**D214 NKM2 Task 2: Data Analytics Report and Executive Summary**

**Chase Seballos (010091804)**

**Western Governors University**



**Table of Contents**

A. Research Question……….………………….………………….………..…….....……………….…… Pg. 3

A1. Summarize the Original Real-Data Research Question……………..……...….……..…....…..…..… Pg. 3

B. Data Collection……………………………………..……………....……….…….……….………..…… Pg. 3

B1. Report on Data-Collection Process…………………………………………………………..……….. Pg. 3

C. Data Extraction and Preparation………….……...…….………………….……..……….………....…… Pg. 4

C1. Data-Extraction and Preparation Process…………………….….……......…............…..….....… Pg. 4 – 10

D. Analysis……………………………………………………………………………..………..……….. Pg. 10

D1. Analysis Techniques used....……………….…..………………………………....…………… Pg. 10 – 34

E. Data Summary and Implications………...…………………………………………….….…….........…. Pg. 34

E1. Summary of Implications of Data Analysis and Limitation…………………………….….………. Pg. 34

F. Web Sources……………………………………………………………………………………….. Pg. 34 - 35

G. In-text Citations and References…………………………………………………………….…………. Pg. 35

**Part I: Research Question**

A.  Summarize the original real-data research question you identified in task 1. Your summary should include justification for the research question you identified in task 1, a description of the context in which the research question exists, and a discussion of your hypothesis.

Referring back to Task 1, the research question is: “Can a time series model using the ARIMA method accurately predict future stock price trends for Walmart with greater than 70% accuracy, based on 50 years of daily historical stock price data?”

To address this question, we must first establish the null and alternative hypotheses. The null hypothesis proposes that the ARIMA model cannot predict stock price trends with 70% or greater accuracy. Conversely, the alternative hypothesis suggests that the model can achieve this level of accuracy.

Given the historical nature of the dataset, time series analysis is an appropriate method for this study. The extensive volume of data allows us to forecast future stock prices to a certain degree of accuracy. The primary focus is to determine whether the ARIMA model can achieve a prediction accuracy of 70% or higher.

**Part II: Data Collection**

B.  Report on your data-collection process by describing the relevant data you collected, discussing one advantage and one disadvantage of the data-gathering methodology you used, and discussing how you overcame any challenges you encountered during the process of collecting your data.

I selected a dataset from Kaggle that contains Walmart’s daily stock prices, comprising of seven columns:

* Date – The date of the stock price
* Open – The opening day’s starting price.
* High – The price of the stock at its highest point of the day.
* Low – The price of the stock at its lowest point of the day.
* Close – The ending price of the stock at the time of the closing of the day.
* Adj Close – The adjusted close price modifies a stock’s closing price to account for corporate actions, providing a more accurate reflection of the stock's value. It is commonly used for analyzing historical returns and past performance in detail.
* Volume – Volume represents the total number of shares traded in a particular stock, index, or other investment over the course of the day.

The dataset spans 50 years, with the number of observations matching the number of days in those years. The 50 years range from as far back as year 1972, all the way up to 2022. As a result, the file totals to over twelve thousand rows of data.

One significant advantage of this dataset is its completeness and simplicity. Upon review, it provides all necessary observations for the analysis without requiring additional datasets from external sources. It is free of NA values and duplicates, and it is conveniently available in a .csv format, making it easy to import into the Jupyter Notebook environment.

However, a notable disadvantage is the lack of verified reliability, credibility, and factual integrity, as it is a public dataset from Kaggle. In real-world scenarios, this could undermine the analysis, making results potentially meaningless and unreliable. Nonetheless, for this project and course, the dataset is suitable for practicing and applying various analysis tools.

**Part III: Data Extraction and Preparation**

C.  Describe your data-extraction and -preparation process and provide screenshots to illustrate each step. Explain the tools and techniques you used for data extraction and data preparation, including how these tools and techniques were used on the data. Justify why you used these particular tools and techniques, including one advantage and one disadvantage when they are used with your data-extraction and -preparation methods.

Since this is submitted in Jupyter Notebook format, my data extraction and preparation processes are fully provided here. The code is commented for clarity, with markdown narratives as needed to detail the data preparation process. I've also included exploratory data analysis for completeness and to assist in the final analysis approach, though it was not required.

The rubric requires an explanation of tools and techniques for data extraction and preparation, including one advantage and disadvantage for each used:

Data Extraction and Data Preparation Process

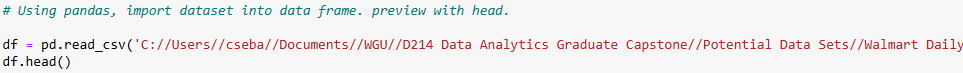
* Jupyter Notebook (Anaconda environment):
  + - Advantage: The lines of code and output make for an easy-to-read experience. Also, the environment is very efficient and detailed in explaining errors in the code and how to correct it.
    - Disadvantage: Can be slow, affecting workflow efficiency.
      * The below snippet includes the initial libraries and packages to start the EDA, and cleaning process.

A computer screen shot of a computer screen

Description automatically generated

Python Libraries:

* Pandas – For loading the data set and dealing with missing values if the case they are present in the data. Also, for the use of transforming the data to the right formats as well as reducing the data frame to keeping only the relevant data for the analysis.
  + - Advantage: One of the most common libraries used in Python to begin cleaning, transforming, and preparing the data to be used for analysis.
    - Disadvantage: It won’t be considered a disadvantage or issue for this project, but it is said that with more complex situations, pandas has limitations in manipulating the data where other libraries and packages may be more suitable for the challenge.
      * This was valuable for the need to load the data set into the Jupyter Notebook environment.



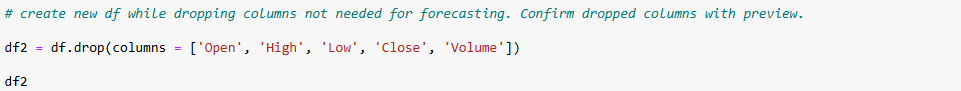
* + - * Use “.info()” code in order to confirm the number of columns, observations, data types, and to determine if any NA’s or nulls exist in the data set.



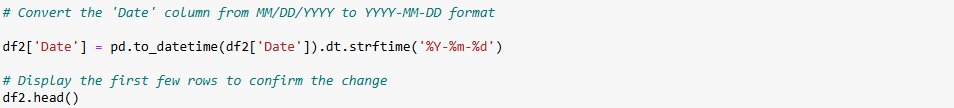
* + - * Run “set\_option” function from pandas to confirm the number of observations or days in the data set.



* + - * For the analysis steps of time series, it is determined that only 2 of the 7 columns will be needed. As a result, I create a new data frame, and with this new data frame, performed the “.drop” function on the irrelevant columns. From here, I previewed the applied changes to confirm it was successfully executed.



* + - * Use “to\_datetime” function in order to convert the format of the date column of data. Then, confirm the changes be previewing.



* + - * Execute “.shape” to confirm the number of rows and columns remaining in the data set. This is just another step for reassurance of the changes applied when dropping the other columns.



* + - * Just to verify that no nulls or NA’s exist in the dataset, perform the “.dropna()” function.



* + - * The next step, we ensure that the “Date” column is appropriately formatted as a datetime type. Also, we run a .dt.tz\_localize code to remove any type of timezone information from the data as it won’t be necessary or relevant for the time series analysis.

A close-up of a white background

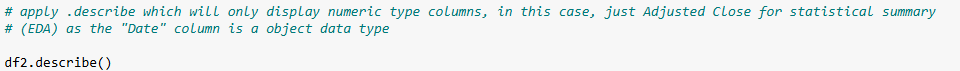
Description automatically generated

* MatPlotLib - The use of matplotlib was to plot the data to give a quicker understanding of patterns and how the data is distributed. This was used throughout the script in order to prep the data for the time series analysis. Additionally, this is a huge help for the exploratory data analysis (EDA) process.
  + - Advantage: Easy to create visualizations.
    - Disadvantage: Despite the simplicity and data projected, the visualizations are rather basic and lack visual appeal.
      * Plot the second half of adjusted closed stock prices for the 25 years since attempting to display the full 50 years of data may be too crowded for the graph. Also, include a trend line in dotted red to help understand the trend of the 25 years.

A screenshot of a computer program

Description automatically generated

* + - * Use “.describe” to review the statistical summary of the adjusted close stock price and perform exploratory data analysis or EDA. This would only apply to the Adj Close column since the date column only consists of dates.



* + - * Now that the time series data set is stationary, move on to plotting the differenced data.

A close-up of a computer screen

Description automatically generated

* + - * The data has proven to center around zero. The next step is to determine the PACF and ACF. This can be possible by importing the “plot\_acf” and “plot\_pacf” from the statsmodels package. This package allows me to plot the Autocorrelation and Partial Autocorrelation.

A screenshot of a computer code

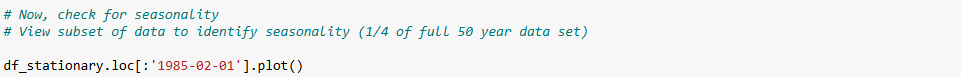
Description automatically generated

* + - * With the PACF and ACF plotted, the p,d, and q are determined (1,1,1). Next, check spectral density by importing the signal package from the scipy library.

A computer screen shot of a math problem

Description automatically generated

* + - * Then, check for seasonality on the Adj Closed Differenced data.



* + - * Next, check for more seasonality using the rcParams function.

A computer code on a white background

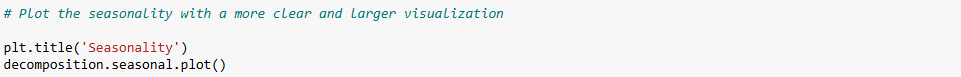
Description automatically generated

* + - * The next step, apply decomposition by importing the seasonal\_decompose package from the statsmodels library. This can be done by plotting the decomposition visualization.

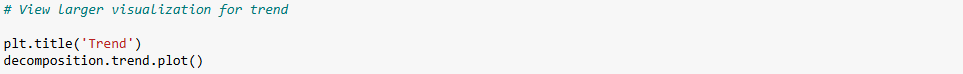
A screen shot of a computer

Description automatically generated

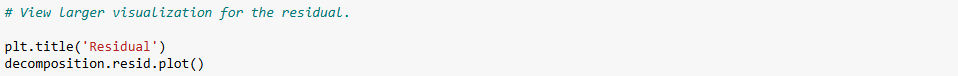
* + - * Following this, give a better visual of the seasonality of the decomposition.



* + - * As support, plot the trend of the decomposition.



* + - * With more support, plot the residual.



* Statsmodel – Needed for the use of Augmented Dickey-Fuller (ADF or adfuller) test to verify stationarity as this is a required step in order to use ARIMA.
  + - Advantage: Makes sure that the data is suitable and fit for the use of a time series analysis.
    - Disadvantage: The results for the test can be inaccurate if it consists of seasonality and trends. However, this disadvantage can be resolved upon checking for it.
      * Perform the Augmented Dickey-Fuller test or ADF for short. This step is essential in order to verify stationarity within the data set.

A white background with blue text

Description automatically generated

* + - * Apply and if and else statement that will easily determine whether the data is stationary by checking if the p-value is less than or equal to 0.05.

A white background with blue and red text

Description automatically generated

* + - * Because the data set failed to reject the null hypothesis, confirming the data is non-stationary, applied differencing on the data set in order to make the data stationary. Also, just in case NA’s are generated, apply “.dropna()” on the code lines to ensure no NA’s exist.

A computer code on a white background

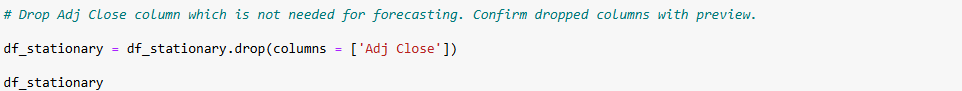
Description automatically generated

* + - * After previewing, the new column of “Adj Close Differenced” has generated NA’s, so run a “.dropna” function to make them go away.

A close-up of a computer screen

Description automatically generated

* + - * Now that “Adj Close Differenced” column exists, we can drop the “Adj Close” column as it won’t be needed for the forecasting.



* + - * Re-perform the Augmented Dickey-Fuller (ADF) test on the data set to confirm if the time series is stationary with a p-value of less than or equal to 0.05. Also, include an if and else statement to make the results apparent.

A screen shot of a computer code

Description automatically generated

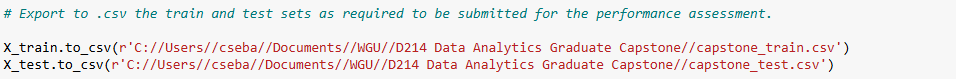
Aside from the libraries and packages mentioned above, the below steps were necessary to finalize the cleaning, preparation, transformation, and EDA steps performed to get the data where it needed to be for the time series modeling steps using ARIMA.

With the data stationary, the PACF and ACF observed, and the seasonality tested with decomposition. Move on to creating a train and test split on the original stock data. From here, print the shape of the split data sets and compare it to the full data set to verify that the appropriate ratio was applied.

A white background with black text

Description automatically generated

Lastly, extract the train and test sets from the Jupyter Notebook environment. The produced .csv files of the prepared data will be included with the submission of the assessment. The process of cleaning, EDA, preparation and transforming of the data is complete. The next steps will be to conduct a time series analysis with the support of ARIMA.



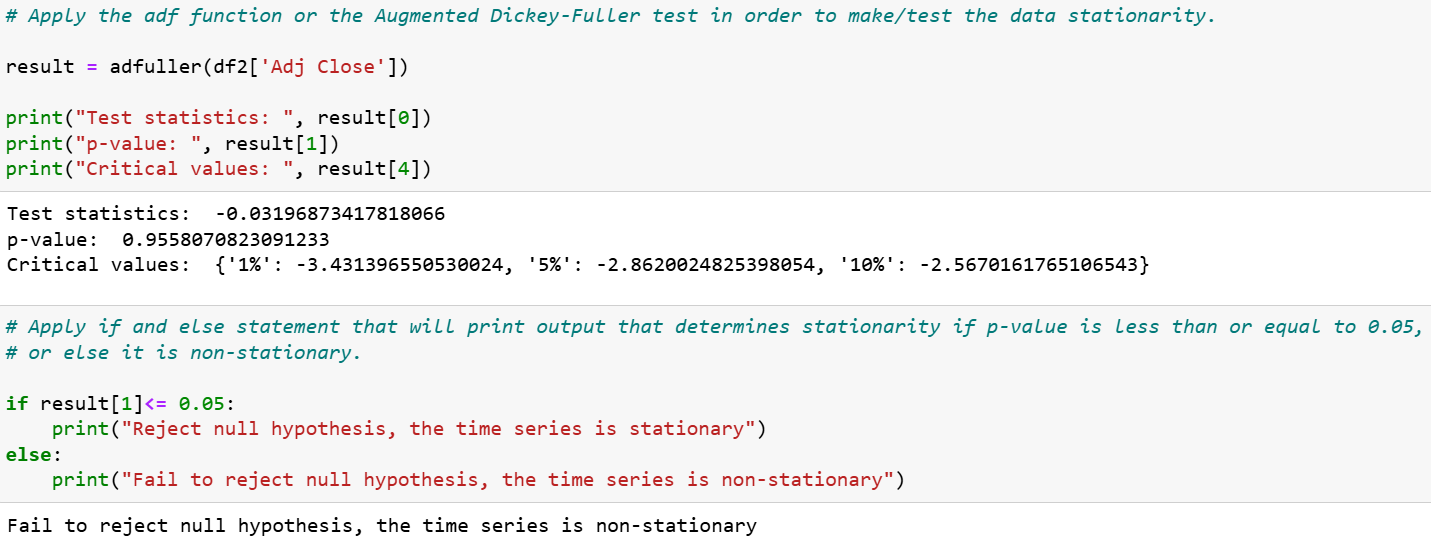
To point out, the preparation steps were tedious and a long process which could’ve been much longer had the data required more and more data from other outside sources. This can be considered a disadvantage for this method of analysis, but ultimately, worth the time and effort as the data is now in the right format and structure to run a time series analysis.

**Part IV: Analysis**

D.  Report on your data-analysis process by describing the analysis technique(s) you used to appropriately analyze the data. Include the calculations you performed and their outputs. Justify how you selected the analysis technique(s) you used, including one advantage and one disadvantage of these techniques(s).

After completing the data-extraction and data-preparation process, we move on to the analysis. The primary analysis technique for the time series model that was utilized was the ARIMA (AutoRegressive Integrated Moving Average) model. This method is well-suited for time series forecasting and involves three main components: autoregression (AR), differencing (I), and moving average (MA). Upon testing for these components, the parameters that are landed on correspond to p, d, and q values.

For the differencing (I), this was determined in the previous section when we executed the Augmented Dickey-Fuller (ADF) test that reveals whether the data is already stationary. On the first test, without applying differencing, the results revealed that the data was not stationary. As a result, we apply differencing to the data, then re-test the data with ADF to see if the application of differencing made the data stationary. In this case, it was successfully stationary, which meant that the value of I or d is 1, because the value of d is decided based on the number of times it took to make the data stationary. Technically, the step of differencing is a step in both transforming the data (pre-processing) as well as the analysis process since this value is needed to move forward with ARIMA. The code was included in the previous step, but the code as well as the output will be shown below with helpful comments and captions documenting the purpose of each line of code for testing stationarity.



A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

After completing the ADF test and confirming the stationarity and differencing of the data with the support of a visualization, the next step is to identify the two components of AR (p) and MA (q). “The ACF and PACF plots provide insights into selecting the appropriate p and q values for an ARIMA model, where significant spikes and drops help in identifying these parameters (Hyndman & Athanasopoulos, 2018).” From the library of scipy, lies a handy package, statsmodels, that consists of the plot\_acf and plot\_pacf features that will be utilized to plot the pacf and acf with lags of 20. Based on the plotted pacf and acf, both visualizations show a spike at lag 1, which leads me to believe that value of both p and q is 1. As a result, (p,d,q) = (1, 1, 1).

A screenshot of a computer code

Description automatically generated

A graph with blue dots

Description automatically generated

With p, d, and q determined, import the signal package from scipy, to test for spectral density. This process is basically another way to confirm stationarity and to check for any potential periodic patterns, which appear to be non-existent based on the code and output below.

A screen shot of a computer

Description automatically generated

Next, check for seasonality on a sample size of the differenced adjusted closed stock price, which indicates linearity.

A screenshot of a graph

Description automatically generated

For further confirmation, plot the autocorrelation and lags for seasonality, which again, shows linearity.

A screen shot of a graph

Description automatically generated

Next, apply decomposition, which is a helpful tool to make all of the complex statistical measures in a time series model into more interpretable components. With the following code, the differenced data is plotted for trend, seasonality, and residuals.

Trend – the trend represents the long-term direction of the data (stock price) over time. Based on the visual, the stock price is gradually increasing over the years.

Seasonality – the seasonality visual represents the seasonal increases or fluctuations based on certain events times that indicate a repetitive pattern. As an example, if stock prices were to decline in the summer months of June through August. However, in the case of the plotted visual, there doesn’t appear to be a strong seasonal pattern present in the data.

Residual – the residual component represents the behavior of the data when removing any trend or seasonality from the index time series. For example, random drops or increase or abnormalities in the data. The visual demonstrates the behavior of the data fluctuating around zero, which implies its random with no other components factoring or influencing the data.

A screenshot of a computer

Description automatically generated

The next few lines of code and outputs just recreates the same visuals on a larger scale for a clearer and larger visualization of the same results, which would be better to showcase in a presentation.

A screen shot of a computer

Description automatically generated

A graph on a screen

Description automatically generated

A screenshot of a computer

Description automatically generated

Now that the p, d, and q are determined, and the visual tests have been completed. The next step was to create a train and set split of the data for further analysis. For the split performed, I used an 80/20 ratio, then confirmed the ratio by displaying the shape which gives the number of rows and columns for confirmation.

A screen shot of a computer

Description automatically generated

Moving on to the exciting part, running ARIMA. The ARIMA feature was imported into Jupyter Notebook from the scipy library. Using the determined p, d, and q of 1, 1, and 1, I ran the test and printed the summary. There are a lot of statistical metrics listed regarding the fitted model, but one of the statistics that I wanted to pay more attention to the Akaike Information Criterion (AIC) of 9378.011, as this value should be as low as possible when comparing it to other model tests, so this number has been recorded and put off to the side to reference later.

Additionally, paying extra attention to the p-value of the AR and MA being 0.00, the statistical figures indicate that the coefficients are statistically significant for the fitted model.

A screenshot of a computer

Description automatically generated

Even though the SARIMAX Results demonstrated a model with a good fit, I ran another ARIMA model using a different set of p, d, and q’s (1, 2, 1) as a comparison to the results in the first test. Looking at the results, even though the p values are zero or close to zero, the AIC is 9409.879, which is higher than the first test, which is enough proof that the initial test had a higher statistical significance and better fit.

A screenshot of a computer

Description automatically generated

Again, re-run the ARIMA test based on different components (0, 2, 5) to compare to the initial test and as a good practice for future time series analysis to get an understanding how the different components influence the changes in the statistics and the fitting of the model. The results were ultimately the same as the second test, which is the AIC was higher than the first test, proving that the initial test had the best fitted model and highest statistical significance.

A screenshot of a computer

Description automatically generated

With three ARIMA tests performed, I moved on to importing auto\_arima from the pmdarima library. Auto\_ARIMA is an alternative method to finding the recommended components (p, d, and q) while also checking for seasonality. As the name implies, auto\_arima runs different combinations of parameters automatically to find the model that has the lowest AIC value.

While attempting to utiliza auto\_arima, I documented code with multiple variations of reducing the data set and optimizations that I tried executing to give more accurate results, however, every attempt aside from the final implemented code resulted in tests that took too long or never finished because the data set is too large. As a result, I had to exclude the seasonality of “m = 252” which represents the number of business days or stock trade days in a given calendar year, and would’ve given a more effective result.

With that being said, the test was able to run successfully despite the limited optimizations, and landed on the best model, being (1, 1, 2), with the lowest AIC being 9376.494. When reviewing the statistics, there are a few key points to consider:

* For the Coefficient portion of the results, the AR and MA’s (ar.L1, ma.L1, and ma.L2), suggest it is a good, fitted model as they are statistically relevant. On the downside, the intercept ended up with a p-value > 0.05, which indicates that it not significant, however, this does not mean the overall model is not a good fit for forecasting, which means we can move forward with the analysis and prediction.
* For the diagnostic statistics, the overall results appear to give good news. The Ljung-Box test is a metric that involves residuals, and with a p-value of 0.98, this suggests no meaningful correlation, which is a good thing. With a p-value of zero for heteroskedasticity, this implies that the variation of the residuals is inconsistent over time, which can affect the model’s forecasting potential.

Overall, the model’s diagnostics indicate a good enough fit to move forward with the prediction, despite a less meaningful intercept and potential heteroskedasticity.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

After the auto-arima and diagnostics, I run ARIMA based on the identified best model parameters to view the SARIMAX results, and to fit to the model for forecasting. The calculated AIC from running ARIMA resulted in the same figure as the auto\_arima, which confirms the best model.

A screenshot of a computer

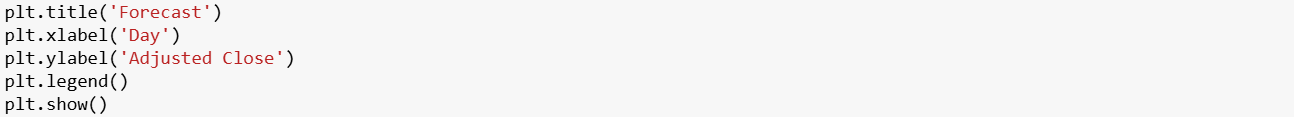
Description automatically generated

With the best parameters for p, d, and q applied to the model, I can now move on to plotting the train, test, forecast, and confidence interval. With the test data consisting of 20% of the model, this translates to 1257 business stock trade days. For the prediction to be accurate, it was determined to try to forecast up to two years of stock prices. In a given month, the average number of business days is 21, to calculate the number of business days in two years, I multiply 21 x 12 = 252, and 252 x 2 years = 504 days, which is what I ended up with. The code below shows the 504 days forecasted, the confidence interval, and the data points for the train and test splits.

Unfortunately, the visualization with the applied best model shows a forecast and confidence interval that stayed consistent over time and did not follow the test’s data points. This likely occurred due to the model not capturing the underlying relationships and variability in the data perfectly. An example if the inaccuracy would be that the data is not stationarity, however, we tested this in the beginning when the data was differenced, so we can eliminate that possibility. Just because the forecast and confidence intervals remained the same, does not mean the model should be disregarded automatically. I can still move forward with the prediction on the index data, but will need to pay closer attention to the model capturing the data patterns adequately, confirm that the residuals are irrelevant, and validate that the forecast is accurate.

A screenshot of a computer code

Description automatically generated



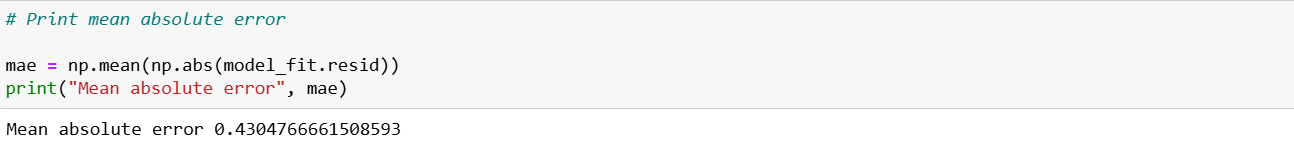
A graph with blue lines

Description automatically generated

Next, I ran the following code to determine the lower and upper limits of the adjusted close stock price for the confidence interval. Followed by calculating the MAE or mean absolute error, which gives the average ratio (0.43) that the predicted values deviate from the actual figures.

A screenshot of a computer

Description automatically generated



Moving on, I ran code that created four diagnostic plots based on the test.

* The standardized residual graph demonstrates errors randomly scattered, but always nearing zero. This indicates that there are no obvious trends.
* For the histogram and density plot, this measures the distribution of the residuals in the data. Based on the graph, the normality or green line shows a healthy bell curve which suggests a distribution that is normal. For the orange line, which represents the density line, the curve has some differences from the normality curve, meaning the residuals are not aligned perfectly, which is understandable.
* The normal Q-Q plot is an alternative graph to the histogram, which displays the difference in the normality and the residuals. The red line represents the normality points, but as displayed, the blue points or residuals appear to fall off around the beginning and end of the data points.
* Lastly, the correlogram mimics a plotted graph earlier in the script for pacf and acf. Specifically, this visual shows the acf of residuals on different lag points. What we would want to see is the coefficients remain close to zero, which they do, and this indicates that the model has identified any potential patterns or trends within the data.

A screenshot of a computer screen

Description automatically generated

For this next step, I run ARIMA on the initial data set model using the determined parameters of (1, 1, 2).

A screenshot of a computer

Description automatically generated

Now, generate predictions on the initial model and calculating the predicted mean based on the same number of days predicted on the test and train split. Then, measure the confidence intervals for the lower and upper limits of the adjusted close stock price to be plotted next.

A screenshot of a computer

Description automatically generated

Finally, move on to plotting the predictions for the initial model. Similar to the test and train test prediction, this graph will include the forecast and the confidence intervals for the lower and upper limits. As expected, the forecast and confidence intervals appear to follow the trend upwards in alignment to the growing stock prices over time, unlike the first prediction on the train and test data that demonstrated a consistent line.

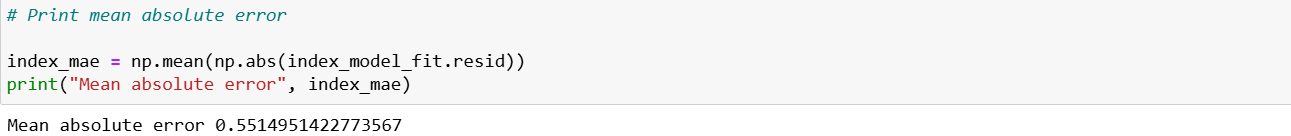
A screenshot of a computer program

Description automatically generated

A graph showing a line

Description automatically generated with medium confidence

Next, calculate the mean absolute error for the index data.



Lastly, re-run the four diagnostic plot tests, but on the index model. The results will be discussed in the next section which will be used as support to the recommendations and course of action following the analysis.

A screenshot of a computer screen

Description automatically generated

Overall, the section included an auto\_arima test, several ARIMA tests, and plotted data as support to the findings of the time series analysis and how the predictions can be interpreted.

Auto\_Arima

* Advantage – can be used to automatically run variations of parameters and highlight the best fit model with the lowest AIC.
* Disadvantage – this test can take very long to run since it does include running multiple variations of parameters for the best fit.

ARIMA

* Advantage – A strong tool to use on historical data to predict future trends.
* Disadvantage – it can be difficult to determine the correct p, d, and q parameters, which can lead to an inaccurate predictive model. However, this disadvantage can be remedied with good practice of utilizing both ARIMA and auto\_arima on time series models.

**Part V: Data Summary and Implications**

E.  Summarize the implications of your data analysis by discussing the results of your data analysis in the context of the research question, including one limitation of your analysis. Within the context of your research question, recommend a course of action based on your results. Then propose two directions or approaches for future study of the data set.

To circle back, the initial research question is, can a time series model using the ARIMA method accurately predict future stock price trends for Walmart with greater than 70% accuracy, based on 50 years of daily historical stock price data? Based on the ARIMA model (1, 1, 2) fit and the diagnostic statistical checks, the following points can be noted:

* The model had successfully identified underlying patterns in the data set, which was indicated by the relevance of the AR and MA coefficients.
* The Mean Absolute Error or MAE was 0.55, which suggested a somewhat average prediction error ratio.
* The many statistical diagnostic plots proved that the residuals were mostly uncorrelated and distributed at random, suggesting a good-fitted model.
* The histograms for both the train/test model and the index model revealed the residuals having variations from the normality of the data points. Also, the Normal Q-Q plots for both tests shows residuals veering off the line in the beginning and end of the data points, which may indicate that outliers exist, or some level of seasonality is affecting the distribution.

With these points in mind, the ARIMA model provided a consistent forecast with a moderate average error, which is useful for making informed decisions based on predicted future prices. However, the model's residual variations from normality suggests there might be some patterns or distinctions in the data that the model did not detect.

One major limitation of the analysis is the assumption of stationarity and the normal distribution of residuals. Although differencing was applied to accomplish stationarity, the residuals still presented heavy tails, indicating probable underlying patterns or anomalies/outliers that were not fully modeled.

Based on the results, the model fails to reject the null hypothesis and wasn’t a strong enough fit to be greater than or equal to 70% of prediction accuracy. It is recommended to use the ARIMA model for short-term forecasting of the adjusted close stock prices. However, it is also advisable to frequently review and update the model with new data to ensure its continued accuracy and dependability. Integrating additional optimizations and running Auto\_Arima on the full data set could potentially improve the model’s performance.

Future studies in response to this analysis may include experimenting with other data sets that may have a similar impact to predicting stock prices or even utilizing other analysis techniques that may perform or predict the stock prices with a higher accuracy percentage. “No one can see the future, but data allows us to harness the past to make a good guess (Confianz, 2022).” Afterall, it’s no denying that the stock prices for Walmart has been increasing over time. With that in mind, “businesses are increasingly looking for data-driven solutions to their problems (Saleem, 2023)” and the greatest course of action would be to continue to study the historical data in different angles as the assigned data analyst to uncover patterns, predict future progresses, and make informed decisions to the benefit of investors, corporate members of Walmart, day traders, or even the competitors.

**Part VI: Sources, In-Text Citations and References**

F.  Acknowledge Sources

Williams, T. A., & Sweeney, D. J. (June 21, 2024) Residual Analysis. Retrieved on June 23, 2024 from <https://www.britannica.com/science/statistics/Residual-analysis>

Hyndman, R.J., & Athanasopoulos, G. (2018) *Forecasting: Principles and Practice*. Retrieved on May 4, 2024 from <https://otexts.com/fpp3/>

Sarkar, S. (April 4, 2020) *Time Series Forecasting and Analysis: Arima and Seasonal-Arima*. Retrieved on May 4, 2024 from <https://medium.com/analytics-vidhya/time-series-forecasting-and-analysis-arima-and-seasonal-arima-cacaf61ae863>

Brownlee, Jason (November 18, 2023) *How to Create an ARIMA Model for Time Series Forecasting in Python.* Retrieved on May 11, 2024 from <https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/>

Dhaduk, H. (July 18, 2023) *Stock market forecasting using Time Series analysis With ARIMA model*. Retrieved on March 20, 2024 from <https://www.analyticsvidhya.com/blog/2021/07/stock-market-forecasting-using-time-series-analysis-with-arima-model/>

Hansen, S. (February 22, 2024) *What Does Walmart’s Stock Split Mean for Investors?* Retrieved on April 7, 2024, from <https://www.morningstar.com/markets/what-does-walmarts-stock-split-mean-investors>

The Investopedia Team (May 8, 2024) *How to Find Historical Prices for Stocks.* Retrieved on May 9, 2024, from <https://www.investopedia.com/ask/answers/03/103103.asp>

Vanzo, R. (March 11, 2024) *Where will Walmart Stock Be in 3 Years?* Retrieved on May 9, 2024, from <https://www.fool.com/investing/2024/03/11/where-will-walmart-stock-be-in-3-years/>

J.  Acknowledge sources, using in-text citations and references, for content that is quoted, paraphrased, or summarized.

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. Chapter 9. Retrieved on May 19, 2024, from <https://otexts.com/fpp3/>

Confianza (May 30, 2022) *Time Series Analysis Forecasting and How It’s Helpful for Your Business*. Retrieved on March 28, 2024, Paragraph 23, from <https://www.confianzit.com/cit-blog/time-series-forecasting/>

Saleem, A. (December 27, 2023) Why Kaggle is the best platform for data scientists? Retrieved on June 13, 2024, Paragraph 1, from <https://datasciencedojo.com/blog/kaggle-data-scientists/>