examples to examine (**Figure 4**).

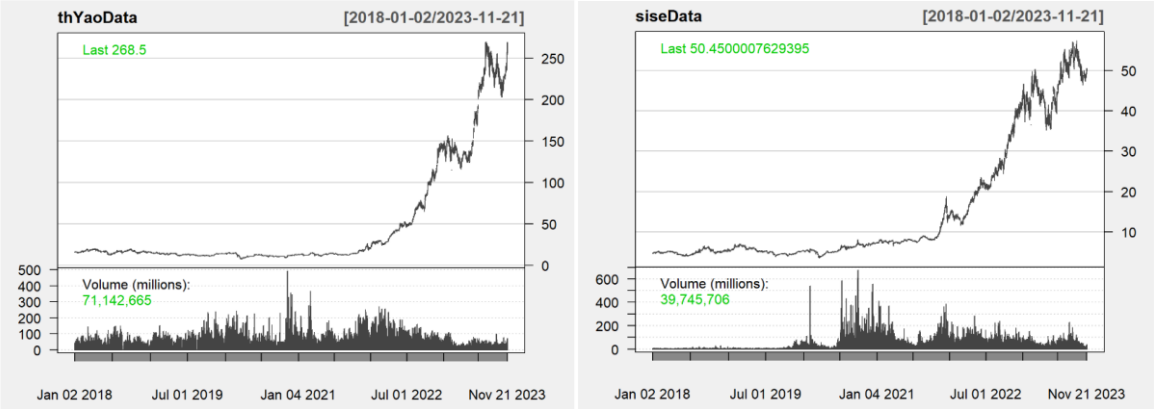

Figure 4. Volume and Price for THYAO and SISE

When there is a peak in volume, a peak is observed in price data, too. Hence, it can be argued that the volume and price data are correlated.

Clustering is done to examine the similarity/dissimilarity between stocks. Clustering analysis has shown that, some stocks are stored in same clusters with minimum distances:

- VKBANK and HLKBNK
- THYAO, SISE, SASA, EREGL
- ASELS, TUPRS, YKBANK, AKBANK

These stocks can be used to predict the price trend of each other.

2.4) Machine Learning Models and Parameter Tuning

Different machine learning models are fitted to the data to select the best performing one to be used for prediction. The data is split into train and test data; the models are fitted into train data, and the models' prediction accuracy is tested on how well they can predict the test data. Weighted Mean Absolute Percentage Error (WMAPE) is used as evaluation metric. WMAPE provides a normalized error rate across all predictions, allowing for a fair comparison between different forecasting models or different data sets. The models tested are decision tree algorithm, random forest algorithm, ARIMA and gradient boosting machine algorithm.

The 'THYAO' stock is selected as an example to try different models.

Parameter Tuning has been conducted to decide on the parameter values to have the best performing model. A manual cross validation is implemented to tune the hyperparameters, for decision tree model, random forest model and GBM model.

For decision tree, parameter tuning has been carried out with assuming the tuning hyperparameter will be only the minimal number of observations per tree leaf and setting complexity parameter to zero and minimum number of observations to split as the twice as the minimal number of observations per tree leaf. It has been decided that the decision tree model with minbucket = 5 is the best performing decision tree model having the smallest RMSE.

For random forest model, parameter tuning has been carried out with assuming the tuning hyperparameter will be only the effect of the ratio of the number of features evaluated at each split (mtry) and setting other parameters as J=500 and the minimal number of observations per tree leaf=5. It has been observed that, the model converges when mtry = 4 (given ntree = 500 & nodesize = 5) is the best performing random forest model in terms of having the smallest RMSE.

For GBM model, parameter tuning has been done focusing on the depth (interaction.depth), the number of trees (n.trees) and the learning rate (shrinkage) of the model. It has been decided that the best performing GBM model has the parameters (Shrinkage = 0.1, depth = 5, n.trees = 500).

2.5) Predicting Future Stock Price

Different machine learning models have been built on the training data. The best performing models for each model have been decided after conducting parameter tuning. To test the performance of each model on unobserved data, predictions have been done on the test data using each model. The aim of this step is to decide on the model with lowest error and use it for predicting the stock price levels during the competition phase of the project. In addition to only looking at the training/test errors of the established models, the model performances in the live system were also evaluated and analyzed to prevent overfitting, and model selection was made accordingly.

RESULTS

Different machine learning algorithms' performances are evaluated on their prediction accuracy performed on the test data. With the training data provided, 4 different Machine Learning and AutoRegressive algorithms (Decision Tree, Random Forest, GBM, ARIMA) were tuned with different parameters using the RepeatedCV method and the results were

analyzed. The metrics used in this analysis and comparison were continuous target variable metrics due to the supervised regression problem. Classical regression metrics such as Mean Absolute Error (MAE), Weighted Mean Absolute Error (WMAE), Root Mean Squared Error (RMSE) and R2 were examined in detail and the performances of the tuned models were compared.

In addition, a metric that shows the weighted error percentages of stocks within the scope of this project was used as the main metric and thus is not affected by scaling. Weighted Mean Absolute Percentage Error (WMAPE) has become more important among the metrics used and has become a metric that provides effective insight for interpretation and comparison.

Within the scope of these studies, we can summarize the performances of each tuned model in the same dataset and environment as in the table below:

Algorithm	**WMAPE**	**MAE**	**RMSE**	**MAPE**
Decision Tree Algorithm	0.982	0.466	1.117	0.958
Random Forest Algorithm	0.728	0.351	0.844	0.682
ARIMA (Autoregressive Integrated Moving Average)	97.152	1615.511	1787.701	7327.988
GBM Algorithm	1.049	0.500	1.069	1.114

- ARIMA is an important technique for both prediction and feature engineering under the stock prediction context. It tries to catch a trend with the given data, considering the stock price. As a result of the analysis, we made in this project, although ARIMA did not perform well in training data, it showed not bad results in live prediction. However, it can still be said that it ranks last in terms of performance compared to other algorithms.
- Although the decision tree algorithm brings good results in terms of algorithm performance, as seen in the metric table, it is an algorithm that tends to result in overfit, and although techniques such as pruning were tried to be used in this project, it could not reach the desired performance.
- Random Forest is the algorithm that achieves the best metric results, but although this algorithm does not result in overfitting, it seems to be affected by feature engineering (especially related to the QuantMod library). Although it gave good results with

training/test data, it was not selected because it produced unrelated results for some stocks in live prediction.

- GBM is an algorithm that creates a strong learner by iteratively performing weak learning construction and improving these weak learners in each iteration. In this project, the performance metrics are as satisfactory as Random Forest and Decision Tree algorithms, it does not overfit like Decision Tree, and it does not produce bad values for some stocks in live results like Random Forest. For this reason, this algorithm was chosen and predictions were made through the tuned GBM algorithm.

With the GBM algorithm used, 300 stock predictions were made live every day for 30 different stocks for the next day. Generally good results (low WMAPE) were achieved, on some days there were bad performances for some stocks. The reason for this is that the stock market and stocks are open to many parameters and manipulations.

DISCUSSION AND CONCLUSION

In this project, data analysis and feature engineering are applied to hourly stock price data for 30 stocks in Borsa Istanbul, to make the data ready for building prediction models. Different Machine Learning models are fitted while implementing parameter tuning and models are tested to select the best performing one to use for prediction in the competition phase. GBM is selected to use for predictions in the competition phase.

Number of Machine Learning models tried in the project can be increased to have a higher possibility of selecting the right model.

During the competition phase, GBM seems to capture the trend in the data. Even though it comes close to predictions of the next day stock prices, there is still room for improvement. The addition of external data such as the frequency of the stock price name search from Google trends could be helpful for improvement in the model.

CODE

In this report, the approaches and analyzes made in the project and the performances found as a result of these approaches were examined in detail. In addition, the repository containing all the studies and codes in detail can be accessed from the link below:

https://github.com/BU-IE-582/fall-23-anillturgut/tree/main/files/Project

REFERENCES

1. Kumari J, Sharma V, Chauhan S. Prediction of Stock Price using Machine Learning Techniques: A Survey. 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N); 2021 17-18 Dec. 2021. 281-284 p.

2. Demirel U, Çam H, Ünlü R. Predicting Stock Prices Using Machine Learning Methods and Deep Learning Algorithms: The Sample of the Istanbul Stock Exchange. GAZI UNIVERSITY JOURNAL OF SCIENCE. 2020 07/01;34. doi:10.35378/gujs.679103.

3. Obthong M, Tantisantiwong N, Jeamwatthanachai W, Wills G. A Survey on Machine Learning for Stock Price Prediction: Algorithms and Techniques. 2020.

4. Ryan JA, Ulrich JM, Thielen W, Teetor P, Bronder S, Ulrich MJM. Package 'quantmod'. 2015.