Improving Price Prediction Accuracy for Magnificent Seven Stocks Utilizing Cross-Stock Historical Data

Independent Study Project

ACENET Microcredential in Advanced Computing

Kunasekaran Nirmalkanna

July 31, 2024

Abstract

This project explores improving stock price prediction accuracy for the "Magnificent Seven" stocks by incorporating historical data from other stocks into traditional AR(1) models. The study finds that including multiple lags of cross-stock prices enhances prediction accuracy in some instances.

1 Introduction

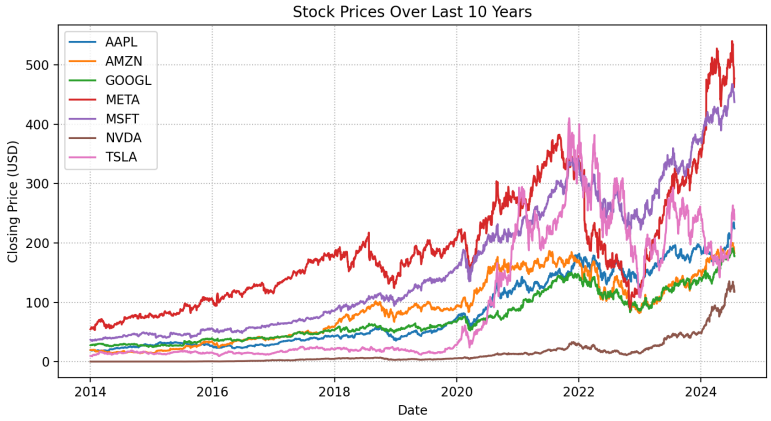
Stock price prediction has been a focus of research for decades, with traditional methods like technical and fundamental analysis as foundational approaches (Fama, 1970; Tsay, 2005). However, the complex and dynamic nature of the stock market has necessitated the exploration of more refined models. Recent literature showcases a surge in the application of machine learning algorithms, including artificial neural networks (Jadhav et al., 2018), support vector machines (Tay and Cao, 2001), and recurrent neural networks (Lipton et al., 2015), to capture intricate patterns within stock price data. Furthermore, researchers have incorporated alternative data sources, such as social media sentiment and economic indicators, to enhance predictive capabilities (Bollen et al., 2011; Hu et al., 2021). Despite these advancements, achieving consistently accurate predictions remains elusive due to financial markets' inherent volatility and noise.

The objective of this project is to explore the potential of improving stock price prediction by incorporating the historical prices of other stocks. The study particularly focuses on the "Magnificent Seven Stocks", which are listed beside Figure 1.

The key research question here is whether the AutoRegressive Integrated Moving Average (ARIMA) method (Ariyo et al., 2014), a popular method for stock price prediction, can be enhanced by using the historical prices of other stocks.

2 Data Collection and Preparation

Market data was obtained using the `yfinance` package in Python (Aroussi, 2019), which provides a convenient way to download data from Yahoo! Finance. The targeted data spans from January 1, 2019,



- **AAPL**: Apple Inc.
- **AMZN**: Amazon.com Inc.
- **GOOGL**: Alphabet Inc. (Google)
- **META**: Meta Platforms Inc. (Facebook)
- **MSFT**: Microsoft Corporation
- **NVDA**: NVIDIA Corporation
- **TSLA**: Tesla Inc.

Figure 1: Last 10 Year Stock Prices

to July 19, 2024, and includes closing prices for the magnificent seven stocks. All the prices are in the US dollar.

3 Analysis

3.1 Preliminary Analysis - Granger Causality

Granger causality (Granger, 1969) is a statistical method used to determine whether one time series can predict another. Unlike true causality, Granger causality assesses if past values of one variable contain information that helps forecast future values of another variable. This is achieved through hypothesis testing, where the null hypothesis states that one time series does not Granger-cause another.

For two time series X_t and Y_t

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p} + \phi_1^* X_{t-1} + \ldots + \phi_p^* X_{t-p} + \epsilon_t, \qquad t-1, \cdots, n, \tag{1}$$

where $\epsilon_t \sim N(0, \sigma^2)$.

The null hypothesis (H_0) states that X_t does not Granger-cause Y_t, implying that the coefficients ϕ_j^* in Model 1 are all zero. The alternative hypothesis (H_1) suggests that at least one ϕ_j^* is non-zero. If the P-value is significantly small, we reject the null hypothesis, indicating that X_t Granger-causes Y_t. `grangercausalitytests` function in `statsmodels` package in python (Seabold and Perktold, 2010) is used to conduct the test.

Figure 2 shows the results of Granger causality tests between different pairs of stock tickers. Each subplot represents a test where the p-values for different lags are plotted. P-values from four different tests (`params_ftest`, `ssr_ftest`, `ssr_chi2test` and `lrtest`) for Granger causality are plotted in the Figure 2. Note that all four tests give similar p-values. The p-values tend to decrease when the lag increases. Notably, all the other stocks, Granger-cause the TSLA stock to at least from lag 5. Meanwhile, for NVDA, small lags of other stocks do not Granger-cause. Other stocks have mixed Granger-causality patterns among them. Specifically, AMZN does not Granger-cause META at any lag level. Overall, these results highlight detailed causal inter-dependencies among these major stocks, particularly at small-lag values.

3.2 Methodology

The methodology section outlines two models. Model 1 (M1), based on the AR(p) model, is defined as:

$$Y_t = \phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p} + \epsilon_t, \quad t = 1, \ldots, n,$$

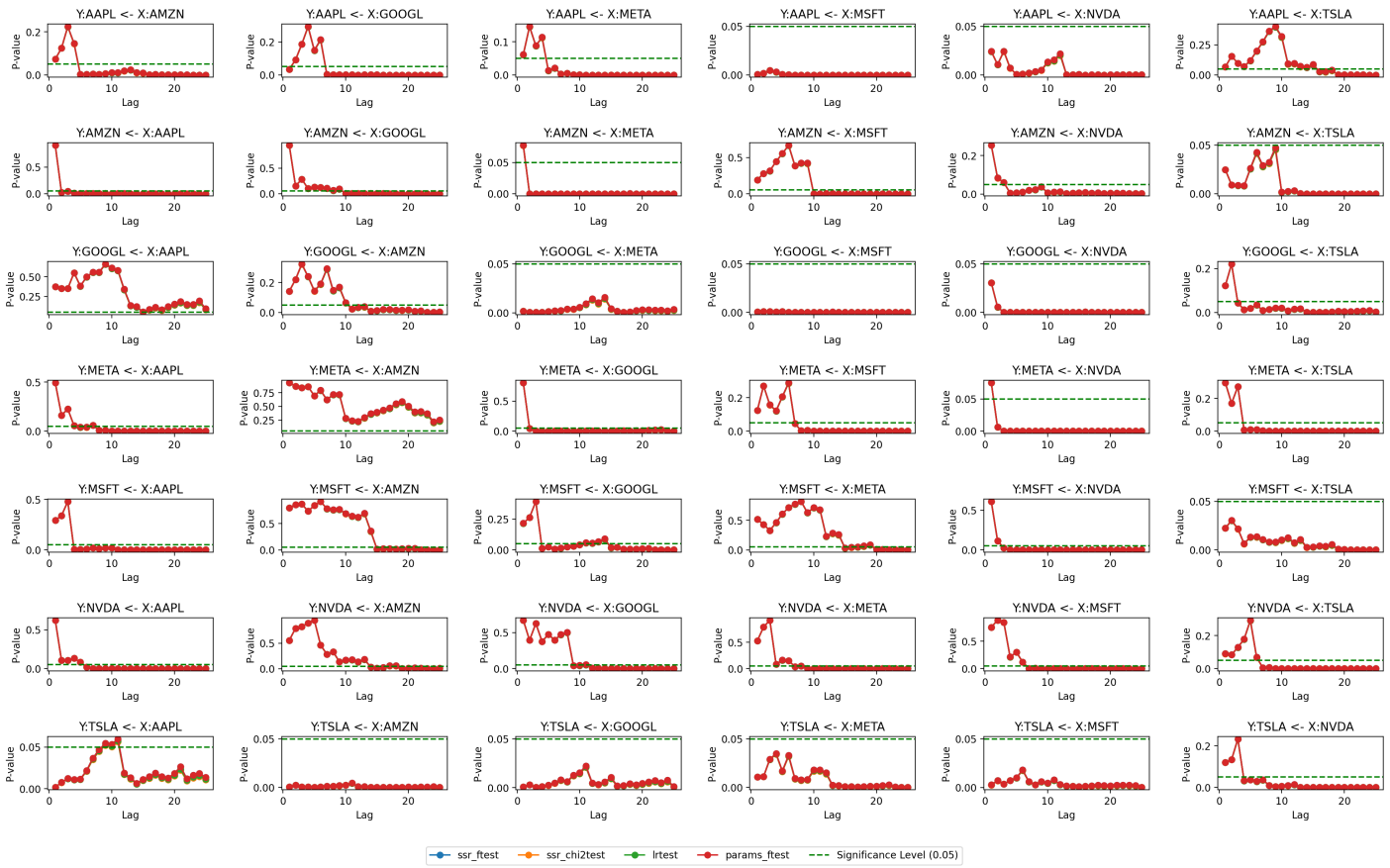


Figure 2: P-values of Granger causality.

where Y_t is the target time-series and $\epsilon_t \sim N(0, \sigma^2)$. The `LinearRegression` function from the `sklearn` package (Pedregosa et al., 2011) is used to fit this linear model, considering $n = 2000$ (approximately 8 years) to estimate $\phi_1, \ldots, \phi_p$. With the estimated ϕs, calculate the Y_{t+1} and denote the estimated value as $\hat{Y}_{t+1}$. The goal is to minimize the root mean square error $\left(RMSE = \sqrt{\frac{\sum_{j=1}^{k} \left(Y_j - \hat{Y}_j\right)^2}{k}} \right)$.

Model 2 (M2) extends the AR(p) model by including external predictors:

$$Y_t = \phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p} + \phi_1^* X_{t-1} + \ldots + \phi_{p_x}^* X_{t-p_x} + \epsilon_t, \qquad t - 1, \cdots, n, \tag{2}$$

and considered both single lag and multiple lags for external predictors time-series X_t. When the RMSE of M2 is less than M1, the prediction of M2 is considered to be improved. For simplicity, this study considers only $p = 1$, assuming that the current price depends only on the previous day's price of target stock. For M2, $p_x = 1, \cdots, 5$ values are investigated.

4 Results

The differences in RMSEs between M1 and M2 (RMSE(M1)-RMSE(M2)) are plotted below to analyze the improvements of M2 over M1. This section investigates two types of M2s, one with a single lag and the other one with multiple lags of other stock (X_t).

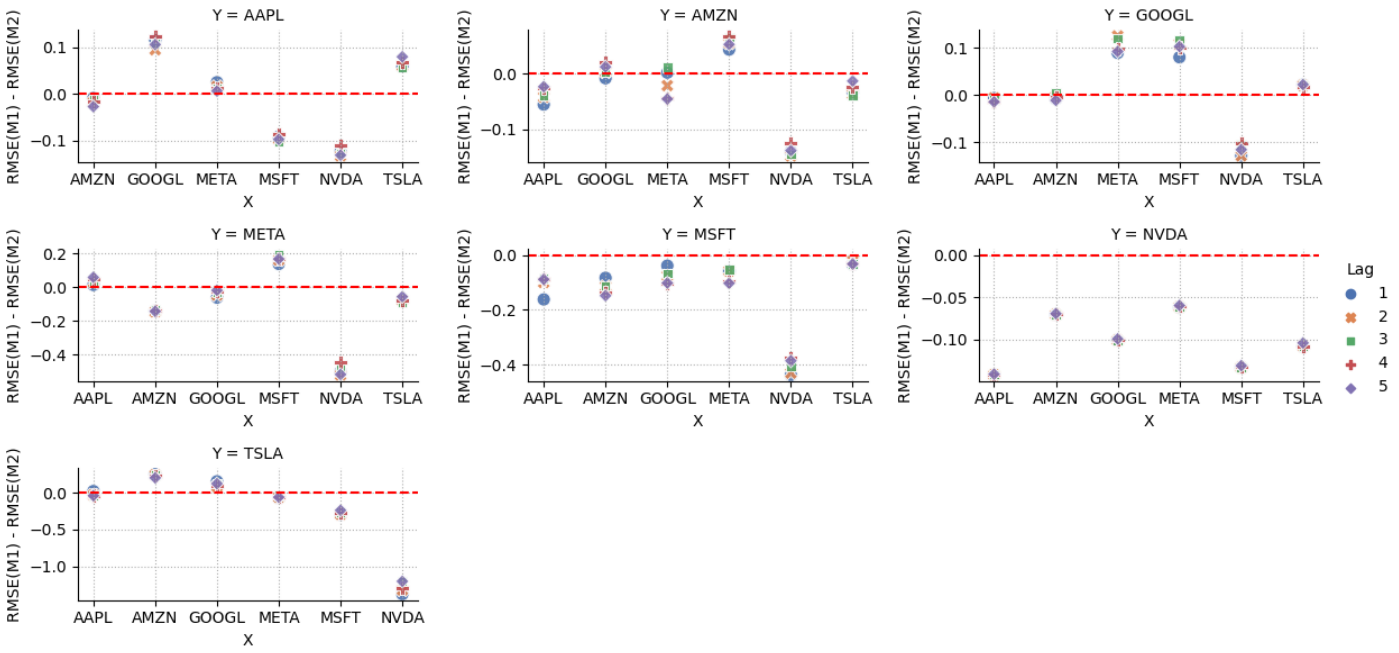


Figure 3: The difference in RMSE (between M1 & M2) across different single lag for each target stock.

4.1 Single Lag of X_t

The plot shows the difference between RMSE values of two models, M1 and M2, for different ticker pairs across single lags (1 to 5). Each subplot represents a different response stock (Y), and within each subplot, various stocks are used as predictors (X). The Y-axis indicates the RMSE difference, RMSE(M1)-RMSE(M2), with a positive value suggesting that M2 performs better than M1, and a negative value indicating the opposite. The red dashed line at zero serves as a baseline, where both models have equal performance. Each lag is depicted with different markers and colors.

According to Figure 3, adding any single lag (1 or 2 $\cdots$ 5) of other stock prices do not result in much difference in model prediction improvement, except for AMZN an MSFT. Single lags of GOOGL or TSLA stock prices can improve AAPL's stock price prediction. Similarly, META and MSFT can improve GOOGL's stock price prediction. Furthermore, MSFT can be utilized to predict both AMZN and META stock prices. The price predictions of NVDA and MSFT do not have any improvement by M2 than M1 with any single lag.

4.2 Multiple Lags of X_t

Adding multiple lags of predictor stock prices has a similar outcome as adding single lags. However, in some cases, there are considerable improvements in the M2 when multiple lags are added. In particular, adding the first five lags of NVDA to M1 drastically improves the prediction of META. Adding the first 2 lags of MSFT or NVDA to M1 drastically improves the prediction of GOOGL. The first five lags of AMZN can also be used to improve the prediction of AAPL.

For example, in Figure 5, both models capture the overall trend and the short-term fluctuations of the GOOGL stock prices well, although neither is perfect in predicting every minor movement. For GOOGL, including META lags (M2) does not significantly improve the predictive performance compared to M1. Both models perform similarly.

Whereas in Figure 6, there are considerable differences between the predictions of M1 and M2, especially in the regions where the stock price has rapid changes with lower RMSE for M2. This suggests that including multiple lagged values of NVDA stock prices (in M2) introduces considerable

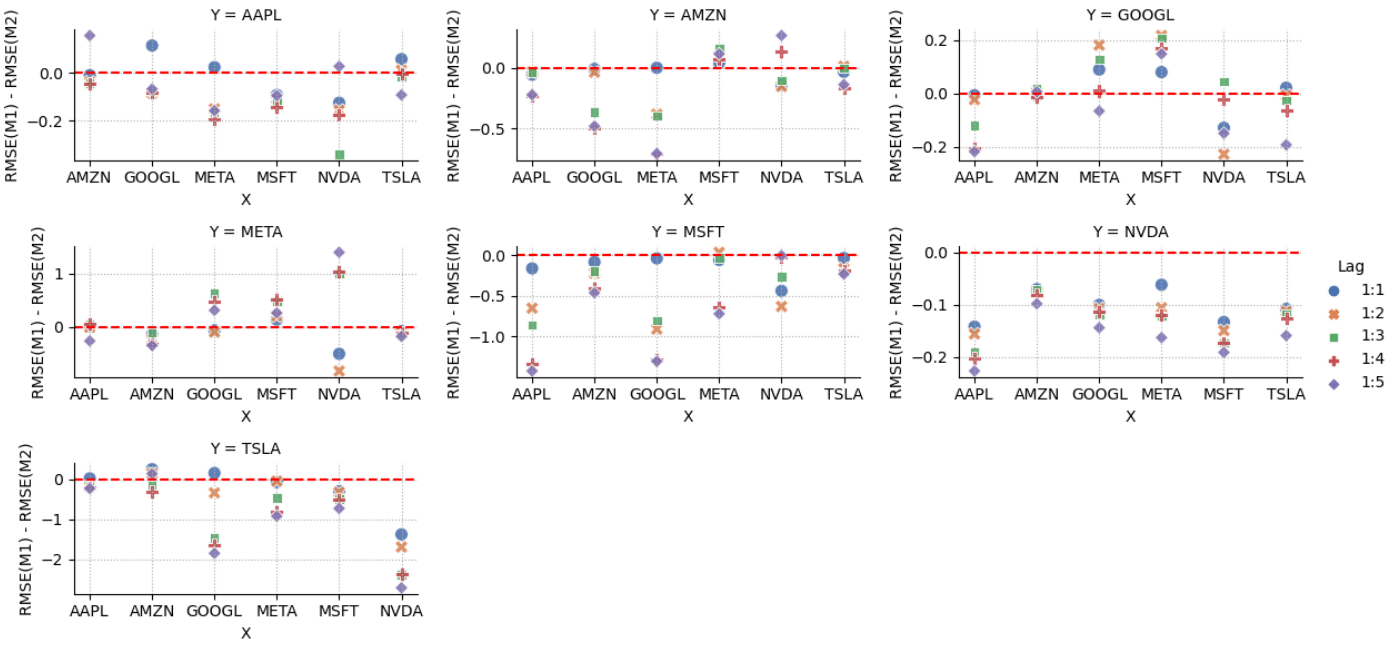


Figure 4: The difference in RMSE (between M1 & M2) across different multiple lags for each target stock.

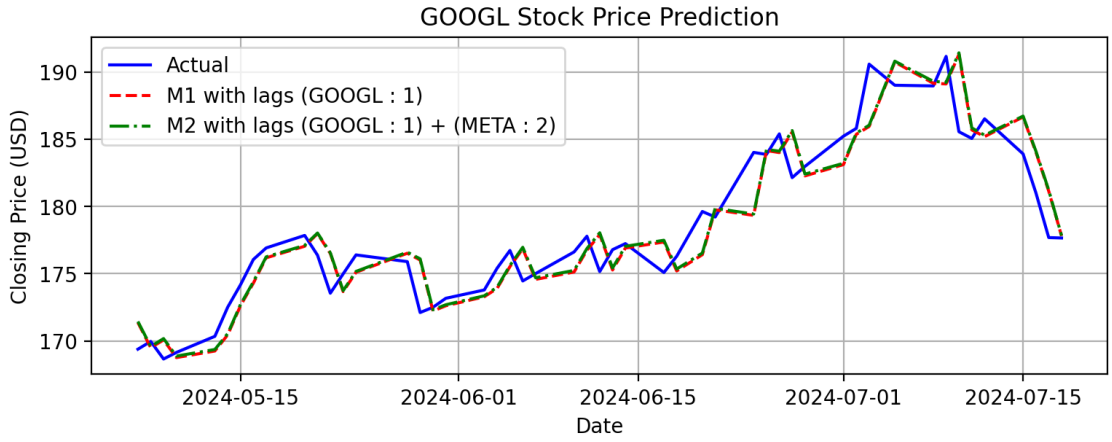


Figure 5: GOOGL Stock Price Prediction

improvement in the predictions compared to using only META lagged values (in M1).

5 Discussion

Since M1 is an autoregressive AR(1) model, initially, `ARIMA` function in `statsmodels` package in python (Seabold and Perktold, 2010) was considered in the study. Later realized that `ARIMA` cannot be used to implement the M2. Therefore, finally, `LinearRegression` function from the `sklearn` package (Pedregosa et al., 2011) is used conduct the whole study.

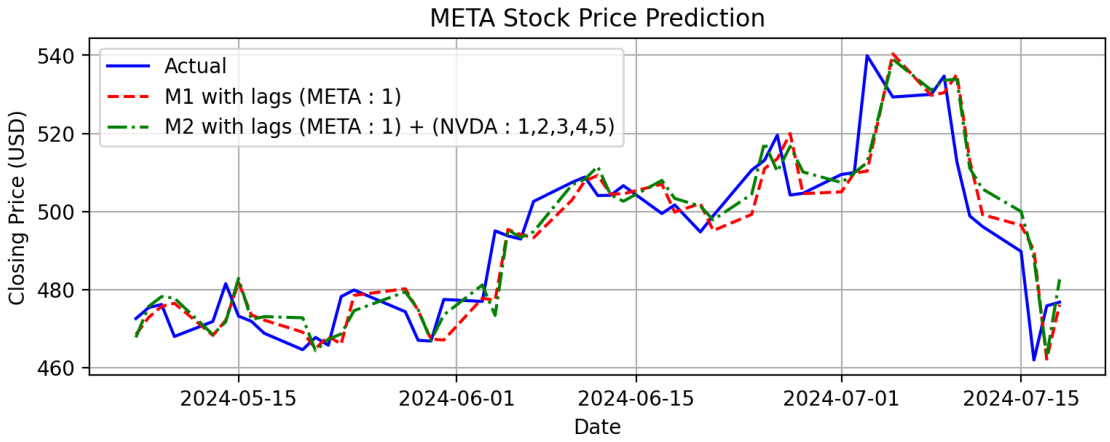


Figure 6: META Stock Price Prediction

6 Conclusion and Future Work

In conclusion, while incorporating other single lagged stock prices into the predictive model does enhance stock price prediction, the improvement is not substantial enough to be considered significant. However, including multiple lags result in much improvement in the prediction model in some cases.

Future work could involve considering different values for n (number of observations) beyond the approximately 8-year span (n = 2000). Exploring various values for p (lags in the model) and extending the number of lags for external predictors p_x could provide further insights. Refining model M2 to include more complex interactions of lagged values and expanding the study to include stocks beyond the Magnificent Seven could also be beneficial.

References

Ariyo, A. A., Adewumi, A. O., and Ayo, C. K. (2014). Stock price prediction using the arima model. In *2014 UKSim-AMSS 16th international conference on computer modelling and simulation*, pages 106–112. IEEE.

Aroussi, R. (2019). yfinance: Yahoo! finance market data downloader. Version 0.2.41.

Bollen, J., Mao, H., and Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8.

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *Journal of Finance*, 25(2):383–417.

Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: journal of the Econometric Society*, pages 424–438.

Hu, Z., Zhao, Y., and Khushi, M. (2021). A survey of forex and stock price prediction using deep learning. *Applied System Innovation*, 4(1):9.

Jadhav, S., Dange, B., and Shikalgar, S. (2018). Prediction of stock market indices by artificial neural networks using forecasting algorithms. In *International Conference on Intelligent Computing and Applications: ICICA 2016*, pages 455–464. Springer.

Lipton, Z. C., Berkowitz, J., and Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Seabold, S. and Perktold, J. (2010). statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*.

Tay, F. E. and Cao, L. (2001). Application of support vector machines in financial time series forecasting. *omega*, 29(4):309–317.

Tsay, R. S. (2005). *Analysis of financial time series*. John Wiley & Sons.

7 Supplementary Material

GitHub link: `https://github.com/Kannastat/ISP_ACENET_knirmalkanna`