Domain Background

What would you have in your mind if you were being asked about today's economic environment? Most people may come up with unemployment rate, Gross Domestic Product (GDP), and unarguably, stock prices or indexes such as Dow Jones Industrial Average. Indeed, when we want to measure the performance of overall market or a company, oftentimes we would look to how well the stock price is moving against others and compared to its historical performances. When companies operate beyond certain scale and require more funding for expansion, one of the most common ways is to "go public", in which the company register itself on one of the major stock exchanges such as New York Stock Exchange and the company would offer certain shares to the public. By offering public stock floatation, company would receive funding from investors, be it retail or institutional ones, and in exchange investor would have voting rights in the board. The stock price of each company is comprised of many aspects, but generally reflects its current operation performance, the business sector, and the macroeconomics. A stock receiving high valuation denotes investors are expecting the company to outperform in the future and thus are willing to buy the stock in the hope that the price would later increase to reflect its value. Vivid examples are that Tesla and Amazon are generating a lot waves in press not only because of their products and services, but also because investors hold high hope in the companies' business endeavors would further push up their stock prices. Other than these much hyped businesses which people generally know a thing or two, investors often ask themselves, "Are there other ways to predict stock prices?"

Stock prices forecasting has always been a fascinating subject, and in recent years applying machine learning algorithms to predict the stock price trend, whether it is a classification issue in the format of prices going either up or down, or try to predict the actual price range. Various researches – including sentiment analysis using semantic frames to predict price movement[1] or predictions under high frequency trading scenarios[2] – have been put into light, and certain innovative approaches have made these tools more accessible to the public. Platforms such as Quantopian is using crowd sourcing method to find algorithms with the best performances, and research advisors for institutional investors such as Lucena Research have made testing trading strategy easier than before.

Problem Statement

With so many solutions available to the public, most investors simply care one thing—Will a stock go up or down? If the answer yes, if would certainly make the case easier for an investor to make investment decision. From my personal experiences, most people who label themselves as occasional investors usually speculated on certain information they received and thus invested in stocks relevant to the implication. Data suggesting that the average holding

[1] Boyi Xie, Rebecca J. Passonneau, Leon Wu, Germ´an G. Creamer. 2013. Semantic Frames to Predict Stock Price Movement. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*.
[2] Michael David Rechenthin. 2014. Machine-learning classification techniques for the analysis and prediction of high-frequency stock direction. University of Iowa.

period for stocks listed at NYSE is eight months[3], therefore it is reasonable to assume most individual investors care more about short-term price movement as opposed to the long-term approach of identifying the intrinsic value.

The project would be building a straightforward tool for investors to inquire the probability of certain stock going up or down in the designated time frame. Therefore, the objective of the project is to predict whether the price of a stock would go up or down in a short time horizon after the time of the query being submitted. The machine learning task is framed as a binary classification task and the objective is to figure out the data input and the derived feature that are relevant in making these predictions.

For instance, if we are going to predict the price of Apple stock in the next five days, should we simply use the historical data of its own or also look at other stocks that are in the same industry? In addition to that, what kind of information other than price, such as volume, should be featured in the dataset? These are all the problems that would be attempted to tackle in the project. For the clarity of the project design and the problem scope, the historical data of Apple would be used as raw input, and features are generated based on commonly used technical indicators. The overall strategy is to test out several classifier algorithms and the one with best performance under evaluation metric would be selected to predict the movement of the stock.

Datasets and Inputs

Input: Using Quandl Python module to load WIKI EOD data, a crowdfunded database that provides End-of-day stock prices, dividends and splits for 3,000 US companies. The raw data extracted from the database is consisted of opening price, intraday high price, intraday low price, closing price, trading volume, ex-dividend amount per share, stock split ratio, adjusted opening price, adjusted intraday high price, adjusted intraday low price, adjusted closing price, and adjusted volume. The benefit of using Quandl library also including that it is easier to get real-time and updated price without loading a pre-existing dataset.

Dataset: Stock market raw data is considered as non-stationary and its behavior may be affected by various factors. Commodities companies might be affected by world demand, while consumer discretionary sector may show seasonal and cyclical patterns. Following the 2008 Financial Crisis turmoil, the market is demonstrating an upward trend since mid-2009. Thus, data points of the stock market are considered as non-stationary. Since the project would allow the user to make inquiries for major U.S. companies, transforming the data to a stationary one using a universal approach may cause biases as not all stocks are showing identical patterns or trends. Therefore the data would not be transformed to stationary in the project.
As highlighted before a period of serious disruption in the stock market happened between 2008 and mid-2009, and thus, the project would only look at data points after January 1, 2011.

[3] Massachusetts Financial Services Company. 2016. In *White Paper Series*.
https://www.mfs.com/wps/FileServerServlet?servletCommand=serveUnprotectedFileAsset&fileAssetPath=/files/documents/news/mfse_time_wp.pdf

Project Design

Data Acquisition: 5 year return is one the commonly used benchmark to evaluate the long-term performance of a stock and it would applied as the range of historical data. Overall, the project would allow user to predict using any data point samples between 2011 and the current date, with a maximum dataset length of 5 years. To test and trial, the examples generated throughout the report would be using the stock of Apple, Inc.(AAPL) from 1/1/2011 to 12/31/2016, which would give us 1,510 data entries to play with.

Feature Engineering: To solve the problem, the fundamental metric to look at is certainly the historical price information. As highlighted earlier that the project aims to predict short term price movement instead of forecasting a target price. Thus, valuation metrics such as Price-to-Earning Ratio and EBITDA—which reflect the nature of the business operation but are more align with finding the intrinsic value, are not considered in the project. Instead, metrics used in Technical Analysis such as Moving Averages and Moving Average Convergence Divergence that are indicative of momentum and trend are applied in the project. Therefore, daily return would be calculated first to generate the technical indicators. In addition, unwanted features such as split ratio, intraday trading prices, ex-dividend amount, and any other columns that become null during the engineering process would be dropped to ensure the data integrity.

Classification Preparation: To transform price trend into a binary result of price going up or down, daily return would be calculated to see if the figure is positive or negative. A prediction of 1 represents a positive return projection, whereas 0 means negative return. A potential elaboration might set foot on return threshold such as only returns that pass absolute value of 0.25% are considered significant, and any movements absolute value below 0.25% would be marked as “Neutral” with a mark of 2 being given. However, the machine learning task would turn to a 3-class classification one under this approach, and I would evaluate if it indeed increase the classification performance.

Classification Algorithm: Support Vector Machine provides an effective mean to dissect high-dimensional data as a result of feature generation, and its memory efficient characteristic provides an economical way to process the five-year span of data. K-nearest neighbor is also another instance-based algorithm that is well suited to solve problem, as the learning is implemented based on the k nearest neighbors of each query point, which is discrete value of generated feature. Ensemble algorithms such as AdaBoost, Gradient Tree Boosting are other robust means to build instances on random subsets of the training set and then aggregate individual predictions to form a final prediction.

Cross Validation: Stock market data is financial time-series data, which would easily cause look-ahead bias if we simply use the randomized 80-20 split. To prevent this issue from happening, roll-forward validation, or TimeSeries Split Cross Validation in Scikit learn would be applied.

Performance Evaluation: Assessing the performance of each algorithm based on evaluation metrics. After choosing the algorithm with the best prediction, the project would offer a

glimpse of the stock performance in the presumed scenario compared to actual world by utilizing back-testing strategies.

User Interface: The query interface would be built in Jupyter Notebook format as investors could adjust their inputs easily. The project would experiment with the feature selection such as PCA, find the best performed classifier, and tweak the parameters to generate the optimal outcome.

Benchmark Model

Several theses and research papers had focused on solving the problem. My benchmark model[4] used single feature of time-series price to predict whether the stock price fall or rise immediately following the preceding five days. The best result it gets from the model is an accuracy of 0.5292 using Gaussian Naïve Bayes classifier.

Evaluation Metrics

The evaluation metric is mean accuracy score. Based on Efficient Market Hypothesis, the prices of stocks have already reflected all the information and the movement should be of absolute randomness. The Hypothesis translates that the expected accuracy should be of 50%. Therefore, the mean accuracy score should be at least 0.50.

[4] Mark Dunne. 2016. Stock Market Prediction. University College Cork Computer Science Department. http://markdunne.github.io/public/mark-dunne-stock-market-prediction.pdf