Predicting Bitcoin Log Returns: A Comparative Analysis of Machine Learning Models

Muyang Liu^{1,a,*}

¹Department of Mathematics, University College London, London, UK a. Jeff.liu.22@ucl.ac.uk *corresponding author

Abstract: As the cryptocurrency market continues its rapid growth in both market value and trading volume, it has garnered significant attention. In response to this trend, this paper evaluates four machine learning models-Support Vector Regression (SVR), Random Forest (RF), Long Short-Term Memory (LSTM) networks, and Extreme Gradient Boosting (XGBoost)-to predict the log return of Bitcoin, one of the leading cryptocurrencies. The primary objective is to determine which model performs best in predicting Bitcoin's log return. During the research, the website Kaggle.com provides a solid database for all of the modeltraining processes and evaluations. Multiple experiments are then conducted to optimize each model's hyperparameters for maximum performance. Afterwards, in order to test whether models have learnt the features of data well, metrics like mean squared error (MSE), mean absolute error (MAE), and R-squared (R²) scores are computed and compared. According to the comparison, SVR outperforms the other models, achieving the best prediction accuracy. This paper offers some insightful information for analysts and researchers in the cryptocurrency domain. Additionally, it advances the area by offering a practical guide for utilizing machine learning algorithms to perform time series analysis, demonstrating their effectiveness in cryptocurrency market prediction tasks.

Keywords: Bitcoin, Machine Learning, Support Vector Regression.

1. Introduction

Nowadays, the cryptocurrency market is experiencing a rapid improvement. There is an exponential grow in not only the number but also the value of digital coins on the market. The market capitalization increased dramatically from approximately \$17.7 billion in early 2017 to about \$700 billion in early 2018 [1]. Blockchain technology is an important idea in financial technology which helps advance the development of cryptocurrency [2]. As a well-known pioneer of cryptocurrency, Bitcoin is classified as a new version of peer-to-peer electronic money. It permits direct transactions to occur between the two parties [3]. Moreover, Bitcoin has some special features like decentralization and scarcity due to the blockchain technology. Therefore, it attracts plenty of attention from investors. Hence, the price and log return prediction of Bitcoin seems to be necessary to be carried out to provide the investors with some suggestions.

Along with the gradual development of blockchain technology and cryptocurrency, many researches were conducted in recent years. Wei investigated the relationship between the liquidity and market efficiency of 456 types of cryptocurrencies and reached a conclusion that there is a

[@] 2025 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

positive correlation between these two factors [4]. The role that trading volume plays in forecasting the volatility and returns of cryptocurrencies was also examined by Elie Bouri et al. It is discovered that trading volume gives some helpful insights on predicting the extremums of cryptocurrencies' return [5]. Another research focused on predicting cryptocurrency log returns by first estimating volatility features using autoregressive conditional heteroskedasticity (ARCH). Besides, generalized autoregressive conditional heteroskedasticity (GARCH) is used as a base model with the closing prices of each cryptocurrency. Among the predicted volatility features, the most significant one was selected to form the foundation for log return prediction. Following this, several time series models, including autoregressive integrated moving average (ARIMA), long short-term memory (LSTM) networks, recurrent neural networks (RNN), and gated recurrent units (GRU), were employed for the prediction task. A comparative analysis of model performance revealed that artificial neural networks consistently produced lower error rates. Additionally, it was observed that simpler models tend to outperform more complex ones in terms of accuracy and efficiency [6]. Jakub Drahokoupil constructed a study on the prediction of Bitcoin's close price value using Extreme Gradient Boosting (XGBoost) to help formulate a trading plan during the Corona Virus Disease 2019 (COVID-19) period. Bayesian optimization was also used aiming to get better strategy. The results showed that XGBoost model has enough prediction power and Bayesian optimization is very suitable for optimizing the weights of XGBoost prediction in the trading strategy [7]. In addition, support vector machines (SVM), logistic regression, artificial neural networks, and random forest classification techniques were employed by Erdinc Akyildirim et al. to assess the return predictability of twelve cryptocurrencies [8].

The key focus of this study is to figure out the most suitable model for predicting Bitcoin log returns. To achieve this, several models are fitted to the Bitcoin log return data, specifically Support Vector Regression (SVR), Random Forest (RF), LSTM networks, and XGBoost. After the initial model fitting, hyperparameter tuning is conducted to enhance model performance and improve generalization to unseen data, thus aiming for more accurate predictions. The models are then assessed based on three key metrics: mean squared error (MSE), mean absolute error (MAE), and R-squared (R²) scores. These metrics are compared to determine which model delivers the best predictive performance. The experimental results reveal that SVR outperforms the other models, making it the most suitable option for predicting Bitcoin log returns. This research contributes to the field by offering a framework that allows for more efficient analysis of Bitcoin log returns. Additionally, it provides investors with more accurate predictions, enabling them to formulate more informed and rational investment strategies. Researchers and analysts can build upon these findings to further refine Bitcoin return prediction methodologies, ultimately improving investment decision-making processes.

2. Methodology

2.1. Dataset Description and Preprocessing

The dataset called BTC-USD used for analysis in this study is generated from Kaggle [9]. It gives details of Bitcoin from September 18th, 2014 to February 19th, 2022. Including its price at the start and end of each day, as well as the adjusted closing, maximum, minimum price and the volume of Bitcoin on each particular day. There is overall 2712 data in this dataset with no missing data. The first five columns of dataset are shown in Table 1.

Date	Volume	Low	High	Close	Open	Adj Close
2014.09.17	21056800	452.421997	468.174011	457.334015	465.864014	457.334015
2014.09.18	34483200	413.104004	456.859985	424.440002	456.859985	424.440002
2014.09.19	37919700	384.532013	427.834991	394.795990	424.102997	394.795990
2014.09.20	36863600	389.882996	423.295990	408.903992	394.673004	408.903992
2014.09.21	26580100	393.181000	412.425995	398.821014	408.084991	398.821014

Table 1: First five columns of BTC-USD.

Since the log return of Bitcoin is not included in BTC-USD and this study focuses mainly on the log return, it is necessary to compute and add it to the dataset. The formula to compute the log return is represented as (1) below:

$$r_d = \ln\left(\frac{P_d}{P_{d-1}}\right) \tag{1}$$

where r_d and P_d represent the log return and the closing price on day d.

After adding the log return value to the dataset, the dataset is divided into two sections. One is called the training part with 2260 data in total, the other is the testing part containing 452 data. Then they are normalized to make sure all of them are in the same interval.

2.2. Proposed Approach

The main methods used in this study focus on machine learning algorithms. The preprocessed training data is fitted to SVR, RF, LSTM and XGBoost models and then use these models to make predictions and compare the performance of each model by performance metrics MSE, MAE and R². Then, hyperparameter tuning is carried out to optimize models' performances and afterwards, the performances are compared to obtain the best model. Figure 1 below illustrates the overall structure of this approach.



Figure 1: Pipeline of the research process.

2.2.1.SVR

SVR is an application of regression analysis based on SVM. The aim of SVR is to find an optimal hyperplane such that all the data points are as close as possible to it while minimizing the residual error. Moreover, kernel function is one of the most crucial parts of SVR. With the help of kernel function, this model can handle with non-linear relationships effectively and the operations are carried out in the input space other than higher-dimensional space [10]. The existence of kernel functions makes SVR more flexible, different kinds of kernel functions can be chosen to fit different types of data. Since this study uses a time series data set, it might be influenced by noise and outliers. With the help of ε -insensitive loss function and slack variables, SVR is robust to noise and outliers and then the prediction will be more accurate.

2.2.2.RF

RF is an integration of multiple decision trees. The training part is partitioned and each tree has a task on a distinct subspace of features. The majority vote from each decision tree determines the ultimate choice. This helps reduce the time to train the data and the risk of overfitting [8]. As described above, it can be spotted that RF has great robustness and may not be impacted much by noise. Even without hyperparameter tuning, RF usually has good performance due to its working principle and the risk of overfitting can be reduced by taking the average of the results from each decision tree. Also, RF is capable of capturing some complex non-linear relationships, while log return often depends on several factors, it may not be always linear. Therefore, RF can be used effectively in this study.

2.2.3.LSTM

There are three fundamental components of LSTM networks: an input layer, one or more hidden layers and an output layer. The forget gate, input gate, and output gate are the three gates found in each cell that stores the state value. The input gate indicates which data will be added to the memory, the forget gate specifies which data should be erased, and the output gate indicates which data should be sent out [11]. LSTM has the ability to detect long-term dependency while Bitcoin data often exhibits long-term dependency. Compared to RNN, LSTM overcomes the problem of excessive weight influence and the disadvantage of being prone to gradient vanishing. Moreover, LSTM processes data in order, which matches the feature of time series data well. Therefore, LSTM is a good choice for the Bitcoin log return prediction.

2.2.4. XGBoost

The XGBoost model is an optimized distributed gradient enhancement library which is developed for speed of execution and better performance. XGBoost allows for the incorporation of a wide range of engineered features. It uses gradient-boosted decision trees for feature selection. This can help build a model with better performance [12]. XGBoost can enhance prediction performance by enabling feature engineering and generating features to identify various patterns at different time in this time series dataset. Moreover, XGBoost uses block structures for parallel learning, which contributes a lot to the training speed. Therefore, the process using XGBoost may have higher efficiency than other models. The addition of regularization helps avoid the overfitting of trees and improves the generalization ability of XGBoost. All the above features of XGBoost show that it is suitable to be chosen for time series prediction.

2.3. Implementation Details

This study works with the help of Python and the Scikit-learn library. In order to obtain models with better performance, several techniques of hyperparameter tuning are used including Grid Search, Random Search and Bayesian optimization. Firstly, random search is used in a large range to help narrow it down. Then, in the relatively smaller range, grid search is used to capture the hyperparameter with the best performance. In some models like LSTM, Bayesian optimization is chosen to reach the best results.

3. Result and Discussion

In the conducted study, four models are assessed using different performance metrics. The first one is MSE, which is the average of squared discrepancies of value before and after prediction. MSE can be presented as:

$$MSE = \frac{1}{m} \sum_{k=1}^{m} (\alpha_k - \hat{\alpha}_k)^2$$
 (2)

MAE is another metric. Absolute value of differences before and after prediction is generated and averaged forming MAE:

$$MAE = \frac{1}{m} \sum_{k=1}^{m} |a_k - \hat{a}_k|$$
(3)

 R^2 illustrates how effectively each model fits the target data. The following shows the way to calculate R^2 :

$$R^{2} = 1 - \frac{\sum_{k=1}^{m} (\alpha_{k} - \hat{\alpha}_{k})^{2}}{\sum_{k=1}^{m} (\alpha_{k} - \bar{\alpha}_{k})^{2}}$$
(4)

where m denotes the number of data points, a_k represents the initial value at the kth observation, $\widehat{a_k}$ is the kth value after prediction, $\overline{a_k}$ indicates the mean of the observed values of the dependent variable.

After building models and making predictions, MSE, MAE and R² are generated. However, some of these results have poor performance. Table 2 displays the initial performances of models.

Model	MSE	MAE	\mathbb{R}^2
SVR	0.000780	0.0181	0.529
RF	0.000391	0.0143	0.764
LSTM	0.00116	0.0281	0.298
XGBoost	0.000673	0.0175	0.593

Table 2: Performance metrics of initial models.

From Table 2 above, it is clear that the R^2 score of LSTM is 0.298, which is quite low. Indicating that the actual value and the predicted value differ significantly. Aiming to achieve the best performance as far as possible, as well as to minimize the error, hyperparameter tuning of each model is carried out.

Model	MSE of Train	MSE of Test	MAE of	MAE of	R ² of	R ² of Test
	Data	Data	Train Data	Test Data	Train Data	Data
SVR	0.000768	8.45 e-05	0.0166	0.00537	0.491	0.949
RF	0.000176	0.000374	0.00643	0.0139	0.884	0.774
LSTM	0.000752	0.000410	0.0163	0.0127	0.502	0.752
XGBoost	0.000734	0.000622	0.0178	0.0181	0.513	0.624

Table 3: Performance Metrics of each model after hyperparameter tuning.

Table 3 shows the performance metrics of both training data and testing data after hyperparameter tuning. It is obvious that SVR has the best performance on predicting testing data since its R^2 score 0.949 is the closest to 1 and the MSE and MAE are both the closest to 0. This represents that the predicted value matches the actual value well, indicating that the SVR is performing well. RF has the best performance on training data, this means that it has successfully and effectively captured the important features of training data. By comparing the results in Table 2 and Table 3, hyperparameter tuning has the maximum enhancement on LSTM, improving the R^2 score from 0.298 to 0.752. This may be caused by the Bayesian optimization. This optimization uses probabilistic models to reach the best hyperparameter which is more strategic than methods like Grid Search. In addition, Bayesian optimization can locate the best range of hyperparameters intelligently, it is more possible to find the accurate range.

However, this study has some limitations as well. Since only MSE, MAE and R^2 scores are included to evaluate the result and they all have some drawbacks. For example, MSE is very sensitive to outliers and R^2 may ignore the problem of bias. Therefore, some more metrics should be included

to give an overall view of the performance of models. In addition, some R^2 scores are too high so they may cause problems of overfitting. Hence, cross-validation seems to be necessary to be conducted.

4. Conclusion

This research focuses on identifying a model with strong performance in predicting the log return of Bitcoin. Four models including SVR, RF, LSTM, and XGBoost were constructed and analyzed. Extensive experiments were conducted using the BTC-USD dataset, and model performance was evaluated based on MSE, MAE, and R² scores. The comparison of results shows that SVR consistently produced the lowest MSE and MAE, indicating it had the least prediction error among the models. Thus, SVR is deemed the best-performing model for this dataset. However, this study has certain limitations that future research should address. Notably, it did not account for the potential issue of overfitting, particularly after hyperparameter tuning. Future work should focus on implementing techniques such as cross-validation to mitigate overfitting. Additionally, incorporating other time series models, such as ARIMA, could provide a more comprehensive analysis. Expanding the study to include the log returns of other cryptocurrencies could also offer insights into whether the models used here are universally applicable across different digital assets. This extension could further validate the generalizability of the findings.

References

- [1] Chen, Y., Liu, Y., Zhang, F. (2024) Coskewness and the short-term predictability for Bitcoin return. Technological Forecasting and Social Change, 200, 123196.
- [2] Zhang, C., Ma, H., Arkorful, G.B., et al. (2023) The impacts of futures trading on volatility and volatility asymmetry of Bitcoin returns. International Review of Financial Analysis, 86, 102497.
- [3] Nakamoto, S. (2008) Bitcoin: A peer-to-peer electronic cash system. Satoshi Nakamoto.
- [4] Wei, W.C. (2018) Liquidity and market efficiency in cryptocurrencies. Economics Letters, 168, 21-24.
- [5] Bouri, E., Lau, C.K.M., Lucey, B., et al. (2019) Trading volume and the predictability of return and volatility in the cryptocurrency market. Finance Research Letters, 29, 340-346.
- [6] Sung, S.H., Kim, J.M., Park, B.K., et al. (2022) A Study on Cryptocurrency Log-Return Price Prediction Using Multivariate Time-Series Model. Axioms, 11(9), 448.
- [7] Drahokoupil, J. (2022) Application of the XGBoost algorithm and Bayesian optimization for the Bitcoin price prediction during the COVID-19 period. FFA Working Papers.
- [8] Akyildirim, E., Goncu, A., Sensoy, A. (2021) Prediction of cryptocurrency returns using machine learning. Annals of Operations Research, 297, 3-36.
- [9] Nagadia, M. (2022) Bitcoin Price Prediction using LSTM. Retried from https://www.kaggle.com/code/meetnagadia/ bitcoin-price-prediction-using-lstm
- [10] Ahmad, Z., Almaspoor, Z., Khan, F., et al. (2022) On fitting and forecasting the log-returns of cryptocurrency exchange rates using a new logistic model and machine learning algorithms. AIMS Math, 7, 18031-18049.
- [11] Wu, C.H., Lu, C.C., Ma, Y.F., et al. (2018) A new forecasting framework for bitcoin price with LSTM. 2018 IEEE international conference on data mining workshops, 168-175.
- [12] Oukhouya, H., Kadiri, H., El, Himdi, K., et al. (2024) Forecasting International Stock Market Trends: XGBoost, LSTM, LSTM-XGBoost, and Backtesting XGBoost Models. Statistics, Optimization & Information Computing, 12(1), 200-209.