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

USC Viterbi School of Engineering

Yu Du, Yikun Chen, Zheng Bao, Zhiyuan Zhou, Chao Li

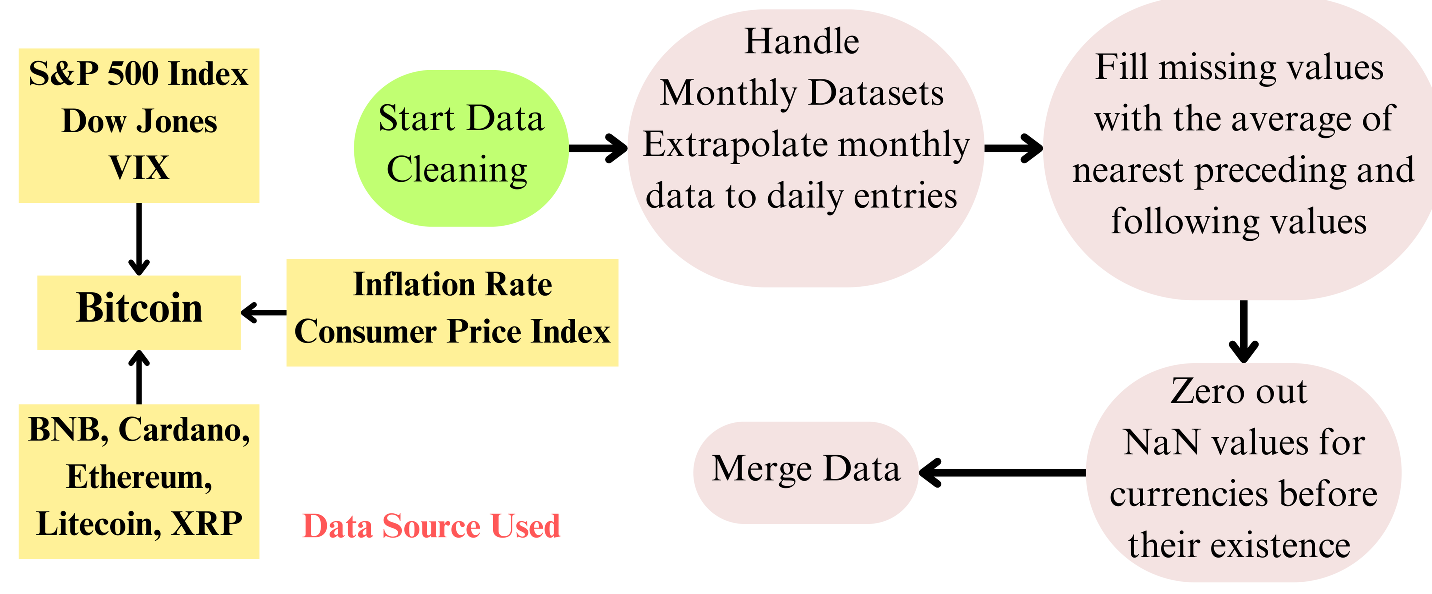
Introduction

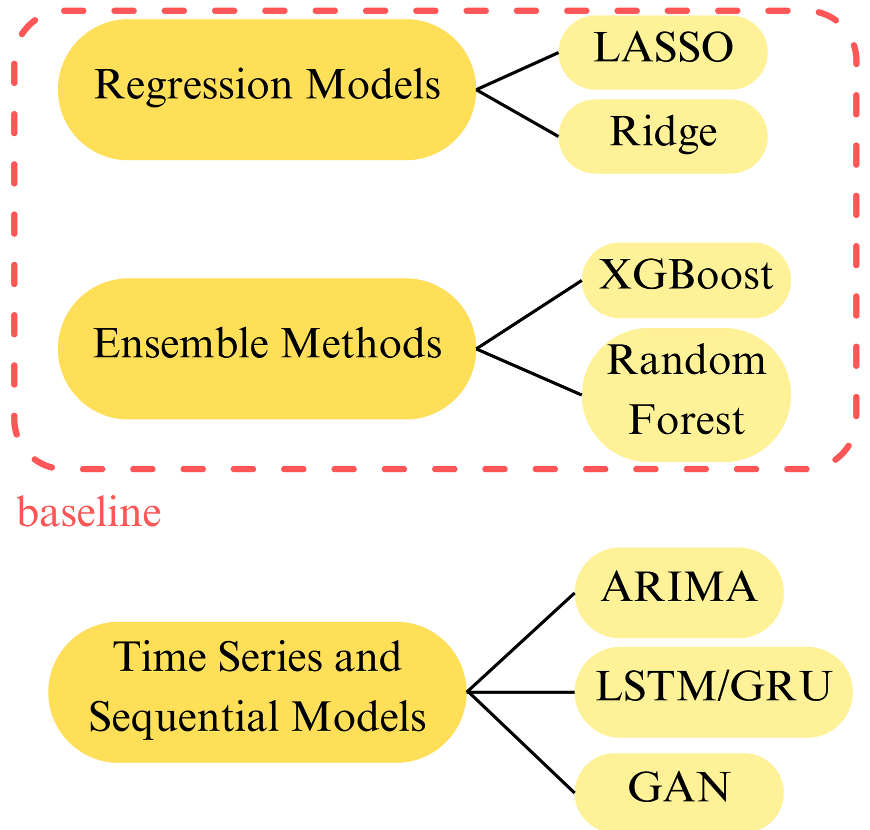
The inception of this project is driven by the critical goal of enhancing the prediction accuracy of Bitcoin prices, a subject of great significance in the financial sector. Existing methodologies in this realm often falter due to the volatile nature of cryptocurrency markets, leading to less reliable and, at times, misleading predictions. Our approach, grounded in a comparative analysis of three important models – Generative Adversarial Networks (GAN), AutoRegressive Integrated Moving Average (ARIMA), and Long Short-Term Memory Networks (LSTM) – seeks to address these challenges. 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 is a careful examination of these models, assessing their accuracy, reliability, and computational efficiency in the context of Bitcoin price forecasting.

Problem Formulation

- Addressing the challenge of accurately predicting Bitcoin prices, crucial for navigating the cryptocurrency market.
- Comparing the efficacy of GAN, ARIMA, and LSTM models to forecast Bitcoin price trends amidst market volatility.
- Identifying the most effective predictive model to capture the dynamic and non-linear nature of cryptocurrency price movements.

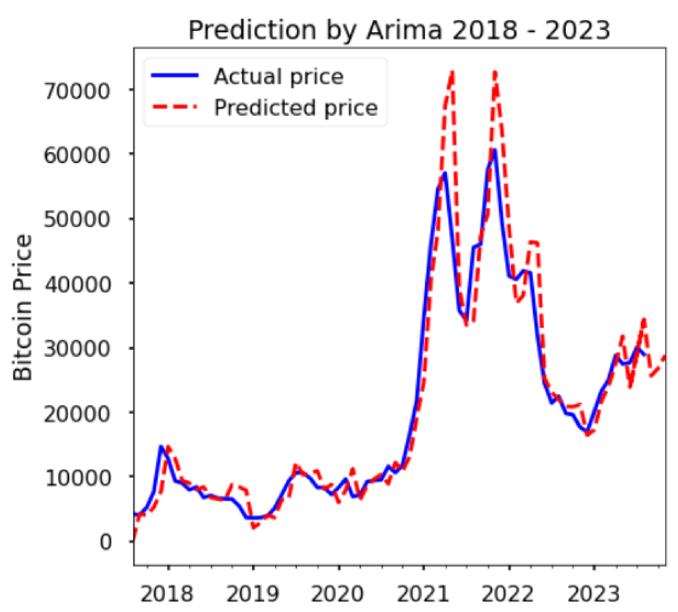
Methodology

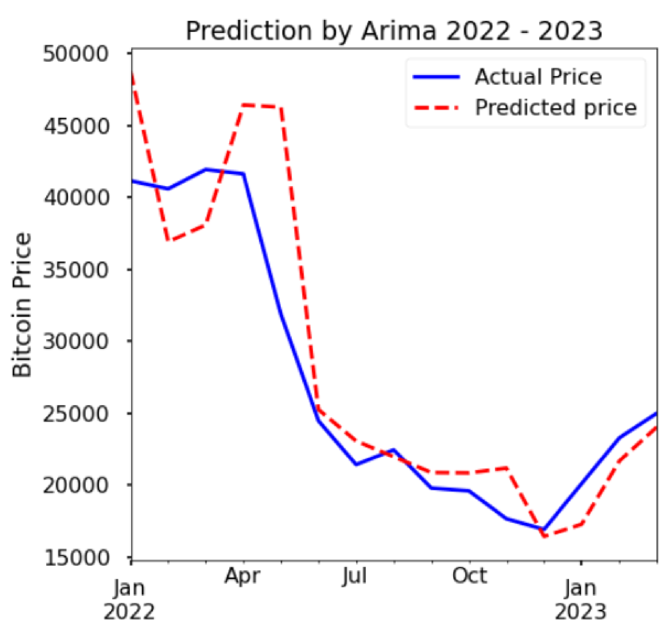




Results and Discussion

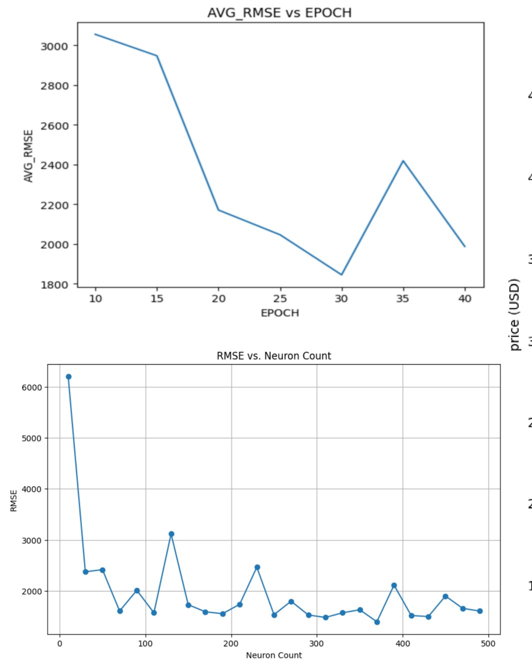
- **ARIMA**

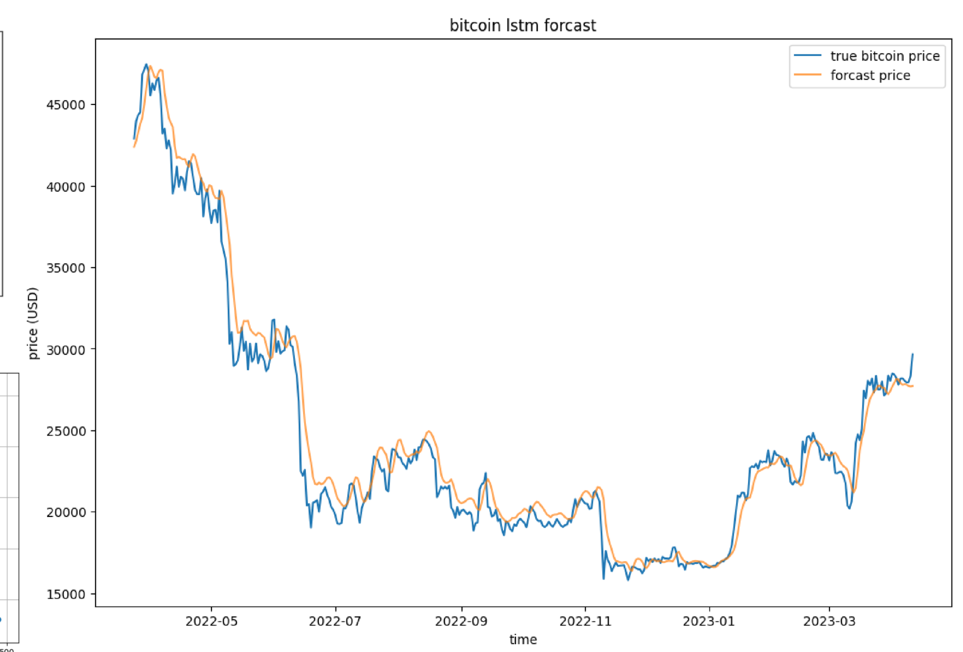




The ARIMA (AutoRegressive Integrated Moving Average) model stands out as a powerful and popular statistical method for time series forecasting, and it has been effectively applied in predicting Bitcoin prices. This model uniquely combines differencing, autoregression, and moving average. By capturing the volatile nature of Bitcoin prices, ARIMA can analyze and predict future values with a significant degree of accuracy.

- **LSTM**



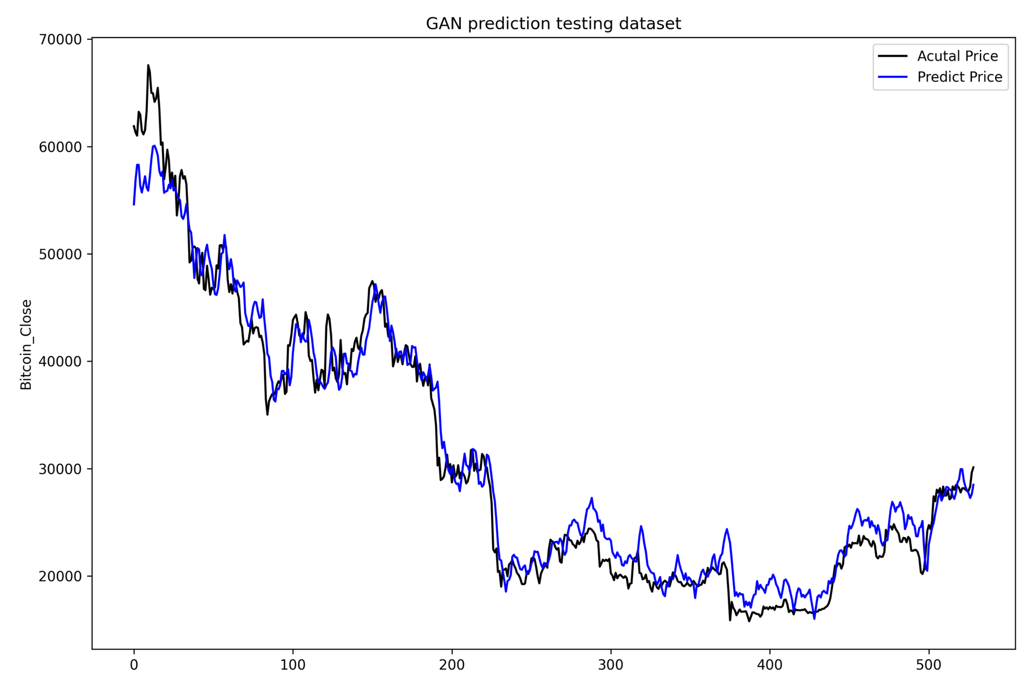


During model configuration, we made key decisions for optimization:

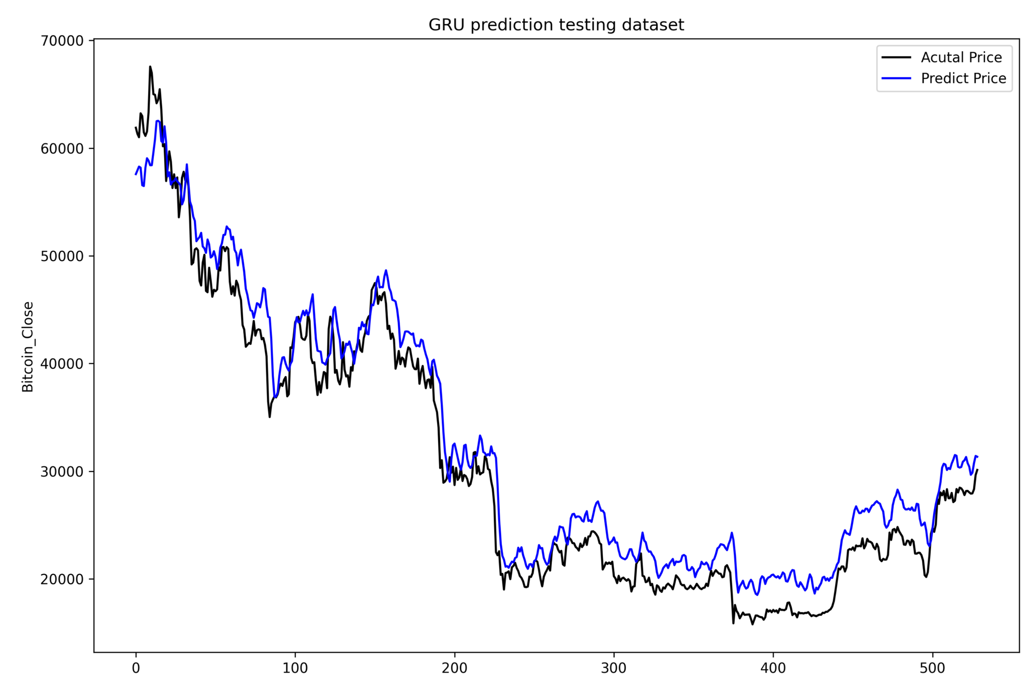
- We used 300 neurons in the LSTM layer, resulting in reduced RMSE and a good balance between complexity and accuracy.
- We applied dropout regularization at each LSTM layer with a 0.05 rate to enhance generalization and prevent overfitting.
- After systematic experiments, we found that training for 30 epochs consistently achieved impressive results on the test dataset, with the lowest RMSE. This indicates the model's ability to capture underlying price patterns effectively.
- A batch size of 32 played a vital role in efficient training and gradient descent convergence, balancing computational efficiency and memory usage. These combined decisions contribute to our LSTM model's overall effectiveness.

Finally, our Network has 2 layers of LSTM with 300 neurons and 2 dense layers.

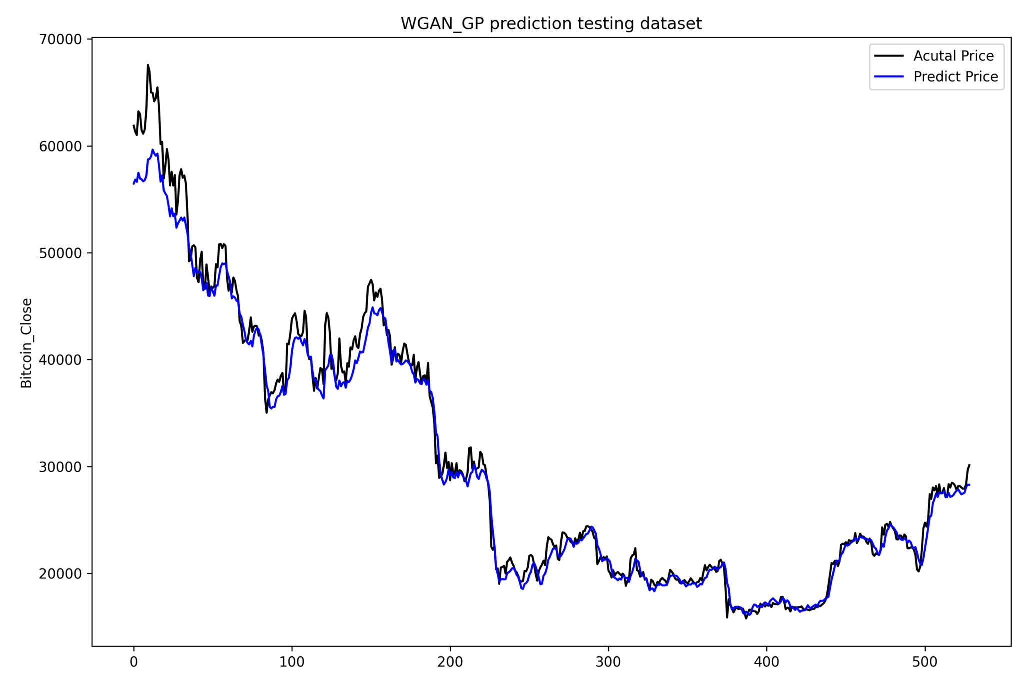
- **GAN**



Our **basic GAN** utilizes KL-JS divergence for loss function. Cross-entropy loss is utilized to minimize distribution differences. The generator is implemented using a **GRU** with three layers (1024, 512, 256) followed by two Dense layers. The discriminator, a Convolutional Neural Network (**CNN**) employs three 1D Convolution layers (32, 64, 128) and three Dense layers (220, 220, 1). Leaky ReLU serves as the activation function, except in the output layer where Sigmoid is applied. The optimizer is Adam with a learning rate of 0.00016, batch size of 128 and training for 200 epochs.

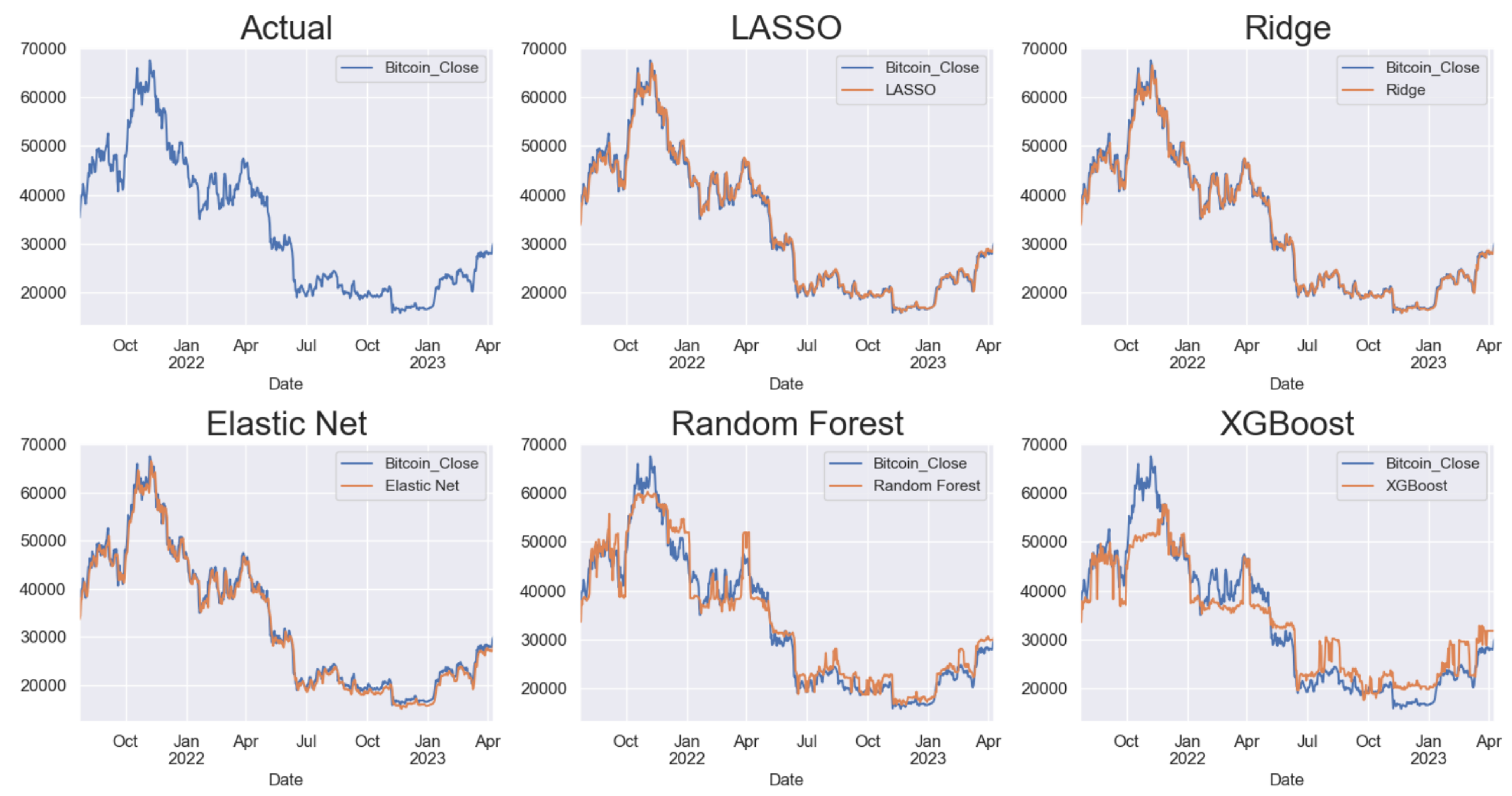


Gated Recurrent Unit (GRU) employs update and reset gates, facilitating selective information updates and resets in hidden states. Unlike conventional RNNs, GRU's adaptive gating mechanism, consisting of an update gate and a reset gate, regulates information flow. The update gate manages information transfer from the previous hidden state to the current one, while the reset gate determines what information to forget. With two layers of GRU, our model utilizes the Adam optimizer with a learning rate of 0.0001 and a batch size of 128, performing training for 100 epochs.



WGAN-GP proposes the use of the **Wasserstein distance**, or EarthMover Distance (EMD), representing the minimum cost of transporting mass between data distributions. Unlike Basic GAN, WGAN-GP operates without a sigmoid function, outputting a scalar score instead of a probability. This score reflects the realism of input data. Additionally, WGAN-GP incorporates a **gradient penalty** in the discriminator. The model adopts the Adam optimizer with a learning rate of 0.0001, a batch size of 128, and undergoes 150 epochs of training on the dataset.

- **Regression and Ensemble Learning**



Regression-based machine learning approaches and ensemble learning methods have been widely used in finance/econometrics analysis. In this project, we primarily studied five of these algorithms, LASSO, Ridge, Elastic Net, Random Forest, and XGBoost. We used these as a baseline and to evaluate the performance of other algorithms.

- **Comparison**

Model	LASSO	Ridge	Elastic Net	XGBoost	Random Forest
RMSE	1475.62	**1390.82**	1571.92	4433.56	2735.38

Model	ARIMA	LSTM	GRU	Basic GAN	WGAN_GP
RMSE	2436.65	**1281.17**	2461.19	1921.42	1439.08

Conclusion

This study provides an in-depth evaluation of the performance of several GAN variants and LSTM models in Bitcoin price prediction and compares them with traditional ARIMA models and linear regression models. Our results show that LSTM model perform well in terms of prediction accuracy, and especially their ability to reduce prediction errors (RMSE) deserves attention. LSTM can more effectively capture and predict the fluctuation trend of Bitcoin prices through their complex time series data processing capabilities. ARIMA performs worse than deep learning-based LSTM and GAN models. This finding highlights the potential of deep learning techniques in financial market forecasting, especially when forecasting compelx and volatile variables.