CSE 482 FINAL PROJECT (Cover Page)

Project Title:
Predicting Future S&P 500 from Economic Indicators using LSTM

Summary of Team Member Participation:

Fill out the following table for each team member of the group.

Name	Participate in data collection	Participate in preprocessing	Participate in data analysis/ experiment	Participate in writing the final report	Participate in creating video presentation	Completed Assigned Tasks
Andreas Frame	x	x	x	x	n/a	yes
Jon Spiwak	x	x	x	x	n/a	yes

Team Member Roles and Contributions:

Name	Roles and Contributions
Andreas Frame	Data collection, primary data preprocessing. Helped with data analysis and experimenting. Primary writing of final report.
Jon Spiwak	Data collection and preprocessing. Primary role in data analysis and experimenting. Helped writing the final report.

I approve the content of the final report (please add your signature below):

Andreas Frame: *Andreas Frame*

Jon Spiwak: *Jon Spiwak*

Predicting Future S&P 500 from Economic Indicators using LSTM

Andreas Frame, Jon Spiwak
Project URL:
https://github.com/akframe123/bigdataproject

ABSTRACT

The S&P 500 Index, hereafter referred to as the S&P, is a stock market index measuring the performance of the top 500 companies listed on stock exchanges in the US. As such, it is an excellent indicator of the US economy as a whole, and is heavily influenced by economic indicators. The project aimed to use indicators such as Treasury bond interest rates, the Effective Federal Funds Rate (DFF) and CBOE Market Volatility Index(VIX) in order to train a Long short-term memory (LSTM) model to predict the future S&P. Using this approach, we were successful in building a model to accurately predict the next days S&P given previous data.

1. INTRODUCTION

The stock market is a massively complex system dependent on a multitude of factors. Despite this, the stock market directly and indirectly affects peoples lives around the country. Particularly in recent times, the volatility and unpredictability of the stock market has been on full display. Consequently, building a model that can predict what the S&P will be when the market next opens is extremely beneficial to traders and normal people alike as it will demystify the movements of the market. As such, modeling and predicting the S&P was a significant undertaking. This was ultimately performed using a LSTM, a specific kind of recurrent neural network designed with the ability to learn long-term dependencies. This was especially useful in this project, as it allowed the model to form dependencies between events over large periods of time.

This project aimed to accurately predict the S&P using historical data from the S&P and other key economic indicators. Economic indicators found to be highly correlated with the S&P include the US treasury bond interest rates, the effective federal funds rate (DFF) and the VIX. These indicators in conjunction with previous S&P data will be extremely useful in the building of an accurate LSTM model.

This project will be achieved through a particular neural network, a LSTM. By training the LSTM using past S&P data as well as the key indicators outlined, the model will be able to draw determinations between the indicators and the S&P over long periods of time.

In order to accurately predict the future S&P, a number of economic indicators needed to be determined and incorporated into the model. It was found that the treasury interest rates, DFF, and VIX were highly correlated with the S&P. The interest rates refer to the amount of interest collected by an individual when purchasing a bond from the US treasury. In regards to this project, the difference between the 10 year bond interest rate and the 1 year bond interest rate was found to be most heavily correlated with the S&P. The DFF, or the effective federal funds rate, refers to the interest rates that banks charge each other on loans. VIX stands for violity index, and is a real-time market index that aims to predict the markets 30 day forward prediction of the volatility of the market. These 3 economic indicators, in addition to previous key S&P data such as open, close, high and more were used to test a LSTM model that was able to accurately predict the future S&P.

Initially we collected a large amount of data on the aforementioned financial indicators. The challenge was more determining what data was relevant and worth collecting and analyzing, this required much research before preprocessing and analyzing the data. Once researched and collected, preprocessing consisted of dropping empty and irrelevant columns and merging the data on valid matching dates. Analyzing was particularly difficult, with many models being attempted before the LSTM was determined to be most promising. Similar to the data collection, this required much research and trial and error before it could be completed.

The project was quite successful in using the model to predict the future S&P close for the next day. This in itself is quite useful, but more forward looking would be informative as well. However, the further in the future we attempted to predict using our model, the more inaccurate it became. We were able to prove the hypothesis that the future S&P close could be predicted using historical data and economic indicators, but also proved how volatile the market is through inaccurate predictions far in the future. We believe that short term movements can be somewhat accurately predicted by data, however long term movements are influenced by factors not always present in the data.

2. DATA

The first dataset that we knew we would need was the S&P historical data[1]. The data was downloaded from the Yahoo Finance website as a CSV file. Following extensive research, the Treasury interest rates on bonds of differing lengths was included as well[2]. This data had to be collected from the US Treasury website[3] and was downloaded as a CSV. This data had to be downloaded a year at a time and compiled into one during preprocessing. The 2 month bonds also had to be dropped as many included null values. In addition to this, the DFF was determined to be a key market indicator[4] and was collected from the Federal Reserve Bank of St. Louis[5]. Finally, VIX data was collected from Yahoo finance[6] following extensive research from the creators of the market indicator, the Chicago Board Options Exchange[7]. Table 1 below shows the data collected during the collection process.

Attribute name	Type	Description
Timestamp	Ordinal	Date data was recorded on.
DFF	Ratio	DFF data as recorded by date.
VIX	Ratio	VIX data as recorded by date.
S&P	Ratio	S&P as recorded by date.
Interest Rates	Ratio	Interest rates for each length of bond by date.

Table 1: Attributes of the data acquired from references.

Each data set collected is a time series, going back 10 years from April 2020. The data on the whole is quite complete, but some challenges must be addressed. Firstly, many columns in each data set are not relevant or incomplete and need to be dropped. This includes the 2 month bonds which do not have information recorded in all previous years. Secondly, and more importantly, the data needs to be compiled such that only dates that exist in every data set will be included in the compiled one. Each raw data set is over 2,000 rows long before any preprocessing and contains many unnecessary columns for our modeling purposes.

Number of observations	10,680
Number of attributes	1335 time steps (Jan 2015-Apr 2020)
% missing values	0%

Table 2: Summary statistics of the raw data from S&P.

Number of observations	18,536
Number of attributes	1327 time steps (Jan 2015-Apr 2020)
% missing values	7%

Table 3: Summary statistics of the raw data from Interest Rates.

Before preprocessing, each data set contained far too much data that would result in overfitting. For each 'set', S&P, DFF, interest rates and VIX, the data had to be preprocessed to remove incomplete data points before merging together into a dataframe. The Interest rates were concatenated from year by year into a single dataframe and 2 month bonds were dropped. Then, the difference between the 10 year and 1 year bond interest rates were extracted. The VIX close was pulled from its set and the rest ignored during preprocessing. The S&P also had too many columns and unnecessary data was dropped, keeping only open, close, high, low, and volume.. Finally, each set was merged on matching dates only to create the data frame. This ensured that only values on dates with all valid data were included. We then used this originally preprocessed and merged data and checked its correlation with the S&P close. We then determined that more preprocessing was required to increase the accuracy and decrease the noise for our model. Through this analysis, we determined the final Dataframe object for the model training should consist of the difference between the 10 year and 1 year bonds as well as S&P historical data (open, close, high, low). This data was then normalized using a min-max scalar before being passed into the model and beginning analysis.

Number of observations	6,625
Number of attributes	1325 time steps (Jan 2015-Apr 2020)
Number of Columns	5
Memory Usage	104,560 mb

Table 4: Summary statistics of the final Dataframe Object

S&P 500 Close	**S&P 500 Open**	**S&P 500 High**	**S&P 500 Low**	**10yr-1yr interest**
0.147148	0.145761	0.150402	0.145721	0.860
0.122987	0.142878	0.132107	0.134133	0.824

Table 5: Top of preprocessed Dataframe Object

Once the preprocessing was completed, the data was split into 75% training and 25% testing. The training data was used to build the LSTM model from the Keras python library. It was used to predict a single column, the closing price for the S&P for the future time step. The model was then applied on the testing data and the predicted closing price was compared to the actual closing price to determine the accuracy.

3. METHODOLOGY

In order to create this model, we decided on implementing a unique recurrent neural network designed to handle long-term dependencies in the data. The preprocessed data was split into 75% training data, and 25% testing. The split was decided to provide enough data for both training and testing, with an emphasis on training the model during less volatile periods of the stock market to ensure it would remain accurate during unpredictable volatile changes like we are seeing currently in the market. The LSTM came from the Keras deep learning library Sequential model in order to predict the S&P closing price. This high level model was then used in a simulated trading market making decisions to buy or sell based off of the predicted values. This was then tested against a simpler LSTM that predicted the closing S&P based only off of yesterday's closing S&P, which too participated in the simulated trading market.

The code can all be found in one Project.ipynb file. At the top, a section is labeled Data collection and preprocessing which entails all of the collecting and preprocessing steps performed during this project. This includes all S&P historical data, VIX, DFF and others that were pulled. It is divided into subcategories where each of these is collected and preprocessed. At the end of this section, the data frame is ready to be passed into the model. In the Analysis section the complex model is trained and tested along with the simple baseline model. The accuracy of predictions was calculated and visualized. The predicted closing prices obtained from the models were then implemented in a simulated market and the results were recorded and visualized.

4. EXPERIMENTAL EVALUATION

This section describes the experimental setup and results you obtain.

4.1 Experimental Setup

This project was performed using Jupyter Notebook with Python, and is reliant on a number of python modules. These include pandas, NumPy, and kiti-learn which are all used through the project in a variety of ways.

In order to ensure the accuracy of our complex LSTM model, we also built a simple LSTM model that was provided with much less information. While the final complex model used historical data from many facets of the S&P as well as the difference between the 10 year and 1 year bond interest rates, the simple model only used historical closing prices of the S&P. These two models were created using the same quantity of data, the first 75% of the previous 5 years, and then applied on the most recent 25%. The predicted values obtained from each model were then fed into a simulated market that used the predicted value of tomorrow's closing S&P to make a decision on whether to buy or sell that day.

Mean Squared Error was used at the penalty while training our models in the LSTM. We chose MSE to be the penalty because in financial modeling loss associated with inaccuracy is not proportional to the inaccuracy. For example, being off by 2 is more than twice as bad as being off by 1. Root-mean-squared-error as well as mean-absolute-error were also used to ensure the accuracy of the predicted results vs the actual market value on the testing dates. To test the overall success, we created a simulated market that executes trades based on the data from our model. Our simulated market allows an investment account to take long and short positions in the S&P based on the predictions of the model.

4.2 Experimental Results

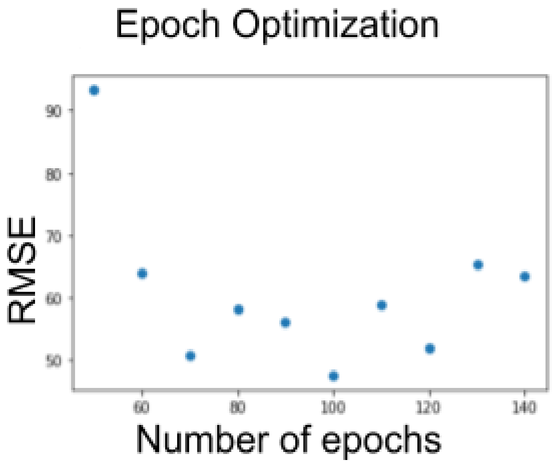

Figure 1: Determining Best number of Epochs

The final model was built using 100 epochs after testing a range of different values and finding the one with the lowest RMSE as seen above in figure 1. This model was then used to predict the closing future S&P and the accuracy was recorded. The model was also tested on a simulated market determining whether to buy or sell based off of the predicted value. Both these results were compared against a baseline model that only used previous S&P close data and accuracy was recorded using RMSE and MAE error metrics.

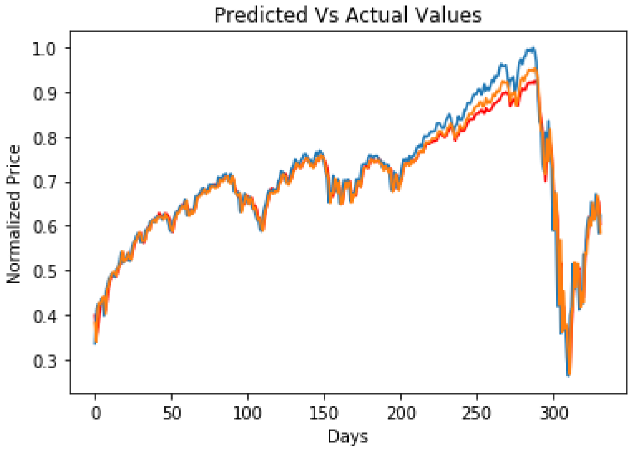

Figure 2: Model predictions and actual closing S&P

Actual in blue, Model in red, Baseline in orange

Interestingly, the simple baseline model was more accurate in predicting the S&P than the complex model. This can be seen above in figure 2. This initial result caused us to believe the complex model was not an accurate predictor, but when we applied it in a mock real world situation using the predicted value to make logical decisions we found it to be far more accurate. Figure 3 below shows how the complex model is able to use the long term dependencies between factors including the S&P highs and lows as well as the interest rate difference on a key bond indicator to profitably buy and sell. Despite the baseline models accuracy at predicting the next days closing, it could not take into account the volatility like the complex model did.

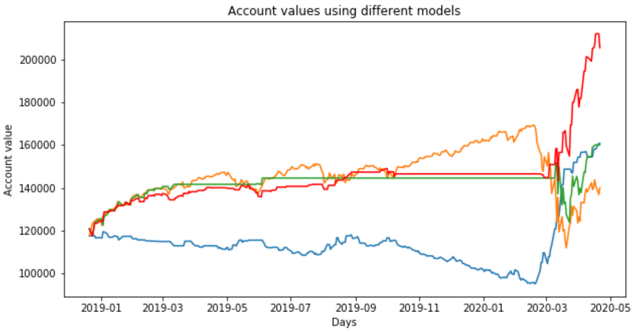

Figure 3: Simulated Market

Market value in orange, Model with long positions in red, model with short positions in blue (the value of the margin is deducted along with interest from the total value), Baseline in green

After training and testing both the complex and simple models on the data we were able to predict the S&P closing accurately enough to profitably buy and sell over long periods of time and beat the growth of the S&P on different simulated accounts.

5. CONCLUSIONS

This project was able to successfully determine relations between economic indicators and historical S&P data in order to profitably engage in the marketplace. Our project was able to beat the S&P 500 benchmark over the course of our test. The S&P grew 19% over the time frame we tested, while our account was able to grow 74% by actively trading with our algorithm. This project can be refined with more indicators that provide a clearer picture of the market such that investors could use it as a tool to increase trading profits. We would do this by using data from all 500 stocks that make up the S&P 500 as well as pulling what's known as daily "candlesticks" on each stock. We were unable to do this because of limitations on data availability.

6. REFERENCES

[1] https://finance.yahoo.com/quote/%5EGSPC/history/

[2] https://www.thebalance.com/treasury-yields-3305741

[3] https://www.treasury.gov/resource-center/data-chart-center/interest-rates/pages/textview.aspx?data=yield

[4] https://www.newyorkfed.org/markets/obfrinfo

[5] https://fred.stlouisfed.org/series/DFF

[6] https://finance.yahoo.com/quote/%5EVIX/

[7] http://www.cboe.com/vix

[8] https://keras.io/