Julian Wiley

CSCI E-116 Dynamic Modeling and Forecasting in Big Data

12/16/22

Final Project – Time Series Analysis and Forecasting in Intraday Stock Data

# Executive Summary

## Introduction

One of the biggest applications for time series forecasting is stock market prediction, which presents its own unique challenges. Despite being an active area of research, there is not as much public information on intraday price prediction, mainly due to the cost of data access and the financial imperative for corporate players to keep developments secret. For my project, I will be examining high frequency stock data provided by AlgoSeek, a retail data provider specializing in machine learning research. I will demonstrate time series analysis techniques along with modern machine learning methods that have been shown to perform well on similar datasets.

## Dataset Description

AlgoSeek providesproprietary high frequency equity data tailored for machine learning model development called the Trade and Quote Minute Bar (TAQ) dataset. The data describes stock trades, options, and investment activity that has been aggregated from the raw, millisecond time scale into minute bars. This dataset is a combination of the trade dataset, which refers to actual trades that have been completed, and the quote dataset, which describes the options for each minute such as the price of different bids/asks for different securities. There is also datapoints on the volume of different trades, both for retail traders and institutional investors allowing us to distinguish between the trading activity for individuals and companies, which has implications for the pricing and volatility of securities. This dataset provides information on premarket, post market, and regular market hours. We split the dataset into three for exploration and modeling, because the pre and post market activities are not bound by constraints that we need in order to forecast. The data that we use is for a single stock, Apple Inc (NYSE: AAPL), from October 14th, 2021 to October 12th, 2022.

## 3 . Preprocessing

In order to conform to the prerequisites for predicting time series datasets, we must create stationary data for forecasting. In order to do this, we calculate different indicators based on the TAQ dataset. These indicators start with returns for 1/5/10/15/20/25/30/60/120/180 minute bars, which reflects the profit you would have made if you had bought the stock in the beginning of that time interval and sold it by the end. Next, we calculated different time series specific values such as moving averages, exponential moving averages (EMA), commodities channel index (cci), Chaiken Money Flow, volatility, and momentum (see Figure 1 for full list of indicators used). We calculated EMAs and Volatility for multiple time frames, based on the different return series. Because the TAQ dataset is not stationary, we focused on the calculated factors for forecasting. When building our models, we split the dataset into train and test sets, training our models on data from before October 2nd, 2022 and testing their performance on the remaining rows.

## Models

For forecasting, we used a combination of time series models, linear regression models, and machine learning methods to predict multiple different return series. Our dependent variable for prediction is the fwd1min variable, which represents the future 1-minute returns for the dataset calculated from the 1-min return variable. Returns represent the relative profit that would have been generated by purchasing the stock at the beginning of the time bar and selling at the closing price. Linear regression was the least successful model type, likely due to its inability to take into account different structural variations in our time series. Our first model used all variables to predict fwd1min and had an adjusted R2 value of 0.0002249. The second linear model used just the return series and had an adjusted R2 value of 0.000247. Our third model only used the EMAs and had an adjusted R-squared value of. We then used volatilities and produced an adjusted R-squared value of -0.00001986. Then, we used the remaining indicators to produce a model with an adjusted R-squared value of 0.00004432. Our sixth and final linear model used lagged predictors and had an adjusted R-squared of 0.0001923. Givent that these models all had incredibly low adjusted R-squared values, we did not calculate the RMSE on the test set. For the next model type, we used time series specific regression models such as ARIMA and structural time series forecasting, which performed much better than the simple linear regression, but still relatively poor. The RMSE value for these models were 0.0009181449 and 0.0009177113, respectively. Afterwards, we used prophet to predict fwd1min, using just the fwd1min values as predictors, which gave us a RMSE of 0.0007595774. The best models as judged by RMSE were neural networks, with the univariate LSTM having a RMSE of 0.0000051017 and the multivariate LSTM having am RMSE of 0.0000051013. However, we were unable to get these models to forecast returns on the test set.

## Conclusion

In building neural networks, we used the Python libraries, TensorFlow and Keras, which are popular deep learning libraries. These all had very small Losses (RMSE) with the best performing models being multivariate. This follows the outcome of several different papers focused on similar predictions. This is likely because these models have architectures that are specifically designed to do very well on problems involving data that is contextual. This means that solving the forecasting problem requires information about each datapoints place within the entire dataset, or a local neighborhood of data points.

# Figures

Text

Description automatically generated

Table

Description automatically generated with medium confidence

Figure - Full Dataframe

Chart

Description automatically generated

Figure - Combined OHLCV Chart

![Chart

Description automatically generated]()

Figure - Trade OHLC Chart

![A picture containing chart

Description automatically generated]()

Figure - Returns

Chart

Description automatically generated

Figure - Zoomed in plot of Returns

Table

Description automatically generated

Figure - Basic Linear Regression Summary

Chart, box and whisker chart

Description automatically generated

Figure - ARIMA Forecasts

Chart, box and whisker chart

Description automatically generated

Figure - Local Linear Structural Model Forecasts

Chart, scatter chart

Description automatically generated

Figure - Prophet Forecasts

A picture containing timeline

Description automatically generated

Figure - Prophet Components plot

Chart, scatter chart

Description automatically generated

Figure - Univariate LSTM Losses

A picture containing timeline

Description automatically generated

Figure - Univariate LSTM Predictions

Shape

Description automatically generated

Figure - Multivariate LSTM Losses

A picture containing text, antenna, screenshot

Description automatically generated

Figure - Multivariate LSTM Forecasts

Table

Description automatically generated

Figure - Model Performance on the test set