Uncertainty Quantification for Cryptocurrency Price Forecasting

Enoch Luk [1] **Cynthia Chen** [1] **Sihan Chen** [1]

Abstract

The prediction of cryptocurrency price has attracted significant attention in recent years due to its potential for large financial gains. However, the inherent volatility of the market poses considerable challenges to accurate forecasting. Traditional forecasting model often overlooks the uncertainty associated with cryptocurrency price predictions. In our paper, we apply ARIMA model and Bayesian Neural Network (BNN) to perform cryptocurrency price forecasts and investigate their prediction uncertainty by implementing a conformal-adjusted confidence interval. We find that conformal prediction provides a robust uncertainty quantification in both cases, offering a direct way to compare the performance of their respective confidence intervals. Using evaluation metrics that take into account price variation, we find that although ARIMA model offers a more effective confidence interval compared to BNN with price inputs, BNN with log price input performs better when dealing with a highly volatile market.

1. Introduction

Uncertainty quantification (UQ) plays a pivotal role in finance, serving as the foundational cornerstone for understanding and managing the intrinsic uncertainties deeply ingrained in financial models and data. Given the rapidly evolving digital age, cryptocurrencies have emerged from the periphery to become a substantial player in the financial landscape. Their appeal, largely predicated on decentralized control and potential for substantial returns, has captivated investors and researchers alike.

However, cryptocurrencies bring with them a unique set of challenges, primarily their pronounced volatility, that make predicting price movements and managing risks an especially arduous task. Their price movements often diverge significantly from a lognormal distribution that is typically assumed in many traditional finance methods. The excess kurtosis for our short-time frame crypto dataset was measured at 543, in comparison to the SP 500 daily returns, which has an excess kurtosis of 10. This demonstrates how the price changes of cryptocurrencies in the short-term exhibit extreme spikes and crashes, straying far from the Gaussian assumption (excess kurtosis = 0) and making them particularly difficult to model and predict.

Addressing this high kurtosis and extreme volatility, the application of UQ techniques in cryptocurrency price prediction and risk management presents a novel opportunity for pioneering innovative solutions and improvements in existing methods. Our investigation involves leveraging well-known techniques - ARIMA (AutoRegressive Integrated Moving Average) modeling, Bayesian neural networks (BNNs), and conformal prediction, known for their ability to quantify uncertainty in complex scenarios.

BNNs extend traditional neural networks by turning them into probabilistic models, providing a distribution of outcomes rather than a single point prediction. This distribution can capture the underlying uncertainty more accurately than point predictions.

ARIMA models offer a robust method for understanding and predicting future points in a series that is based on historical data. These models are tailored to capture the intricate temporal dependencies often present in time-series data, which makes them particularly suitable for financial data like ours. By considering autoregressive terms, differencing operations, and moving average terms, ARIMA models can effectively capture a wide range of temporal structures.

Conformal prediction, on the other hand, provides a way to quantify the confidence in the prediction of a model, which aligns well with the notion of uncertainty. Together, these techniques offer a robust and comprehensive approach to tackle the inherent uncertainty in short-time frame cryptocurrency data.

Therefore, the primary objective of our project is to look into the application and comparative performance analysis of these mathematical methods for predicting prices and

[1]Caltech. Correspondence to: Enoch Luk, Cynthia Chen, Sihan Chen <eluk@caltech.edu, cchen6@caltech.edu, sihanc@caltech.edu>.

CS 159: Advanced Topics in Machine Learning, Pasadena, California, USA. Spring 2023. Based on the ICML 2023 submission template.

trends in short-time frame cryptocurrency data. Our findings and contributions will provide valuable insights into uncertainty quantification in the highly volatile and challenging landscape of cryptocurrency markets.

2. Related Work

The use of machine learning algorithms in stock price forecasts has received much attention in recent years. In order to quantify the uncertainty, Chandra and He applied Bayesian neural networks for stock price prediction before and during COVID (Chandra & He, 2021). While their approach achieves results pre-Covid, the true stock prices do not fall within their model's confidence regions during COVID, a time of higher volatility. To our knowledge, there are no existing well-calibrated BNN approaches for predicting more volatile stock prices, and there are no BNN approaches for crypto markets, which are inherently more volatile. In addition to a Bayesian approach to UQ, researchers have recently experimented with a frequentist approach with conformal prediction (CP). It has the advantage of offering a robust confidence interval without assuming a specific prediction model, such as in the typical Gaussian processes approach. Kath and Ziel employed CP to forecast short-term electricity prices and proposed a path-based guideline for different markets (Kath & Ziel, 2021). Wisniewski, Lindsay, and Lindsay applied inductive conformal prediction using the window approach proposed in Kath and Ziel to generate a market maker's net position over time (Gammerman et al.). Chernozhukov, Wüthrich, and Zhu use a CP model to predict stock price return based on market volatility (Chernozhukov et al., 2021).

Despite the vast literature employing machine learning methods to study stock price behavior, there has not been much attention devoted to the cryptocurrency market. One reason is that most of the studies were performed using decade-long stock market data, and the cryptocurrency market rose in popularity only in recent years. We wish to bridge the gap by employing existing ML methods on short-term crypto market data.

3. Methods

3.1. Dataset

We use three months of Bitcoin tether order book data (bid/ask prices and volumes every millisecond) collected from June 2021 to September 2021. We compute open, close, low, and high prices for ten-minute intervals from the millisecond data. The dataset contains 15,019 rows, with each row representing a ten-minute interval of data.

We split the dataset into training and test sets. We use approximately 70 percent for training and 30 percent for testing.

3.2. BNN

For our modeling approach using a Bayesian neural network, we used two different representations of data. First, we simply used the price as the target variable. The dataset consists of 15 columns, including the timestamp and various features representing the price changes at a certain number of timesteps before (prev-0, prev-1, ..., prev-99). The target variable, price$_i$, is denoted as y.

Then, we ran the BNN again using the log price changes between intervals as features where y = log(price$_{i+1}$ / price$_i$). This would more accurately reflect commonly held beliefs in traditional finance modeling. We transformed the output back into price predictions by multiplying the predicted change by the current price.

For the implementation of Bayesian Neural Networks (BNNs), we utilized the numpyro library supported by PyTorch. To find optimal parameters, our grid search explored three different values for the number of dimensions of the hidden layers: 10, 20, and 30. For each of these configurations, we tested three different learning rates: 1e-5, 1e-3, and 1e-2, yielding a total of nine unique parameter configurations. The BNN was constructed with two hidden layers, both of which had the same number of dimensions, and an output layer of dimension 1.

The BNN was trained using the Stochastic Variational Inference (SVI) method with the Adam optimizer and the Evidence Lower Bound (ELBO) as the loss function. The training process ran for 20,000 iterations.

Overall, the results suggested that a 'dim' of 10 and a 'lr' of 1e-2 was the optimal configuration for this BNN architecture and dataset, as it achieved a lower ELBO loss with relatively less uncertainty across the different configurations tested.

3.3. ARIMA

We use a traditional statistical method, AutoRegressive Integrated Moving Average (ARIMA) to predict the next time step open price. ARIMA(p, d, q) model combines differencing with autoregression and moving average models (Hyndman & Athanasopoulos, 2018), and the full model is written as

$$\begin{aligned} y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} \\ + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t, \end{aligned}$$

where $y_t^{'}$ is the difference series (to order d), ϵ_{t-i} are past forcast errors, ϵ_t is white noise, p is the order of the autoregressive parts, d is the degree of the first differencing involved, and q is the order of the moving average part.

Both AR and MA models are only applicable to stationary time series. To make our price data stationary, we perform differencing and ran the augmented Dickey-Fuller test until our time series are stationary. Stationarity is achieved after we differenced our data once.

The model is trained on all previous open price. We find the best ARIMA model ARIMA(0, 1, 0) using auto ARIMA from pmdarima. ARIMA(0, 1, 0) reduces to a random walk model, where

$$\hat{y}_t = \mu + \hat{y}_{t-1}.$$

A 95% confidence interval is given by $\hat{y}_{t+1|t} \pm 1.96\hat{\sigma}$, where $\hat{\sigma}$ is standard deviation of the residuals.

3.4. Conformal Prediction

We apply conformal prediction using $\alpha = 0.1$ to allow for comparison of the two models' confidence intervals. We define our nonconformity score as

$$s(x, y) = \max\{\hat{t}_{\alpha/2} - y, y - \hat{t}_{1-\alpha/2}(x)\}$$

and the prediction interval as

$$[C(x) = \hat{t}_{\alpha/2} - \hat{q}, \hat{t}_{1-\alpha/2}(x) + \hat{q}].$$

Due to the inherent nature of time series data, the data distribution changes over time. We apply weighted conformal prediction under distribution shift as in (Angelopoulos & Bates, 2022), where

$$\hat{q} = \inf\{q : \sum_{i=1}^{n} \tilde{w}_i \mathbb{1}\{s_i \leq q\} \geq 1 - \alpha\}, \text{ guaranteeing}$$

$$P(Y_{test} \in C(X_{test}) \geq 1 - \alpha - 2\sum_{i=1}^{n} \tilde{w}_i \epsilon_i,$$

where $\epsilon_i = d_{TV}((X_i, Y_i), (X_{test}, Y_{test}))$ and $\tilde{w}_i = \frac{w_i}{w_1 + \ldots + w_n + 1}$. We use a rolling window of size $K = 100$. Then $w_i = \mathbb{1}\{i \geq n - K\}$ and $\tilde{w}_i = \frac{1}{K+1}$, so

$$\hat{q} = \lceil \frac{(K+1)(1-\alpha)}{K} \rceil.$$

We also apply conformal prediction without accounting for distribution shift (unweighted CP), where we just use the first 1000 data points of the test set for calibration as a point of comparison.

3.5. Evaluation

We evaluate the confidence intervals of the BNN and ARIMA models after standardizing their coverage using conformal prediction. To achieve this end, we define three different metrics. The first metric to examine is the gap in the confidence interval. A more accurate forecast is expected to have a smaller confidence interval. Denoting the upper/lower confidence interval bound at time step t as $\text{UCI}_t/\text{LCI}_t$, we define the metric s_1 that quantifies the gap of the confidence interval as

$$s_1 = \frac{1}{N} \sum_{t=1}^{N} (\text{UCI}_t - \text{LCI}_t),$$

where N is the total number of predictions made by our model.

In addition, we observe that stock price tends to exhibit a periodic variation in the variance of price, with larger movements in trends followed by smaller movements during the consolidation period. The standard deviation of price σ_t is computed in a trailing fashion, using the mean price of the past n timesteps starting at time t. A good confidence interval is expected to capture the change in price variation, with increasing variation corresponding to a widening of confidence intervals and decreasing variation corresponding to a shrinking of confidence intervals. Define the metric s_2 that tracks the change in confidence interval gap with price variation as

$$s_2 = -\frac{1}{N} \sum_{t=1}^{N} \frac{\Delta_{t+1} - \Delta_t}{\Delta_t} \frac{\sigma_{t+1} - \sigma_t}{\sigma_t},$$

where $\Delta_t = \text{UCI}_t - \text{LCI}_t$. The negative sign is included so that the lower the s_2 score the better, as an opposite change in confidence interval gap with price variation contribute positively to the sum.

We can further incorporate price variance in accessing the gap of the confidence interval. Our end goal is to create a profitable trading strategy, which oftentimes is triggered when the price shows volatile movement in a specific direction. Therefore, we would like to place more emphasis on the performance of confidence intervals during times of large price variance. The variance-assisted confidence interval gap s_3 is defined as

$$s_3 = \frac{1}{N} \sum_{t=1}^{N} \frac{(\text{UCI}_t - \text{LCI}_t)\sigma_t}{\sum_{t'} \sigma_{t'}},$$

where t' goes through all predicted data points.

4. Results

4.1. Initial Model Price Forecasts and Confidence Intervals

We perform one step ahead price forecasts and obtain confidence intervals from each model (Figure 2). The intial coverages for each model are shown in 1 under the Before CP column. We observe that BNN with price inputs and BNN with log price change inputs both have very high initial coverage, although the confidence intervals of the BNN with log price change inputs are significantly narrower, indicating that log price changes are better input features than raw prices.

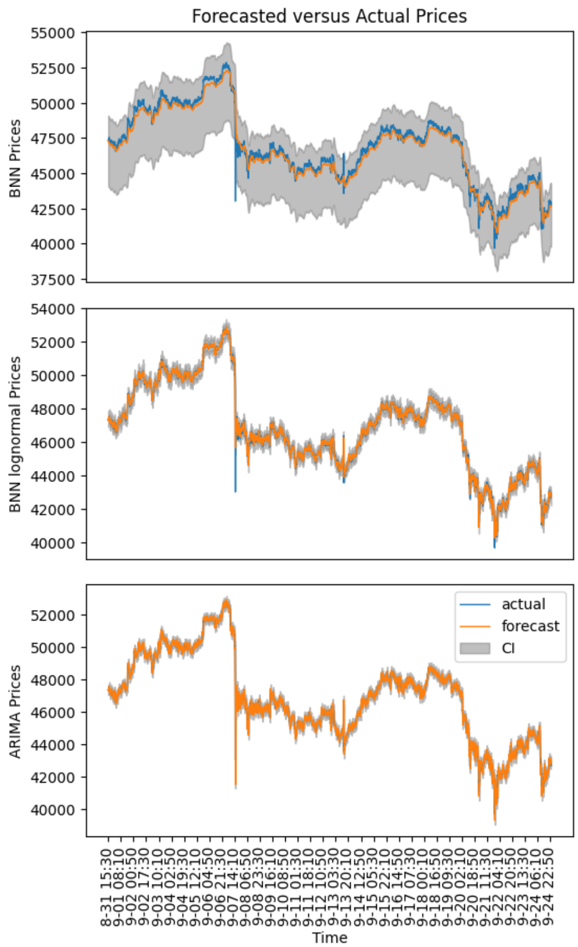

Figure 1. One step ahead price forecasts and confidence intervals for BNN, BNN with log price change inputs, and ARIMA.

4.2. Comparison of unweighted CP and weighted CP for Time Series Data

To compare the results of using unweighted CP and weighted CP for time series data, we compute the coverage for our models before conformal prediction, after unweighted conformal prediction, and after weighted conformal prediction with a rolling window of size K = 100 over the whole test set (Table 1). We also compare the coverages of each model before CP, after unweighted CP, and after weighted CP over a rolling window of size M = 500 (Figure 2).

While the coverage is approximately 0.9 after both unweighted and weighted CP over the entire test interval, the coverage fluctuates significantly under unweighted CP, while remaining stable about 0.9 under weighted CP. Thus, using a rolling window for CP effectively accounts for the distributional shift of time series data.

	Before CP	Unweighted CP	Weighted CP
BNN	0.9997	0.888	0.887
BNN (log input)	0.9917	0.8930	0.8999
ARIMA	0.9307	0.8982	0.9019

Table 1. Coverage comparison before CP, after unweighted CP, and after weighted CP calculated over entire test set

4.3. Confidence Intervals after weighted CP

In order to better compare the confidence intervals of each model, we obtain conformal-adjusted confidence intervals for each model using $\alpha = 0.1$ and rolling window CP. Figure 3 shows the before and after CP confidence intervals of each model over a 24 hour time period. We observe that the confidence intervals of both BNNs narrow significantly after CP as expected.

4.4. Metric Evaluation for Confidence Intervals

Using our evaluation metrics, we examine the performance of confidence interval from BNN with price inputs, BNN with log price changes, and ARIMA model, and the results are shown in Table 2. The ARIMA model outperforms both BNNs under s_1, but under s_2 and s_3 (after taking price variance into account), the BNN trained on log price change inputs outperform the other two methods.

	s_1	s_2	s_3
BNN	2362	-0.00005845	0.6581
BNN (log input)	307.3	-0.001623	0.09288
ARIMA	444.6	-0.0001053	0.1432

Table 2. Confidence Intervals Evaluation

The price forecast and confidence of intervals of ARIMA(0,1,0) and BNN after applying conformal prediction with distribution drift are shown in Fig. 4. ARIMA models provides a much narrower confidence interval than BNN. Furthermore, the ARIMA's point forecast lies around the center of the predicated confidence interval, whereas BNN tends to have a wider confidence interval on the downside.

As shown in Fig. 5, the change of confidence interval gap of ARIMA(0,1,0) follows more closely with the change in price's standard deviation when compared to BNN, which is also captured quantitatively in Table 2. This is noticeably

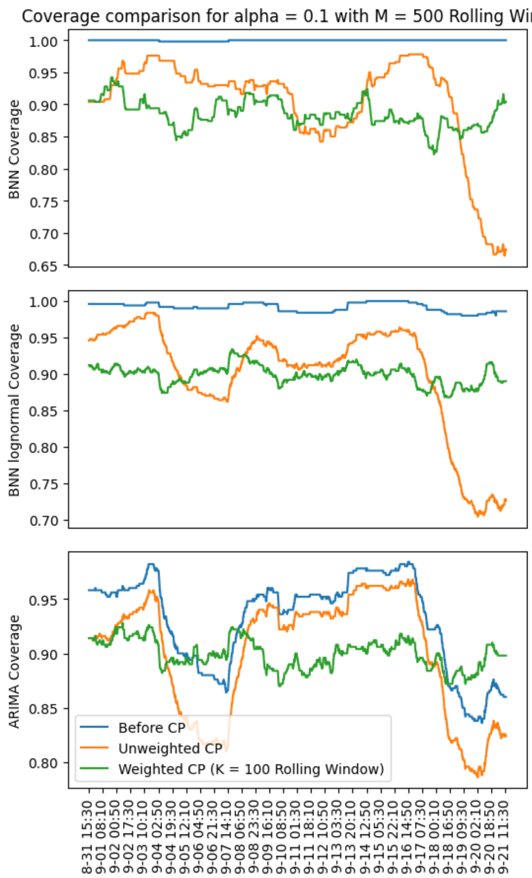

Figure 2. Coverage comparison before CP, after unweighted CP, and after weighted CP calculated over a rolling window of size M = 500.

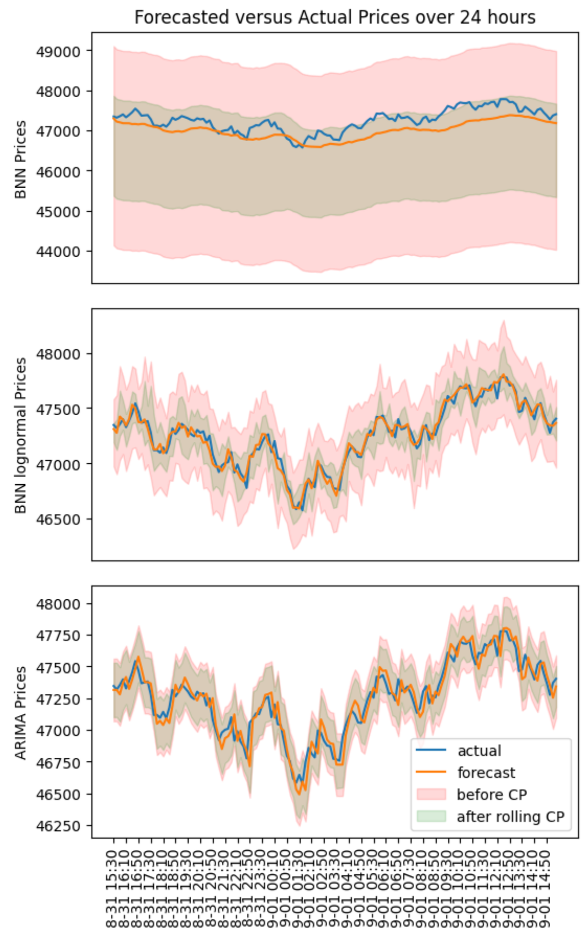

Figure 3. Forecasted prices over 24 hours with confidence intervals before CP and after rolling CP

true at time $t \approx -0.3$ where as spike in price's standard deviation occurs. The ARIMA model is quick to adapt to the abrupt change in price movement whereas BNN models experienced a longer lag time.

5. Discussion

Our study revealed that both BNN and ARIMA models possess strong predictive potential for cryptocurrency price forecasting, albeit with certain limitations. The BNN model achieved a high coverage before conformal prediction with extremely wide intervals, and CP brought the confidence closer to its desired accuracy The ARIMA model, on the other hand, showed a more consistent performance before and after CP, but with lower initial coverage.

The use of conformal prediction in both models provided a valuable measure of the uncertainty and variability in cryptocurrency price forecasting and our comparison between weighted and unweighted CP demonstrated the effects of distribution shifts in time-series data on coverage levels.

In comparing Bayesian Neural Networks (BNNs) to ARIMA models for time series forecasting, it is important to understand how each model processes data and their respective strengths and limitations.

ARIMA is a class of models specifically designed for time series forecasting. ARIMA models take into account three aspects: the autoregressive component (AR), the integrated component (I), and the moving average component (MA). The AR component is the regression of the time series onto itself, the I component corrects the non-stationarity of the data, and the MA models the errors based on past errors. One of the key aspects of ARIMA is that it explicitly models the temporal dependencies in the data, which is crucial for time series analysis.

On the other hand, BNNs, like most neural network models, do not intrinsically model temporal dependencies unless they are specifically designed to do so, such as Recurrent Neural Networks (RNNs) or Long Short-Term Memory networks (LSTMs). The assumption of exchangeability,

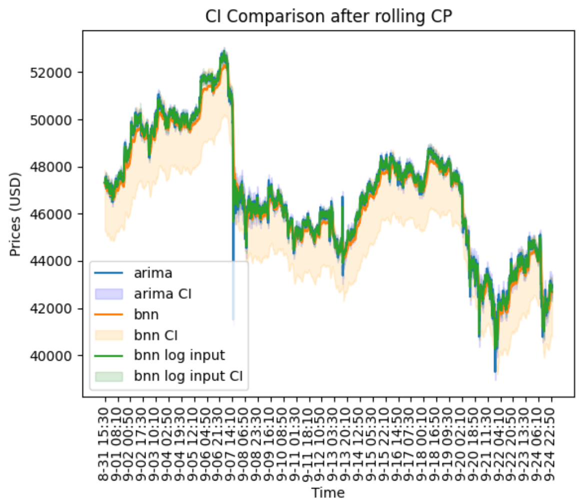

Figure 4. Plots of ARIMA(0,1,0) and BNN model price forecast with conformal-adjusted confidence intervals

which is the presumption that the order of observations doesn't matter, is generally true for BNNs. This assumption is violated in time series data, where the order and timing of observations matter significantly.

The performance of ARIMA and BNNs can vary depending on the characteristics of the data. If the time series data exhibits strong temporal dependencies and is stationary (i.e., its properties such as mean and variance remain constant over time), ARIMA models can perform better as they are designed to handle such data.

6. Conclusion

In conclusion, our research showed the potential of utilizing BNN and ARIMA models for uncertainty quantification in cryptocurrency price forecasting. While each model has its strengths and weaknesses, they both offer valuable insights into the volatile nature of cryptocurrency prices and provide a foundation for managing risk and making informed decisions in this rapidly evolving market.

Future work should explore additional methods for handling time-dependent data, such as using the current variance as an additional calibration parameter for price intervals. Another possible improvement could be incorporating both technical and fundamental information, such as order book depth or relevant news, which could provide further predictive power. Additionally, a more extensive study with a larger dataset spanning a longer period may provide further insights into the long-term trends and patterns in cryptocurrency prices.

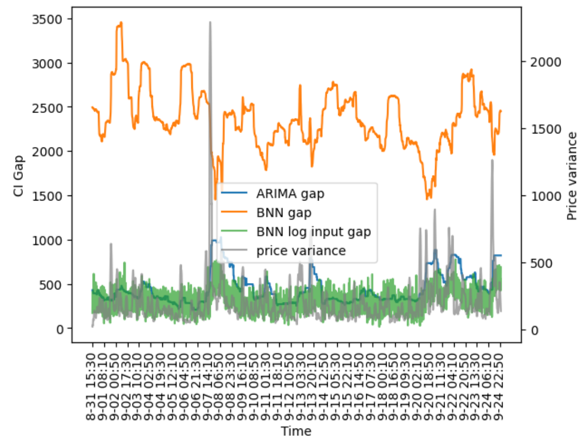

Figure 5. Plot shows confidence interval gap of ARIMA(0,1,0) and BNN model, along with standard deviation of price at each predicted timesteps

7. Acknowledgements

The authors would like to thank the CS159 instructors for useful lectures on related topics and would also like to thank Anastasios Angelopoulos for helpful discussions on conformal predictions.

References

Angelopoulos, A. N. and Bates, S. A gentle introduction to conformal prediction and distribution-free uncertainty quantification, 2022.

Chandra, R. and He, Y. Bayesian neural networks for stock price forecasting before and during covid-19 pandemic. *PLoS One*, 16(7):e0253217, 2021. doi: 10.1371/journal.pone.0253217.

Chernozhukov, V., Wüthrich, K., and Zhu, Y. An exact and robust conformal inference method for counterfactual and synthetic controls, 2021.

Gammerman, A., Vovk, V., Luo, Z., Smirnov, E., and Cherubin, G. (eds.). *Proceedings of the Ninth Symposium on Conformal and Probabilistic Prediction and Applications*, volume 128 of *Proceedings of Machine Learning Research*. PMLR.

Hyndman, R. J. and Athanasopoulos, G. *Forecasting: Principles and Practice*. OTexts, Melbourne, Australia, 3rd edition, 2018. ISBN 978-1-94392-216-0. URL https://otexts.com/fpp3/.

Kath, C. and Ziel, F. Conformal prediction interval estimation and applications to day-ahead and intraday power markets. *International Journal of Forecasting*, 37(2):777–799, apr 2021. doi: 10.1016/j.ijforecast.2020.

09.006. URL https://doi.org/10.1016%2Fj.ijforecast.2020.09.006.