Machine Learning Engineer Nanodegree

Capstone Project: Forecasting Stock Price Movement Direction

Ku Wei Chiet

July 2017

I. Definition

a) Project Overview

Predicting the trend of future stock price has always been an attractive topic in many fields including trading, finance, statistics and computer science. The primary objective is to determine the market timing to buy, hold, or sell a certain stock. The task is challenging because there are many uncertainties involved and many factors that influence the stock price. These include macroeconomic factors such as political events, firms' policies, general economic conditions, investors' expectations, institutional investors' choices, movement of other stock market, and psychology of investors etc.[1]. Therefore, stock market prices are susceptible to quick changes and are generally regarded as dynamic, non-parametric, chaotic and noisy in nature [2].

Financial markets around the world are becoming more integrated as the results of globalization. When subprime mortgage crisis in U.S. spiraled out of control, it caused the meltdown of the US economy and subsequently other countries. U.S. stock market suffered significant drop as shock over the United Kingdom voters' move to exit the European Union in 2016 [3]. These are the evidences that no financial market is completely isolated today. Political instability, economic factors, war and terrorism events in oversea could influence the performance of domestic markets. It is envisioned that the prices of global stock markets are correlated to each other.

In this project, the use of global stock data as input features to machine learning algorithms to predict the movement of S&P 500 is proposed. Stock market from around the globe has different closing time with U.S. market. For instance, Hong Kong's Hang Seng Index closes 12 hours before S&P index. Its stock price data is available before the beginning of the U.S. market trading time. This project would like to explore whether the correlation of oversea stock indices and U.S. market could be utilized as significant predictor for future movement of S&P 500.

b) Problem Statement

The objective of this project is to predict whether the close price of S&P 500 today will be higher or lower than yesterday. It is formed as binary classification problem with output be '1' if S&P 500's close is predicted to be higher than yesterday, '0' otherwise. Several supervised learning classification techniques is used to address the objective, namely Decision Tree, Logistic Regression, Support Vector Machine and Multilayer Perceptron (MLP)

Neural Network. The input data to the classifier is the stock price of major world indices, while the output is the predicted trend of S&P 500. The performance of these classification models is compared to select the best model.

After the classification model has been trained, its predicted output is used to generate buy/sell signals in a customized trading strategy that trades S&P 500. The performance of the proposed trading strategy is benchmarked against a low risk buy-and-hold strategy of S&P 500. The trading strategy based on the prediction outcomes could be consider success if its portfolio return is higher compared with buy-and-hold strategy.

c) Evaluation Metrics

For binary classification problem in this project, there are four possible outcomes of the prediction:

$$True\ Positive: S\&P\ 500's\ close\ is\ higher\ than\ yesterday\ and\ model's\ output\ is\ 1$$
$$False\ Positive: S\&P\ 500's\ close\ is\ lower\ than\ yesterday\ and\ model's\ output\ is\ 1$$
$$True\ Negative: S\&P\ 500's\ close\ is\ lower\ than\ yesterday\ and\ model's\ output\ is\ 0$$
$$False\ Negative: S\&P\ 500's\ close\ is\ higher\ than\ yesterday\ and\ model's\ output\ is\ 0$$

Accuracy will be used to quantify the performance of the classification model. It is defined as follow:

$$Accuracy = \frac{True\ Positives + True\ Negatives}{Total\ Number\ of\ Predictions\ Made}$$

In addition, F1 Score of the models will be compared because it is the weighted average of precision and recall. It is defined as:

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

where:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

For the problem in this project, precision indicates the hit rate among rising predictions while recall reflects the proportion of predicted rising days among all rising days.

To compare the portfolio performance of buy-and-hold strategy and the proposed trading strategy based on the prediction results, Sharpe Ratio and return rate will be used. These metrics are defined as follow:

$$Return = \frac{Ending\ Value\ of\ Portfolio}{Beginning\ Value\ of\ Portfolio} - 1$$

$$Sharpe\ Ratio = \frac{\bar{r}_p - r_f}{\sigma_p}$$

where:

$$\bar{r}_p = Mean\ return\ of\ portfolio$$

$$r_f = Risk\ free\ rate$$

$$\sigma_p = Standard\ deviation\ of\ portfolio$$

The Sharpe Ratio is the average return earned in excess of the risk free rate per unit of volatility [4], where the risk free rate usually is the federal interest rate in the U.S.. It has become an industry standard for measuring strategy performances and it is also great at comparing performances across different strategies.

II. Analysis

a) Data Exploration

For this project, 9 stock indices from 1st January 2006 to 31st December 2016 are obtained from Yahoo Finance, with 4 of them are U.S. indices, the rest of 5 are non-U.S. indices. Table 1 shows the indices with their respective closing time. The closing time of the indices in the table shows that non-U.S. stock close early than U.S. market due to time difference. Each stock index retrieved has the following attributes:

a) Date: in days

b) Open: price of the stock at the opening of the trading

c) High: highest price of the stock during the trading day

d) Low: lowest price of the stock during the trading day

e) Close: price of the stock at the closing of the trading

f) Adj Close: price of the stock at the closing of the trading adjusted with dividends and stock splits.

g) Volume: amount of stocks traded during a given trading day

Table 1: Stock indices used in this project

Index	Description	Closing Time (EST)	Hours Before S&P Close
SPDR S&P 500 ETF Trust (SPY)	An exchange-traded fund (ETF) that tracks the S&P 500 stock market index.	1600	0
NYSE Composite (NYA)	Stock market index covering all common stock listed on the New York Stock Exchange (NYSE).	1600	0
NASDAQ Composite (IXIC)	Market capitalization-weighted index of approximately 3,000 common equities listed on the Nasdaq stock exchange.	1600	0
Dow Jones Industrial Average (DJI)	Price-weighted average of 30 significant stocks traded on the NYSE and the NASDAQ.	1600	0
London FTSE (FTSE)	Share index of the 100 companies listed on the London Stock Exchange with the highest market capitalization.	1130	4.5
Frankfurt DAX (GDAXI)	Stock market index consisting of the 30 major German companies trading on the Frankfurt Stock Exchange.	1130	4.5
Tokyo Nikkei (N225)	Stock market index comprised of Japan's top 225 blue-chip companies traded on the Tokyo Stock Exchange.	0200	14
Hong Kong Hang Seng (HSI)	Market capitalization-weighted index of 40 of the largest companies that trade on the Hong Kong Exchange.	0400	12
Australia All Ordinaries Index (AORD)	Share prices for 500 of the largest companies listed on the Australian Securities Exchange.	0200	14

Adjusted Closing Price

Figure 1 shows the adjusted closing price of the stock indices. Because the indices operate on scales differing by orders of magnitude, the data has been normalized by its first value on 3rd January 2006 (the first trading day) for better visualization. Over the eleven-year period, SPY is obviously correlated with other indices. It is noticeable that financial crisis in 2009 caused sudden drops in SPY also happened globally to all indices. The correlation of the adjusted closing price of the stock indices is also shown in Figure 2. Generally, all indices are strong correlated with SPY, especially U.S. and European indices.

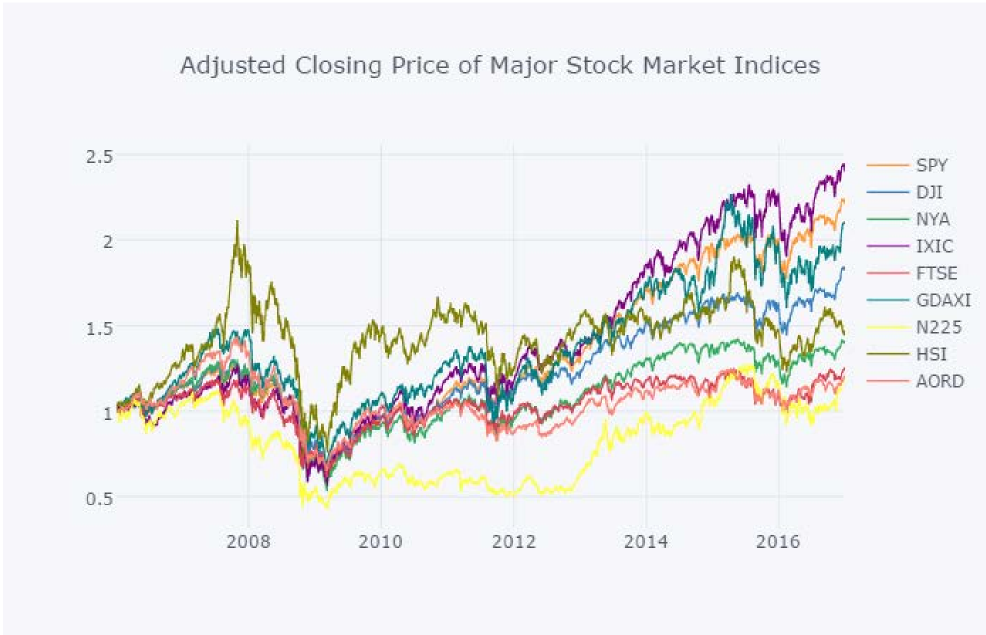

Figure 1: Adjusted closing price of major stock market indices

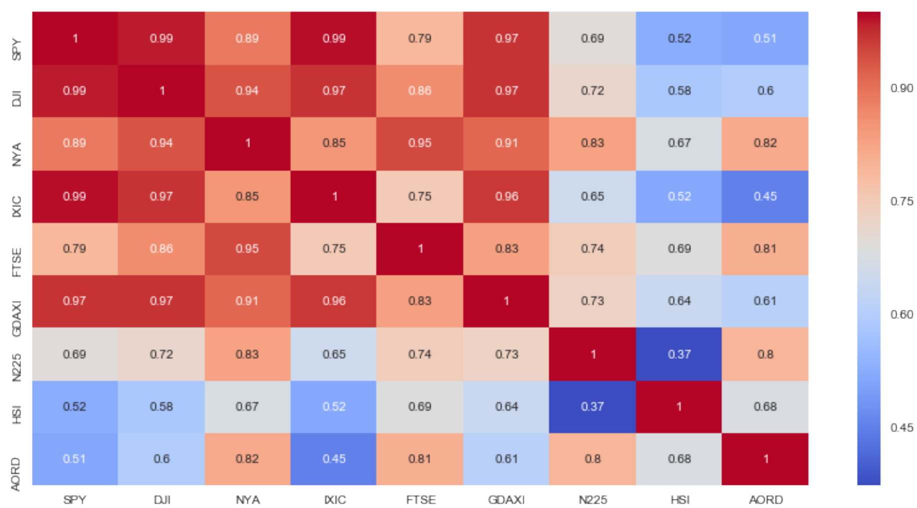

Figure 2: Heatmap of correlation of adjusted closing price of stock indices

<u>Daily Returns</u>

While trend exists in the adjusted closing price of the indices, but its actual value is not very useful in modeling binary classification problem in this project. Therefore, daily returns which is computed as follow is chosen for this purpose:

$$Return\ (t) = \frac{Adjusted\ Closing\ Price(t)}{Adjusted\ Closing\ Price(t-1)} - 1$$

The daily returns plot in Figure 3 does not show any particular pattern or visible trend in the data. They tend to fluctuate randomly around zero.

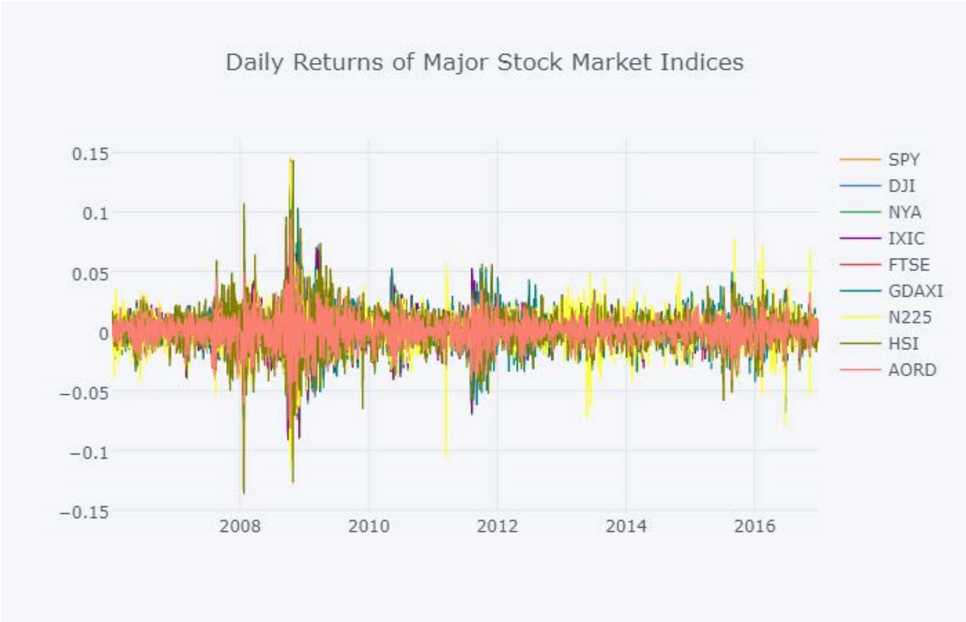

Figure 3: Daily returns of major stock market indices

Figure 4 provides a glimpse at pairwise relationships of the daily returns of the indices. At first glance, all U.S. indices are linearly correlated to each other and relationship exists between SPY and European indices. The correlation is quantified in the heatmap in Figure 5. From the heatmap, it is noticeable that daily returns of U.S. indices are strongly correlated, as expected. SPY returns are correlated with European indices at around 0.62-0.64 for the FTSE and GDAXI, which is also a strong correlation. For Asian/Oceanian indices, their correlation with SPY is at around 0.15-0.42, which is a significant less correlated.

Figure 6 shows an autocorrelation plot for the daily returns of SPY stock. The autocorrelations determine correlations between current values of the index and lagged values of the same index. The goal is to determine whether the lagged values are reliable indicators of the current values. It can be seen from the plot that the autocorrelation of SPY daily trend is close to zero excepts at the origin. This leads to conclusion that individual financial market is approximately a Markov process and yesterday's values are no practical help in predicting today's return.

Figure 4: Pairwise relationships of the daily returns of the indices

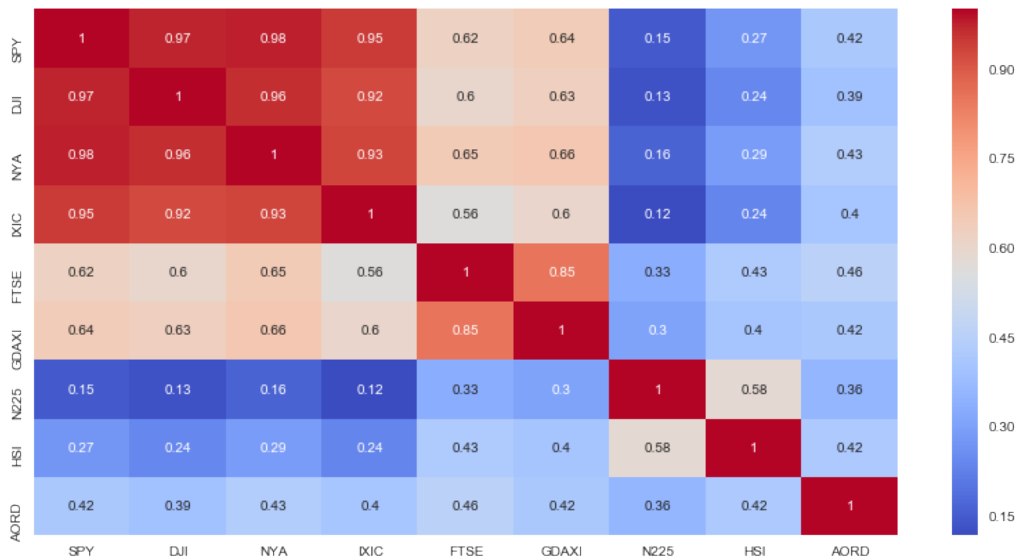

Figure 5: Heatmap of correlation of daily returns of stock indices

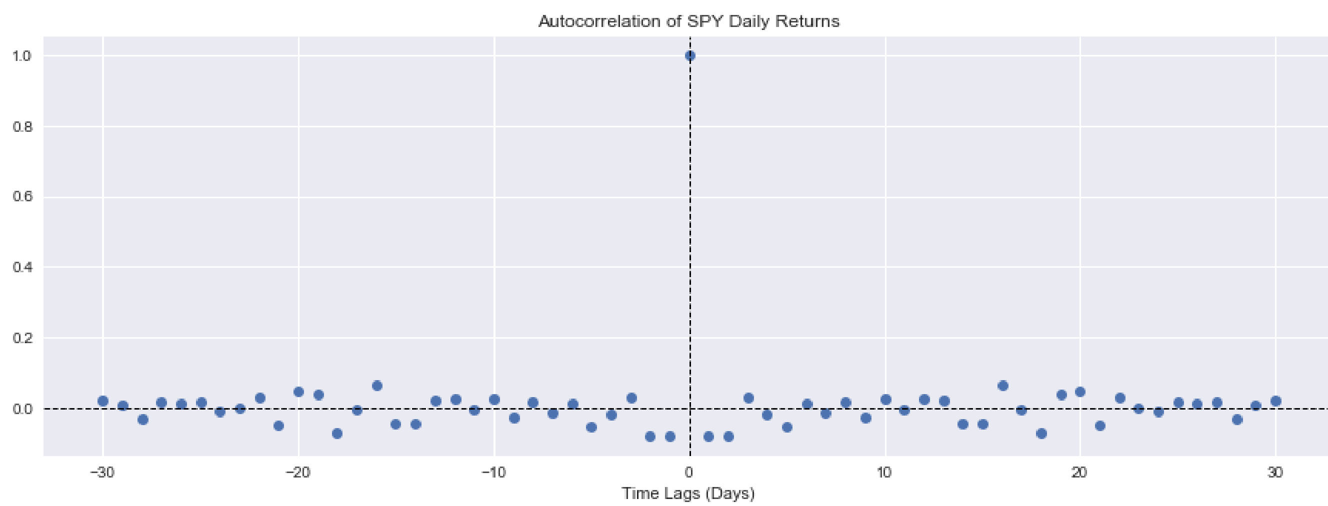

Figure 6: Autocorrelation plot of SPY daily returns

Figure 7 depicts the cross-correlation of daily returns of SPY and other indices. The cross-correlation values at the origin are same as the values in first row of Figure 5. Correlation also exist between SPY and other European/ Asian index values from the previous day, as shown in the plot at T-1. Other than that, it is safe to assume that past values of other indices are not good predictors for the SPY close because the cross-correlation values are close to zeros at other time lags.

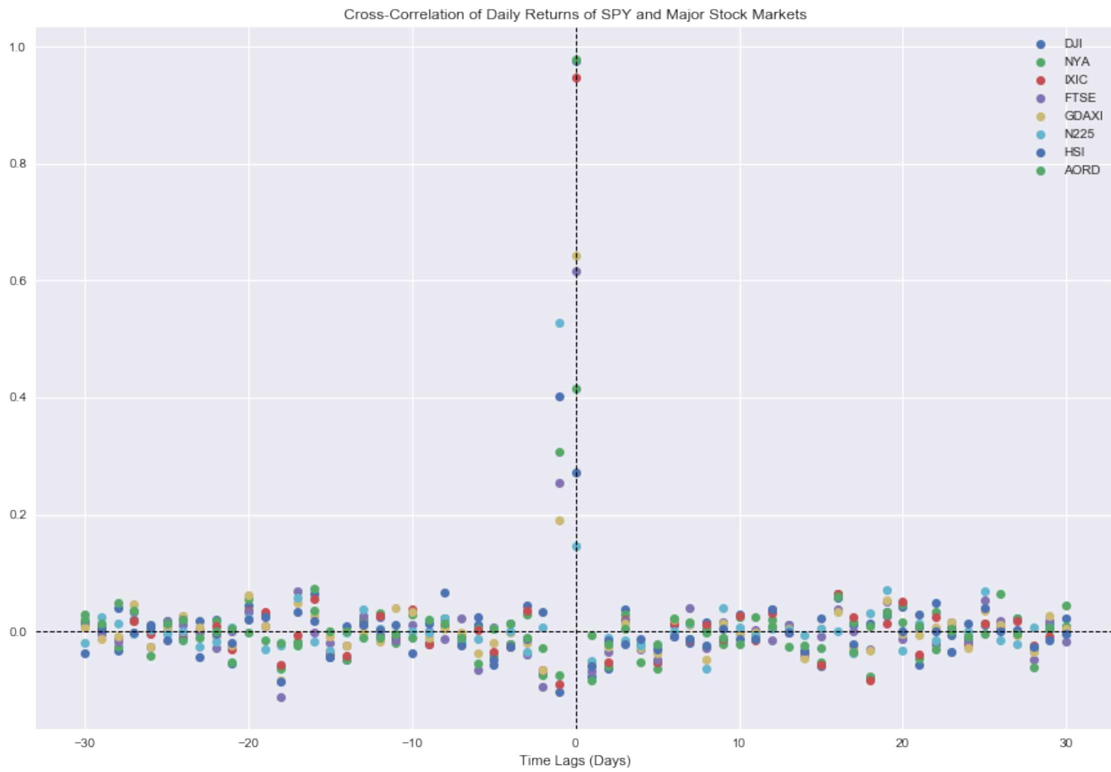

Figure 7: Cross-correlation of daily returns of SPY and other stock indices

Summing up the above exploratory data analysis:

- European indices from the same day are strong predictors for the SPY daily returns.
- Asian/Oceanian indices from the same day are significant predictors for the SPY daily returns.
- Indices from previous days are not good predictors for the SPY daily returns.

Although U.S. indices have strong correlation with SPY daily returns, given the data are on same day, they are not available by the time they are needed for prediction. However, data sources such as European indices and some other markets are promising features for machine learning model as they have relative high correlation with SPY at the origin and their data is available before or at the beginning of the U.S. market trading time. This observation corroborates the previous belief about the inter-connection between global markets and how the information reflected by their movements can be beneficial to the prediction of U.S. stock markets.

b) Algorithms and Techniques

<u>Input Features and Output Label</u>

The problem to be solved is predicting whether the close price of SPY today will be higher or lower than yesterday. It can be modeled as binary classification problem with output be '1' if the return of SPY close is positive, and '0' otherwise. For prediction of the SPY movement on day T, the input features to the machine learning models would be:

- Returns of non-U.S. indices available on day T (5 features)
- Returns of all stock indices available on day T-1 (9 features)

These features are selected based on the findings from previous exploratory data analysis, namely that returns from other regions on a given day are strongly correlated with the return of the SPY. The returns of other indices on yesterday are also added as additional input features for the classification models. To reduce the dimensionality of the data, Principal Component Analysis (PCA) is applied on the input features. The number of dimensions necessary for the problem is determined by examining the cumulative explained variance ratio of by each of the components.

<u>Classification Models</u>

Four supervised learning classifiers are chosen in this project:

- Decision Tree
- Logistic Regression
- Support Vector Machine (SVM)
- Multilayer Perceptron Neural Network (MLP)

Decision tree is a tree structure-based learning algorithm that decides the target value of a new sample based on various attribute values of the available data. The internal nodes of a decision tree denote different attributes, the branches between the nodes indicate the possible values that these attributes can have in the observed samples, while the terminal nodes are the final value (classification) of the dependent variable.

Linear regression attempts to model the relationship between two variables by fitting a linear equation to observed data. The outcome is continuous and can have any one of an infinite number of possible values. Logistic regression is a special case of linear regression that allows the prediction of binary outcome. It predicts the probability of occurrence of an event by fitting data to a logit function.

Support Vector Machines use certain kernels to transform the original problem so that linear classification techniques can be applied to non-linear data. Applying the kernel equations arranges the data instances in such a way within the multi-dimensional space, a hyper-plane that separates data instances of one kind from those of another can be found.

Artificial neural network is a computational model that is inspired by the way biological neural networks in the human brain process information. MLP is a class of feedforward artificial neural network that consists of at least three layers of nodes. Patterns are presented to the network via the ‘input layer’, which communicates to one or more 'hidden layers' where the actual computation is done via a system of weighted connections. The hidden layers then link to an ‘output layer’ where the outcome of the network can be seen.

Decision tree, logistic regression and SVM are constructed by using scikit-learn (sklearn) modules while the MLP is built by using TensorFlow library. Next, grid search cross validation (GridSearchCV via sklearn) is conducted on the first three models to optimize their hyper parameters. For MLP model, several variations of the number of network layers and number of neurons per layer are tested to find the optimal model. The model with highest accuracy and F1 Score is selected as the best classification model.

Trading Strategy

Figure 8 demonstrates a customized trading strategy that uses machine learning model predictions to generate buy/sell orders. The strategy is designed to take full advantage of the model's predictions and consider the existing portfolio position when generating trade orders. If the model predicts today's return is higher than yesterday, and it is already hold a long position, which cumulates from previous buy signals, or have no position at all, it makes sense to buy more units of share or enter a long position. If it is already at short position, the strategy suggests go aggressively to exist all short position to avoid the lost and start entering a new long position to profit from the price increase.

```
If model predicts up
    If at long position or no position
        Long 300 shares
    If at short position
        Buy back all short position units and buy additional 300 shares
    Set current position to long

If model predicts down
    If at short position or no position
        Short 300 shares
    If at long position
        Sell all long position units and sell additional 300 shares
    Set current position to short
```

Figure 8: Detail illustration of customized trading strategy

Backtesting

To test how the proposed trading strategy performs, a market simulator that allows placing buy/sell orders, executing the orders at market price including trading cost, and evaluating portfolio value at daily frequency is needed. Quantopian [5] is chosen in this project because it enables users to perform a much more realistic simulation than one might otherwise be able to carry out. Quantopian can correctly model costs that a real trading algorithm would incur, such as commission charged by the stock broker and slippage. Slippage is the difference between the expected price of a trade and the price at which the trade is actually executed [6]. Slippage might occur when placing a large order that could shift the market, or when market volatility is high. The proposed trading strategy algorithm is written in Python by using Quantopian's web-based Interactive Development Environment (IDE). The prediction results of the machine learning model are saved as a single CSV file and uploaded to Dropbox. The trading strategy which is executed on the web-based IDE imports the prediction results to determine buy/sell signals.

c) Benchmark

Out of eleven years of stock indices data from year 2006-2016, the first nine years of data (2006-2014) is used as training set. Over these nine years, the probability of SPY closed higher (daily return is positive) is 52.80%, which is greater than the probability of SPY closed lower (daily return is negative). Based on this observation, the most naïve prediction for any given day would be that the market would increase. If the model were to use this assumption to make prediction on the test set data, it would have an accuracy of 48.61% and an F1 Score of 0.65. These values will be served as the benchmark of accuracy and F1 Score of the machine learning models for this project.

The performance of the proposed trading strategy based on machine learning model's prediction results is compared with a passive buy-and-hold strategy [7]. In this strategy, the trader buys the index on day one and then held the index until the end of the test period. This strategy is consistent with benchmark predictions where the index is always closes higher than yesterday during test period.

III. Methodology

a) Preprocessing

After the raw data of indices has been imported from Yahoo Finance, any missing values or NaN values is imputed by using 'forward-fill' method, followed by 'backward-fill' method [8]. Next, daily returns of the indices are computed by using adjusted closing price and normalized to zero mean and unit variance.

As mentioned in previous section, to predict whether the SPY close today will be higher or lower than yesterday, a total of 15 features based on today and yesterday returns of global stock indices are chosen as input features of the machine learning models. PCA is applied on these features to reduce the dimensionality of the feature space. Figure 9 shows the cumulative explained variance ratio of the PCA components. 92% of the data is explained by the first 7 principle components. This reduce the dimensionality of the input space of the machine learning models from 15 to just 7.

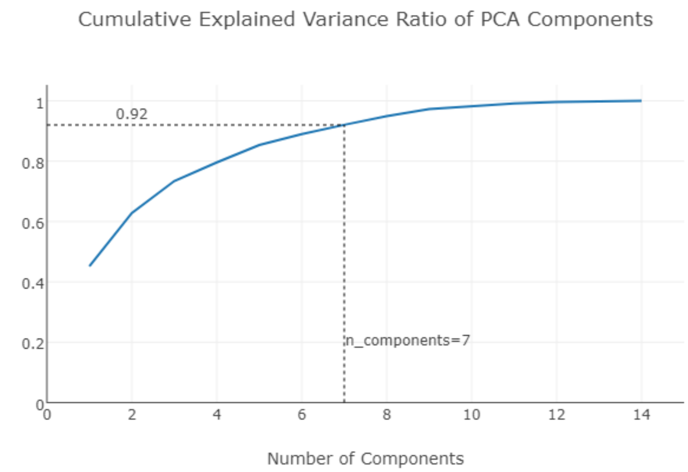

Figure 9: Cumulative explained variance ratio of PCA components

b) Implementation

The data is split in training set and test set, with all trading days from 1st January 2006 to 31st December 2014 (2268 data points) as training set, and all trading days from 1st January 2015 to 31st December 2016 (504 data points) as test set. Because the data set is a time series, the data is not shuffled prior splitting between the training and sets to avoid look-ahead bias [9].

Decision Tree, Logistic Regression and Support Vector Machine classifiers are implemented by using default settings of sklearn module functions. The neural network is built by using TensorFlow library and consists two hidden layers with 25 neurons and 50 neurons respectively. The learning rate is set to 10^{-4} with 2000 training epochs.

The initial results for the four models are presented in Table 2 and Figure 10. All models exhibit accuracy and F1 scores above the benchmark values. Among these models, logistic regression, SVM and MLP perform significantly better than benchmark result with nearly a 20% improvement of accuracy compared to benchmark value.

Table 2: Initial classification performance of machine learning models

Model	Accuracy (%)	F1 Score
Benchmark	48.61	0.6542
Decision Tree	63.49	0.6320
Logistic Regression	70.44	0.7140
Support Vector Machine (SVM)	69.65	0.7086
Multilayer Perceptron (MLP)	68.87	0.7004

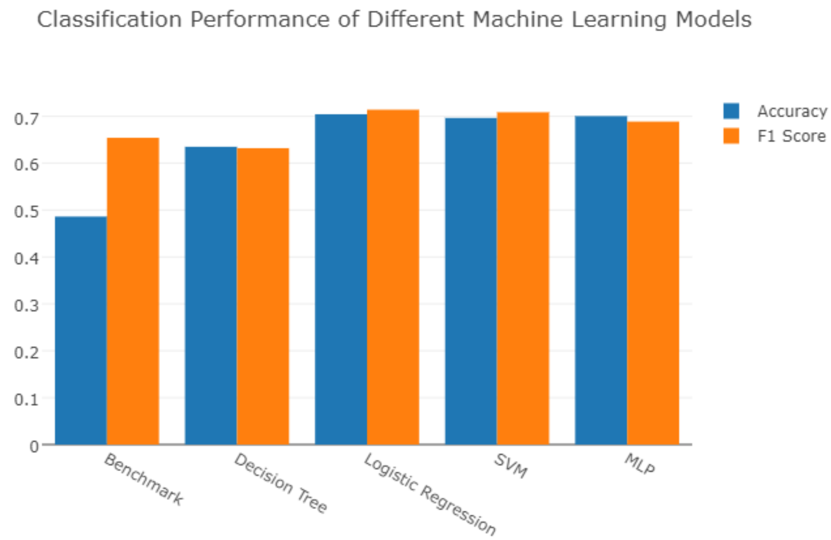

Figure 10: Initial classification performance of machine learning models

c) Refinement

To improve the performance, grid search cross validation (GridSearchCV via sklearn) is conducted on the decision tree, logistic regression, SVM models. For decision tree classifier, the maximum depth of the tree and the minimum number of samples required to be at a leaf node are two parameters to be searched. For logistic regression model, C parameter is tuned. For SVM model, C and gamma is adjusted to find the best estimators. The neural network is tuned by tweaking the network architecture and the number of neurons in each hidden layer. Table 3 compares the performance of these tuned models with the benchmark values. Overall, slight performance improvement can be found on decision tree model with about 4.57% increase of accuracy and 0.0542 increase of F1 Score. For other models, there is no much changes in the accuracy and F1 Score.

Table 3: Classification performance after fine-tuning

Model	Fine-tuned Parameters	Accuracy (%)	F1 Score
Benchmark	N.A.	48.61	0.6542
Decision Tree	min_samples_leaf = 7 max_depth = 6	68.06	0.6862
Logistic Regression	C = 0.1274	70.04	0.7113
Support Vector Machine (SVM)	C = 2.9763, gamma = 0.003162	69.44	0.7094
Multilayer Perceptron (MLP)	3 hidden layers, with number of neurons = {50, 100, 150}	70.04	0.6925

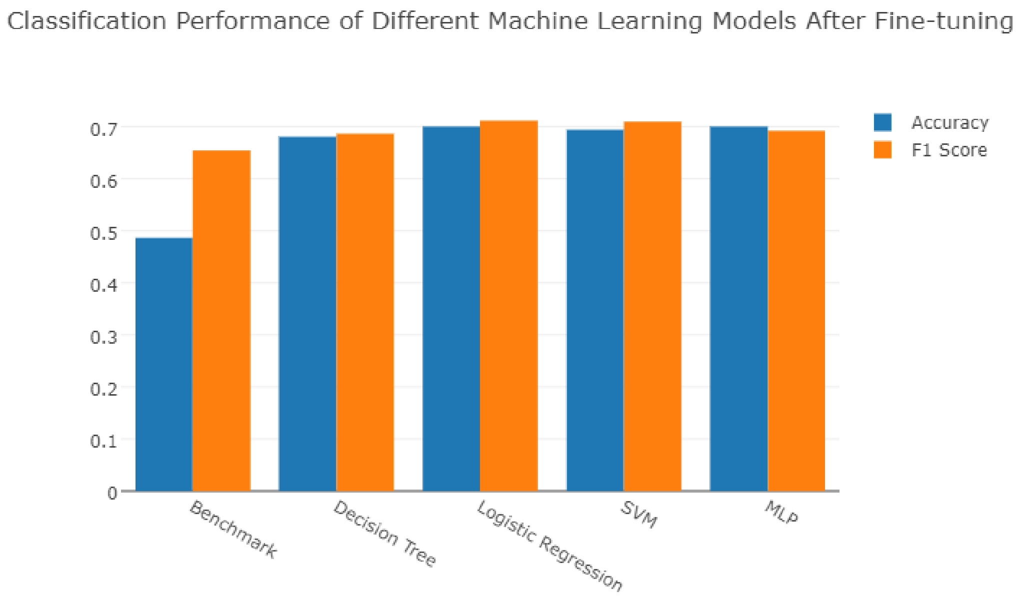

Figure 11: Classification performance after fine-tuning

IV. Results

a) Backtesting Trading Strategies

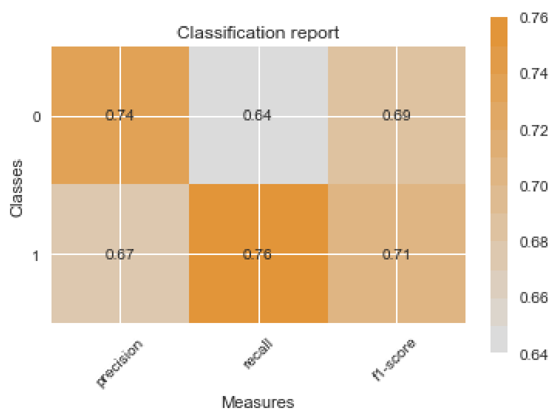

Figure 12: Classification report of logistic regression model

Among all machine learning models, logistic regression gives the best performance in predicting index movement, whose accuracy is 70.04% and F1 Score is 0.7113. Both accuracy and F1 score outperformed benchmark values, which is simply always guessing that the market will close higher than yesterday. The final classification performance is summarized in Figure 12.

To verify the robustness of the final model, cross-validation is applied on the training data to ensure the model generalizes well without overfitting. StratifiedShuffleSplit of sklearn is used to split the data into training set and

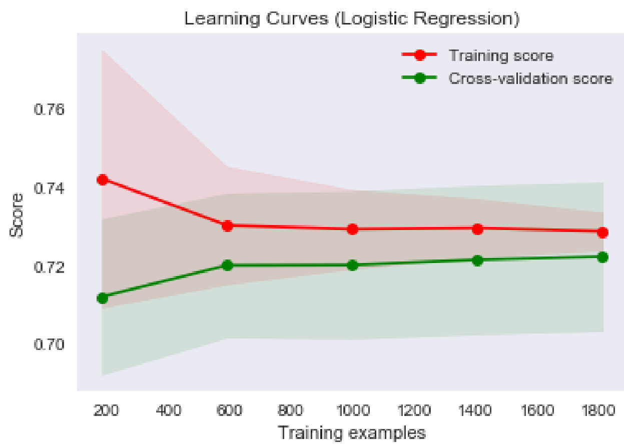

Figure 13: Learning curve of cross-validation of logistic regression model

validation set for cross-validation. The cross-validated training and test scores are plotted in Figure 13. The plot reveals that adding more training data does not boost the performance significantly. Therefore, the number of training samples chosen in this project is sufficient for the problem. Both training score and cross-validation score are consistently above 0.70. The gap between training score and cross-validation score is also not significant, indicating that the model is not suffering from overfitting problem. As a result, the proposed model is an effective and robust solution for predicting the trend of stock index.

To fully exploit the advantage of nearly 20% improvement of model's prediction accuracy, the prediction results are used to generate buy/sell signals in a customized trading strategy outlined in Figure 8. The input of the prediction model includes closing price of other stock indices, which have different closing time with U.S. market. One important assumption of the trading strategy is that the closing price of other stocks are available before the strategy is executed. Therefore, the strategy is scheduled to be executed daily at 4 hours before U.S. market close, which is half an hour after closing time of European markets.

The proposed trading strategy starts with initial capital with $100,000 dollars and its performance is compared with buy-and-hold strategy. For buy-and-hold strategy, all initial capital is invested to purchase SPY shares on first trading day of year 2015 and the shares are held until the end of year 2016.The performance charts of the proposed trading strategy and buy-and-hold strategy generated by Quantopian are presented in Figure 13 and Figure 14 respectively. The details of the backtesting results can be accessed on [10] and [11]. The red line indicates the performance of default benchmark used by Quantopian which is also a buy-and-hold but with reinvested dividends. It can be ignored in following performance analysis.

Initially, the cumulative returns of proposed strategy are lower than buy-and-hold strategy. The strengths of the proposed strategy based on prediction results becomes apparent after mid of August 2015, when the portfolio returns become positive and stays in positive after that. For buy-and-hold strategy, the portfolio returns fluctuate between positive and negative during entire test period and apparently is more volatile than proposed trading strategy. At the end of test period, the SPY shares at hands based on proposed trading strategy worth $129,182 dollars, resulting in 29.2% of total returns. It is better than buy-and-hold strategy with total returns of 11.9%. Besides high profit, the proposed strategy also has the advantage of higher risk-adjusted return with Sharpe ratio of 0.60 compared with 0.46 of buy-and-hold strategy.

Figure 14: Performance of proposed trading strategy simulated on Quantopian

Figure 15: Performance of buy-and-hold strategy simulated on Quantopian

V. Conclusion

a) Reflection

Predicting stock market movements has always been a challenging task considering the trends being affected by various random factors such as investors' sentiments or other macroeconomics factors. However, the interaction among global financial indices can be captured to make profit in stock trading. In this project, the use of machine learning algorithms to predict the stock index movements based on data collected from different global financial markets is proposed. The solution and the findings of this project can be summarized as follows:

- Data Retrieval and Preprocessing
    - Historical stock price data of major global markets is retrieved from Yahoo Financial. Missing values in the data are filled before performing normalization.
- Feature Extraction and Dimension Reduction
    - Exploratory data analysis is applied on the data to select strong predictors for the model. The analysis reveals that strong correlation exists between the U.S. stock indices and global markets such as European markets that close before U.S. trading time.

    - PCA is performed on the input features to reduce the input dimension of the models.
- Model Construction and Optimization
    - Several machine learning models are constructed for predicting daily trend of SPY stock.
    - The parameters of the models are fine-tuned by using grid search method to improve the prediction accuracy.
    - All models outperformed benchmark values, with Logistic Regression achieved best performance with accuracy of 70.04% and F1 Score of 0.71.
- Trading Strategy Backtesting
    - A customized trading strategy is proposed to exploit the prediction results of the machine learning model. The backtesting results based on a realistic market simulator indicate both cumulative returns and Sharpe ratio of proposed strategy are higher than the benchmark buy-and-hold strategy.

b) Improvement

There are several aspects of the implementation of this project could be improved. First, while the performance of the machine learning models is better than benchmark results, the best accuracy that can be achieved is 70.04% by Logistic Regression model. The accuracy and F1 Score do not increase significantly even after fine-tuning of the model parameters. This suggests that other input features other than daily returns of the stock indices should be included to improve the prediction performance.

Next, the input features of machine learning models are based on closing price of non-U.S. markets on same day of prediction, which are only available a few hours before U.S. market closes. The proposed trading strategy could not be executed before the input data is available, which may not efficient enough for some investors. The machine learning models could incorporate other features that provide indicators of tomorrow direction of movement of stock index. Then, the trading strategy could be executed when market open daily to fully exploit the advantage of having insights of today market trend. For instance, technical indicators such as moving average, momentum and stochastic reflecting the condition stock price index used in [12] and [13] seem to provide promising results. By utilizing these input features, the original problem can also be extended to obtain prediction for longer terms. How the machine learning models perform under different time spans need to be investigated. Since the proposed trading strategy is designed based on daily trading signals, it also need to be adjusted to adapt to the changes in original problem.

References

[1] T. Z. Tan, C. Quek, and G. S. Ng, "BIOLOGICAL BRAIN-INSPIRED GENETIC COMPLEMENTARY LEARNING FOR STOCK MARKET AND BANK FAILURE PREDICTION1," *Computational intelligence*, vol. 23, pp. 236-261, 2007.

[2] Y. S. Abu-Mostafa and A. F. Atiya, "Introduction to financial forecasting," *Applied Intelligence*, vol. 6, pp. 205-213, 1996.

[3] U.S. stocks hammered as Brexit shock rocks markets. Available: https://www.usatoday.com/story/money/markets/2016/06/24/brexit-bombshell-torpedoes-global-markets/86323890/

[4] Investopedia-Sharpe Ratio. Available: http://www.investopedia.com/terms/s/sharperatio.asp

[5] Quantopian. Available: https://www.quantopian.com

[6] Investopedia-Slippage Defination. Available: http://www.investopedia.com/terms/s/slippage.asp

[7] Investopedia-Buy And Hold. Available: http://www.investopedia.com/terms/b/buyandhold.asp

[8] pandas 0.20.2 documentation ."pandas.DataFrame.fillna". Available: http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.fillna.html#pandas.DataFrame.fillna

[9] Investopedia.com-Look-Ahead Bias. Available: http://www.investopedia.com/terms/l/lookaheadbias.asp

[10] Poposed Trading Strategy Simulated on Quantopian. Available: https://www.quantopian.com/posts/proposed-trading-strategy

[11] Buy-and-Hold Strategy Simulated on Quantopian. Available: https://www.quantopian.com/posts/buy-and-hold-strategy

[12] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques," *Expert Systems with Applications*, vol. 42, pp. 259-268, 2015.

[13] N. Masoud, "Predicting direction of stock prices index movement using artificial neural networks: The case of Libyan financial market," *British Journal of Economics, Management & Trade*, vol. 4, pp. 597-619, 2014.