Linear vs. ANN Models for Bitcoin Price Prediction

Elon Ezra and Avi Rahimov

Abstract

The volatility of Bitcoin's price poses a formidable challenge for analysts seeking to forecast its future movements. In this study, we conduct a comparative analysis of two predictive modeling approaches: linear regression and Artificial Neural Networks (ANN). Drawing on historical Bitcoin price data, our research evaluates the performance of these models in predicting price differentials, shedding light on their respective strengths and limitations.

Through a rigorous examination of model performance metrics such as Mean Squared Error (MSE) and R-squared scores, we delve into the intricacies of linear regression and ANN models in capturing the complex dynamics of cryptocurrency markets. Our analysis aims to provide insights into the effectiveness of these models without prescribing specific financial strategies.

By presenting a comprehensive assessment of experimental results and theoretical considerations, this study contributes valuable knowledge to the field of financial forecasting, offering researchers and practitioners a nuanced understanding of predictive modeling techniques in the context of cryptocurrency price prediction.

Introduction

Cryptocurrencies have garnered significant attention in recent years, with Bitcoin emerging as a prominent player in the digital asset landscape. The volatility inherent in Bitcoin's price dynamics presents a formidable challenge for analysts seeking to forecast its future movements. In this paper, we propose a machine-learning approach to predict the price differential between the opening and closing prices of Bitcoin. Our study draws on a comprehensive dataset spanning historical Bitcoin price data since January 1st, 2014, encompassing key metrics such as daily high and low values, opening prices, and trading volumes.

Unlike traditional financial assets, Bitcoin's pricing mechanism operates in a decentralized and largely unregulated environment, contributing to its unique market dynamics. Our objective is to develop a predictive model that can accurately estimate the difference between the open and close prices, leveraging machine-learning techniques to capture the underlying patterns in the data.

To achieve this goal, we preprocess the dataset to extract relevant features and engineer a target label representing the price differential between the opening and closing values. This regression task entails predicting a continuous value, posing a distinct set of challenges given Bitcoin's inherent volatility and unpredictability.

In our methodology, we partition the dataset into training and testing subsets, allocating 80% of the data for model training and reserving the remaining 20% for evaluation. This ensures robustness and generalizability of the model, guarding against overfitting and enabling assessment on unseen data.

While Bitcoin's closing price serves as a stable reference point reflecting market consensus at the end of each trading day, it is subject to fluctuations driven by various factors including investor sentiment, market speculation, and regulatory developments. Our predictive model aims to elucidate these complex dynamics, offering insights into the underlying trends and patterns governing Bitcoin price movements.

By harnessing the power of machine learning, we endeavor to enhance our understanding of Bitcoin's price behavior and contribute to the growing body of research in cryptocurrency forecasting. Through empirical analysis and model evaluation, we seek to uncover valuable insights that can inform investment strategies and risk management practices in the ever-evolving landscape of digital assets.

Project description

As we delve deeper into the comparison between linear and artificial neural network models for Bitcoin price prediction, it becomes essential to consider the intricacies of each approach. While linear models are based on simple linear regression, ANN models leverage complex neural networks that can capture nonlinear patterns in the data.

One of the main advantages of linear models is their simplicity and interpretability. They provide a clear understanding of the relationship between input variables and the target output. However, they may struggle to capture the complex and non-linear nature of Bitcoin price movements, especially when considering a wide range of influencing factors.

On the other hand, ANN models have the capability to handle intricate patterns and relationships within the data. Their ability to learn from non-linear and complex features makes them a compelling choice for Bitcoin price prediction. By utilizing hidden layers and activation functions, ANN models can uncover intricate patterns that may go unnoticed by linear models.

As we move forward in this comparison, it is important to delve into the specific features and characteristics of both models, considering their performance in various market conditions and their ability to adapt to changing trends and dynamics in the cryptocurrency market.

To compare the two approaches, historical Bitcoin price data is gathered, followed by feature engineering and model training in different ways. The next steps include evaluating the performance of both linear and ANN models using metrics such as mean squared error and R-squared. Additionally, it is important to assess the robustness and generalization ability of these models through cross-validation.

In the investigation of predictive model performance, the baseline model, referred to as the "dummy" model, served as a control for comparative analysis. The coefficient of determination (R^2) obtained for the training dataset was -0.34146908823426037, and for the test dataset, it was -0.40308013789713826. The model exhibited a Mean Squared Error (MSE) of 1.2575376553076052, indicating the variance between the predicted and actual values.

Subsequent analysis was conducted using a linear regression model, which underwent 53,623 epochs and was divided across five folds. The average loss, as measured by the mean squared error for this model during the proposed training, was 0.18480867. The R^2 for the training dataset reached 0.8151917102590073, demonstrating a substantial proportion of the variance in the dependent variable being predictable from the independent variables. In contrast, the R^2 value for the test dataset was slightly lower at 0.778332252047607, suggesting a slight decrease in predictive accuracy outside the training dataset. The difference in R^2 values between the training and test datasets was approximately 0.04, indicating a more precise prediction capability without significant overfitting.

Further examination was conducted with an Artificial Neural Network (ANN) model, which was trained over 27,390 epochs and similarly divided across five folds. This model yielded an average training loss of 0.17909063. The R^2 value for the training set was found to be 0.8209095720568673, while for the test set, it was 0.7940919189726352. These results demonstrate a consistent predictive performance with minimal overfitting, highlighting the model's robustness and reliability in prediction.

This comparative analysis underscores the enhanced predictive accuracy and generalization capability of the linear regression and ANN models over the baseline "dummy" model, with both advanced models showing minimal evidence of overfitting.

Linear vs. ANN Models for Bitcoin Price Prediction

Elon Ezra and Avi Rahimov

אוניברסיטת אריאל בשומרון

model	Test-MSE score	Train R-Squared scores	Test R-Squared scores
Dummy model	1.257537655307	-0.3414690882342	-0.4030801378971
Linear Regression	0.1845490783452	0.815191710259	0.794091918972
ANN	0.1986739933490	0.820909572056	0.77833225204

Table 1. Models score comparison

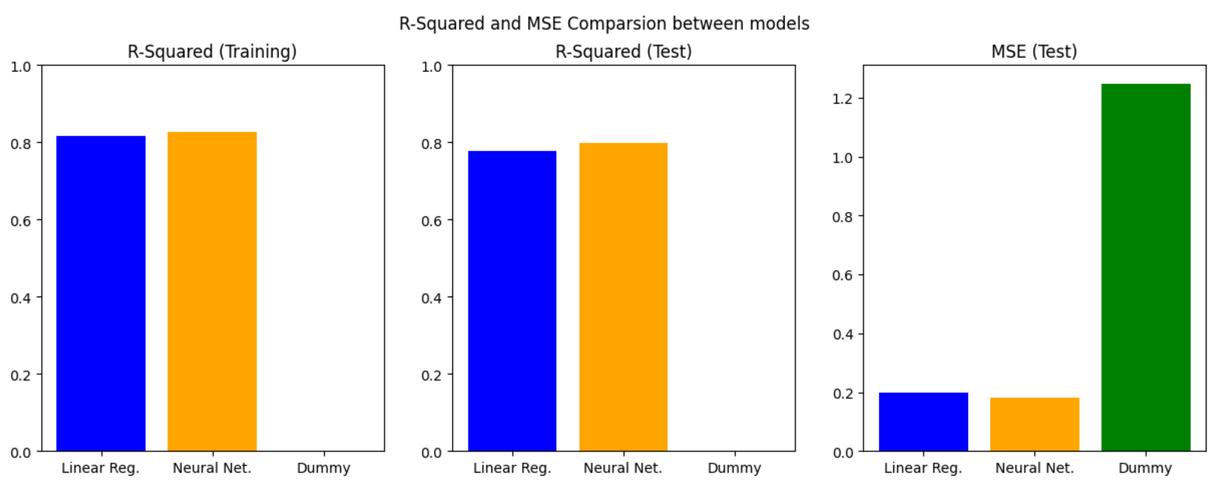

Figure 1. Comparison between the models

Throughout the experimental phase detailed in the subsequent chapter on experiments and simulation results, the coefficient of determination (R^2) was employed to ascertain the reliability and accuracy of the predictions, while the Mean Squared Error (MSE) was utilized to estimate the degree to which the model's predictions deviated from the actual data. Another method that facilitated a more visual assessment of the model's quality involved examining the predicted values as a function of the actual values.

In the graphs presented, a linear line depicted in cyan represents the ideal model, where the test predictions perfectly match the actual data. Our objective was to align the model's performance as closely as possible with this linear representation in the graph. It is observable that, in the case of the baseline model, the predictions, shown in green, are misaligned with the actual values, indicating a discrepancy between the predicted and observed data. Conversely, both the linear and Artificial Neural Network (ANN) models exhibit a more linear alignment, closely approximating the ideal function represented in cyan. There are still instances of background noise and incorrect predictions, which align with the R^2 results, indicating areas where model predictions diverge from actual outcomes.

This comparative analysis underscores the importance of using both statistical metrics and visual methods to evaluate model performance comprehensively. The closer a model's predictions align with the ideal linear path, the more accurate and reliable it is deemed to be, with deviations highlighted through both R^2 values and graphical representation.

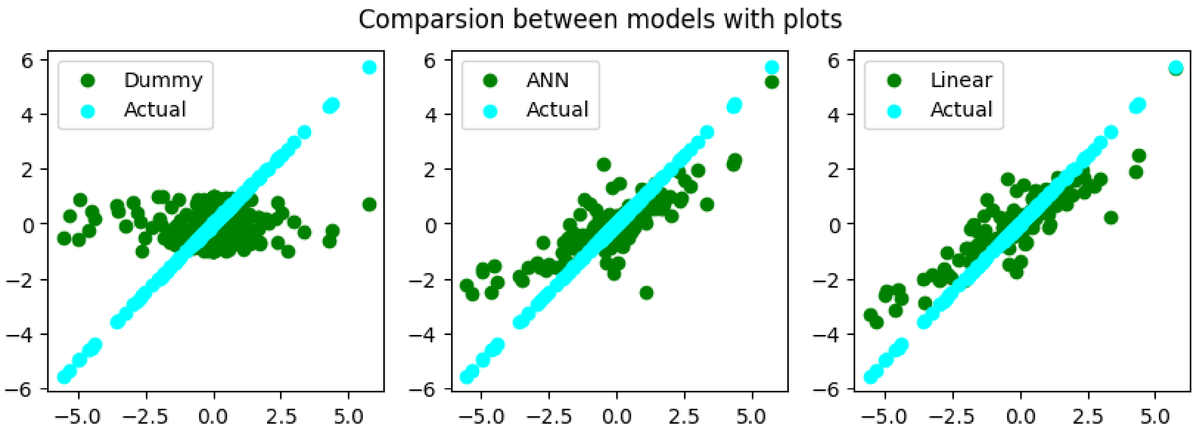

Figure 2. Comparison between the models

Experiments and simulation results

In this section, we will review our experiments starting from the examination of the dataset, through trial and error of model training, and drawing conclusions, culminating in the final model we constructed as presented in the previous chapter.

Initially, we tried to train the model on the details from the data set. Due to the large value of the data, we had to perform normalization. The normalization we performed was simple, and it is described by the following expression:

$$x' = \frac{x - \text{mean(x)}}{\text{variance}}$$

The results we obtained were as follows:

- For linear regression, we obtained:

$$R^2 = 0.9919877932512889$$
$$NRMSE = 0.022056943475388894$$

- For ANN, we obtained:

$$R^2 = 0.9323165141124011$$
$$NRMSE = 0.022056943475388894$$

These results are not satisfactory because they are not realistic. The price of Bitcoin is volatile, and the linear regression provides a result that is too good, even though it is simpler than the ANN.

After we obtained these results, we investigated the reason for their highness compared to reality and the quality of the model. We presented the correlation between the features and the closing price. The relationship between the features can be seen in the heat map we presented. Note that the correlation value between the closing price and the rest of the features is 1. These results indicate a very strong connection between the features and the closing price, which explains the over-fitting we obtained after the training.

Due to the strong correlation, we had to think of another way to indirectly predict the closing price. First, we tried to take the average of the opening and closing prices.

$$\frac{open - close}{2}$$

But as can be seen in the heatmap, the correlation to most of the features is 1, as opposed to the trading volume. So this is also not reliable enough for us to use. We then tried to take the difference between the opening and closing prices.

$$open - close$$

According to the map, we see better correlation values that are close to 0. High correlation between the features and the closing price means that the features explain most of the change in the closing price, but also that they contain a lot of redundant or overlapping information. This can lead to a problem called overfitting, which occurs when the model fits too well to the training data and fails to generalize the trends to new data.

Low correlation between the features and the closing price means that the features do not explain all the change in the closing price, but also that they contain unique and complementary information. This can lead to a problem called under-fitting, which occurs when the model does not fit well enough to the training data and fails to capture the complex structure of the data. Therefore, we need to find a balance between overfitting and under-fitting, and choose features that have a suitable correlation to the closing price, so that the model will be efficient and accurate. One of the ways to do this is to use methods of feature selection or feature extraction, which aim to reduce the dimensionality of the data and leave only the important and significant features.

Linear vs. ANN Models for Bitcoin Price Prediction

Elon Ezra and Avi Rahimov

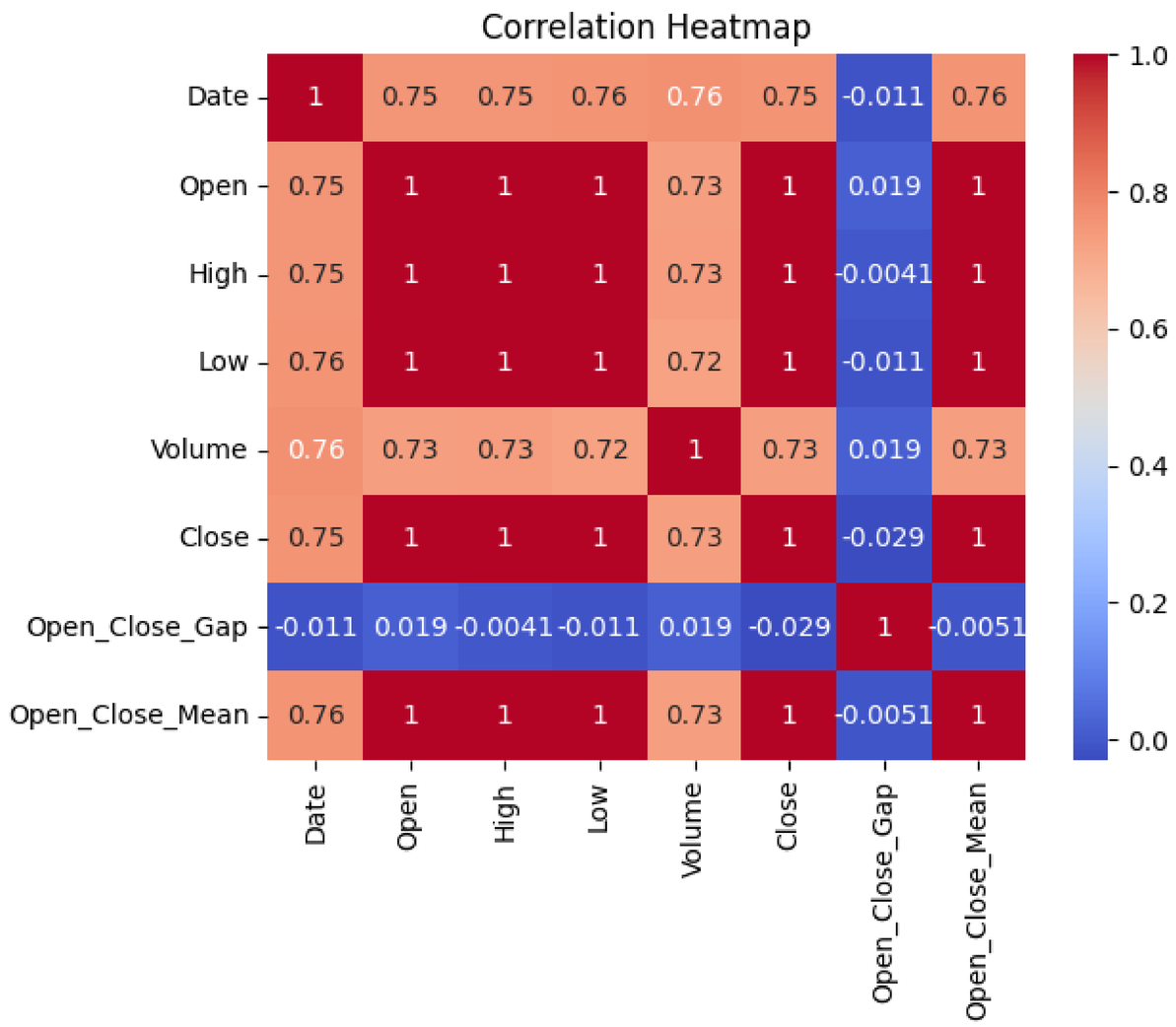

Figure 3. Correlation Heatmap

In the group of graphs depicted in Figure 4, the visual representation illustrates the relationship among various features. As the points gather more closely around a linear line, the correlation becomes stronger. For instance, a notable correlation can be observed in the average opening-closing feature, as it is directly influenced by both the closing and opening prices.

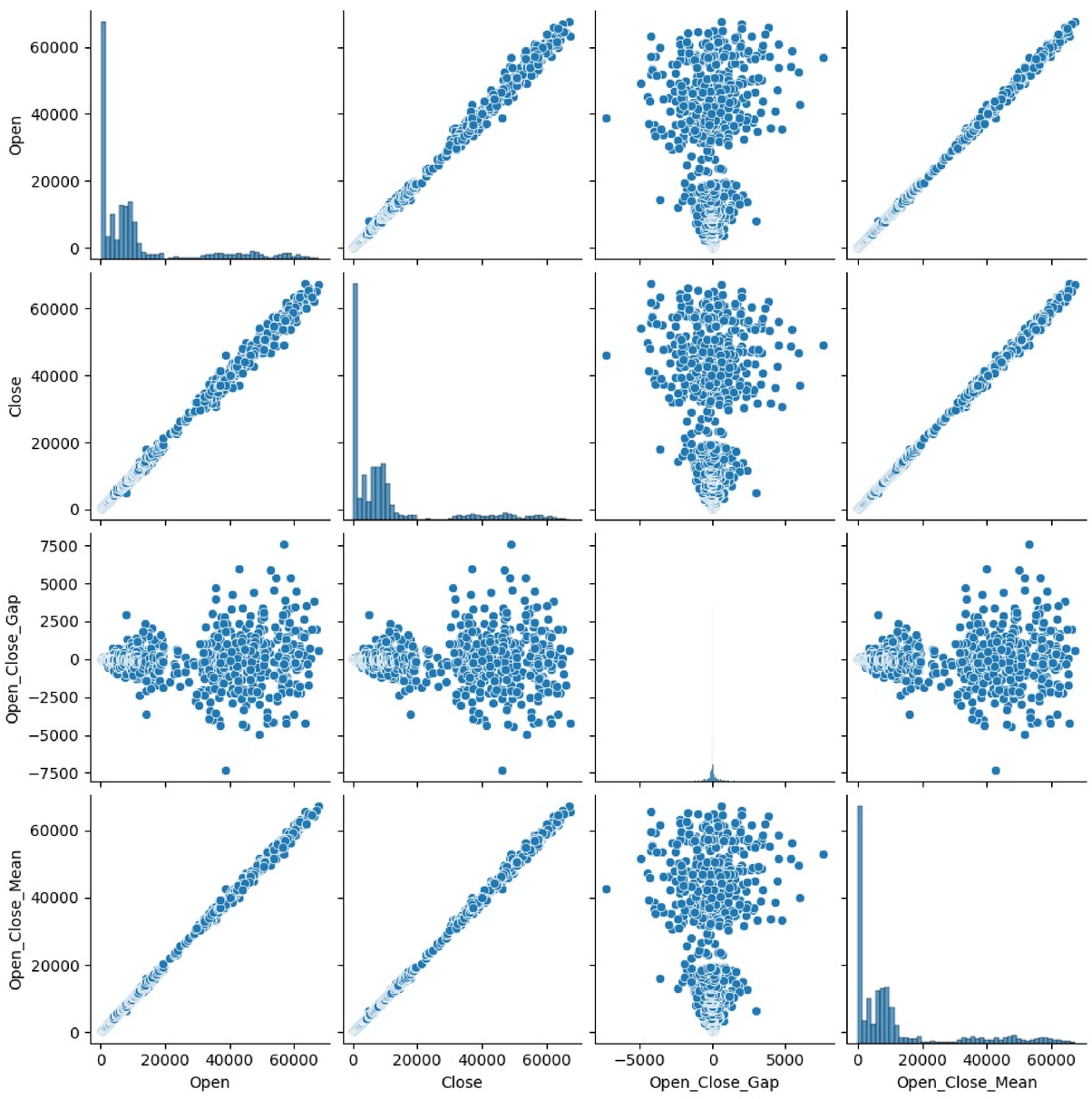

Figure 4. Comparison between the models

Furthermore, an interesting observation is the strong correlation between the opening and closing prices. This correlation likely stems from market sentiment, wherein if the opening price rises, the closing price tends to increase correspondingly, reflecting an upward trend in the market. Conversely, the same principle applies in the opposite direction.

On the other hand, when examining the gap between the opening and closing prices, a dispersion of points across the space is evident, even in relation to the opening-closing average. This dispersion ensures lower correlation and more reliable predictions.

These visual insights emphasize the intricate relationships among the features and provide valuable cues for understanding the dynamics of the dataset. Such observations inform the selection of relevant features and contribute to the development of a more effective and accurate predictive model.

Based on the findings from both the correlation heatmap and the series of graphs depicting the relationships in the dataset, it is preferable, unlike previous instances, to predict the gap between the opening and closing prices compared to relying solely on the closing price or its average. However, if there is a specific need to ascertain the closing price itself, we can deduce it by subtracting the predicted gap from the opening price. This approach leverages the observed relationships in the dataset and offers a more nuanced and potentially more accurate prediction of the closing price.

Upon revising our methodology to focus on predicting the discrepancy between opening and closing prices rather than solely the closing price, we encountered initial results that necessitated further refinement of our approach. Specifically, the Linear Regression model demonstrated an R-Squared score of -0.009098117873280742, coupled with a Mean Squared Error (MSE) of 0.9044236540794373 on the test set. Concurrently, the Neural Network (ANN) model exhibited an R-Squared score of -1.0380318673999493, with an MSE of 1.8266254663467407 on the test set. These outcomes suggested a potential underfitting of our predictive models, indicative of an insufficient number of training epochs.

To address this, we integrated an early stopping mechanism within our training regimen. Early stopping is a form of regularization used to avoid overfitting when training a learner with an iterative method, such as gradient descent. This technique halts the training process if the model's performance on a validation set does not improve for a given number of epochs, thereby preventing the model from learning noise in the training data.

The incorporation of early stopping into our model training protocol yielded substantial improvements in predictive accuracy. Post-implementation, the Linear Regression model's R-Squared score markedly increased to 0.7756200579730448, with the MSE on the test set reducing to 0.20110484957695007. Similarly, the Neural Network model's performance improved, achieving an R-Squared score of 0.7930579359242167 and an MSE of 0.18547581136226654 on the test set. For context, a baseline Dummy Model generated an R-Squared score of -0.3675316464329903 and an MSE of 1.2256766336894482 on the test set, underscoring the effectiveness of our refined approach.

Notably, the training cessation for the Neural Network model occurred after epoch 120,306, whereas the Linear Regression model concluded its training after epoch 195,703. These results not only validate the efficacy of early stopping in enhancing model performance but also highlight its critical role in mitigating the risk of overfitting, thereby ensuring that the model retains its generalizability when applied to unseen data sets.

	Without Early Stopping		With Early Stopping	
	Test-MSE Score	Test R-Squared Score	Test-MSE Score	Test R-Squared Score
Linear Regression	0.9044236540	-0.0090981178	0.201104849576	0.77562005797
ANN	1.8266254663	-1.0380318673	0.185475811362	0.7930579359
Dummy Model	1.2148727436	-0.3554773564	1.225676633689	-0.3675316464

Table 2. Comparative Performance of Models With and Without Early Stopping

Linear vs. ANN Models for Bitcoin Price Prediction

Elon Ezra and Avi Rahimov

In our pursuit of refining the predictive models and enhancing their robustness, we incorporated cross-validation into our methodology. Cross-validation is a widely employed technique in machine learning for assessing the performance and generalizability of predictive models. The fundamental principle behind cross-validation involves partitioning the dataset into complementary subsets, performing model training on a subset of the data (training set), and validating the model on the remaining data (validation set). This process is repeated multiple times, with each subset serving as both the training and validation set, thereby providing a more comprehensive evaluation of the model's performance.

The necessity of cross-validation arises from its ability to address potential biases and variance in the model's performance estimation. By systematically rotating through different subsets of the data for training and validation, cross-validation helps mitigate the risk of overfitting to a particular subset and provides a more accurate assessment of the model's generalization ability to unseen data. Moreover, cross-validation aids in identifying and mitigating issues such as data heterogeneity and sample size discrepancies, which can significantly impact model performance.

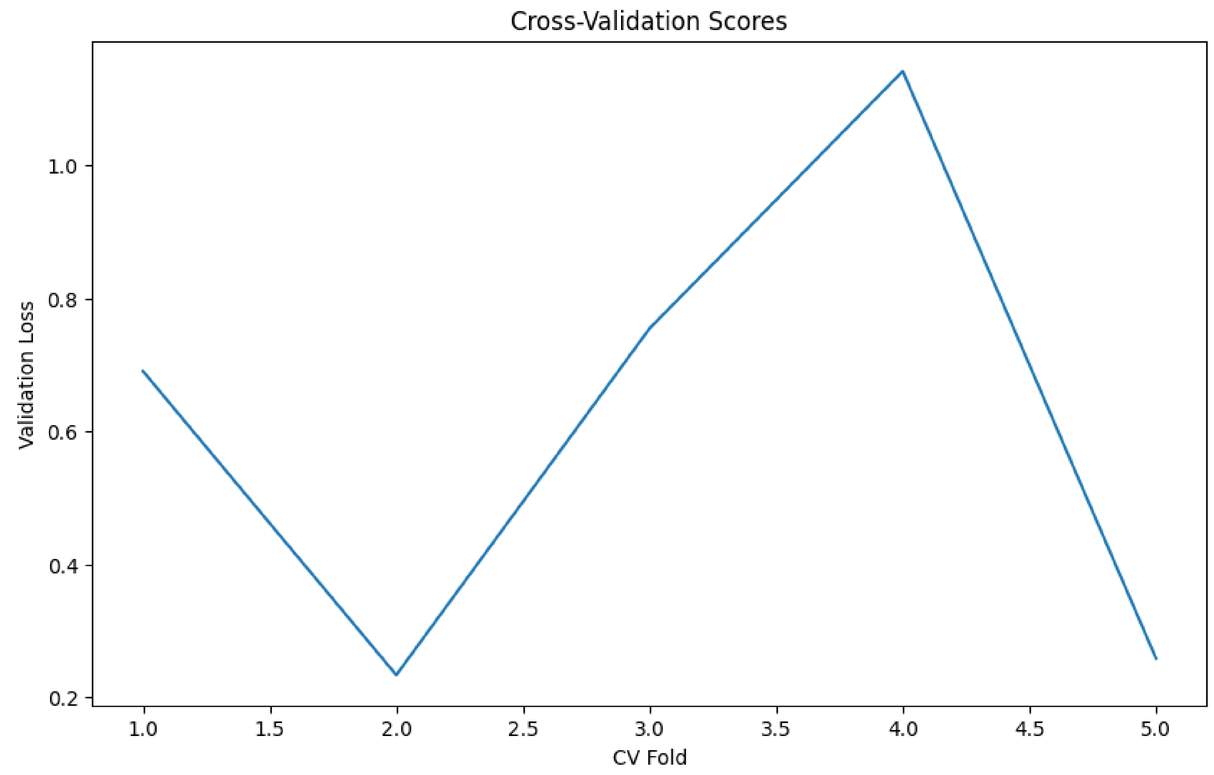

Figure 5. cross validation scores for the ANN model

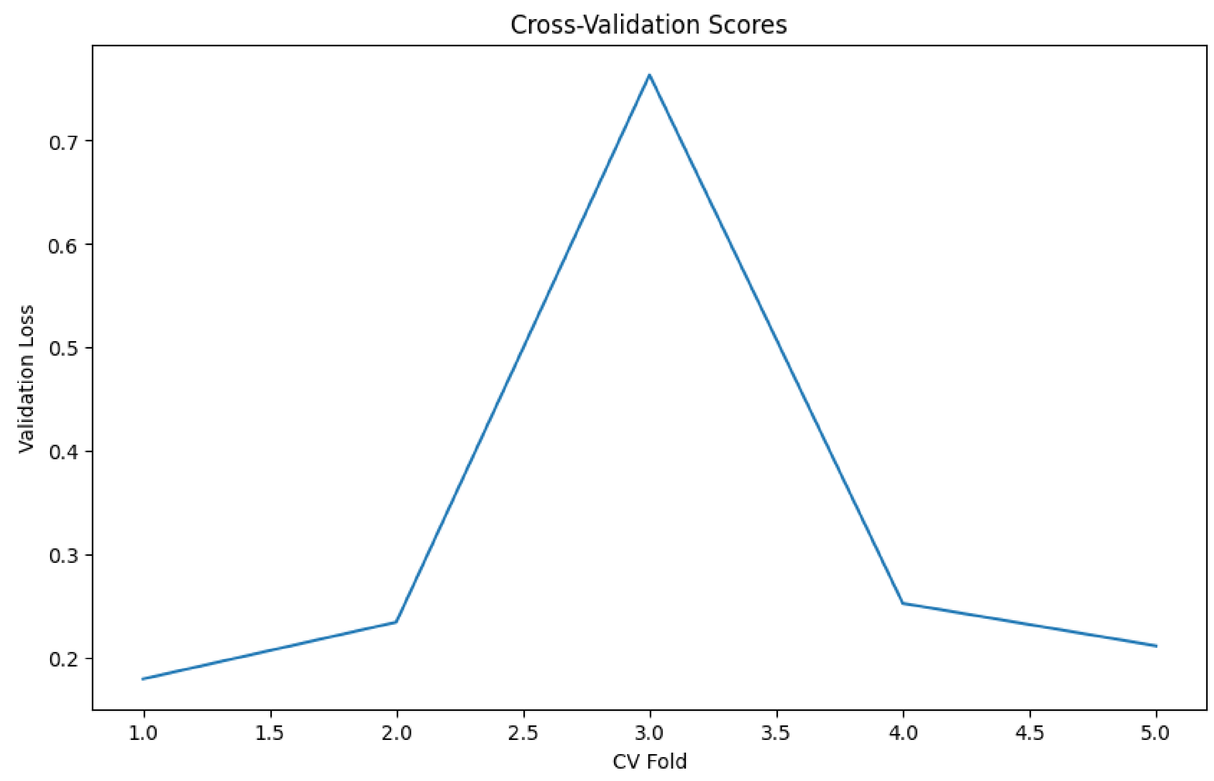

Figure 6. cross validation scores for the ANN model

The incorporation of cross-validation into our modeling approach yielded discernible improvements in predictive accuracy and reliability. By leveraging multiple iterations of training and validation on distinct subsets of the data, cross-validation facilitated a more robust estimation of the model's performance metrics, including the Mean Squared Error (MSE) and R-Squared score. Specifically, after implementing cross-validation, the MSE for Linear Regression was reduced to 0.1845490783452, and for ANN to 0.1986739933490. Furthermore, the R-Squared score for Linear Regression improved to 0.794091918972, and for ANN to 0.77833225204.

	Without Cross-Validation	
	MSE Score	R-Squared Score
Linear Regression	0.201104849576	0.77562005797
ANN	0.185475811362	0.7930579359
Dummy Model	1.225676633689	-0.3675316464
	With Cross-Validation	
	MSE	R-Squared Score
Linear Regression	0.1845490783452	0.794091918972
ANN	0.1986739933490	0.77833225204
Dummy Model	1.257537655307	-0.403080137897

Table 3. Performance Metrics With and Without Cross-Validation

Overall, the integration of cross-validation served as a crucial enhancement to our modeling methodology, providing a more comprehensive and reliable assessment of predictive model performance. By systematically validating the models across multiple subsets of the data, cross-validation contributed to a more accurate estimation of predictive accuracy and generalization ability, thereby enhancing the overall quality and reliability of our predictive models.

Conclusions

In this study, we conducted a comprehensive analysis comparing linear regression and Artificial Neural Network (ANN) models for Bitcoin price prediction. Our investigation into the predictive capabilities of these models revealed valuable insights into their performance and suitability for forecasting cryptocurrency prices.

Through rigorous experimentation and analysis, we found that both linear regression and ANN models exhibited improved predictive accuracy compared to a baseline "dummy" model. The incorporation of sophisticated machine learning techniques, particularly ANN models, enabled us to capture complex nonlinear patterns in the Bitcoin price data.

However, it's important to note that the models we built may not be directly applicable in real-world trading scenarios due to the reliance on historical data. In practical trading environments, real-time data such as today's high and low prices, trading volume, and other market indicators would be crucial inputs for accurate prediction. Our study serves as a foundational exploration of predictive modeling techniques for Bitcoin price forecasting, laying the groundwork for future research and development in this area.

Furthermore, our exploration of feature engineering and model refinement highlighted the importance of selecting appropriate features and addressing issues such as overfitting and underfitting. By leveraging techniques such as early stopping and cross-validation, we were able to enhance the robustness and generalization ability of our predictive models.

Moving forward, future research could explore additional factors and techniques to further improve predictive accuracy, such as sentiment analysis of news and social media data, integration of external market indicators, and advanced deep learning architectures. By continuing to refine and advance predictive modeling techniques, we can contribute to a better understanding of cryptocurrency markets and support informed decision-making in this rapidly evolving domain.