STAT- S 650 TIME SERIES ANALYSIS FINAL PROJECT
Bitcoin Price Prediction: AN ARIMA Approach

Viswa Suhaas Penugonda

May 5, 2023

1 Abstract

The use of digital currencies, especially Bitcoin, has increased significantly in recent years. However, the volatile nature of Bitcoin's price has created a need for a prediction method that can help users make informed decisions about investing in Bitcoin. This project aims to develop a predictive method for Bitcoin prices by recognizing patterns in its time series data. To achieve this, data was collected from Bitstamp from January 1, 2012, to March 31, 2021. The data underwent a preprocessing stage that included removing attributes, conducting a stationary test, and differencing. The model candidates were then determined using the correlogram method. Two prediction methods were used, namely Autoregressive Integrated Moving Average (ARIMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA), both of which have proven to be effective in short-term predictions. The results of the study indicated that the ARIMA (1,1,0) model and SARIMA (1,0,0,1) model produced predictions with the smallest Akaike Information Criterion (AIC) score, indicating high accuracy. Therefore, these models are feasible for predicting Bitcoin prices in the short term. This project provides a promising approach for predicting Bitcoin prices, which can be useful for investors and traders in making informed decisions. However, it is important to note that Bitcoin's volatile nature makes long-term predictions challenging, and therefore, the results of this study should be interpreted with caution.

2 Introduction

ARIMA (Autoregressive Integrated Moving Average) and SARIMA (Seasonal Autoregressive Integrated Moving Average) are time series models used to forecast future values based on past observations.

ARIMA models are used for time series data that is not seasonal, while SARIMA models are used for data that exhibits seasonal patterns.

In ARIMA, "Autoregression" refers to the use of past values in the same series to predict future values, "Moving Average" refers to the use of past forecast errors to predict future values, and "Integrated" refers to the process of differencing the data to make it stationary (i.e. having a constant mean and variance) before modeling it.

SARIMA extends ARIMA by incorporating seasonality. In SARIMA, "Seasonal" refers to the use of past values from the same season in previous years to predict future values, in addition to the past values from the same series. The seasonality is captured by adding extra terms to the ARIMA model that correspond to the seasonal patterns.

ARIMA and SARIMA models are commonly used in fields such as finance, economics, and engineering for time series forecasting and trend analysis. They require the input of time series data and the specification of several parameters, including the order of autoregression, the order of moving average, and the degree of differencing.

3 Dataset Description

The dataset consists of data for the time period of Jan 2012 to March 2021, with minute-to-minute updates of OHLC (Open, High, Low, Close), Volume in BTC, and weighted bitcoin price. Timestamps are in Unix time. Timestamps without any trades or activity have their data fields filled with NaNs. Open, High, Low, Close (OHLC) are the opening, highest, lowest, and closing prices of the cryptocurrency within a particular time, such as a minute or a day. Volume in BTC refers to the total amount of Bitcoin that has been traded during a specific time. This metric provides an idea of the level of activity in the Bitcoin market, as well as the liquidity of the market. Weighted bitcoin price is a metric that calculates the average price of Bitcoin by considering the trading volume at different prices. This contrasts with a simple average, which treats all prices equally. The weighted average is calculated by multiplying each price by the volume at that price, adding up the results, and dividing by the total volume. This metric can provide a more accurate representation of the prevailing market price for Bitcoin.

4 Hypothesis

Null Hypothesis- There is no significant relationship between the weighted price of bitcoin with its future prices.
Alternate Hypothesis- There is a significant relationship between the weighted price of bitcoin with its future prices.

5 Methodology

5.1 Data Pre-processing

Due to the size of the dataset (contains 48,57,377 rows), I have resampled the time-series data of Bitcoin prices in different time frames. Firstly, the Timestamp column is converted to datetime format. Then, the resample() function is used to resample the data to different time frames. The 'D' argument is passed to the function to resample the data on a daily basis. The resulting dataframe contains the mean value of Bitcoin open, high, low, close prices, volume in BTC, volume in currency, and the weighted price for each day.

Similarly, the data is resampled for monthly, quarterly, and annual frequencies using the resample() function and passing arguments 'M', 'Q-DEC', and 'A-DEC', respectively. The resulting dataframes contain the mean values of Bitcoin prices for each month, quarter, and year.

Resampling is a useful technique to change the time granularity of the data, which can help in analyzing the data at different levels of granularity. By resampling the data to a lower granularity, we can identify long-term trends and patterns in the data. Conversely, resampling to a higher granularity can help in identifying short-term fluctuations and patterns in the data.

5.2 Time Series Plots

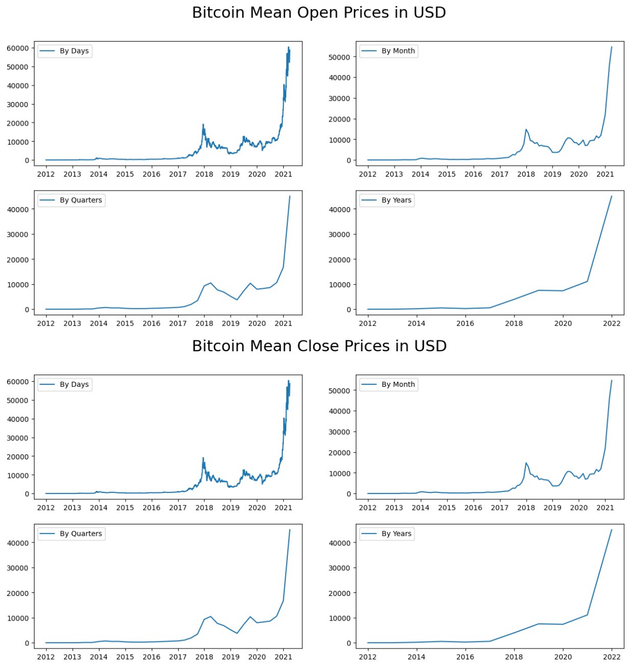


Figure 1: Time series plots of Open and Close Variables

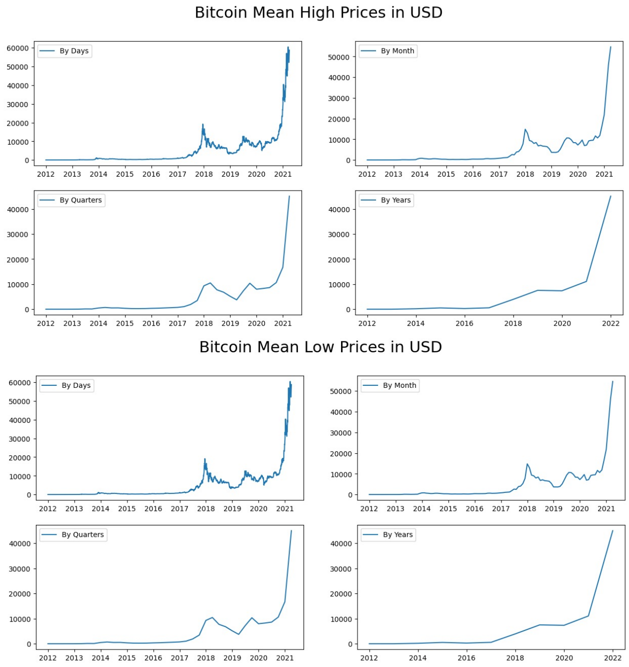


Figure 2: Time series plots of High and Low Variables

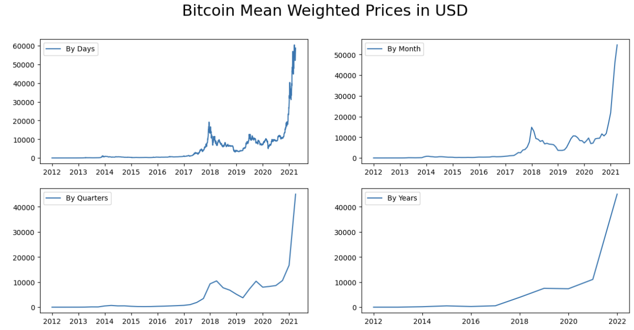


Figure 3: Time series plots of High and Low Variables

Looking at the plotted time series graph, each series consists of prices generalized by day, month, quarters and years.

5.3 Test for Stationality - Augmented Dickey-Fuller (ADF)

Stationary Time Series - The observations in a stationary time series are not dependent on the passage of time. If a time series lacks trend or seasonal impacts, it is said to be stationary.

Non-Stationary Time Series - tObservations from a non-stationary time series display seasonal effects, trends, and other time-dependent features.

Traditional time series analysis and forecasting techniques focus on finding and eliminating seasonal influences in order to make non-stationary time series data stationary.

Augmented Dickey-Fuller (ADF) - Strong assumptions are made about your data during statistical tests. They can only be used to determine whether or not a null hypothesis can be rejected. For a particular problem, the result must be understood in order to have any value. A unit root test is a sort of statistical test that includes the Augmented Dickey-Fuller test. A unit root test, according to the underlying theory, determines how strongly a time series is determined by a trend.

The time series is not stationary (has some time-dependent structure), and it can be represented by a unit root, according to the null hypothesis of the test. The time series is stationary, rejecting the null hypothesis, according to the alternative hypothesis.

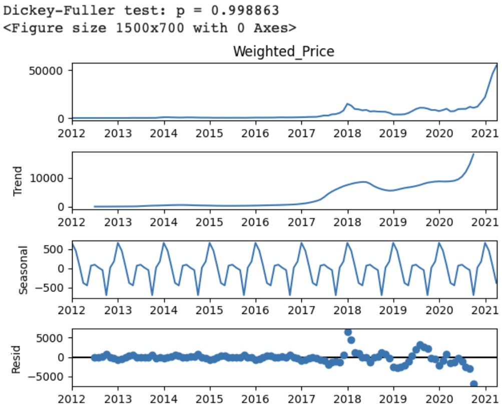
Null Hypothesis (H0): If it cannot be rejected, it indicates that the time series has a unit root, implying it is not non-stationary.

Alternate Hypothesis (H1): If the null hypotheses can be rejected; it indicated that the time series does not have a unit root, implying it is stationary.

If the **p-value greater than 0.05**, we fail to reject the null hypothesis (H0), Hence the data has a unit root and is non-stationary.

If the **p-value less than or equal to 0.05**, we reject the null hypothesis(H0), the Hence data does not have a unit root and is stationary.

Since we are taking the Weighted Price of Bitcoin as our target variable, let's check if the series is stationary or non-stationary.



```
Dickey-Fuller test: p = 0.998863
<Figure size 1500x700 with 0 Axes>
```

Figure 4: ADF Test for Weighted Price

The obtained p-value is greater than 0.05, Hence we can say that the series is non- stationary.

To convert the series to stationary, Differentiation is done.

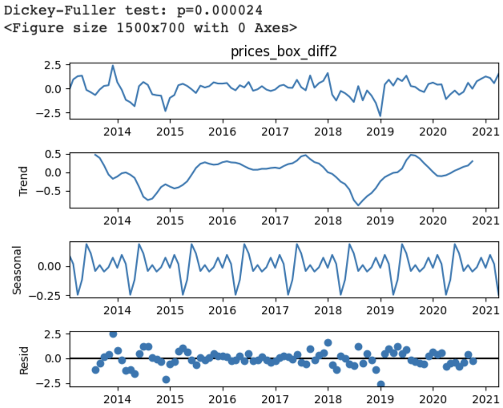
Differencing a time series involves computing the difference between consecutive observations. This removes the trend and seasonality in the data, leaving behind only the random fluctuations. By removing the trend and seasonality, the resulting series is more likely to be stationary.

Box-Cox Transformation: Box-Cox transformation is a method for transforming non-normal dependent variables into a normal shape. It involves applying a power transformation to the variable, where the value of the power (lambda) is determined by maximizing the likelihood function. This can be used to improve the accuracy of statistical models that rely on normally distributed data.

Seasonal Differentiation: Seasonal differentiation is a process of differencing a time series at a fixed time interval, typically equal to one period of the seasonal cycle. It is used to remove the seasonal component of a time series, making it stationary and more suitable for time series analysis and forecasting.

Regular Differentiation: Regular differentiation is a mathematical operation used to compute the derivative of a function. It involves calculating the rate at which the output of a function changes with respect to its input. The result of regular differentiation is a new function that describes the rate of change of the original function.

By doing,
Box-Cox Transformation, the p-value got reduced to **0.998863**
Seasonal Differentiation, the p-value got reduced to **0.444282**
Regular Differentiation, the p-value got reduced to **0.000024**



```
Dickey-Fuller test: p=0.000024
<Figure size 1500x700 with 0 Axes>
```

Figure 5: ADF Test for Weighted Price

5.4 Auto-Correlation and Partial Auto-Correlation

An ACF (Auto-Correlation Function) plot measures the correlation between the time series and its lags. It indicates the relationship between the current value of a series and its past values. The plot shows the correlation coefficient on the y-axis and the lag on the x-axis. It can help to identify the order of the MA (Moving Average) term in an ARIMA model.

A PACF (Partial Auto-Correlation Function) plot measures the correlation between the time series and its lags, after removing the effect of the intermediate lags. It indicates the direct relationship between the current value of a series and its past values. The plot shows the correlation coefficient on the y-axis and the lag on the x-axis. It can help to identify the order of the AR (Auto-Regressive) term in an ARIMA model. From the ACF plot of the differenced weighted price variable, we observe gradual decreases and increases in the graph like waves, which indicates the presence of seasonality in the time series data. The pattern of waves in the ACF plot corresponds to the repeating patterns of seasonal cycles. The gradual decrease in the autocorrelation coefficients indicates a decreasing influence of the seasonality, while the gradual increase indicates an increasing influence of the seasonality.

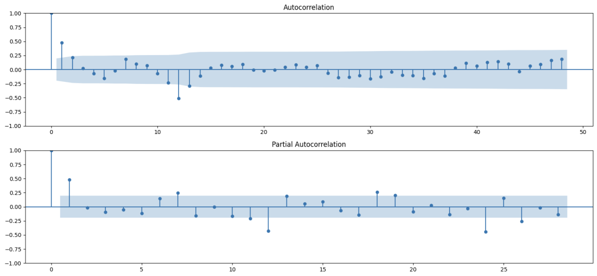


Figure 6: ACF for the differentiated Weighted Price

From the PACF plot of the differenced weighted price variable, it suggests that there may be a significant relationship between the current value and the value at lags beyond 10 and 20. This could indicate that an autoregressive (AR) model may be appropriate for modeling the data.

5.5 Spectral Analysis

The power spectral density (PSD) plot is used to identify the dominant cycles or frequencies present in a time series variable. It shows how the variance of the time series is distributed across different frequencies. By analyzing the PSD plot, we can identify the periodicity of the data, the presence of trends, and the cycles that exist within the data.

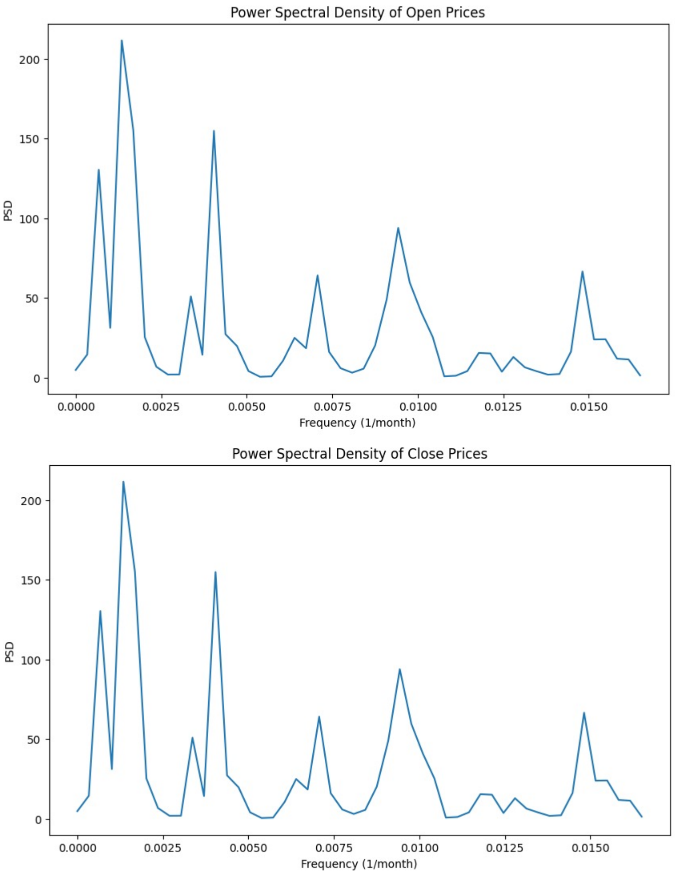


Figure 7: PSD plot for the differentiated Open Close Prices

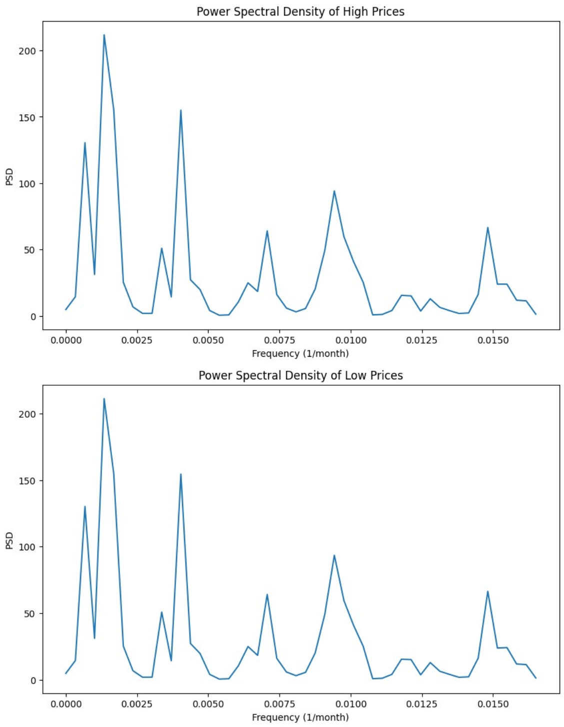


Figure 8: PSD plot for the differentiated High Low Prices

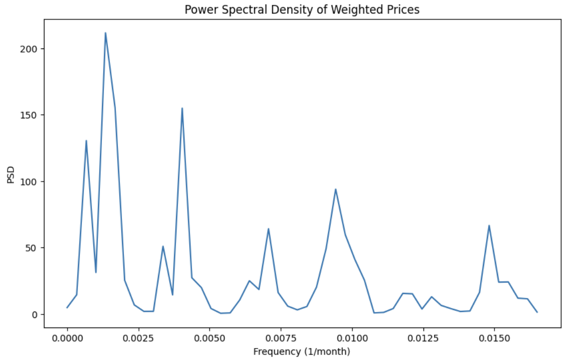


Figure 9: PSD plot for the differentiated Weighted Prices

Looking at the PSD plots all the variables, we can see there are some dominant peaks at frequencies 0.0015, 0.0020, 0.0035, 0.0070, 0.0090, 0.0150 indicating periodicity and cycles within the data.

5.6 Results and Discussion

ARMAX Model:

An ARMAX model, also known as Autoregressive Moving Average model with Explanatory Variables, is a time series model that is used to make predictions based on past data and explanatory variables. It is an extension of the ARMA model, which includes explanatory variables in addition to autoregressive and moving average terms.

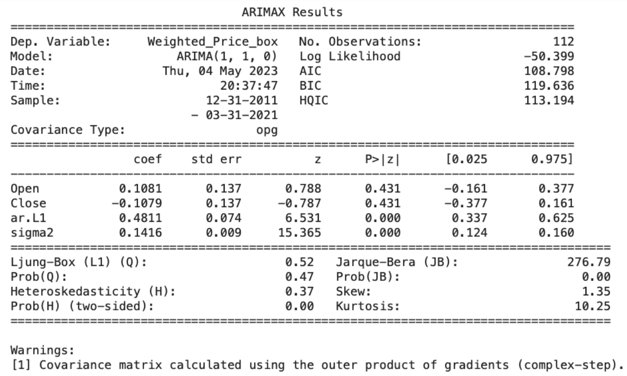
The ARMAX model consists of three components: an autoregressive (AR) component, a moving average (MA) component, and an exogenous (X) component. The autoregressive component uses past values of the dependent variable to predict future values, while the moving average component uses past errors to make predictions. The exogenous component incorporates other variables that may have an influence on the dependent variable, such as economic indicators or weather data.

To fit an ARMAX model, the coefficients of the autoregressive, moving average, and exogenous components are estimated using a maximum likelihood method. The model is then validated by comparing the predicted values to the actual values in the data set. The model can be refined by adjusting the coefficients and including or excluding variables as necessary.

The ARMAX model has several advantages over other time series models. First, it can incorporate external factors that may affect the dependent variable, making it more flexible and able to capture complex relationships. Second, it can handle non-stationary data, which is common in economic time series. Finally, the model can be used to forecast future values, allowing for more informed decision-making.

However, the ARMAX model also has some limitations. It requires a large amount of data to accurately estimate the coefficients, and it may be difficult to identify the correct lag structure for the autoregressive and moving average components. Additionally, the model assumes that the error terms are normally distributed, which may not always be the case in practice.

In summary, the ARMAX model is a powerful tool for analyzing and predicting time series data with explanatory variables. It can be used in a variety of applications, from economic forecasting to weather prediction, and provides a flexible and accurate method for modeling complex relationships.



```
                               ARIMAX Results
==============================================================================
Dep. Variable:     Weighted_Price_box   No. Observations:                  112
Model:                 ARIMA(1, 1, 0)   Log Likelihood                 -50.399
Date:                Thu, 04 May 2023   AIC                            108.798
Time:                        20:37:47   BIC                            119.636
Sample:                    12-31-2011   HQIC                           113.194
                         - 03-31-2021
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Open           0.1081      0.137      0.788      0.431      -0.161       0.377
Close         -0.1079      0.137     -0.787      0.431      -0.377       0.161
ar.L1          0.4811      0.074      6.531      0.000       0.337       0.625
sigma2         0.1416      0.009     15.365      0.000       0.124       0.160
===================================================================================
Ljung-Box (L1) (Q):                   0.52   Jarque-Bera (JB):               276.79
Prob(Q):                              0.47   Prob(JB):                         0.00
Heteroskedasticity (H):               0.37   Skew:                             1.35
Prob(H) (two-sided):                  0.00   Kurtosis:                        10.25
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

Figure 10: Summary of results for ARMAX model

The ARMAX model is a type of time series model that includes both autoregressive (AR) and moving average (MA) components along with exogenous variables. In this particular model, the dependent variable is the weighted price of Bitcoin, and the exogenous variables are the Open and Close prices.

The results of the model show that the ARIMA(1, 1, 0) model is a good fit for the data, as indicated by the log likelihood (-50,399), AIC (108.798) and BIC (119.636) values. The coefficient for the AR(1) term is 0.4811, which indicates that the model includes a significant autoregressive component. This means that past values of the dependent variable have a significant impact on the current value.

The coefficients for the Open and Close prices are not significant, as indicated by the p-values being greater than 0.05. This suggests that these exogenous variables do not have a significant impact on the dependent variable.

The sigma2 value of 0.1416 indicates that there is some variance in the residuals of the model, but this is to be expected in any model. The Ljung-Box (L1) and Jarque-Bera (JB) tests are used to test for residual autocorrelation and normality of the residuals, respectively. The p-values for both tests are greater than 0.05, indicating that the residuals are independent and normally distributed.

The Heteroskedasticity (H) test is used to test for heteroskedasticity in the residuals. The p-value for this test is greater than 0.05, indicating that there is no evidence of heteroskedasticity in the residuals.

Overall, ARAMAX model with Weighted Prices as the target variable and Open, Close as the exogenous variable makes a good fit to the data with (1,0,1) as the noise.

Residual Analysis of the best ARMAX Model

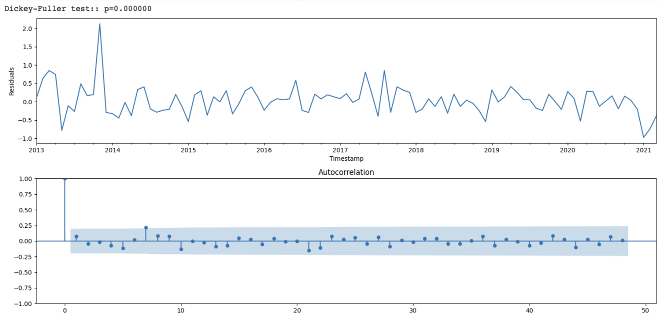


Figure 11: Residual ACF plots for best ARMAX model

Looking at the residual plot, we cannot evaluate the goodness of fit of the model based on this plot. We can see there's a trend in the residual plot, which is a bad indicator. However, we can look at other diagnostic statistics such as the Ljung-Box test for autocorrelation of residuals and the Jarque-Bera test for normality of residuals to ensure that the residuals are uncorrelated and normally distributed.

Looking at the ACF of the Residuals, we can however see that are no correlations and the residuals are indeed white noise.

Predictions of the Best Fit ARMAX Model

First, an inverse Box-Cox transformation function is defined to convert the log-transformed data back to its original scale. Then, a list of future dates is created, and a DataFrame 'future' is constructed with these dates, which will be used to store the forecasted values. The ARMAX model is then fitted to the data using the best model object, and the 'predict' function is used to generate predictions for the future dates. Finally, the predicted values are converted back to the original scale using the inverse Box-Cox transformation, and the results are plotted against the original data.

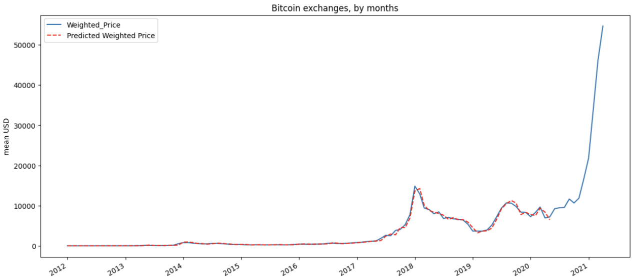


Figure 12: Predictions for best ARMAX model

The plot shows that the predicted Weighted Price values are close to the actual values, indicating that the ARMAX model is performing well in forecasting the Bitcoin Weighted Price. Although, the model is good at making predictions, the model failed to predict the values to the end.

The model also failed to address the trends as seen in the residual plot (Figure. 11).

Hence, I've decided to perform Seasonal ARIMA to better deal with the seasonality in the data to get a better fit in making better predictions in future bitcoin prices.

SARIMA Model:

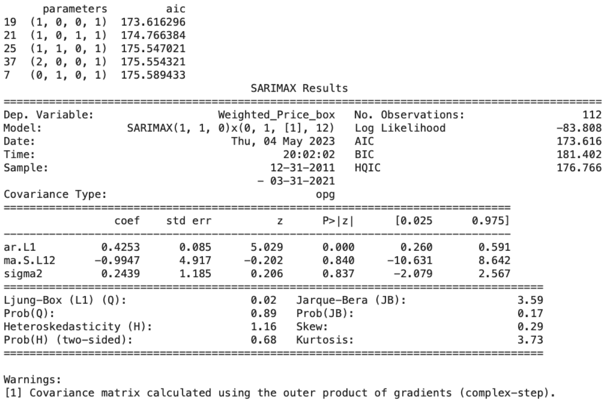
SARIMA (Seasonal Autoregressive Integrated Moving Average) is a time series forecasting model that extends the ARIMA (Autoregressive Integrated Moving Average) model to include seasonal components. This model is particularly useful when dealing with time series data that exhibit seasonal patterns, such as sales data that typically experience higher sales during certain months of the year.

The SARIMA model involves the selection of three parameters: p, d, and q, which correspond to the autoregressive, integrated, and moving average components of the model, respectively. Additionally, there are three more parameters: P, D, and Q, which correspond to the seasonal autoregressive, seasonal integrated, and seasonal moving average components, respectively. The model also requires a specification of the seasonal period, which is the number of time periods in a complete seasonal cycle.

Like the ARIMA model, SARIMA relies on the identification of the order of differencing required to make the time series stationary. The seasonal order of differencing is determined by applying the seasonal difference operator to the series.

SARIMA models can be fitted using various techniques, including maximum likelihood estimation, which involves selecting the model parameters that maximize the likelihood of observing the data given the model. The model can then be used to make forecasts by iteratively generating predicted values for each time step using the estimated model parameters.

Overall, the SARIMA model provides a powerful tool for time series forecasting, particularly for data with seasonal patterns. With careful selection of the model parameters, it can be used to generate accurate predictions that can inform decision making in various fields such as finance, economics, and marketing.



```
     parameters          aic
19  (1, 0, 0, 1)  173.616296
21  (1, 0, 1, 1)  174.766384
25  (1, 1, 0, 1)  175.547021
37  (2, 0, 0, 1)  175.554321
7   (0, 1, 0, 1)  175.589433
                                     SARIMAX Results
==============================================================================================
Dep. Variable:                  Weighted_Price_box   No. Observations:                     112
Model:             SARIMAX(1, 1, 0)x(0, 1, [1], 12)   Log Likelihood                     -83.808
Date:                             Thu, 04 May 2023   AIC                                173.616
Time:                                     20:02:02   BIC                                181.402
Sample:                                 12-31-2011   HQIC                               176.766
                                      - 03-31-2021
Covariance Type:                               opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.4253      0.085      5.029      0.000       0.260       0.591
ma.S.L12      -0.9947      4.917     -0.202      0.840     -10.631       8.642
sigma2         0.2439      1.185      0.206      0.837      -2.079       2.567
===================================================================================
Ljung-Box (L1) (Q):                   0.02   Jarque-Bera (JB):                 3.59
Prob(Q):                              0.89   Prob(JB):                         0.17
Heteroskedasticity (H):               1.16   Skew:                             0.29
Prob(H) (two-sided):                  0.68   Kurtosis:                         3.73
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

Figure 13: SARIMAX SUMMARY model

The output above is the summary of a SARIMAX model, a seasonal ARIMA model with exogenous variables. The model was built to forecast the weighted price of Bitcoin based on historical data. The model has an order of (1, 1, 0)x(0, 1, [1], 12), meaning it includes one autoregressive term, one differencing term, and no moving average term for the non-seasonal component, while it includes a seasonal order of 0 for autoregressive and moving average terms, and a seasonal differencing term of 1 with a seasonal period of 12. The model was fitted to 112 observations and has an AIC value of 173.616, indicating that it is the best model among the models considered in the analysis.

The model coefficients table shows that the autoregressive term is statistically significant, with a coefficient of 0.4253 and a p-value of 0.000, while the moving average term is not significant, with a coefficient of -0.9947 and a p-value of 0.840. The sigma2 value of 0.2439 represents the variance of the error term, which is relatively low, indicating that the model is a good fit for the data.

The Ljung-Box test result shows a Q-statistic of 0.02 and a p-value of 0.89, indicating that there is no evidence of autocorrelation in the residuals. The Jarque-Bera test result shows a JB statistic of 3.59 and a p-value of 0.17, indicating that the residuals are normally distributed. Finally, the Heteroskedasticity test result shows a p-value of 0.68, indicating that there is no evidence of heteroskedasticity in the residuals.

In summary, this SARIMAX(1,0,0,1) model with exogenous variables is a good fit for the Bitcoin weighted price data and can be used for forecasting purposes. However, it is important to note that the model is only as good as the data it is trained on, and future changes in the data generation process may require re-fitting or adjusting the model.

Residual Analysis of the best SARIMAX Model

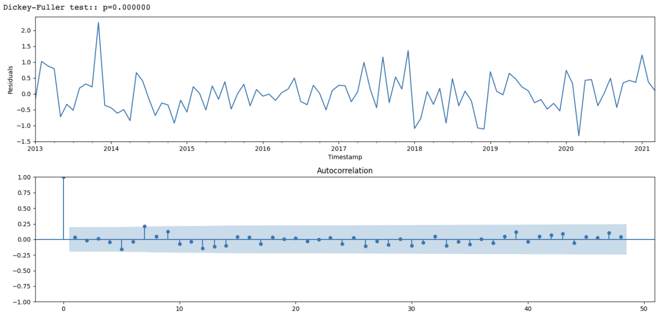


Figure 14: Residual ACF plots for best SARIMAX model

Looking at the residual plot, There is no observable trend or cycles in the residuals of the data, which is a better indication that the model is a good fit for the data.

Looking at the ACF of the Residuals, we can see that are no correlations and the residuals are indeed white noise.

Predictions of the Best Fit SARIMAX Model

Firstly, an inverse Box-Cox transformation function is used to predict Bitcoin prices in USD for future dates. The Box-Cox transformation is a statistical technique used to transform non-normal data into a normal distribution. This transformation can be useful in improving the accuracy of predictive models, as models typically assume normality of the data.

The inverse Box-Cox transformation function, which takes as input a vector of data and a lambda parameter. The lambda parameter is used to determine the type of transformation applied to the data. If the lambda parameter is zero, the function returns the exponential of the input data. Otherwise, the function applies a more complex transformation involving the natural logarithm and the lambda parameter.

A new DataFrame is created, that contains a subset of the original Bitcoin price data and a set of future dates for which to make predictions. The best model object is used to generate predictions for the future dates, and the inverse Box-Cox transformation function is applied to convert the predictions back to their original scale.

The resulting predictions are plotted alongside the actual Bitcoin prices in USD using the matplotlib library. The plot shows a line graph of the actual Bitcoin prices in blue and the predicted prices in red. The graph provides a visual representation of the accuracy of the predictive model. If the predicted prices closely follow the actual prices, the model is considered to be accurate. On the other hand, if the predicted prices diverge significantly from the actual prices, the model may need to be refined or improved.

In summary, the Box-Cox transformation can be used in conjunction with predictive models to generate accurate predictions of Bitcoin prices in USD. The resulting visualizations can be used to evaluate the accuracy of the predictive model and make improvements as needed.
The plot shows that the predicted Weighted Price values are close to the actual values, indicating that the SARIMAX model is performing well in forecasting the Bitcoin Weighted Price.

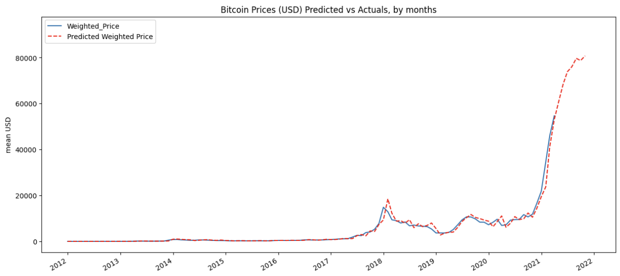


Figure 15: Predictions for best SARIMAX model

5.7 Conclusion

The ARIMA and SARIMA models were used to predict weighted price in this project. Although the best model for ARIMA, ARIMA(1,1,0) gave a better fit to predict the future prices of bitcoin, it failed to take the seasonality present in the data into account which resulted in a slightly worse trend in predictions. However the SARIMAX(1,0,0,1) model performed the best in predicting the future prices of Bitcoin. Which brings us to our initial hypothesis, that we reject the null hypothesis (since p value is less than zero) which states that there is no significant relationship between the weighted prices and the future prices of bitcoin and say that there is indeed a relationship between the weighted price and the future prices of bitcoin.

5.8 References

1. ”Bitcoin price forecasting with ARIMA, LSTM, and Prophet models” by N. M. Khandaker, et al. (2020). This paper compares the performance of ARIMA, LSTM, and Prophet models for bitcoin price prediction and provides a detailed analysis of the results.

2. ”Bitcoin Price Prediction Using ARIMA Model” by S. Gupta and R. Singh (2021). This paper explores the application of ARIMA models for bitcoin price prediction and compares the results with other machine learning techniques. 3. ”Bitcoin price forecasting using ARIMA and SARIMA models” by S. A. Almutairi and S. A. Al-Sharif (2021).

This paper investigates the effectiveness of ARIMA and SARIMA models for bitcoin price prediction and proposes a hybrid model that combines both techniques.

4. ”Bitcoin Price Prediction with ARIMA and GARCH models” by A. Rahman and M. Uddin (2021). This paper applies ARIMA and GARCH models to bitcoin price prediction and compares the results with other machine learning algorithms.

5. Yuan, X., Wu, Z., Ji, Q., Peng, Y. (2021). Bitcoin price forecasting based on a SARIMA-GARCH model

6. Shahbaz, M., Chaudhary, A. R., Sharif, A., Balcilar, M. (2021). Bitcoin price forecasting with high-frequency data using SARIMA-ARCH models.

5.9 Appendices

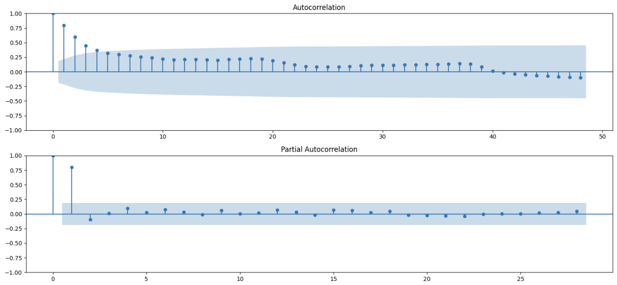
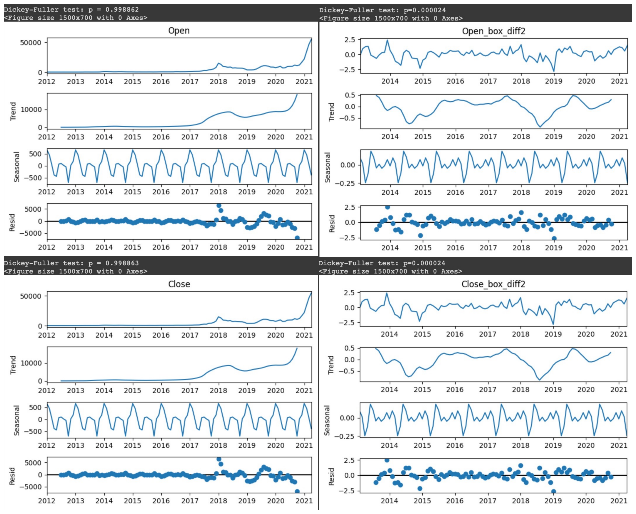


Figure 16: ACF plot Un-Differenced Weighted Price



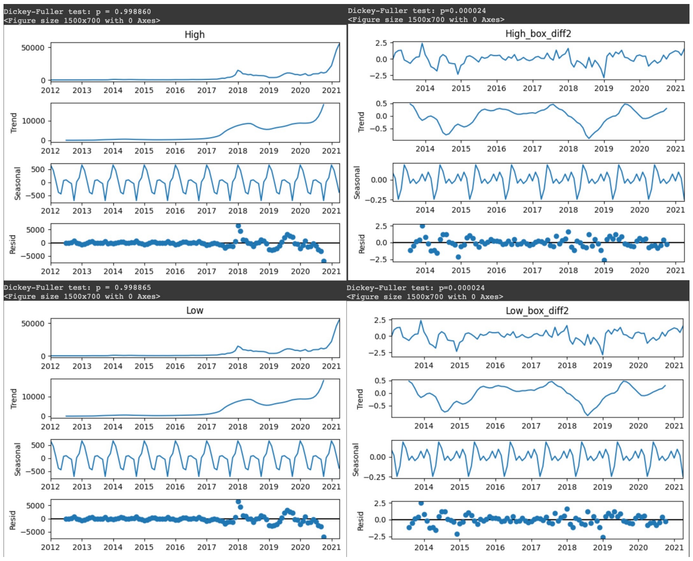
```
Dickey-Fuller test: p = 0.998862
<Figure size 1500x700 with 0 Axes>
```

```
Dickey-Fuller test: p=0.000024
<Figure size 1500x700 with 0 Axes>
```

```
Dickey-Fuller test: p = 0.998863
<Figure size 1500x700 with 0 Axes>
```

```
Dickey-Fuller test: p=0.000024
<Figure size 1500x700 with 0 Axes>
```

Figure 17: ADF Test for Open and Close Prices



```
Dickey-Fuller test: p = 0.998860
<Figure size 1500x700 with 0 Axes>
```

```
Dickey-Fuller test: p=0.000024
<Figure size 1500x700 with 0 Axes>
```

```
Dickey-Fuller test: p = 0.998865
<Figure size 1500x700 with 0 Axes>
```

```
Dickey-Fuller test: p=0.000024
<Figure size 1500x700 with 0 Axes>
```

Figure 18: ADF Test for High and Low Prices