Performance Evaluation of GAN, ARIMA, and LSTM Models in Bitcoin Price Prediction

Yu Du, Yikun Chen, Zheng Bao, Zhiyuan Zhou & Chao Li
Department of Computer Science
University of Southern California
Los Angeles, CA 90017, USA
{ydu35952, yikunche, baojames, zzhou534, cli49003}@usc.edu

1 Introduction

The inception of this project is driven by the critical goal of enhancing the prediction accuracy of Bitcoin prices, a significant financial topic. Existing methodologies in this domain frequently encounter challenges stemming from the inherent volatility of cryptocurrency markets, resulting in predictions that may lack reliability and, on occasion, prove misleading. In response to these issues, our strategy involves conducting a comparative analysis of multiple models, including AutoRegressive Integrated Moving Average (ARIMA), Long Short-Term Memory Networks (LSTM), and Generative Adversarial Networks (GAN). Additionally, we incorporate traditional linear regression and ensemble learning models as our baseline to provide a comprehensive evaluation framework. We observed that while individual models have been explored in isolation, a comprehensive comparative study that encapsulates the strengths and weaknesses of these models in tandem is lacking. Our solution involves a careful examination of these models. We assess their accuracy, reliability, and computational efficiency specifically for Bitcoin price forecasting.

2 Related Work

2.1 AutoRegressive Integrated Moving Average (ARIMA)

There are many research works done previously on predicting Bitcoin price using ARIMA. Latif et al. (2023) evaluated the performance of ARIMA(3, 1, 3), which can capture the general trend of Bitcoin Price. However, it fails to predict the direction of the price. Si (2022) presented two ARIMA models, namely ARIMA(5, 2, 1) and ARIMA(0, 2, 2), and assessed their performance on Bitcoin Price Prediction. Both the models passed residual tests, while ARIMA(0, 2, 2) has lower values of AIC, AICc, and BIC than ARIMA(5, 2, 1) does. In our project, we are going to pick the model with the smallest AIC possible.

2.2 Least Absolute Shrinkage and Selection Operator (LASSO), Tikhonov Regularization (Ridge), Elastic Net, Random Forest, and Extreme Gradient Boosting (XGBoost)

Chevallier et al. (2021) outlined a few machine-learning algorithms, and we selected some of them, specifically LASSO, Ridge, Elastic Net, SVM, and Random Forest to serve as a comparison baseline. These models have been widely used in econometrics and quantitative finance-related research. Chevallier et al. (2021) concluded that Bitcoin prices differ from traditional financial derivatives and react more to news related to other cryptocurrencies. Chen (2023) studied Bitcoin price prediction with random forest regression and LSTM. They showed that random forest was more accurate than LSTM in this task. Ranjan et al. (2022) researched Bitcoin price prediction at high-dimensional features and high-frequency data. They concluded that Logistic regression and Linear Discriminant Analysis models are superior for data with high-dimensional features and XGBoost outperforms other algorithms when trained with high-frequency data.

2.3 Generative Adversarial Network (GAN)

Limited research has explored stock price prediction using Generative Adversarial Networks (GANs), with inconsistent outcomes observed across various combinations of generator and discriminator choices. Zhou et al. (2018) presents a framework integrating LSTM and CNN with adversarial training, showcasing improved accuracy in predicting stock price direction and reduced forecast error, validated through simulated trading scenarios. In contrast, Romero compared GAN performance with LSTM against the specific task of predicting one-day price increases, finding no significant differences and, in some cases, slightly worse performance for GAN. Zhang et al. (2019) propose a GAN model using LSTM as the generator and MLP as the discriminator, demonstrating superior performance in predicting one-day closing stock prices compared to baseline LSTM. Additionally, Gulrajani et al. (2017) introduce a GAN architecture with a generator based on Gated Recurrent Unit (GRU) and a discriminator employing CNN, along with an enhanced discriminator using Wasserstein distance Gradient Penalty (WGAN-GP). Experimental results highlight the superiority of the proposed GAN model over traditional approaches, with both basic GAN and WGAN-GP exhibiting enhanced performance.

2.4 Long short-term memory (LSTM)

López-Cabarcos et al. (2021) provided insights into how Bitcoin volatility is influenced by investor sentiment, S&P 500 returns, and VIX (Volatility Index) returns, highlighting Bitcoin's role as a potential safe haven or speculative asset under different market conditions. This paper prompted our team to incorporate S&P 500 and VIX data into our deep learning models to potentially enhance their predictive capabilities. In addition, as described by Cheng et al. (2024), the use of a combination of ML, SARIMA, and Facebook Prophet models in forecasting Bitcoin prices provided a new way of addressing the under-explored area of forecasting Bitcoin volatility. This study underscored the criticality of incorporating volatility as a pivotal dimension in our analysis and elucidated how disparate models exhibit varied performances in response to unpredictable events, such as warfare and global pandemics.

In the realm of financial forecasting, particularly for Bitcoin prices, LSTM (Long Short-Term Memory) networks have emerged as a prominent tool. Ferdiansyah et al. (2019)'s study "A LSTM-Method for Bitcoin Price Prediction: A Case Study Yahoo Finance" developed an LSTM-based model to tackle the volatility of Bitcoin in the stock market, showcasing the capability of LSTM as an advanced RNN module. Complementing Li (2022), "Prediction of Bitcoin Price Based on LSTM" employed transaction data from 2014 to 2017 to differentiate the effectiveness of single-feature and multi-feature LSTM models, aiming to reduce the unpredictability of external influences on Bitcoin prices. Further enhancing the predictive accuracy, "Bitcoin Price Forecasting using LSTM and 10-Fold Cross Validation" combined various neural network approaches, including RNN and LSTM, with 10-fold cross-validation, demonstrating the potential of LSTM in refining predictions in the volatile cryptocurrency market according to Tandon et al. (2019).

3 Problem Formulation

The primary challenge in Bitcoin price prediction lies in its unpredictable market fluctuations, driven by various global economic factors, investor sentiments, and market trends. Accurately forecasting these price movements is not only challenging but also essential for traders and investors. The hypothesis of this study is rooted in examining the effectiveness of GAN, ARIMA, and LSTM in predicting Bitcoin prices, each model bringing its methodology to the forefront of financial forecasting.

4 Methodology

4.1 Data Pre-processing

Our dataset consists of historical Bitcoin price data spanning from Sep 18th, 2014 to April 11, 2023. Key features include opening price, closing price, high, low, and trading volume. To prepare this data for model training, we perform data cleaning. For cryptocurrencies other than Bitcoin that were NaN before they came into existence, values (daily high and low, opening and closing price) were zeroed out to maintain dataset integrity. In cases where datasets like the Dow Jones Industrial Average, S&P 500, and VIX presented missing values, either sporadically or consistently, these were addressed by filling the missing values with the average of the nearest preceding and following values. This approach mitigates the impact of data gaps while preserving the overall trend and characteristics of the time series. For monthly datasets, such as the 10-Year Breakeven Inflation Rate, Sticky Price CPI, and US CPI, the values were extrapolated to provide daily entries, assigning the month's value to each day within that month. This extrapolation ensures a uniform temporal resolution across all datasets, thereby facilitating more accurate and consistent time-series analysis. Additionally, we structure the data into suitable time steps, allowing the models to learn from previous price movements to predict future prices. We employ a chronological split of the dataset into training and validation segments. The model is trained with hyperparameter tuning on the training set, the first 80% of the data, while model evaluation are performed using the validation set, the last 20% of the data.

4.2 Lasso, Ridge, and Elastic Net

Lasso regression, while using L1 regularization, penalizes each β (regression coefficient) equally. It shrinks the less important features' coefficients to zero and removes them from the model, which makes it an ideal model for feature selection. The cross-validated alpha (as in sklearn.linear_model.Lasso) for Lasso is 20.

Ridge regression, on the other hand, shrinks less important features towards zero without removing them. A large β is penalized more using β^2 than $|\beta|$, as in Lasso. The cross-validated alpha for Ridge is 20.

Elastic Net effectively combines the advantages of both Lasso and Ridge. It features both variable selection of Lasso and stronger shrinkage of Ridge, and it typically uses two milder penalties instead of a strong one. The cross-validated alpha and l1_ratio are both 0.

The packages used are Lasso, Ridge, and ElasticNet from sklearn.linear_model.

4.3 Random Forest and XGBoost

A Random Forest is a "forest" consisting of many decision trees. It independently draws a sample of data from the full dataset and fits a regression tree to the data. This step is repeated many times and the results are taken as average forecasts of all trees. Since each tree is trained individually and parallelized, it is often fully grown and does not require pruning. The cross-validated parameters are max_depth: 10, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 50. The package used is sklearn.ensemble.RandomForestRegressorr.

XGBoost is a gradient-boosting algorithm that builds Gradient Boosting Decision Trees (GBDT) sequentially, allowing each tree to learn from previous weak learners. Its objective function contains L1 and L2 regularization terms, which helps prevent overfitting. Trees are pruned bottom-up, removing ones that do not improve overall performance. The cross-validated parameters are colsample_bytree: 1.0, learning_rate: 0.67, max_depth: 5, n_estimators: 200, subsample: 1.0. The package used is xgboost.

4.4 ARIMA

The ARIMA model, known for its effectiveness in time series forecasting, is widely used in predicting Bitcoin prices. This model, which stands for AutoRegressive Integrated Moving Average, combines three key elements: differencing to ensure the time series is stationary, autoregression to link an observation with its past values, and a moving average to relate an observation to past forecast errors. Its ability to handle the unpredictable nature of Bitcoin prices allows it to forecast future trends with notable accuracy, making it a crucial tool for investors and analysts in the fast-paced cryptocurrency market.

4.5 GAN AND WGAN-GP

Generative Adversarial Networks (GAN) embody a minimax problem in non-cooperative games, utilizing a generator and discriminator in continuous adversarial interplay. The generator aims to produce realistic synthetic examples, while the discriminator differentiates between genuine and generated samples. This dynamic competition results in highly authentic data representations.

In the basic GAN, the KL-JS divergence-based loss function is minimized using cross-entropy during training. To enhance GAN stability and performance, we utilized WGAN-GP, mentioned in work by Lin et al. (2021), utilizing the Wasserstein distance (EarthMover Distance) to measure the minimum cost of transporting mass between data distributions. WGAN-GP, unlike Basic GAN, includes a gradient penalty in the discriminator, altering the discriminator loss as follows:

Basic GAN	$-\frac{1}{m}\sum_{i=1}^{m}\left[\log(D(y^i) + \log(1 - D(G(x^i)))\right]$
WGAN-GP	$-\frac{1}{m}\sum_{i=1}^{m}\left[(D(y^i) - D(G(x^i)) + \lambda E(\|\nabla D_{y^i \tilde{x}^i}\|_2 - 1)^2\right]$

4.6 LSTM

LSTM networks and Gated Recurrent Unit (GRU) are types of recurrent neural network (RNN) designed to mitigate the vanishing gradient problem in sequential data learning, particularly suited for time series forecasting due to their ability to capture temporal dependencies and manage long-term data sequences using an update gate and a forget gate. Unlike traditional RNNs, the update gate controls the flow of information from the previous hidden state to the current state, while the reset gate determines information to be forgotten. In the context of Bitcoin price prediction, the volatile and unpredictable nature of the cryptocurrency market makes LSTM an appealing choice.

The LSTM model in this study is designed with multiple layers to effectively capture complex patterns in the data. Two LSTM layers consist of a set number of nodes, and we employ dropout layers after each LSTM layer to prevent overfitting. Finally, we use dense layers to generate fixed-sized output. The model's architecture is fine-tuned to balance between capturing intricate details in the data and generalizing well to new, unseen data. We conduct controlled experiments to find the best set of hyperparameters (batch size, epoch, number of neurons in the LSTM layer, drop probability) that yield the lowest average Root Mean Square Error (RMSE). For comparison, we utilize Mean Squared Error (MSE) as the loss function, and the model is optimized using an appropriate optimization algorithm like Adam.

5 RESULTS AND DISCUSSION

The prediction by ARIMA is demonstrated in Figure 1. Box-cox transformation, seasonal differentiation, and regular differentiation are performed on the daily price of Bitcoin, to obtain stationary data for the later analysis. The package used for ARIMA is sm.tsa.statespace. The parameters of the model with the best performance are (1, 1, 0) $\times$ (2, 1, 0, 12), which has the smallest value of AIC.

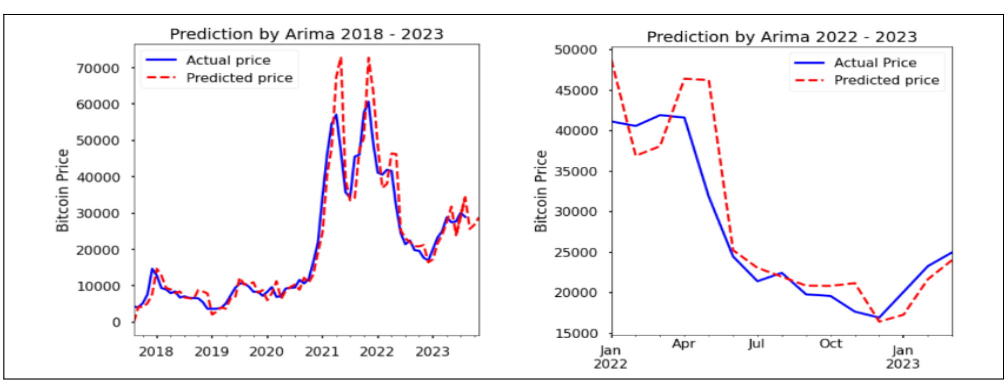

Figure 1: Price Prediction by ARIMA

The baseline models, including LASSO, Ridge, Elastic Net, XGBoost, and Random Forest, were evaluated for their performance in predicting Bitcoin prices, as shown in Figure 2. Among the linear regression models, Ridge stands out with the lowest RMSE of 891.04, showcasing its effectiveness in capturing the underlying trends in the dataset.

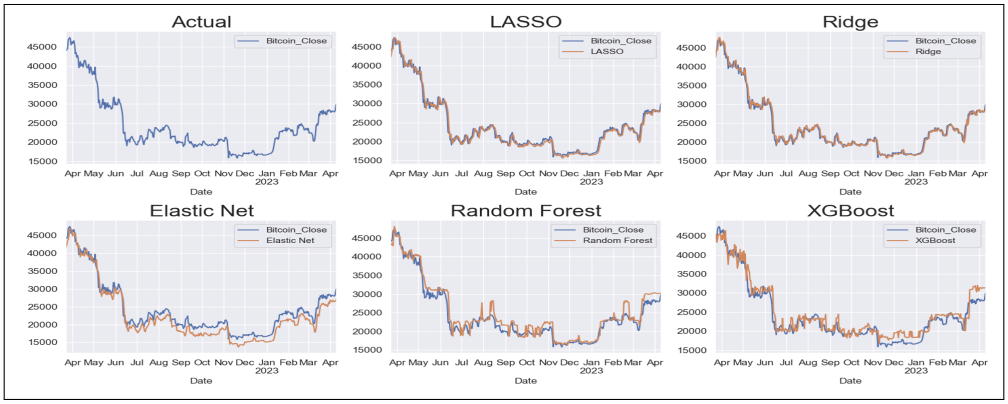

Figure 2: Bictoin Real price vs Predicted price (Baseline)

Based on our experiments, we automated the selection of the number of LSTM neurons and the dropout value. We found that the optimal configuration includes 300 neurons per layer with a dropout rate of 0.01 for the LSTM (Figure 6 in appendix). Additionally, we determined that 35 training epochs and a batch size of 32 are suitable shown in the Figure 5. Our LSTM model achieved an RMSE of 1281.17 when predicting Bitcoin prices. This result is based on an average of 10 model fittings to ensure reliability and accuracy, highlighting the model's precision. The final LSTM Bitcoin prediction results are displayed in Figure 3, depicting real values compared to predicted values. The accompanying graph visually demonstrates our model's ability to capture essential trends. To

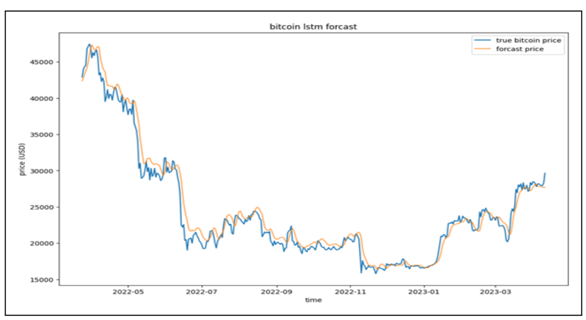

Figure 3: Bictoin Real price vs Predicted price (LSTM)

evaluate the standalone performance of GRU, which serves as the generator for GANs, we conducted individual assessments.The performance of Basic GAN and WGAN-GP, in conjunction with GRU, is depicted in Figure 4. GRU, Basic GAN, and WGAN-GP achieved RMSE values of 2461.17, 1921.42, and 1439.08, respectively. Despite the potential demonstrated by GANs in generating realistic data, their effectiveness in the realm of time-series prediction remains a formidable challenge. The results of our study indicate that incorporating Wasserstein distance and gradient penalty in the discriminator component significantly enhances the performance of GANs in the context of Bitcoin price prediction. However, the findings also underscore the need for further refinement and exploration to unlock the full predictive potential of GANs and align them with the reliability demonstrated by traditional methods in Bitcoin price forecasting.

According to Table 1, while LSTM stands out as the top performer with an RMSE of 1281.17 among the time-series models, the heightened volatility of Bitcoin, especially during global events, challenges the predictability of all time-series models. Linear regression models, especially Ridge with an RMSE of 891.04, consistently outperform their time-series counterparts.

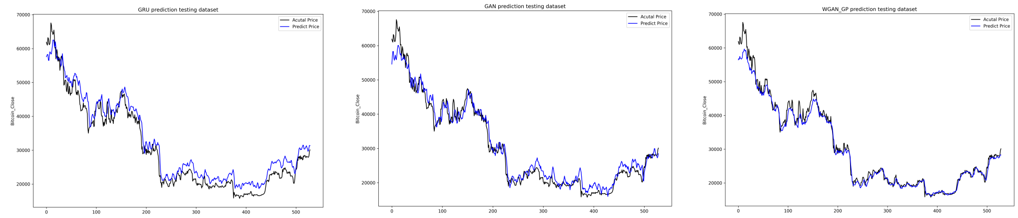

Figure 4: (a) GRU (b) Basic GAN (c) WGAN-GP

Table 1: Root Mean Square Error (RMSE) of all models

Model	LASSO	Ridge	Elastic Net	XGBoost	Random Forest
RMSE	964.63	**891.04**	2278.85	1709.45	1940.51
Model	ARIMA	GRU	LSTM	Basic GAN	WGAN-GP
RMSE	2436.65	2461.19	**1281.17**	1921.42	1439.08

6 Conclusion and Future Work

In the realm of cryptocurrency price prediction, the choice between linear regression models and time-series models, such as LSTM, hinges on key considerations. Linear regression models stand out for their straightforward interpretability, offering clear insights into the contribution of each feature to the predicted outcome. This interpretability holds significant value, particularly in financial markets, where understanding the factors influencing predictions is crucial for informed decision-making. On the other hand, time-series models, exemplified by LSTM, are often perceived as black boxes, posing challenges in interpreting the intricacies of their decision-making processes.

In our project, we collected several time-series models mentioned in recent published papers, conducting experiments to evaluate their feasibility and performance, and comparing them with traditional linear regression approaches. In our study, the LSTM model exhibits the best overall performance among the time-series models. However, with the heightened volatility of Bitcoin, particularly during global events, time-series models encountered difficulties in forecasting the correct direction and magnitude, in the face of sudden and extreme price fluctuations that deviate from previously learned patterns. Their ability to adapt quickly to new market dynamics is a potential limitation, due to the fact that they cannot capture the real-time situation of the market with a lack of focus on external factors, where ARIMA is a great example of the failure.

In contrast, linear regression models, designed to be less sensitive to short-term market shifts, assigning reasonable weights to all factors in the dataset, demonstrate more stable performance during periods of heightened turbulence. Moreover, the effectiveness of time-series models, especially deep learning models like LSTM, relies heavily on the availability of extensive historical data for training. If the historical data for Bitcoin is limited or subject to abrupt changes in market dynamics, time-series models may struggle to generalize effectively. In such scenarios, linear regression models emerge as more robust contenders, showcasing their capacity to perform well even with constrained data, provided the underlying relationships remain relatively stable. These considerations underscore the nuanced nature of model selection, emphasizing the need to align the model's characteristics with the specific challenges and characteristics of the cryptocurrency market.

This project compares the performance of various time-series models with traditional linear regression models, regarding the ability to forecast Bitcoin price. The results of the project open up new avenues for future research, particularly in the choice of the basic models and potential optimizers, when exploring combined forecasting models. Future research work should focus on expanding the data sets, experimenting with various LSTM architectures, incorporating market sentiment, and exploring hybrid models. This approach aims to enhance the accuracy and reliability of financial market predictions using advanced machine-learning techniques.

REFERENCES

Junwei Chen. Analysis of bitcoin price prediction using machine learning. *Journal of Risk and Financial Management*, 2023. doi: https://doi.org/10.3390/jrfm16010051.

J Cheng, S Tiwarix, K Djebbouri, M Mahendru, and U Shahzad. Forecasting bitcoin prices using artificial intelligence: Combination of ml, sarima, and facebook prophet models. *Technological Forecasting and Social Change*, 198:122938, 2024. doi: 10.1016/j.techfore.2023.122938.

Julien Chevallier, Dominique Guégan, and Stéphane Goutte. Is it possible to forecast the price of bitcoin? *Forecasting*, pp. 377–420, 2021. doi: https://doi.org/10.3390/forecast3020024.

F. Ferdiansyah, S. H. Othman, R. Zahilah Raja Md Radzi, D. Stiawan, Y. Sazaki, and U. Ependi. A lstm-method for bitcoin price prediction: A case study yahoo finance stock market. In *2019 International Conference on Electrical Engineering and Computer Science (ICECOS)*, pp. 206–210, 2019. doi: 10.1109/ICECOS47637.2019.8984499.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans, 2017.

N. Latif, J. D. Selvam, M. Kapse, V. Sharma, and V. Mahajan. Comparative performance of lstm and arima for the short-term prediction of bitcoin prices. *The Australasian Accounting Business and Finance Journal*, 17(1):256–276, 2023. doi: 10.14453/aabfj.v17i1.15.

T. Li. Prediction of bitcoin price based on lstm. In *2022 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE)*, pp. 19–23, 2022. doi: 10.1109/MLISE57402.2022.00012.

HungChun Lin, Chen Chen, GaoFeng Huang, and Amir Jafari. Stock price prediction using generative adversarial networks. *Journal of Computer Science*, pp. 188–196, 2021. doi: https://doi.org/10.3844/JCSSP.2021.188.196.

M.Á. López-Cabarcos, A. M. Pérez-Pico, J. Piñeiro-Chousa, and A. Šević. Bitcoin volatility, stock market and investor sentiment. are they connected? *Finance Research Letters*, 38:101399, 2021. doi: 10.1016/j.frl.2019.101399.

Sumit Ranjan, Parthajit Kayal, and Malvika Saraf. Bitcoin price prediction: A machine learning sample dimension approach. *Computational Economics*, pp. 1617–1636, 2022. doi: https://doi.org/10.1007/s10614-022-10262-6.

Ricardo Alberto Carrillo Romero. Generative adversarial network for stock market price prediction. URL `https://cs230.stanford.edu/projects_fall_2019/reports/26259829.pdf`.

Yang Si. Using arima model to analyse and predict bitcoin price. *BCP Business Management*, 34:1210–1216, 2022. doi: 10.54691/bcpbm.v34i.3161.

S. Tandon, S. Tripathi, P. Saraswat, and C. Dabas. Bitcoin price forecasting using lstm and 10-fold cross validation. In *2019 International Conference on Signal Processing and Communication (ICSC)*, pp. 323–328, 2019. doi: 10.1109/ICSC45622.2019.8938251.

Kang Zhang, Guoqiang Zhong, Junyu Dong, Shengke Wang, and Yong Wang. Stock market prediction based on generative adversarial network. *Procedia Computer Science*, 147:400–406, 2019.

Xingyu Zhou, Zhisong Pan, Guyu Hu, Siqi Tang, and Cheng Zhao. Stock market prediction on high-frequency data using generative adversarial nets. *Mathematical Problems in Engineering*, pp. 1–11, 2018. doi: https://doi.org/10.1155/2018/4907423.

A Appendix

A.1 Author Contributions

Yu Du: Constructing and fine-tuning the baseline models (Lasso, Ridge, Elastic Net, Random Forest, and XGBoost).
Chao Li: Responsible for data cleaning and preparation tasks.
Yikun Chen and Chao Li: Collaboratively working on optimizing the LSTM model.
Zheng Bao: Constructing and fine-tuning the GRU, GAN, and WGAN-GP models.
Zhiyuan Zhou: Constructing ARIMA model, tuning parameters to find the best performance ARIMA model.

A.2 GitHub page

`https://github.com/mezzy33/csci-567-group-project`

A.3 Figures

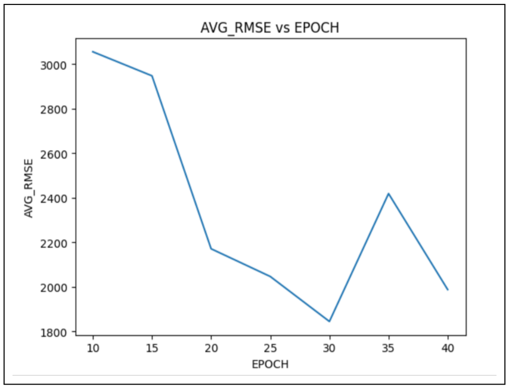

Figure 5: LSTM number of Epochs vs RMSE

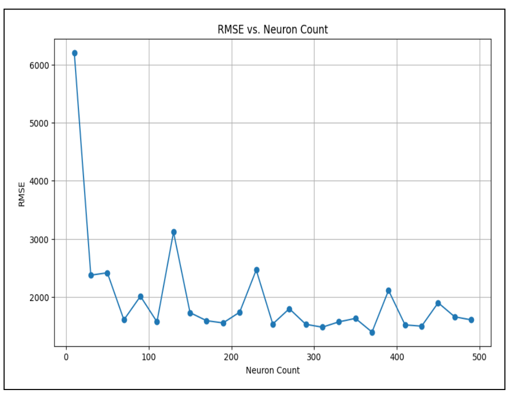

Figure 6: LSTM number of Neurons vs RMSE