Predicting Stock Market Volatility

Marc, Mitchell, and Andrew

The Problem

Using daily stock data from the S&P500 dating back many years, can we accurately predict the volatility of the market as measured by the VIX (Volatility Index), a measure of expected price fluctuations in the S&P 500 Index options over the next 30 days.

Initial Analysis

We initially identified the dataset we wanted to use, a kaggle dataset of "**Daily Historical Stock Prices (1970-2018)"**

Our X features (predictors)

and a DataHub dataset of **VIX (1993-2018)**.

Our y features (to be predicted)

Data Cleaning Predicted

We performed initial functions on the VIX data, and identified no null values. We dropped all columns except:

- **VIX Closing**
- **Datetime**

For future merging with the other dataset.

Data Cleaning Predictors

We performed initial functions on the stock price data and realized that the dataset included **20,000,000 data points**, and **6 features**.

- The data was free of null values.
- We dropped the adjusted close column, because we already had a closing price feature.

20,000,000 data points is far too much for our machines to process, and would lead to highly unbalanced classes for our target dataset. As such, we decided to **cherry pick predictive stocks** which are representative of different sectors in the stock market.

Feature Engineering

We decided to split up our stock data into the 11 main sectors of the stock market:

- **Energy**, **Materials**, **Industrials**, **Consumer Discretionary**, **Consumer Staples**, **Health Care**, **Financials**, **Information Technology**, **Telecommunication Services**, **Utilities**, **Real Estate**

Feature Engineering

The features we use to predict are the:

open price, **close price**, **high price**,

low price, and **trading volume**

Indexed by day.

Feature Engineering

For each sector we identified the 3 stocks with the highest trading volumes, and added together each of their feature values into a single value to representing the sector.

For example, for the IT sector we concatenated the data from **AAPL**, **AMD**, and **MU**, into one dataframe indexed by datetime. Then for each feature, we added together each stocks value, and made a new column. So for instance, **Opening Price AAPL**, **Opening Price AMD**, and **Opening Price MU**, we added together into a column **Opening Price IT**, and then dropped the individual stocks feature column. We did this for all features.

Feature Engineering

After creating 11 separate dataframes for each sector, with each sector having 5 engineered features, we concatenated all the features into one dataframe, totalling **56 features**, and **3150 data points**.

We then concatenated the **VIX** dataframe on the date column.

We then converted the date column to **datetime**, so we could use an **RNN Model**.

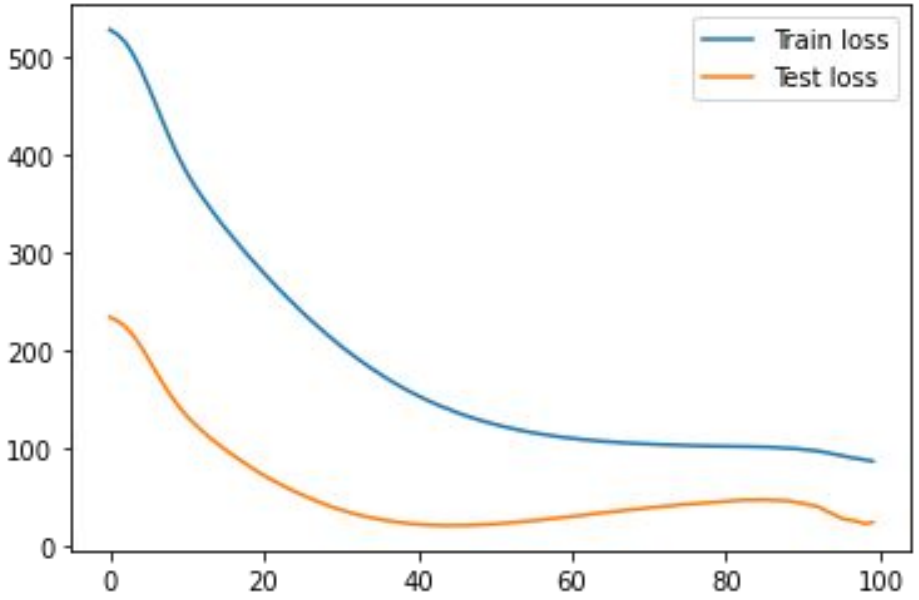
Fitting Model

- In order to fit our model we first had to convert our features to display percent change from day to day.
- Next we ran our train test split, scaled our data and then converted our dataset into a time series

Model 1

GRU RNN using **MSE** as loss function

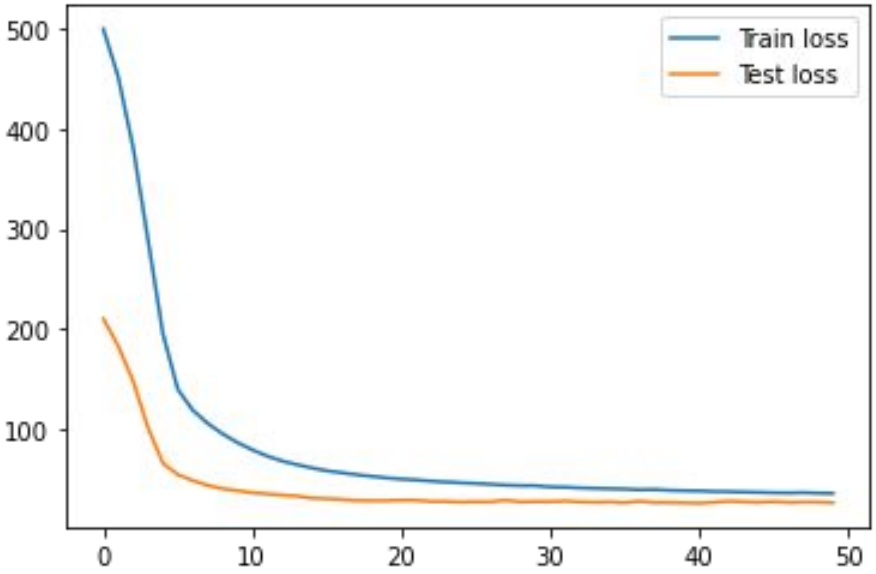
GRU input -> Dense hidden layer with **relu activation** -> Dense output layer



Model 2

Basic RNN with dense input layer and **MSE** loss function

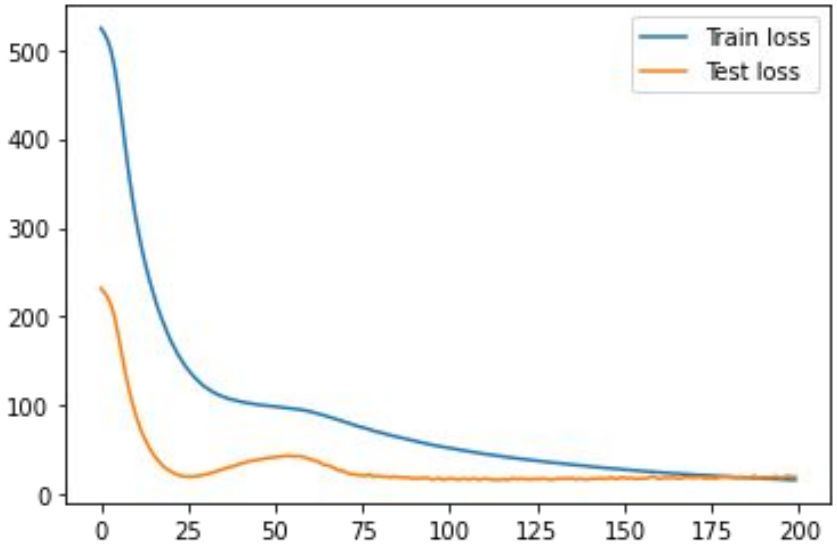
Dense input -> Dense hidden with **relu activation** -> Dense output with **linear activation**



Best Model

LSTM and **MSE** loss function

LSTM input with **tanh activation** -> Dense hidden layer with **relu activation** -> Dense output with **linear activation**



Limitations/Difficulties of the Process

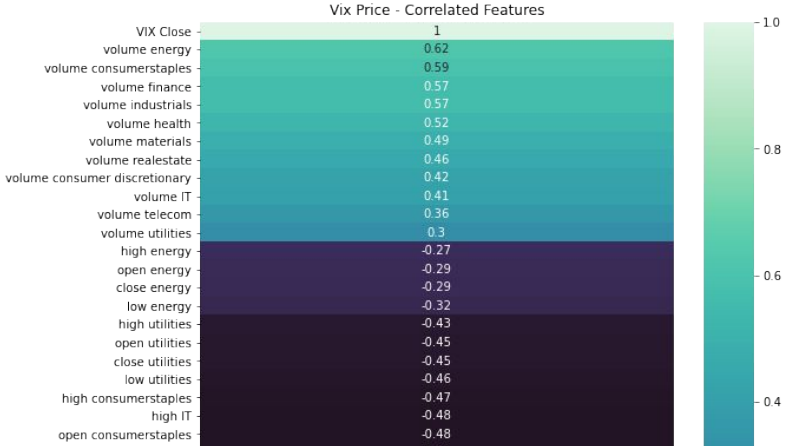
- One of the major limitations of our project is that we used specific stocks to represent sectors of the stock market rather than using our entire dataset.
- Another limitation is that our data cleaning and merging process reduced our dataset to 3,160 data points and 55 features. Ideally we would have significantly more data points than our features squared.
- Given these constraints, our model had surprisingly accurate predictive power.

Limitations/Difficulties of the Process

- Engineering the features into 11 separate data frames by sector proved to be quite tedious given the sheer magnitude of data we had to sift through.
- It additionally required many lines of code with many possibilities for error as we soon realized.

Conclusion

- Stock volume in the energy, consumer staples, and finance sectors were most positively correlated with market volatility
- Stock volume of telecom and utilities were least correlated to market volatility
- We found that higher trading price of stocks was generally correlated with less market volatility



Conclusions

- We found that we were able to use stock data from the S&P500 to accurately predict market as measured by the VIX index
- Specifically, we used the 3 highest volume stocks of 11 main sectors of the stock market as representations of those sectors
    - We used our representations of those 11 sectors as an analog for volatility in the stock market at large
- We were able to create a low bias, low variance, accurate model by using these modeling methods paired with time series and neural net analysis

Next Steps

- Our next steps would be to rerun our models with higher computational resources so that we could use the entire original dataset or more stocks per sector
- Next we would like to see how far in the future we could get our model to maintain its predictive accuracy