Definition

1. Project Overview

Stock market, Forex, cryptocurency trading, etc are some of the options where people can invest money into in order to generate some profit. No matter which option from listed above is chosen, the main idea of the a long-term or a short-term investment remains the same: buy an financial instrument at low price and sell at higher price. Although the main trading principle seems to be quite simple, there are a lot of people who not only didn't make any profit during trading but lost a lot of money. There are many reasons that could lead to such a sad outcome. Making profitable trades in the stock or in any market is a very difficult task because there a huge number of factors that influence the direction of the price such as economic condition of the company(bringing to the market new products or services will strength the company's stock price), political situation inside the company(new CEO can rebuild/ruin the reputation of the company, a corporate reorganization), investors/customers sentiment can be positive or negative(that also affect the price), even economical or political decisions of the government can impact the performance of the company(overseas sells) and many other factors. As a result, the stock markets are subjected to a very quick price change generating random fluctuation and high volatility of the price turning profitable investments into a challenging task. Despite stock's market complexity, hedge funds, investment companies and individuals input a huge amount of time, money and efforts in order to understand the market behavior for placing profitable investments. A big number of fundamental indicators(price-to-earnings ratio, earnings per share, to name a few) which describe economical and financial condition of a company, were invented. I addition to that, a wide range of technical indicators (Relative Strength Index, Moving Average Convergence Divergence to name a few) were invented for better understanding of dynamic of the stock price. Also, with the rise of the social media, it's sentiment analysis plays an important role in making investments based on people's thoughts and feelings about certain company. Gathering, processing and analyzing such huge amount of data seems impossible even for a big group of trained and skilled people. This is where computers come into play. For the last 20 years, computing power and data storage made a huge step forward allowing to gather, store, process and analyze enormous amount of data at a high speed and at a low cost. Cloud technologies provides an easy access to scalable clusters of computers which are affordable for companies and individual researchers. Along with computing power, a big breakthrough was made in the field of machine learning and artificial intelligence. Self-driving cars, computer vision, voice recognition, text translation, medicine(robots-surgeons, disease detection), recommendation systems are few fields where machine learning shines. The investment field was not left aside. Now-days, computers not just placing buy/sell orders. Algorithms are scanning thousands of stocks to find the most promising ones in terms of investment, they perform pattern

recognition by scanning historical data, they can predict predict future price or market direction[1,2], they can select the most important indicators out of hundreds available, they can perform trading in automated way (algotrading) without emotions! As a result, companies and individuals take advantage of machine learning capabilities to builds semi-automated or fully automated trading systems to generate profit for customers or to get a financial freedom.

2. Problem Statement

The goal of this project is to predict the stock price movement such as up or down for a short-term investments(3-15 days) using machine learning. To tackle this problem two approaches will be taken:

1. Prediction of the stock price movement based on Linear Regression. Open price will serve as input data and Close price(for the same day) is a target which will be predicted. But making a decision about price movement(up or down) based on predicted Close price as continuous output may lead to unfortunate outcome. For example, if at time t the Close price for a stock is $605 and for t+1 is $608 but if the predicted Close prices for t is $602 and for t+1 is $601, then there is a problem with a trend direction although the prediction error is small. Real trend is up($605 < $608) but the predicted trend is down($602 > $601). So, no matter how small the error is in Linear regression, the continuous value is not desired format to succeed in the trend prediction. A way to get this around is to convert the continuous data into categorical one. Regression problem is converted into a classification task. If Price(t+1) > Price(t) then it is an uptrend(label 1), if Price(t+1) < Price(t) then it is a downtrend(label -1). The same procedure applies to the predicted prices. Having both sets of labels an accuracy score can be calculated taking into account the number of matched and mismatched labels in the sets.

2. Prediction of the stock price movement based on Classification. In contrast to the first approach, the classification algorithm will use multiple input features and predict the price trend(based on Close price) for the next day. If Price(t+1) > Price(t) then it is an uptrend(label 1), if Price(t+1) < Price(t) then it is a downtrend(label -1). The input features will be calculated based on Open, Close, High, Low prices, on their ratios, to be precise. In addition some technical indicators will be used as input features too.

After getting the performance metrics for both models trained on the same training and testing dataset, a conclusion can be made on which model predicts the price trend the best.

3. Metrics

Both approaches belong to a classification problem. One of the frequently used metrics for classification is accuracy score which measures a fraction of the classifier's predictions that are correct. The accuracy score can be calculated by the following formula:

$$\text{Accuracy} = \frac{\text{True positive + True negative}}{\text{Total dataset}}$$

In some cases, when dataset is unbalanced, accuracy is not a good choice as a metric but for this project is it suitable since many stocks have almost 50/50 ups and downs.

Analysis

1. Data Exploration

Financial data (open, close, high, low prices) will be automatically acquired from Yahoo Finance by a build-in Python functionality. Both machine learning models will use a single stock ("Amazon") but a user can choose any stock available from Yahoo Finance. A time period for a training dataset 2010-2015 but a user can choose any time period for training. The predictions will be made within 3-15 days right after the last date in the training set. Some technical indicators will be used as inputs for a classification. All indicators will be calculated with TA-Lib library. The following indicators will be included:

EMA – exponential moving average

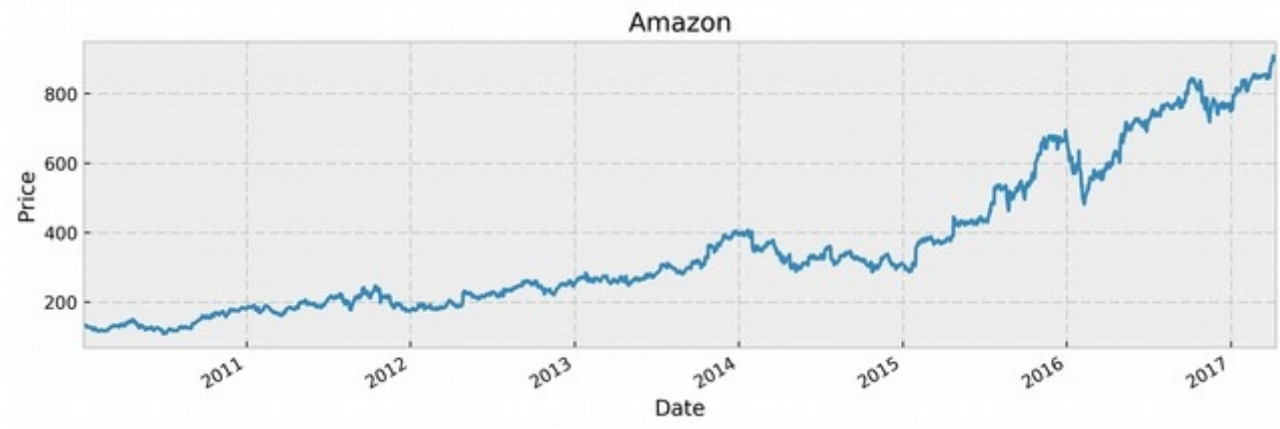
SMA – simple moving average

BBANDS – Bollinger Bands

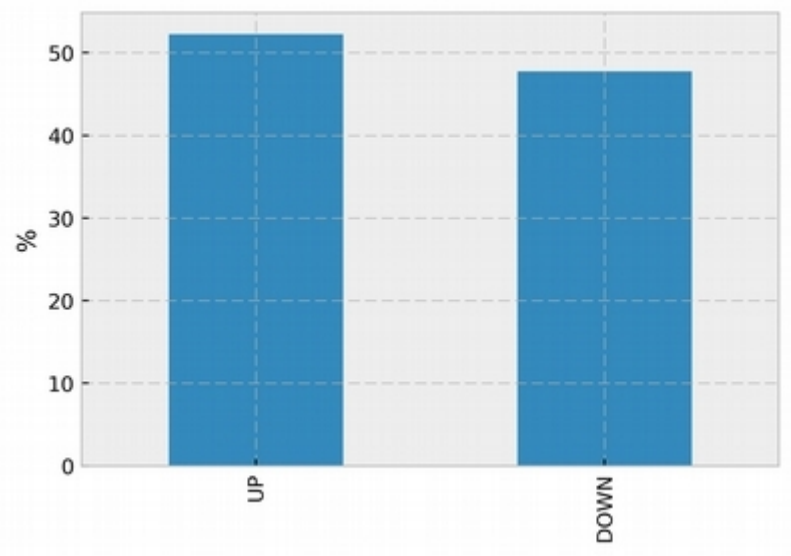
RSI – relative strength index

MACD – moving average convergence divergence

2. Exploratory Visualization



The plot above displays Amazon stock price from 2010 through 2017. As was previously mentioned, accuracy score is going to be a metric of choice but in some cases, for example when data is unbalanced, accuracy score is not preferred. The plot bellow confirms that the data (Amazon) is balanced and accuracy can be use as a metric. Many stocks have almost 50/50.



3. Algorithms and Techniques

The first step of the first approach is to perform a Linear Regression on Open price in order to predict the Close price. Before performing a regression, the dataset was divided into a training and testing datasets. Since we are interested in predicting the price trend, the continuous values of real Close and predicted Close prices are converted into labels, 1 and -1. A positive label represents an uptrend(next value is greater then the current value) and a negative label indicates on a downtrend(next value is less than current value). Having both labeled dataset, an accuracy score is calculated.

The second approach is based on Classification problem with multiple input features. To generate a feature matrix, in addition to Open, Close, High, Low prices some technical indicators are calculated too. The target variable for this classification problem is a price trend based on the Close price with was converted to labels(1 and -1) the same way as in the first approach as well as dividing data into the training and testing dataset. After training the model, an accuracy score is calculated based on true labels and predicted labels. Results are compared with the first approach.

4. Benchmark

There are a couple of options on how to create a benchmark model for the classifiers. One of them is a “majority vote” model which assigns class labels for prediction by taking into account a majority class found in the training set. Another option is to use one of the classifiers used for prediction as a benchmark model.

Methodology

1. Data Preprocessing

There is no need for preprocessing data for Linear Regression but predicted continuous values of the Close price and values for the real Close price were converted to labels(1 and -1) before calculating the accuracy score. For the classification task raw prices(Open, Close, High, Low) were used to generate features and some technical indicators were calculated.

2. Implementation

Work on this project consists of the following steps:

First approach:

1. Get stock prices(Amazon) from Yahoo Finance and select Open and Close prices.
   Open price is input feature(x) and Close price is a target variable(y). Split both columns into training and testing datasets(x_train, y_train, x_test, y_test) but preserving the sequence.
2. From sklearn import a regression model. Fit the model with the training datasets(x_train, y_train) and run prediction on x_test to get y_pred. Optional: get R2 score and plot true vs predicted Close prices.
3. Convert continuous values in y_test and y_pred into labels 1 and -1 by subtracting the current price from the next price(Price(t+1) – Price(t)). If the difference is positive – label is 1, if negative – label is -1.
4. Get the accuracy score(built-in function in sklearn). Compare to benchmark model or wait with comparison when second model is done.

Second approach:

1. Get stock prices(Amazon) from Yahoo Finance. Use Open, Close, High, Low prices to generate input features. Calculate some technical indicators and add those values to the input features. Do not forget to drop NaN which were generated during calculations. Convert continuous Close values into labels(see approach 1). Since Close(t+1) is predicted variable, shift Close prices by 1. Split the inputs and target into training and testing datasets preserving the sequence(see approach 1).
2. Select several classifiers(Logistic regression, SVC, Random Forest Classifier) and train the classifiers on the training datasets. Make predictions on x_test.
3. Get an accuracy score for each classifier and compare it among them and then compare with the accuracy score from the first approach.
4. Draw a conclusion about performance of approaches to predict the price trend.

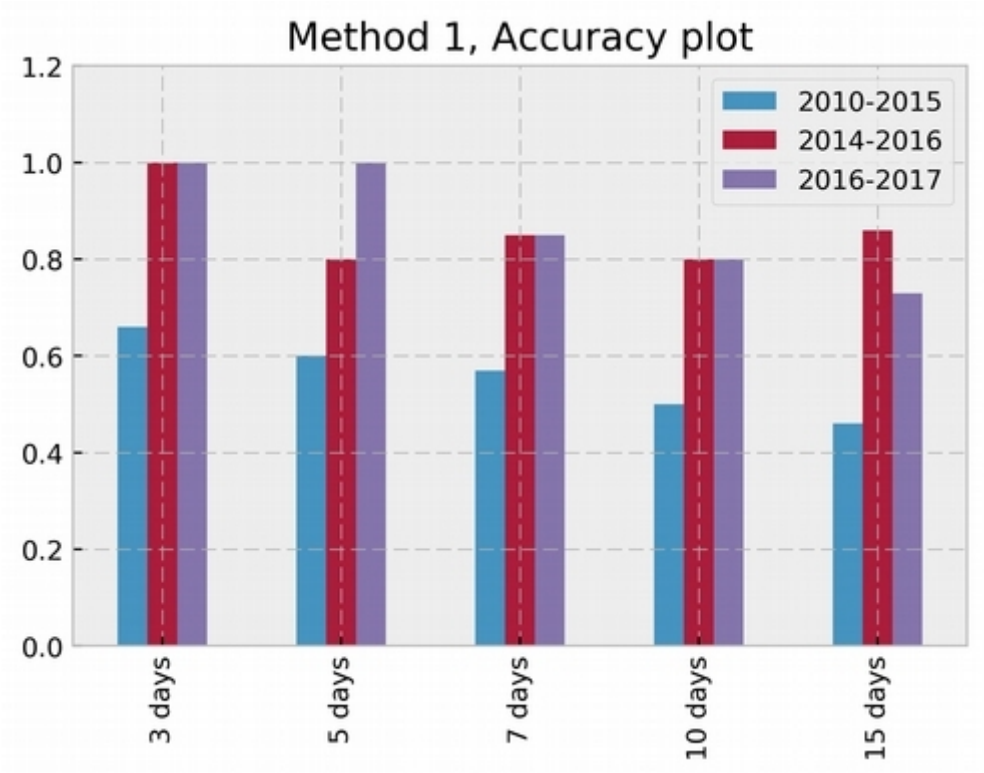
Results

Both methods are evaluated on the stock prices of Amazon company. To test effectiveness of both models, a set of three time periods with different time frame were taken into account: 2010/01/05 – 2015/01/02(5 years), 2014/01/02 – 2016/01/04 (2 years), 2016/01/04 – 2017/01/03(1 year).

1. Bellow are result obtained by the first approach - Linear regression, than classification.

Accuracy table for a short-term (3-15 days) forecasting

Period	3 days	5 days	7 days	10 days	15 days
2010-2015	0.66	0.6	0.57	0.5	0.46
2014-2016	1.0	0.8	0.85	0.8	0.86
2016-2017	1.0	1.0	0.85	0.8	0.73

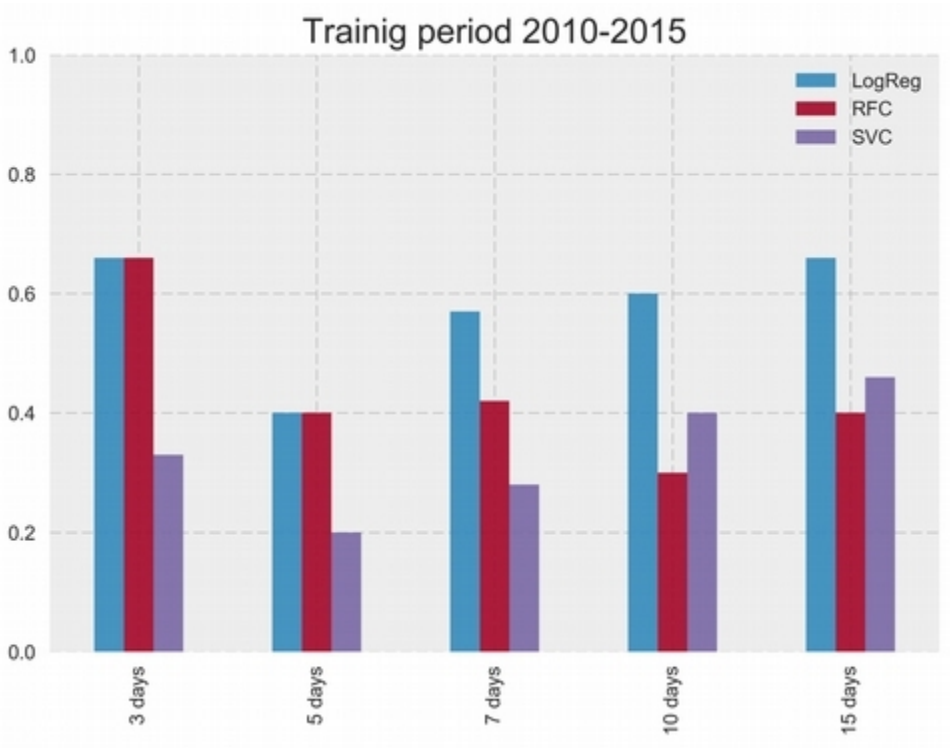


The results suggest that this method the price trend with moderate accuracy for a 5-year period data as a training set and shows much greater performance with the training time periods of 1 and 2 years.

2. Now let's take a look at the result obtained by the second method where 3 algorithms (Logistic Regression, Support Vector Classifier, Random Forest Classifier) were used for a classification problem under the same conditions as method 1.

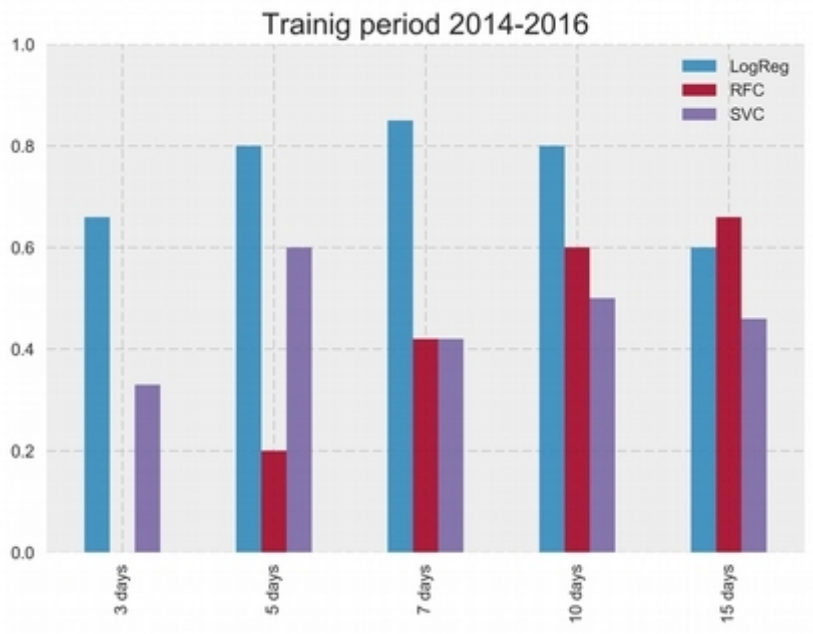
Accuracy table for Trainig period 2010-2015

Model	3 days	5 days	7 days	10 days	15 days
LogReg	0.66	0.4	0.57	0.6	0.66
SVC	0.33	0.2	0.28	0.4	0.46
RFC	0.66	0.4	0.42	0.3	0.4



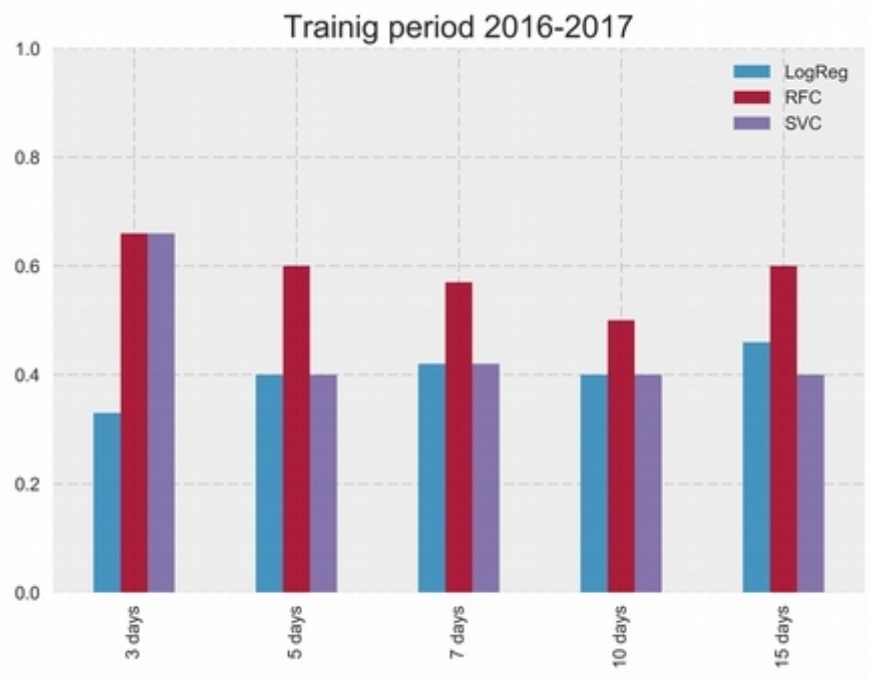
Accuracy table for Trainig period 2014-2016

Model	3 days	5 days	7 days	10 days	15 days
LogReg	0.66	0.8	0.85	0.8	0.6
SVC	0.33	0.6	0.42	0.5	0.46
RFC	0.0	0.2	0.42	0.6	0.66



Accuracy table for Trainig period 2016-2017

Model	3 days	5 days	7 days	10 days	15 days
LogReg	0.33	0.4	0.42	0.4	0.46
SVC	0.66	0.4	0.42	0.4	0.4
RFC	0.66	0.6	0.57	0.5	0.6

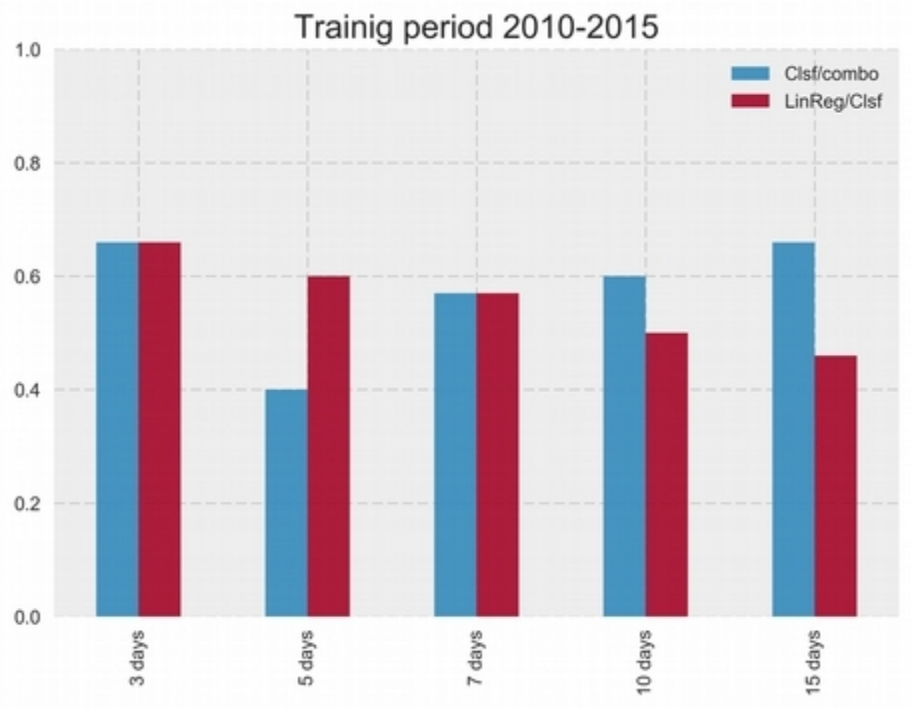


Out of three classifiers, Logistic regression performs the best on the training periods of 5 and 2 years. It does a poor job on the 1-year period. Random Forest Classifier gives the beast result on a short training period(1year) and outperforms SVC on a 2-year period (10, 15 days) and on a 5-year period (3, 5, 7 days). SVC have overall a low performance.

To come to a certain conclusion about the performance of both approaches, we need to compare the accuracy tables. Since we tried 3 classifiers in the second part, we will choose the highest accuracy out of all three classifiers for the specific time period.

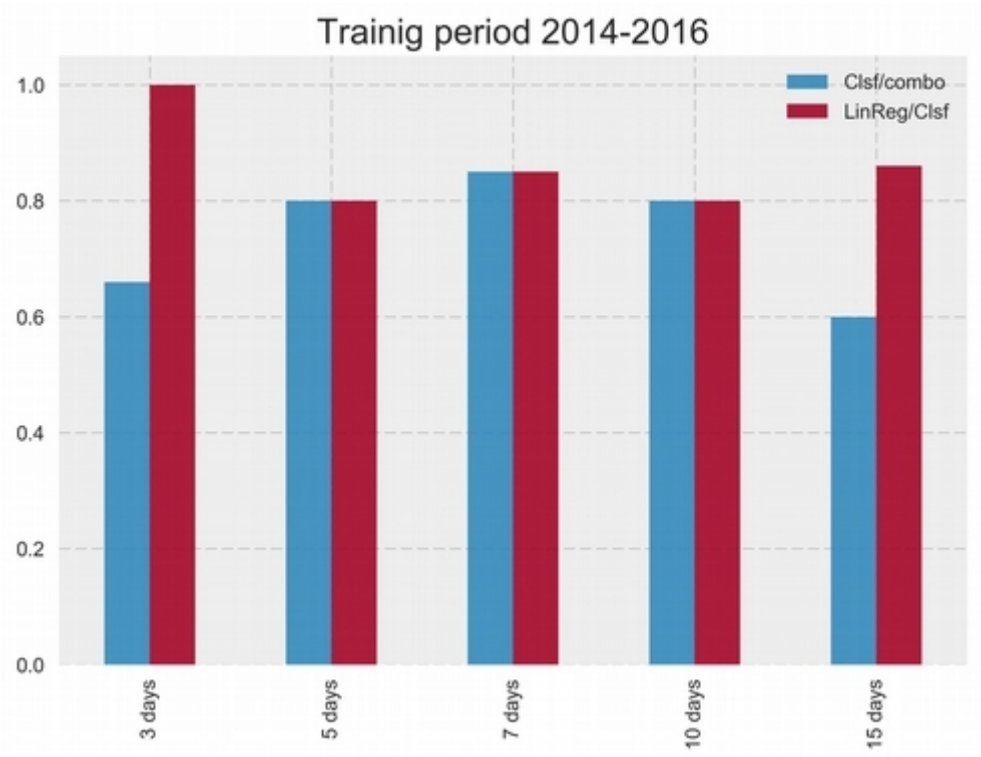
Accuracy table for Trainig period 2010-2015

Approach	3 days	5 days	7 days	10 days	15 days
LinReg/Clsf	0.66	0.6	0.57	0.5	0.46
Clsf/combo	0.66	0.4	0.57	0.6	0.66



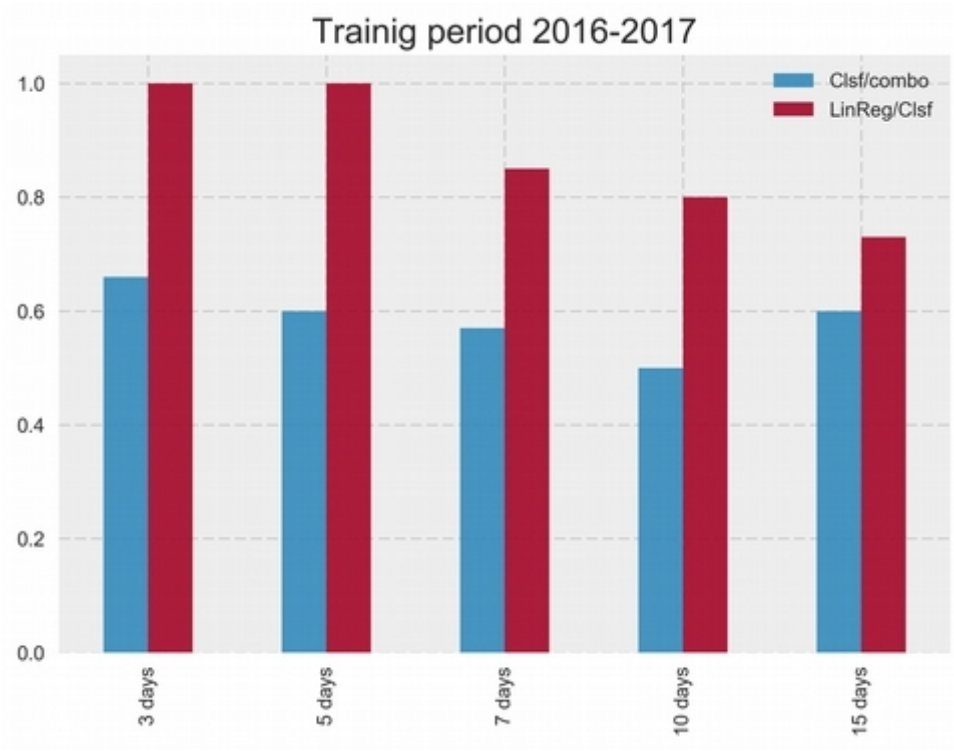
Accuracy table for Trainig period 2014-2016

Approach	3 days	5 days	7 days	10 days	15 days
LinReg/Clsf	1.0	0.8	0.85	0.8	0.86
Clsf/combo	0.66	0.8	0.85	0.8	0.6



Accuracy table for Trainig period 2016-2017

Approach	3 days	5 days	7 days	10 days	15 days
LinReg/Clsf	1.0	1.0	0.85	0.8	0.73
Clsf/combo	0.66	0.6	0.57	0.5	0.6



Conclusions

Results form the tables shows that for the period of 5 years, the team of classifiers gives a little higher accuracy for 10 and 15-day periods but for a 5-days period the accuracy is lower in comparison to linear regression/classification method.

For a 2 year period, both methods have identical accuracy score for 5,7,10-days periods but for 3 and 15-day periods, method 1 performed much better.

Data from a 1-year period clearly shows that method 1 significantly outperformed the trio of classifiers.

Putting all together, we can conclude that the first approach in which we used linear regression to predict the Close price followed by classification task gives better results(accuracy score) than a combination of three classifiers which used technical indicators to predict the price trend. Analysis was done on stock prices of Amazon company with three training periods (1, 2, and 5 years).

Future improvements

For both approaches there is a room for improvements. Specifically, if we need to increase accuracy for method 1 at 5-year period, we may include a wider range of features, for example, volume of shares and some technical indicators as well as testing several other regression models.

For the second approach addition of other technical indicators may I also improve the accuracy. Another trick is to use PCA to reduce dimensionality hoping that not many feature are equally important for the model. Also, trying many others hyper-parameters using gridsearch increase the change to get a better accuracy.