COMS-W4995: Applied Machine Learning

Group 19: Kennis Kong (kk3513), Harshini Ramanujam (hr2538), Saravanan Thanu (st3523), Hunter Joseph Agnew (hja2121)
Professor Pappu

Project Report - Stock Price Prediction

Introduction

Stock and financial data have historically been an important and popular topic to analyze and draw inferences from. In this project, we will attempt to use machine learning on stock prices and financial data to predict one-day ahead stock prices. A large issue that comes from trying to use machine learning on financial data is that they are very noisy, and it is challenging to distinguish between signal and noise. We plan to predict stock prices from 5 companies (Pfizer (PFE), Exxon Mobil (XOM), Wells Fargo (WFC), Microsoft (MSFT), and McDonald's (MCD)) using their historical closing price data from the New York Stock Exchange from 2010 - 2017.

Data Exploration

There are 4 datasets: *prices.csv* and *prices-split-adjusted.csv* contain stock prices as daily prices with opening/closing and high/low values. *fundamentals.csv* contains fundamental indicators/aggregates of companies based on annual SEC filings, and *securities.csv* contains descriptions about each company.

Our primary dataset, *prices.csv*, does not contain any missing data, but *fundamentals.csv* had 6 columns of missing data, while *securities.csv* had only 1 column missing 39% of its data. For our prices, we scaled the prices using a MinMaxScaler and changed the data type of the dates into DateTime objects for better manipulation and to resolve inconsistent data types.

We saw that close prices for our 5 stocks generally all had a positive trend while the volume traded tends to decrease across the companies over time. We used moving average graphs to smooth out the curve with less noise and better identify upward or downward trends. Earnings per share was another metric we looked at for each company, but there wasn't a consistent trend we saw across the companies. We also measured volatility for each stock to compare the variation of stock prices and returns over time across the companies. Our data shows that MCD has the lowest volatility, while WFC and MSFT have the highest volatility.

Models

We performed stock price predictions using a LSTM, rolling regression, and ARIMA. To prepare our data for our models, we had to keep our data in sequential order to preserve the time element. Therefore, we used close prices from 2010 to 2015 as our training set, 2015 to 2016 as

our validation set, and 2016 to 2017 as our test set. We implemented separate models for each of our chosen stocks - PFE, XOM, WFC, MSFT, and MCD.

We implemented a LSTM neural network that would use the past 60 days of closing stock prices (MinMaxScaled) to predict the next day's closing stock price. The predicted score would be inverse-transformed to obtain an actual predicted price. The architecture consists of 2 LSTM layers and 2 dense layers. The first LSTM layer has 128 hidden units which feed the information into another LSTM layer that has 64 hidden units. The information is then fed into a dense layer with 25 units and then ultimately a layer with just 1 unit for the price prediction. The model's chosen optimizer was Adam, and the loss/metric was mean squared error. It was trained for 20 epochs with a batch size of 50; hyperparameters were chosen to avoid overfitting.

We implemented rolling linear regression with two models - 1) simple linear regression with alpha as 0 and 2) ridge regression with tuning the alpha value and type of solver to maximize the R-square score using grid search. Since the number of features or rows were not particularly high - it made sense to model with ridge regression.

We implemented the ARIMA (Auto Regressive Integrated Moving Average) model to forecast the closing stock price based on past data. The training data was initially converted to a stationary time-series using first-order differencing and verified using the augmented Dickey-Fuller test. The auto-ARIMA model was used to perform hyperparameter tuning to determine the optimal values for p (Order of the autoregressive model), d (Degree of differencing), and q (Order of moving average model) for each stock.

Analysis

For the LSTM model, the chosen hyperparameters achieve a good balance between computational efficiency and performance. Plotting the MSE over the epochs for all the stocks (figure 1), we see the MSE for both the training data and validation data decrease continually for the 20 epochs. When using the model to predict, we looked at RMSE and MAE metrics (figure 3) to see that the model generally performed decently for PFE, XOM, WFC, and MSFT, but it did not perform as well for MCD. It performed best with PFE, but in all cases, the model seemed to predict prices that corresponded with the price a few days before. There is some lagged error, and this is especially apparent with MCD with price predictions lagged a few weeks behind (figure 2).

For the rolling linear regression models:

1. For each stock chosen, on tuning the ridge regression model - the alpha tended to be close to 0. Upon increasing alpha to higher values (above 0.1), the performance was

lesser. Essentially, the "best" model from grid search was chosen to be a simple linear regression model with alpha 0.

2. Therefore, when comparing the best model from the tuned ridge model and regular linear regression model, the performances were identical. So we will report the scores and stick to analyzing our visualizations and metrics from the simple rolling linear regression model.

Each model reported a score of above 90% using R^2 values (figure 5). The model with the poorest performance (~90%) is for the stock XOM. In our visualization, when we check the plot of the closing prices for each stock over time (figure 4), we can see that these results make sense. All the companies we have chosen have an upward growing trajectory of stock prices which seem to form a linear line. XOM varies the most from a straight line trajectory with there being a dip between 2014-2015. This explains why the model performs poorly for this stock prediction.

However, in general from looking at the actual v/s predicted plots for each stock (figure 6)- there does seem to be some overfitting. Considering, most of the stocks were increasing constantly, it might be possible that on predictions with recessions the models won't perform as well.

For the ARIMA model, the hyperparameters determined by the auto-ARIMA function were optimal. This is confirmed by the plot of residuals and correlograms (figure 7) that shows the stationarity of the time series. An interesting observation is that the degree of differencing(d) parameter is 1 for all the stocks, whereas the other parameters p and q differ. We used RMSE and MAE metrics (figure 8) to validate the trained model. We can see that the model performed best with PFE and slightly worse with XOM and MCD. From the actual vs predicted plots (figure 9), we can observe that the predictions are quite good and are able to capture the trends correctly.

Conclusion

Of our three models, it seems that rolling regression performs the best in terms of RMSE and MAE while LSTM and ARIMA performed similarly to each other but worse than the rolling regression. The LSTM model could further be improved by possibly feeding in more data and trying different architectures. We could also add in additional covariates using fundamental information from the other data sets. Overall, our models performed fairly well in capturing the data in the limited time periods that we had.

Appendix

Figure 1. LSTM MSE Over Epochs for PFE, XOM, WFC, MSFT, and MCD

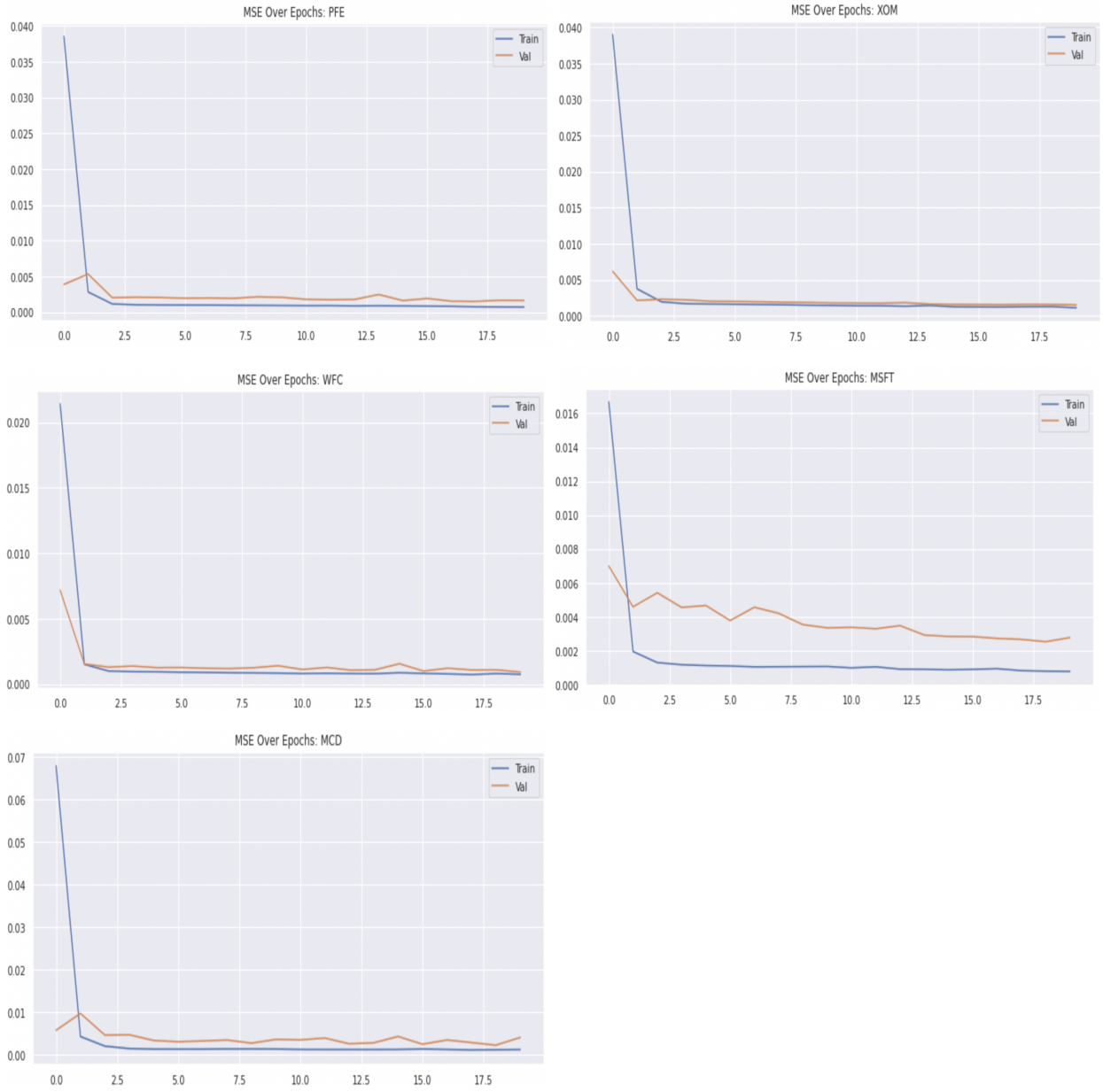


Figure 2. LSTM Stock Price Predictions for PFE, XOM, WFC, MSFT, and MCD

Figure 3: LSTM Model Metric Scores

Company	RMSE	MAE
PFE	0.765	0.577
XOM	1.400	1.054
WFC	1.416	1.003
MSFT	1.289	1.035
MCD	3.782	3.300

Figure 4: Plot of closing prices (USD) over time for our selected stocks

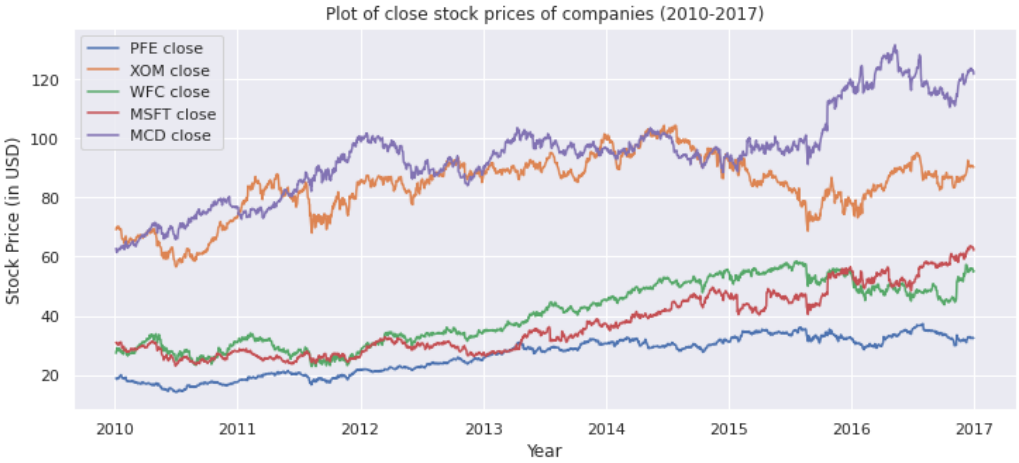


Figure 5: Simple Linear Regression Model Metric Scores

Company	RMSE	MAE	Score (R^2)
PFE	4.34×10^{-4}	1.46×10^{-2}	0.944
XOM	3.54×10^{-4}	1.35×10^{-2}	0.904
WFC	4.70×10^{-4}	1.57×10^{-2}	0.947
MSFT	7.08×10^{-4}	1.86×10^{-2}	0.966
MCD	7.33×10^{-4}	1.99×10^{-2}	0.951

Figure 6: Rolling Linear Regression Stock Price Predictions for our selected stocks

Figure 7: Plot of Auto-ARIMA during Hyperparameter tuning

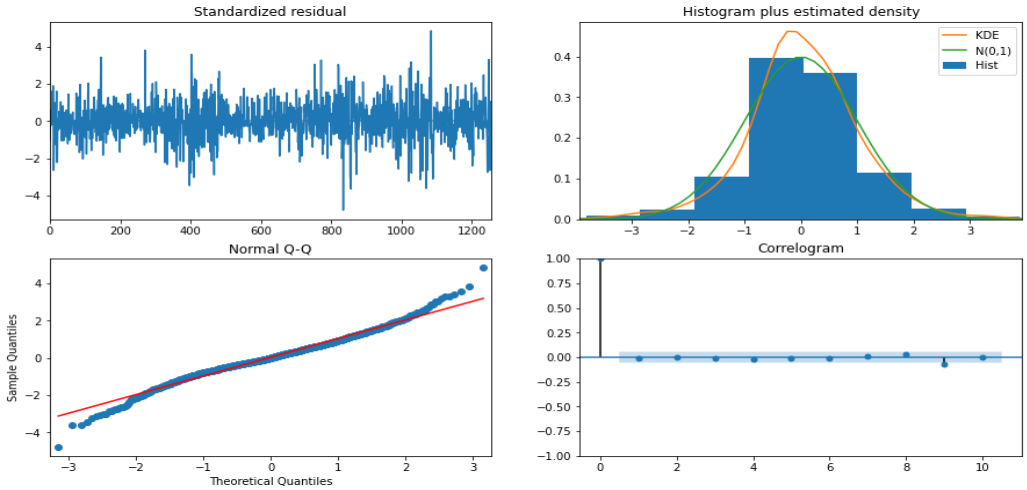
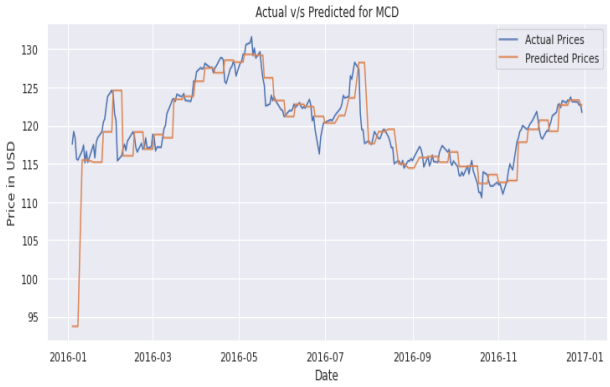


Figure 8: ARIMA Model Metric Scores

Company	RMSE	MAE
PFE	0.625	0.425
XOM	2.715	1.423
WFC	1.412	0.956
MSFT	1.608	0.983
MCD	3.956	1.921

Figure 9. ARIMA Stock Price Predictions for PFE, XOM, WFC, MSFT, and MCD


Actual v/s Predicted for MCD
Actual Prices
Predicted Prices
Price in USD
130
125
120
115
110
105
100
95
2016-01
2016-03
2016-05
2016-07
2016-09
2016-11
2017-01
Date