Context-Enriched Hybrid Modeling for Cryptocurrency Price Prediction: Integrating ARIMAX and Deep Learning Architectures

Gerasimos Vonitsanos[1], Andreas Kanavos[2] and Phivos Mylonas[3]

[1] *Computer Engineering and Informatics Department, University of Patras, Patras, Greece*

[2] *Department of Informatics, Ionian University, Corfu, Greece*

[3] *Department of Informatics and Computer Engineering, University of West Attica, Athens, Greece*

mvonitsanos@ceid.upatras.gr, akanavos@ionio.gr, mylonasf@uniwa.gr

Keywords: Cryptocurrency, Deep Learning, Time Series Forecasting, Neural Networks, Price Prediction, ARIMA, Market Sentiment, Financial Modeling

Abstract: The inherent volatility and nonlinear dynamics of cryptocurrency markets present significant challenges to accurate price forecasting. This study explores a hybrid modeling approach combining classical time series analysis and deep learning techniques to enhance prediction accuracy in the context of Bitcoin price movements. We comprehensively evaluate ARIMA, ARIMAX, Support Vector Machines (SVM), and Long Short-Term Memory (LSTM) networks using high-resolution historical market data from 2019 to 2024. Emphasis is placed on integrating exogenous variables such as market volume, capitalization, and moving averages to enrich model inputs. The experimental results demonstrate that hybrid models, particularly ARIMAX, outperform standalone statistical and machine learning methods in terms of Root Mean Squared Error (RMSE) and R^2 score, achieving superior alignment with actual market trends. The findings underscore the utility of synergistic frameworks that leverage historical statistical regularities and deep learning's capacity to model nonlinear temporal dependencies. This research contributes to the advancement of robust, data-driven tools for financial forecasting in highly dynamic and speculative digital asset markets.

1 INTRODUCTION

The emergence of digital currencies has completely changed the structure of contemporary financial systems, swiftly evolving from specialized technology innovations to essential parts of international transaction networks. Among these, cryptocurrencies—digital tokens supported by blockchain technology and cryptographic mechanisms—have gained much popularity because of their transparency, decentralization, and impenetrability Narayanan et al. (2016). Bitcoin, introduced in 2009, laid the foundation for a rapidly evolving ecosystem now comprising over 5,000 active cryptocurrencies, including dominant platforms such as Ethereum (ETH) and Ripple (XRP) Pintelas et al. (2020). The diversity and velocity of this growth underscore the emergence of a dynamic and volatile market, one increasingly targeted by speculative investors and academic researchers.

One of this domain's most challenging yet intriguing problems is accurately forecasting cryptocurrency prices. While the intrinsic volatility and susceptibility to external shocks render predictive accuracy inherently elusive, the endeavor to model price trajectories remains invaluable. Effective prediction systems can significantly enhance strategic investment decisions, inform macro-financial policy, and deepen insights into the behavioral dynamics of digital financial markets Urquhart (2016).

The academic community has broadly converged on two distinct paradigms for approaching this problem. The first treats cryptocurrency valuation as a canonical time series forecasting task, adopting methodologies historically applied in econometrics and signal processing. Traditional statistical models, particularly the Auto-Regressive Integrated Moving Average (ARIMA), rely on historical patterns and autocorrelation structures to extrapolate future values. Despite their interpretability and simplicity, these models often fail to capture the complex nonlinearities inherent in high-frequency financial datasets.

In contrast, machine learning frameworks—especially those grounded in deep learning—offer a data-driven alternative capable of modeling latent, nonlinear dependencies. These approaches utilize multilayered neural architectures to extract hierarchical features from temporal data, enabling robust function approximation even under conditions of noise and volatility Siami-Namini et al. (2018). Deep learning models, encompassing recurrent structures such as Long Short-Term Memory (LSTM) networks and their variants, have demonstrated superior performance in various domains by leveraging their ability to learn representations across different time scales and data granularities LeCun et al. (2015).

The interaction between the data properties and the model architecture introduces additional levels of complexity. Exogenous factors, such as macroeconomic signals, regulatory changes, and investor sentiment, specifically impact cryptocurrency markets. Hybrid modeling techniques that combine statistical regularities and contextual awareness are required to capture these diverse contributions. This partnership makes the development of adaptable models that can more accurately forecast and generalize in various market scenarios possible.

This study differentiates itself from prior work by proposing a hybrid modeling framework that integrates classical statistical methods with advanced deep learning architectures. Specifically, we incorporate exogenous variables such as market volume, capitalization, and moving averages into ARIMAX and LSTM models to enhance predictive performance and robustness in highly volatile cryptocurrency markets. This multivariate approach enables more context-aware forecasting, bridging the gap between interpretability and accuracy. Our comparative analysis demonstrates that hybrid models offer measurable improvements over standalone techniques, particularly in scenarios characterized by structural shifts and nonstationary behavior.

The remainder of the paper is organized as follows. Section 4 reviews the literature, discussing previous studies and developments in cryptocurrency forecasting to establish the research context. Section 3 details the data preprocessing techniques and methodologies used in our study. Section 4 explains the practical application of the models within our computational framework. Section 5 presents the results of our experiments, providing a comparative analysis of model performances. Finally, Section 6 summarizes the findings and discusses potential directions for future research in cryptocurrency price prediction.

2 RELATED WORK

Forecasting cryptocurrency prices has emerged as a significant research challenge due to the high volatility, nonstationary behavior, and complex market dynamics inherent in digital assets. Traditional statistical approaches, such as Auto-Regressive Integrated Moving Average (ARIMA), initially dominated the field. These models offered a framework for identifying linear patterns in time series but struggled with the nonlinearities and abrupt structural changes typical of cryptocurrency markets Pintelas et al. (2020).

Researchers adopted machine learning techniques such as Support Vector Machines and Random Forests to address these limitations. These methods demonstrated improved flexibility in handling diverse features extracted from price, volume, and technical indicators. However, their inability to model temporal dependencies restricted their effectiveness in capturing sequential dynamics.

This led to a shift toward deep learning, particularly the use of Recurrent Neural Networks (RNNs) and their gated variants—Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks. These models are specifically designed to learn long-term temporal dependencies and nonlinear transformations, allowing them to capture the intricate structure of cryptocurrency time series. In one prominent study, LSTM architectures in Ethereum price prediction yielded marked improvements over classical statistical models and underscored the model's ability to cope with highly volatile environments Zoumpekas et al. (2020).

Further comparative studies have shown that GRU models often achieve similar levels of predictive accuracy to LSTM, while offering lower computational overhead—an advantage for applications requiring rapid inference or operating under resource constraints [51]. Moreover, recent contributions emphasize deep networks' interpretability, ability to generalize across different cryptocurrencies, and adaptability to high-frequency financial data LeCun et al. (2015).

A notable work applied multiple deep learning architectures—including Multilayer Perceptrons (MLP), RNN, LSTM, and Bidirectional LSTM—to large-scale, high-frequency time series of various cryptocurrencies. This study systematically compared the predictive accuracy of each architecture, revealing that advanced recurrent models consistently outperform simpler feedforward networks. Additionally, the research provided methodological insights into the preprocessing and design considerations necessary for effective financial forecasting using neural networks Vonitsanos et al. (2024).

Complementary advancements include hybrid models that integrate traditional time series decomposition techniques with deep neural networks, enabling improved denoising, trend extraction, and resistance to nonstationary effects. These models combine the filtering capabilities of statistical methods with the pattern recognition strength of deep learning to enhance generalization across regimes Narayanan et al. (2016).

Exogenous information has also proven beneficial in enhancing model performance. Features derived from news, sentiment analysis, and macroeconomic indicators are now frequently incorporated into neural architectures, especially in time-sensitive applications where market psychology plays a critical role. Techniques from natural language processing have been leveraged to quantify sentiment from platforms like Twitter and Reddit, enabling models to anticipate price movements driven by investor behavior Siami-Namini et al. (2018).

3 METHODOLOGY

3.1 ARIMA

The Auto-Regressive Integrated Moving Average (ARIMA) model is a foundational statistical technique for univariate time series analysis, designed to model autocorrelations under the assumptions of linearity and stationarity. Its formulation, denoted as ARIMA(p, d, q), integrates autoregressive lags, differencing operations, and moving average smoothing to construct a detailed model of temporal dependencies. Each component contributes a distinct function: the autoregressive term reflects persistence in past values, the differencing step ensures the removal of stochastic trends to attain stationarity, and the moving average term captures serial correlation within the forecast errors.

ARIMA's methodological appeal lies in its mathematical tractability and interpretability, which have supported its longstanding use across macroeconomic and financial forecasting contexts. Parameters are typically optimized through likelihood-based approaches, and model selection is refined via statistical criteria such as the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC), which prioritize goodness of fit and model parsimony Box et al. (2015).The general forecasting equation for an ARIMA(p,d,q) model, after differencing the original time series Y_t d times to achieve stationarity (denoted as y_t), is given by:

$$\hat{y}_t = \mu + \sum_{i=1}^{p} \phi_i y_{t-i} - \sum_{j=1}^{q} \theta_j e_{t-j} \tag{1}$$

In cryptocurrency forecasting, the application of ARIMA has been found to have significant limitations. The model's reliance on linear approximations is ill-suited for capturing the nonlinear and chaotic behavior typical in digital asset markets, which are often influenced by regime shifts, speculative bubbles, and external shocks Azari (2019). Furthermore, the ARIMA model's reliance on strict stationarity assumptions frequently necessitates successive data differencing, which can obscure underlying structural patterns and diminish the interpretability of the resulting model. This over-reliance on differencing may also introduce the risk of overfitting, particularly when modeling complex and volatile financial time series such as cryptocurrencies. Additionally, ARIMA models are inherently limited in their capacity to incorporate exogenous variables—such as macroeconomic indicators, social media sentiment, and regulatory developments—which are often critical in shaping market behavior. These constraints reduce the model's practical applicability in dynamic and information-rich financial environments, where adaptive and multifactorial modeling approaches are required for accurate forecasting Petrică et al. (2016).

In contemporary empirical research, ARIMA is predominantly utilized as a baseline model for assessing the performance of more complex predictive frameworks, including deep learning and hybrid time series models. While it can exhibit adequate forecasting capabilities under conditions of temporal regularity and limited stochastic disturbance, its applicability is notably constrained in financial environments characterized by volatility clustering and non-constant variance—phenomena extensively analyzed in the context of conditional heteroskedasticity Bollerslev (1986).

3.2 Garch

A well-respected statistical modeling method for precisely forecasting short-term returns on financial assets, particularly when such returns appear uncertain, is the Generalized Autoregressive Conditional Heteroscedasticity (GARCH) model. An ARMA process can be used to model the error variances in a GARCH process. One benefit of the GARCH model is its ability to effectively lessen the excessive kurtosis that exists in returns Franses and Van Dijk (1996).

Let ε_t denote the innovation process. It is conditionally normally distributed given past information, denoted by ψ_{t-1}, as follows:

$$\varepsilon_t \mid \psi_{t-1} \sim \mathcal{N}(0, h_t)$$

In this expression, h_t denotes the conditional variance, ψ_{t-1} refers to the information set available at time $t-1$, and $\mathcal{N}$ indicates the conditional normal distribution Fałdziński et al. (2020).

The GARCH(p,q) model, also referred to as GARCH-n, is formulated as:

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j h_{t-j}$$

where $\alpha_0 > 0$, $\alpha_i \geq 0$, and $\beta_j \geq 0$ for $i = 1, 2, \ldots, q$ and $j = 1, 2, \ldots, p$.

Except for volatility variation, which indicates the lack of a trend or seasonal pattern, GARCH models predict future volatility and assume that the time series is stable. While retaining an unaltered unconditional variance, these models exhibit conditional heteroskedasticity and mean reversion.

3.3 SVM

Supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning are the four basic methods that make up the area of machine learning. Support Vector Machines (SVM) are supervised learning models that analyze data for regression and classification using particular techniques. One kind of supervised learning is classification, which uses several example input-output pairings gathered during a training phase to map input data to output data. Features from sample observations are used to train a decision function that correctly assigns class labels. These characteristics range widely, from social media posts to functional neuroimaging data. Using the previously identified trends, the decision function ”classifier” can automatically assign class labels to fresh, unseen observations once they have been formed. The capacity of the SVM to learn data classification patterns with equal accuracy and consistency is its strongest point. Although SVM is sometimes used for regression (Support Vector Regression), it has become increasingly popular as a flexible method for classification in various data formats. Pisner and Schnyer (2020).

SVM offer several advantages, including their effectiveness when the number of dimensions exceeds the number of samples, their ability to work well when there is a clear margin of separation between classes, and their overall effectiveness in high-dimensional spaces. However, SVMs also have notable drawbacks. They tend to perform poorly with vast datasets due to high computational complexity, and their performance can be significantly reduced when dealing with noisy datasets that contain a large amount of irrelevant information. An SVM decision function is an ideal ”hyperplane” that efficiently classifies observations into several groups according to informational patterns, or features. The hyperplane can determine the most likely label for unobserved information. Typically, the features used to infer the hyperplane are not in their original form; instead, they are often derived data that was interpolated during the feature selection process. As support vectors, features are located and recognized by considering their relationships with each other Vapnik (1999).

3.4 LSTM

Developing algorithms that improve with practice is the primary goal of machine learning. Ideally, as the learning process is repeated, the algorithm’s performance gets better. Using the given training data, the learning algorithm is in charge of producing a classifier function. This created classifier is then used on unseen data to assess its effectiveness. Long Short-Term Memory Recurrent Neural Networks (LSTM-RNN) are well known for being very good dynamic classifiers. They are used in machine translation, speech-to-text transcription, language modeling, and many other fields Staudemeyer and Morris (2019).

RNNs are limited to investigating the past in time steps of about ten Staudemeyer and Morris (2019). This happens when the feedback signal either exponentially increases or decreases to zero. With the introduction of LSTM-RNN, this issue was settled. The memory block in the recurrent hidden layer is the central part of an LSTM network. This block comprises memory cells that can preserve temporal information because of their self-connections Pintelas et al. (2020). Adaptive gate units also control the information flow within the block. Depending on how complicated the network architecture is, LSTM may learn more than 1,000 time steps by treating the hidden layer as a memory unit. It can also efficiently handle dependencies in time-series data over short and long periods.

4 IMPLEMENTATION

4.1 Dataset

A CSV file containing the following columns—timeOpen, timeClose, timeHigh, timeLow, name, open, high, low, close, volume, marketCap, and timestamp—contains the dataset that was utilized to train and test the models. With timestamps showing the exact data capture period, each column offers a distinct perspective on the market's activity on a particular day, including opening and closing prices, daily highs and lows, volume traded, and market capitalization.

CoinMarketCap, a provider of several cryptocurrency market statistics, was the source of the information, guaranteeing its correctness and dependability. Cross-referencing with other sources and confirming the consistency of the data points were just two of the several actions taken to ensure the data's integrity. Before being utilized on each model, the dataset was cleaned and preprocessed. This involved addressing missing numbers, fixing any apparent irregularities, and ensuring the dataset was appropriately formatted for analysis.

The dataset covers Bitcoin's daily market activity from September 2019 to February 2024. It offers a strong basis for evaluating the intricate price trends of cryptocurrencies and forecasting future movements. The richness and breadth of the information are essential factors in creating a reliable model for predicting Bitcoin prices. The dataset offers a daily picture of the Bitcoin cryptocurrency's performance, with each record representing a day's worth of market activity. These specifics make a thorough examination of pricing movements throughout time possible.

4.2 Tools

We created our predictive models using Scikit-learn and TensorFlow, two popular machine learning frameworks. The open-source Python package Scikit-learn is widely used because it provides practical and intuitive tools for traditional machine learning algorithms, such as support for statistical modeling and preprocessing methods Silaparasetty and Silaparasetty (2020). Google created TensorFlow, an extensive framework for general machine learning and deep learning applications. Its design uses data flow graphs and efficient tensor operations and is ideal for large-scale numerical computation.

5 EXPERIMENTAL EVALUATION

5.1 Experimental Setup

To ensure a consistent and fair comparison, all models were trained and evaluated on the same dataset comprising daily Bitcoin market data from September 24, 2019, to February 6, 2024. The dataset was divided into training (80%) and testing (20%) subsets. Preprocessing steps included handling missing values, converting timestamps to datetime format, and normalizing features where necessary. Each model—ARIMA, ARIMAX, SVM, and LSTM—was implemented using Python and trained using their respective optimal configurations. Evaluation was performed using standard metrics to assess predictive performance and generalization on unseen data.

5.2 Statistical Tests

The four models evaluated in this study span statistical, machine learning, and deep learning paradigms. **ARIMA** is a univariate model that forecasts future values based on past observations and their errors. It operates under the assumption of stationarity, thus requiring differencing for non-stationary time series. **ARIMAX** extends ARIMA by incorporating exogenous variables—specifically volume, market capitalization, and a 7-day moving average of closing prices—allowing the model to learn from additional market indicators and potentially improve predictive performance. SVM for regression was implemented using a radial basis function (RBF) kernel, enabling the model to capture complex, non-linear relationships in the data. This approach is particularly effective for small- to medium-sized datasets with high dimensionality. Lastly, LSTM networks, a variant of recurrent neural networks (RNNs), were employed due to their ability to learn long-range dependencies in sequential data. LSTM models are especially suited for financial time series forecasting, as they can handle the inherent volatility and noise present in such datasets.

To assess and visualize each model's forecasting accuracy, we compared the predicted and actual Bitcoin prices. Figure 1 presents the prediction results for ARIMAX, SVM, and LSTM models against actual values. The ARIMAX model demonstrates the highest alignment with the observed price trends, particularly during periods of strong market movement. LSTM also exhibits robust performance, capturing the general trajectory of price changes and adapting well to nonlinear patterns. In contrast, the SVM model performs adequately but struggles to maintain accuracy during sudden price fluctuations, underscoring the challenges of modeling high-volatility assets with non-temporal models.

Additionally, Figure 2 illustrates the time series plot of Bitcoin prices from 2019 to 2024, contextualizing the volatility and irregular seasonal trends present in the dataset. These fluctuations emphasize the complexity of the forecasting task and the im-

Figure 1: Comparison of predicted vs. actual Bitcoin prices using ARIMAX, SVM, and LSTM models. ARIMAX demonstrates the closest alignment with real market trends, followed by LSTM.

portance of choosing models capable of recognizing long-term dependencies and structural shifts.

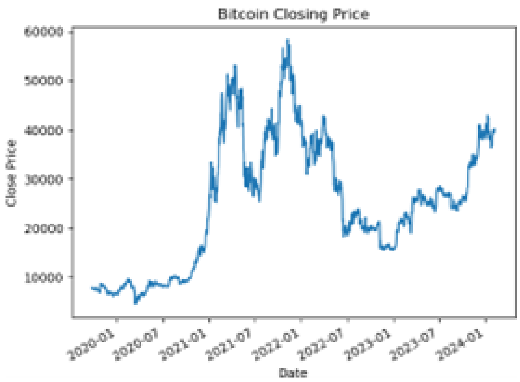

Figure 2: Daily closing prices of Bitcoin from 2019 to 2024. The time series exhibits high volatility and irregular patterns.

Finally, we include Figure 3 to display histograms of residuals for the ARIMA model. The distribution exhibits skewness and leptokurtic behavior, indicating a departure from normality. This non-Gaussian error structure supports adopting more advanced models such as LSTM and ARIMAX, better suited for handling non-linear and non-stationary data with irregular error patterns.

5.3 Evaluation Metrics

The models were evaluated using two key quantitative metrics for a robust comparison. The Root Mean Squared Error (RMSE measures the average magnitude of the prediction error and is particularly sensitive to large deviations, making it effective for highlighting significant inaccuracies. The R^2 Score, also known as the Coefficient of Determination, indicates the proportion of variance in the dependent variable explained by the model, thus providing a measure of its explanatory power.

In addition to these metrics, further statistical diagnostics were explicitly applied to the ARIMA and ARIMAX models. The **Augmented Dickey-Fuller (ADF) test** was employed to assess the stationarity of the time series data, which is a critical assumption for the validity of these models. To evaluate whether the residuals followed a normal distribution, the **Jarque-Bera test** was conducted. Furthermore, the **Ljung-Box test** and **Autocorrelation Function (ACF) plots** were used to detect any autocorrelation remaining in the residuals, ensuring the adequacy and reliability of the model fit.

Table 1 presents the performance metrics for each forecasting model. The evaluation used the Root Mean Squared Error (RMSE) and the Coefficient of Determination (R^2 score). Among all models, ARIMAX achieved the best performance with the lowest RMSE and the highest R^2 value, indicating strong predictive accuracy and generalization capability. The SVM model also demonstrated solid performance, slightly outperforming the LSTM model. In contrast, the ARIMA model yielded significantly higher errors, underscoring its limitations in capturing Bitcoin prices' complex and volatile behavior.

Table 1: Performance Metrics of Forecasting Models

Model	**RMSE**	**R^2 Score**
ARIMA	7012.59	–
ARIMAX	**508.45**	**0.9920**
SVM	793.32	0.9806
LSTM	943.17	0.9732

6 CONCLUSIONS AND FUTURE RESEARCH

With an emphasis on Bitcoin, this research aimed to examine how well sophisticated deep learning algorithms anticipate the price movements of cryptocurrencies. The aim was to find the best model for the Bitcoin Historical Price time-series dataset. We trained and tested the models using data spanning four years. The bitcoin market is going through a crucial stage at this time.

Through a thorough analysis of the three distinct models—ARIMA, SVM, and LSTM—this study aims to assess each one's quality and predictive capacity for future Bitcoin prices. Two key metrics—root mean square error (RMSE) and coefficient of determination (R^2 score)—formed the basis of the comparison analysis. Visual graphs were then used to illustrate the models' accuracy, error rates, and capacity to capture variance.

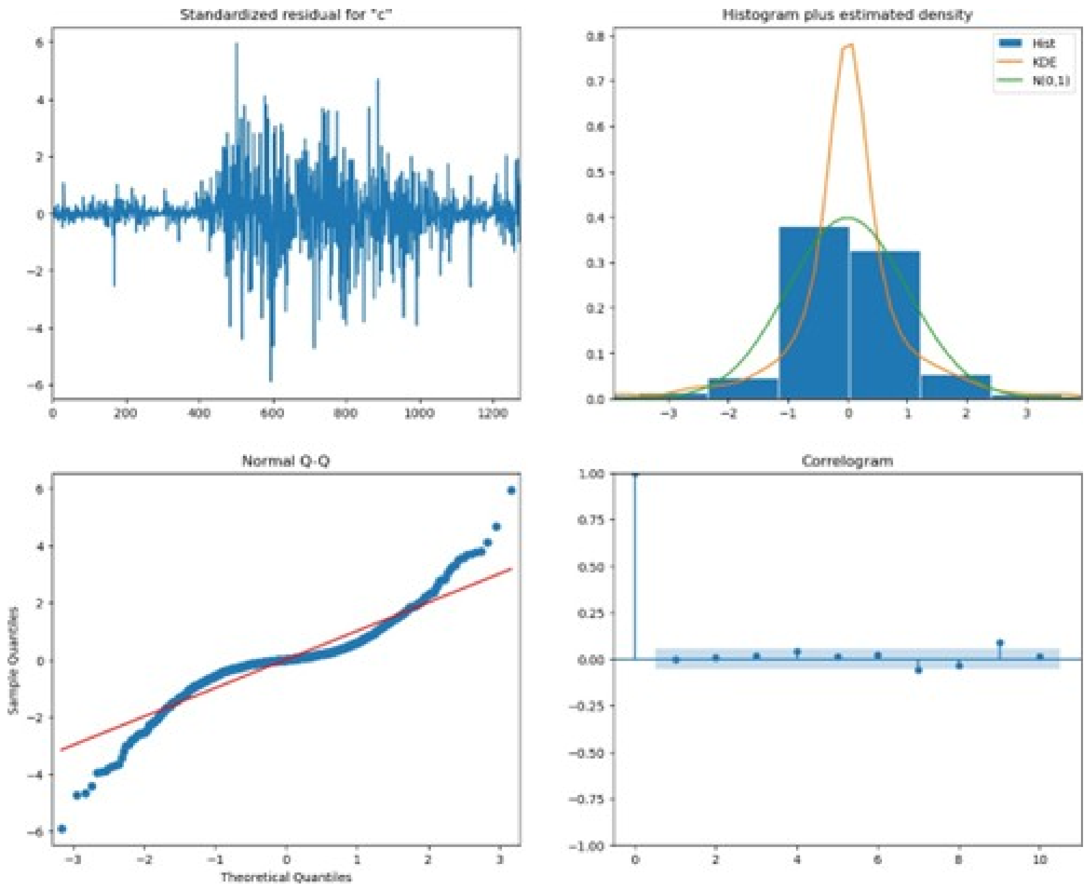

Figure 3: Histogram and Q-Q plot of ARIMA residuals. The results reveal significant deviations from normality, supporting more robust forecasting models.

The study emphasizes the importance of choosing the right forecasting model depending on specific requirements and circumstances. The ARIMAX model is an excellent option for situations requiring high accuracy and precision because it can produce improved predictions on our Bitcoin dataset. However, because of its remarkable capacity to explain variance, the SVM model is also a viable option for applications that demand a compromise between computing efficiency and accuracy. Last, the LSTM model might be favored in circumstances requiring the recognition of intricate patterns across time or real-time prediction abilities, even if it has greater error rates. In contrast to prior studies that emphasize deep learning architectures alone, our work establishes that statistical-deep learning hybrids, when equipped with exogenous market features, can significantly outperform standalone deep models. This finding suggests a paradigm shift from model-centric optimization to data-centric integration for financial forecasting tasks.

Future studies could concentrate on creating hybrid models that further improve prediction efficiency and accuracy by combining the benefits of SVM, LSTM, and ARIMAX. Additionally, adding more cryptocurrencies and extended periods to the data set may offer a deeper understanding of how well the models perform in various market scenarios. It may also be possible to increase forecast accuracy by looking at the effects of outside variables like significant market occurrences or regulation changes.

REFERENCES

Azari, A. (2019). Bitcoin price prediction: An arima approach. *arXiv preprint arXiv:1904.05315*.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327.

Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.

Fałdziński, M., Fiszeder, P., and Orzeszko, W. (2020). Forecasting volatility of energy commodities: Comparison of garch models with support vector regression. *Energies*, 14(1):6.

Franses, P. H. and Van Dijk, D. (1996). Forecasting stock market volatility using (non-linear) garch models. *Journal of forecasting*, 15(3):229–235.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

Narayanan, A., Bonneau, J., Felten, E., Miller, A., and Goldfeder, S. (2016). *Bitcoin and cryptocurrency technologies: a comprehensive introduction*.

Petrică, A.-C., Stancu, S., and Tindeche, A. (2016). Limitation of arima models in financial and monetary economics. *Theoretical & Applied Economics*, 23(4).

Pintelas, E., Livieris, I. E., Stavroyiannis, S., Kotsilieris, T., and Pintelas, P. (2020). Investigating the problem of cryptocurrency price prediction: a deep learning approach. In *Artificial Intelligence Applications and Innovations: 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings, Part II 16*, pages 99–110.

Pisner, D. A. and Schnyer, D. M. (2020). Support vector machine. In *Machine learning*, pages 101–121.

Siami-Namini, S., Tavakoli, N., and Namin, A. S. (2018). A comparison of arima and lstm in forecasting time series. In *2018 17th IEEE international conference on machine learning and applications (ICMLA)*, pages 1394–1401.

Silaparasetty, N. and Silaparasetty, N. (2020). The tensorflow machine learning library. *Machine Learning Concepts with Python and the Jupyter Notebook Environment: Using Tensorflow 2.0*, pages 149–171.

Staudemeyer, R. C. and Morris, E. R. (2019). Understanding lstm–a tutorial into long short-term memory recurrent neural networks. *arXiv preprint arXiv:1909.09586*.

Urquhart, A. (2016). The inefficiency of bitcoin. *Economics Letters*, 148:80–82.

Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999.

Vonitsanos, G., Kanavos, A., Grivokostopoulou, F., and Sioutas, S. (2024). Optimized price prediction of cryptocurrencies using deep learning on high-volume time series data. In *2024 15th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pages 1–8.

Zoumpekas, T., Houstis, E., and Vavalis, M. (2020). Eth analysis and predictions utilizing deep learning. *Expert Systems with Applications*, 162:113866.