**D213 Performance Assessment Task 1**

**TIME SERIES MODELING FOR MEDICAL DATA**

Fahim A. Akbar Student ID 001434895 Masters Data Analytics (January 1, 2021) Program Mentor: Mandy Rasmuson (385) 428-5957x5957 [fakbar3@wgu.edu](mailto:fakbar3@wgu.edu)

**Part I: Research Question**

**A. Data Mining Report Justification**

**A1. Research Question:**

The research question that this analysis will focus on is if time series modeling can accurately predict daily revenues for the hospital compared to the actual observed daily revenues.

**A2. Goal**

The goal of the analysis is to predict daily revenues based on the provided hospital data. To do so, 80% of the dataset will be used as a training set to predict the remaining 20%. This approach will generate values as a point of comparison, and help to avoid creating future predictions that cannot be evaluated using the given dataset. This project will be conducted using an ARIMA time series model to assess the 80% of data. Our aim is to accurately project the values for the remaining 20% of data.

**Part II: Method Justification**

**B. Assumptions of Time Series Model**

For this project, we will be conducting a time series analysis. () Time series analysis using ARMA or ARIMA models operates on the following assumptions of the data being analyzed:

**Stationarity:**

* The data used in the times series analysis must be stationary, and no trends or seasonality should be present.
* The mean of the series should be the same for all time periods
* The variance should remain constant over time
* The strength of the relationship between the series and its lagged values (autocorrelation) should be constant.
* The mean and variance are stable when centered at zero after detrending the data, highlighting their importance in analyzing lags and differences as they represent specific points in time.
* There are no outliers or any other abnormal values present in the dataset.
* The data is univariate, and represents a singular variable intended for modeling
* Past datapoints are indicative of the behavior of future datapoints

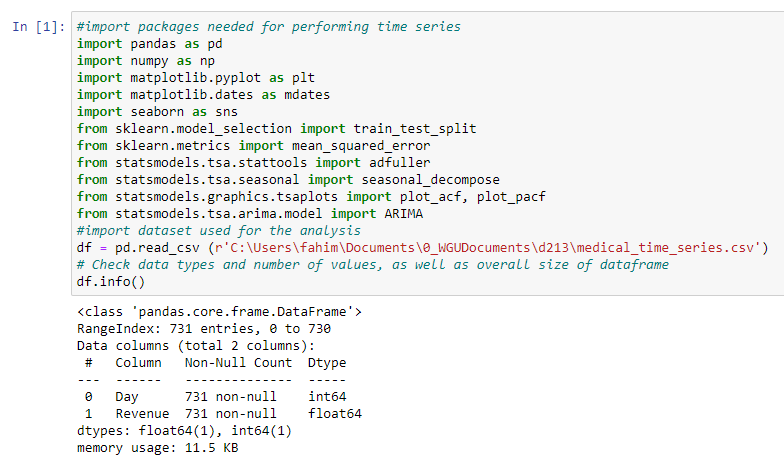
**Autocorrelation:** degree to which a time series is correlated with its own past values.

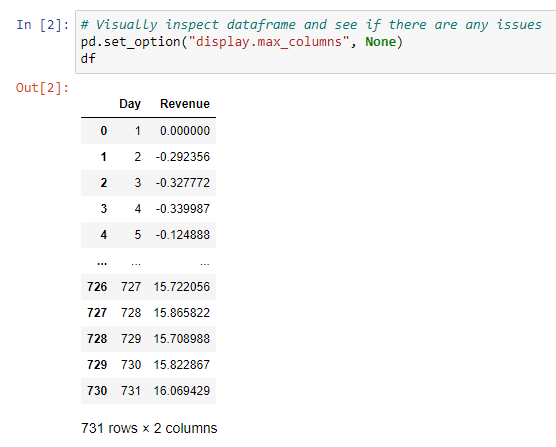
* **Autocorrelation Function (ACF)**: plot that represents the correlation coefficient between a time series and its lagged values. In a stationary time series, ACF drops to zero relatively quickly, indicating that past values do not predict future values once the appropriate lag is reached.
* **Partial Autocorrelation Function (PACF):** PACF represents the correlation between a time series and its lagged values, controlling for the effect of other lags in between. PACF helps in determining the relationship between specific lags without the influence of other lags.

(Tyagi, 2021)

**Part III: Data Preparation**

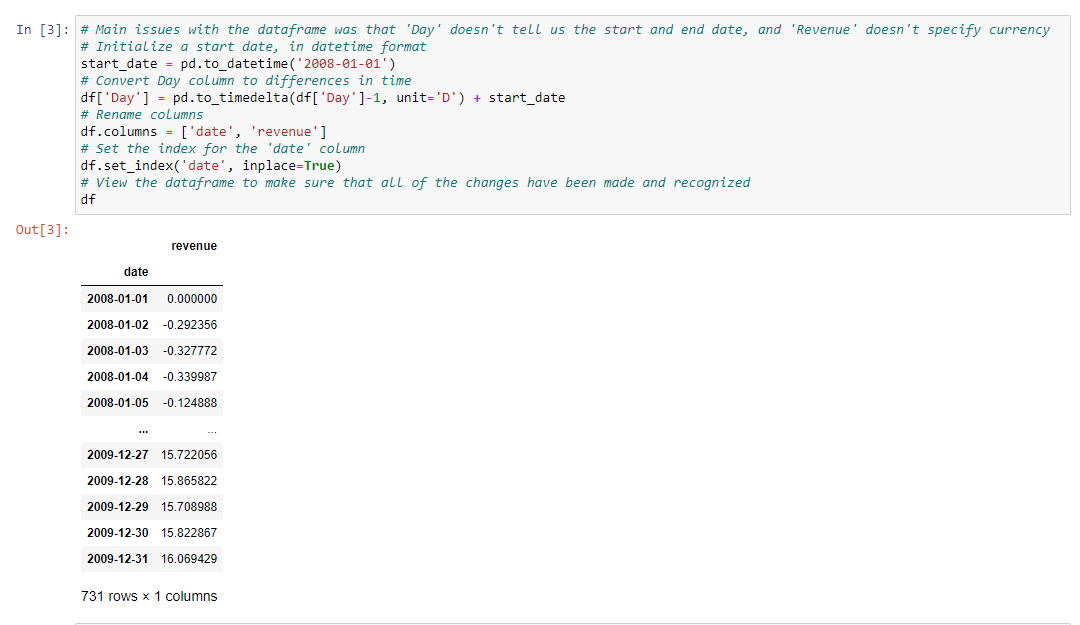
**C. Summary of the data cleaning process**

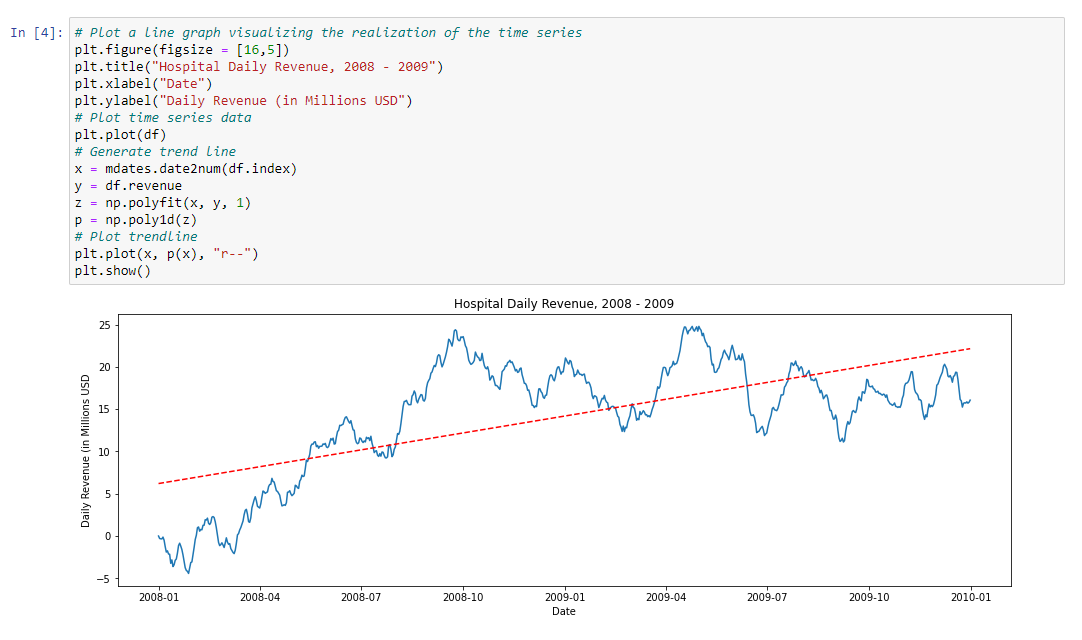
**C1. Time Series Visualization**

****

The two column dataset used in this analysis provides us with records of the first two years of revenue from the hospital. The first column, 'Day,' spans from Day 1 of the initial two-year period to Day 731. The second column, 'Revenue,' represents the daily financial performance for the respective day, indicating the hospitals monetary gains or losses in millions.

One limitation of the dataset present is the nature of the day column. The 'Day' column should be converted to a DateTime object for better usability, but the dataset lacks context about when the data was collected. It could represent two years starting from a random date, not necessarily recent years, and includes a leap year, resulting in 731 daily observations instead of the expected 730 for a two-year period. Since the start date for the data is never disclosed in the data dictionary as well, we will be using January 1 2008 as our starting point. Furthermore, it’s not specified what currency is used for revenue, so we will assume USD for this analysis.





After the data was cleaned and labeled, the graph provided above was generated to illustrate the daily revenue in millions of dollars USD for the hospital over the course of two-years, from2008 - 2009. The graph illustrates an increase in daily revenue over these years, which is further emphasized by the red trendline.

**C2. Time Step Formatting**

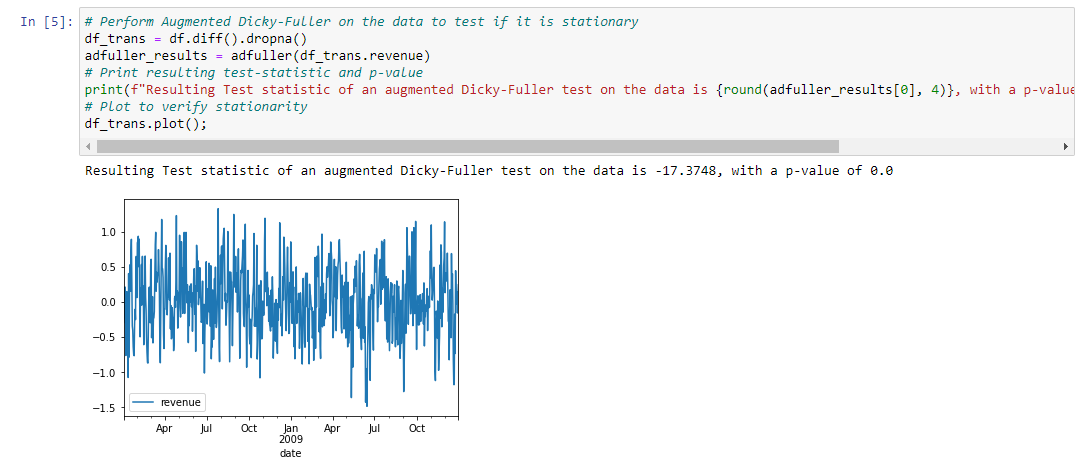
The time series is composed of a DateTime index and daily revenue observations in millions of dollars USD for the hospital. Each day, the DateTime index advances by one day, spanning from January 1, 2008, to December 31, 2009, with no missing data points. There are 731 data points in the sequence, including the leap year in 2008. Representing the data as a DateTime object allows for diverse aggregations such as revenue by week, month, or year, as well as other data manipulations.

**C3. Series Stationarity**

Stationary time series are characterized by the absence of trends. Based on the resulting plot shown in C1, the data here demonstrates a lack of stationarity. The dataset shows that the hospitals daily earnings started at 5 million dollars during the initial months, and then latter leveled 15-20 million dollars per day. This confirms a continuous increase in daily revenue. To validate this trend, a trendline was incorporated into the plot. The rising red dashed trendline further emphasizes the upward trend. Therefore, it is apparent that this data is not stationary and needs to be prepared before analysis.

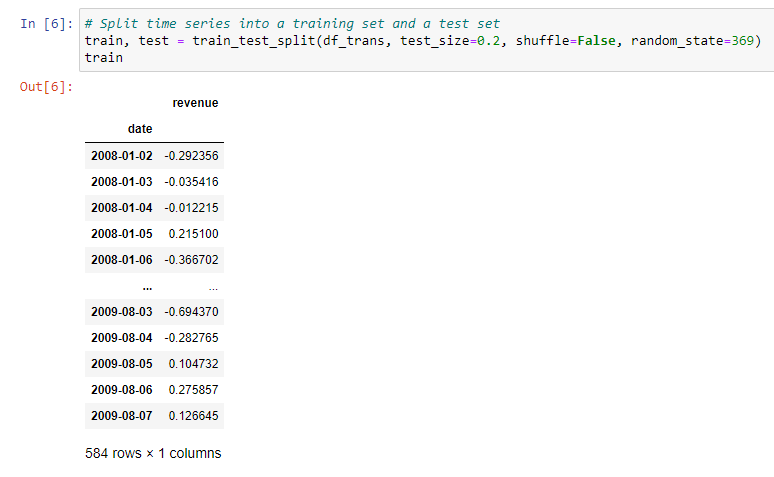
**C4. Data Preparation and Explanation**  
In order to use the data for time series analysis, the following steps will be taken for data preparation:

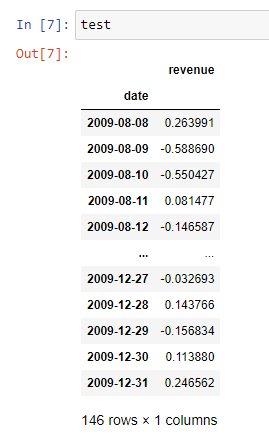
1. Convert the "Day" column into a valid DateTime format.
2. Set the revised "Day" column as the index for the time series (as explained in C1 above).
3. Transform the "Revenue" data into a stationary time series, removing any trend or seasonality.
4. Divide the data into distinct training and test sets.

****The plot and augmented Dicky-Fuller results shown above confirm that the data is now stationary, and we can continue with splitting the data into training and test sets. To ensure that both the train and test sets are provided and the time series retains its order, SciKit-Learn's train\_test\_split will be used with shuffle=False. This will retain the time series data’s order and prevent data rearrangement.

**C5: Copy of Prepared Data Set**

The data was successfully split into both a training and a test set. The training set comprises of 80% of the data and the test set comprises of 20% of the data. The training set is saved as 'D213Task1\_train\_clean.csv', while the test set is saved as 'D213Task1\_test\_clean.csv'.



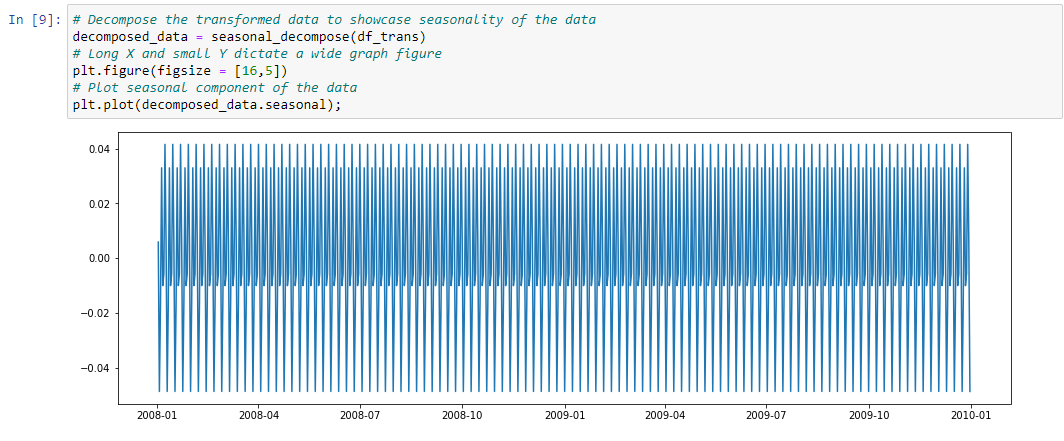




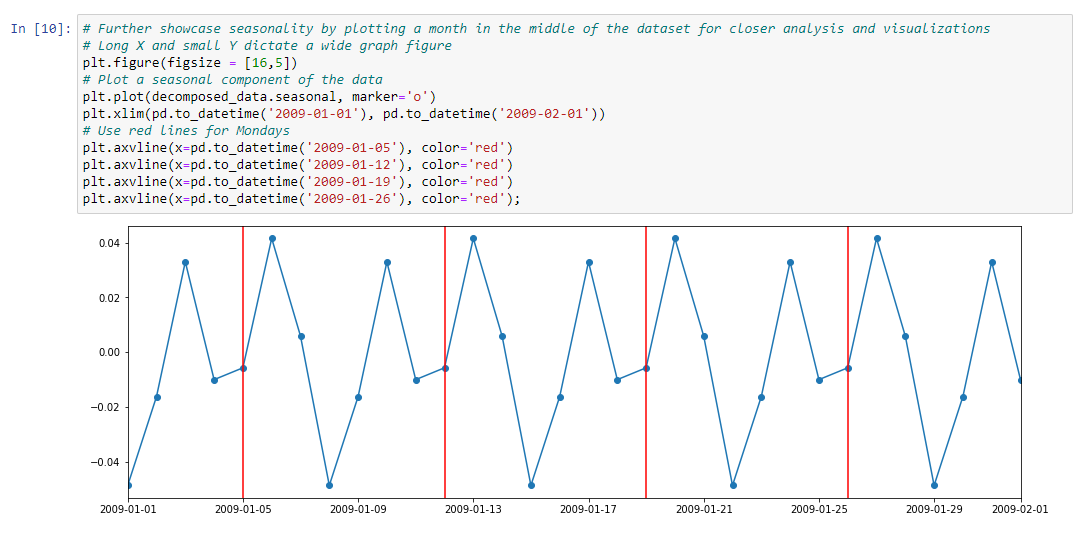
The preparation of the dataset was performed in Python using a Jupyter notebook environment. The Jupyter notebook file is attached to the task submission. A pdf copy of the notebook and a txt. file of code used is provided with the task submission as well. A copy of the cleaned dataset is also provided with the task submission. Lastly, the entire code used is also provided at the end of the document.

**Part IV: Model Identification and Analysis**

**D1: Annotated Findings & Visualizations**

**Seasonality**

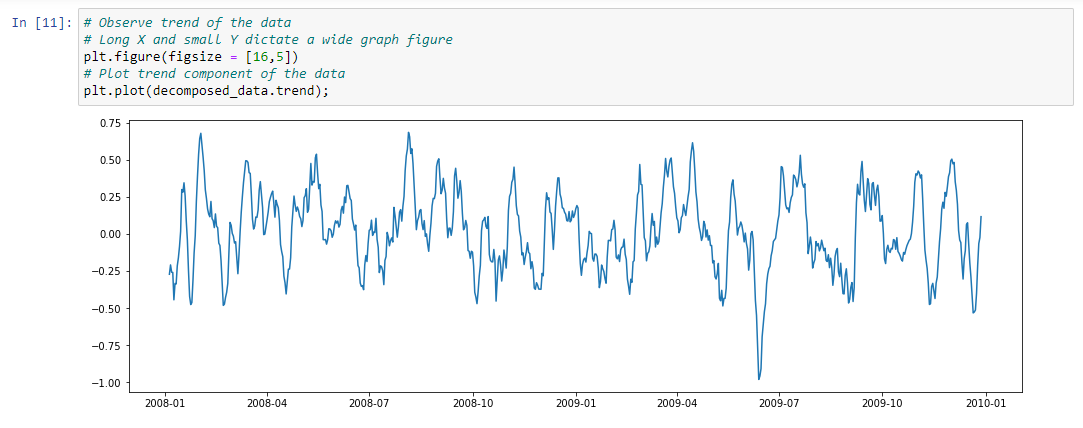
Decomposition of the data shows a repetitive pattern, indicating seasonality. While the seasonal component's magnitude is exceptionally small at around 0.04 in either direction from zero at its most extreme, it is in stark contrast to the dataset's overall magnitude, which ranges from 1.0 to 1.5. (The magnitude is shown on the y-axis).This is likely due to certain days of the week. To confirm this, a month in the middle of the dataset was plotted for closer inspection of the data.



The above graph was generated to look at the seasonality from January 1 2009 to February 1 2009. Based on a 2009 calendar, the red lines occur on Mondays. We can observe data points to the left or right to determine values for different days of the week. We see that there are two peaks every week: one on Saturday (represented by two points to the left of the Monday lines) and another on Tuesday (represented by one point to the right of the Monday lines). Similarly, there are two noticeable dips in each week: one on Sunday (indicated by one point to the left of the Monday lines) and another on Thursday (indicated by three points to the right of the Monday lines).

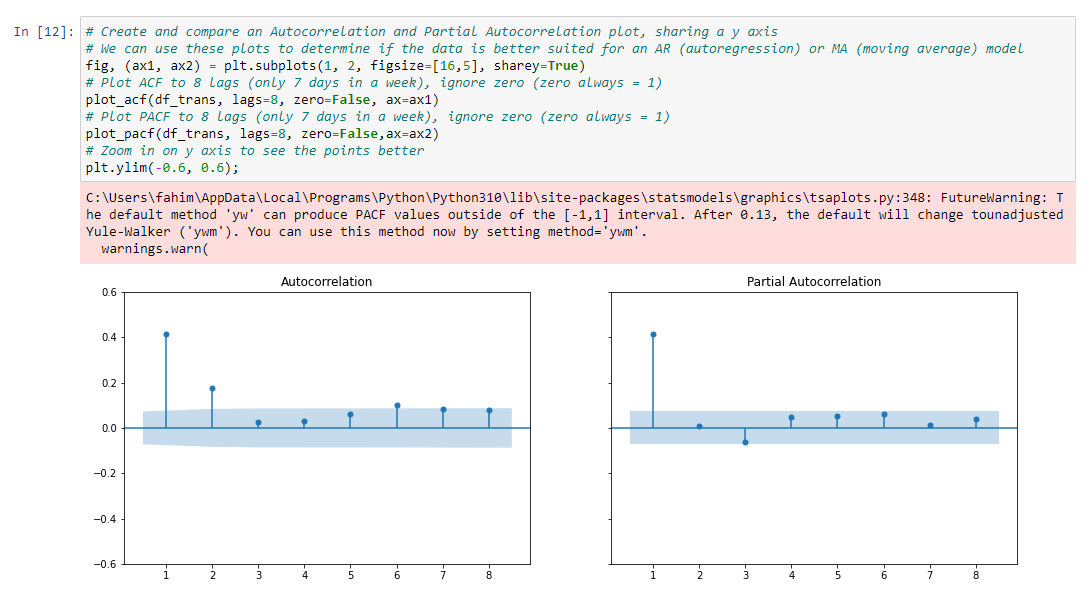
The influence of this seasonality appears minimal when compared to the transformed data. The outcomes of the previously conducted augmented Dicky-Fuller test confirm that the data is stationary.

**Trends**



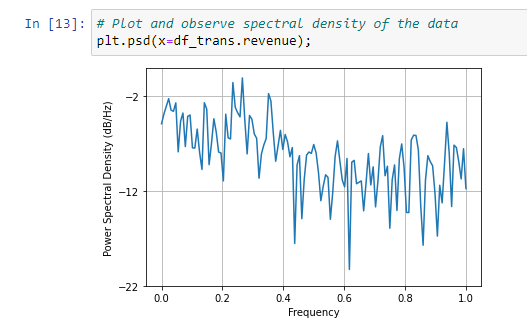
The graph shows the lack of an overall trend in the dataset. There does appear to be an outlier in the summer of 2009.

**Auto Correlation Functions**



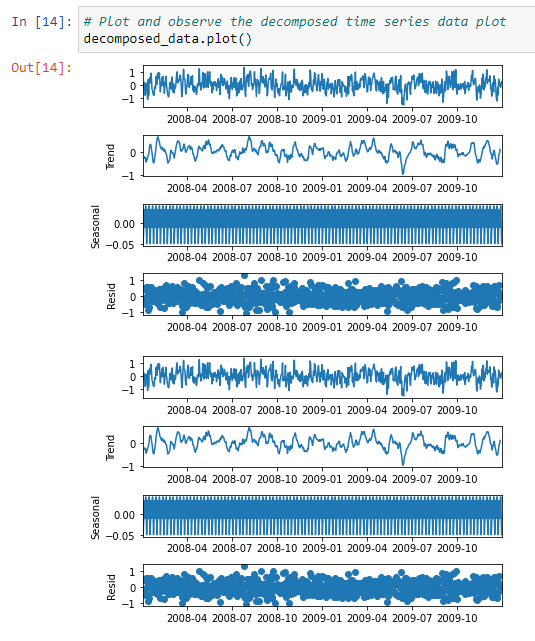
Here, ACF and PACF plots were generated to determine if the data corresponds to an AR (autoregression) or MA (moving average) model. The shaded area in the plot signifies statistically insignificant outcomes, allowing exclusion of those datapoints from consideration. It is observed that the ACF tapers off around 2 (although this determination carries a degree of subjectivity), whereas the PACF distinctly stops at 1 (Monigatti, 2022). When the ACF gradually diminishes and the PACF abruptly ceases after lag p, it signifies an AR model of order p. Based on the behavior exhibited by this data, it is most fitting for an AR(1) model.

**Spectral Density**

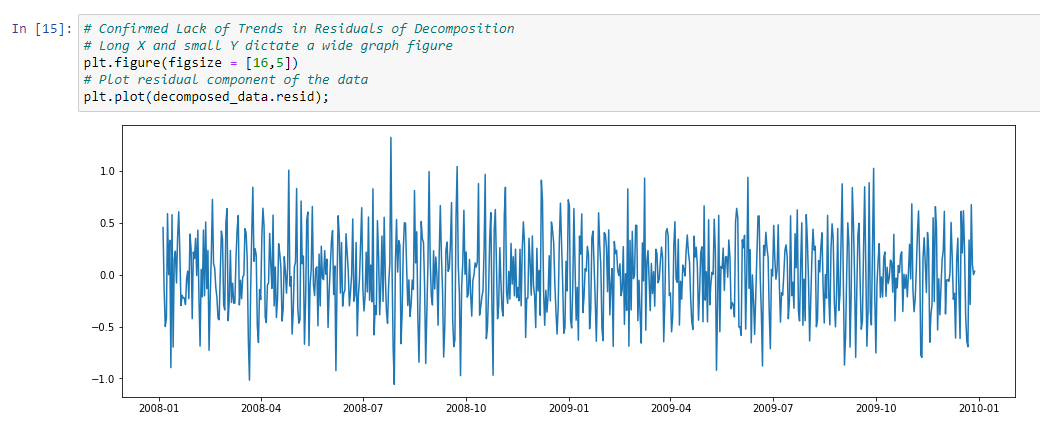


A plot of the power spectral density of the transformed time series is displayed above.

**Decomposed Time Series**



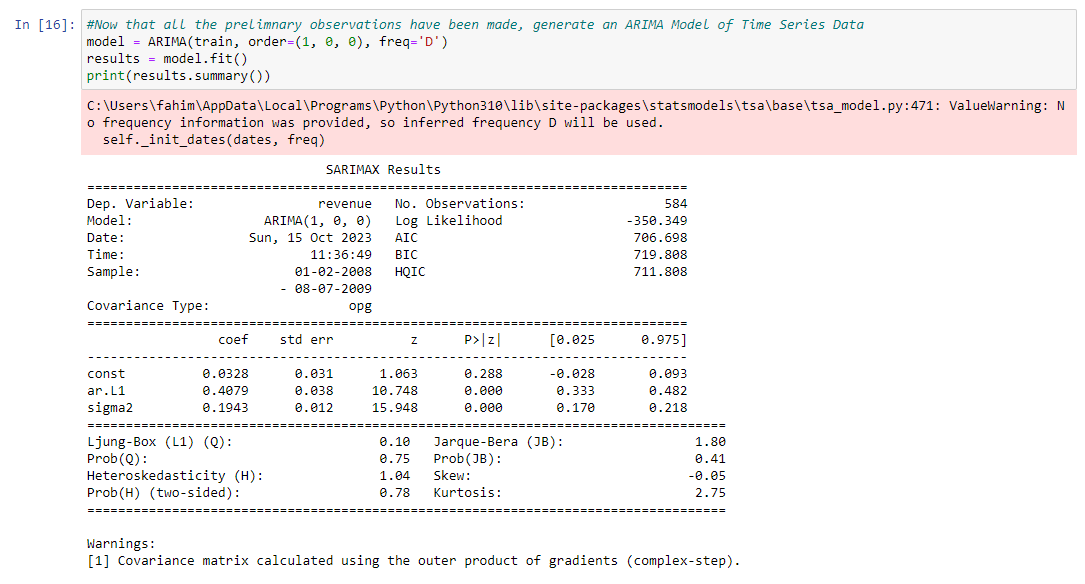
**Confirmed Lack of Trends in Residuals of Decomposition**



The residuals of the decomposition lack apparent trends, as shown above.

**D2: ARIMA Model of Time Series Data**

Based on the trends within the ACF and PACF plots, the data is best suited to an AR(1) model.

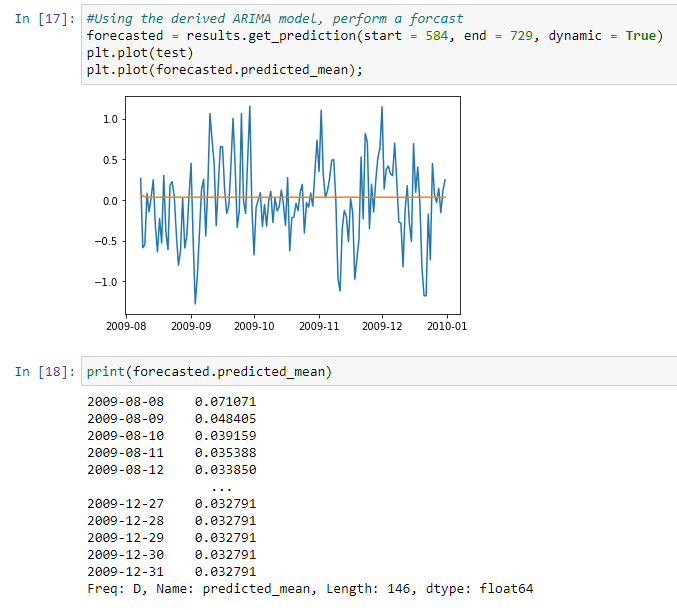


The above coefficients derived from the model give us the resulting ARIMA equation:

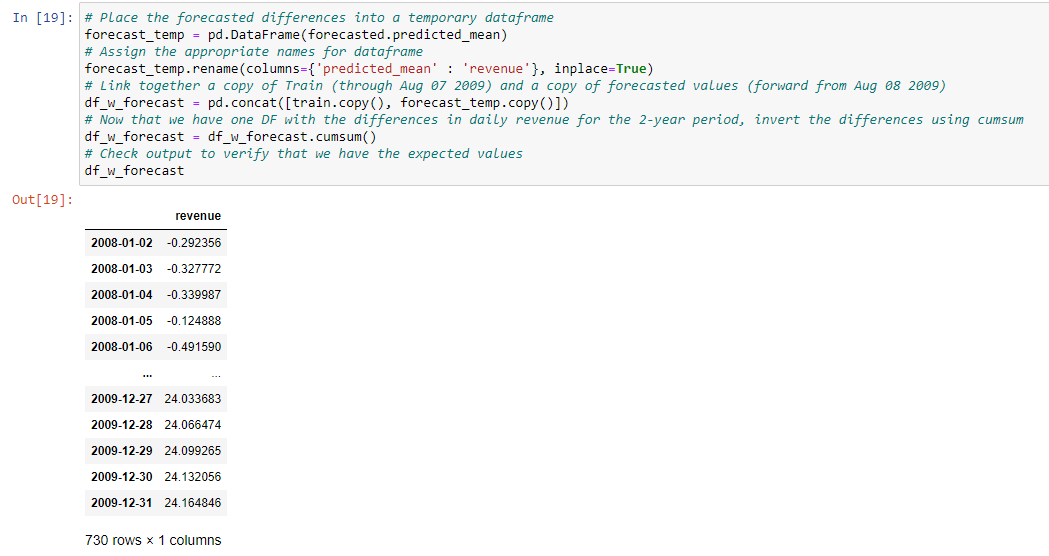
*Xt* = (0.4079(*X(t-1)*) + 0.0194 + at

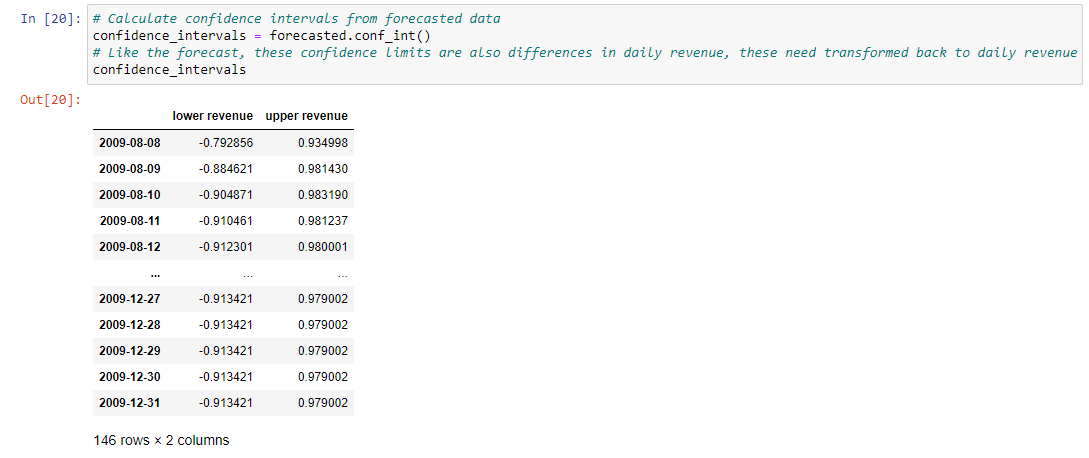
**D3: Forecast**

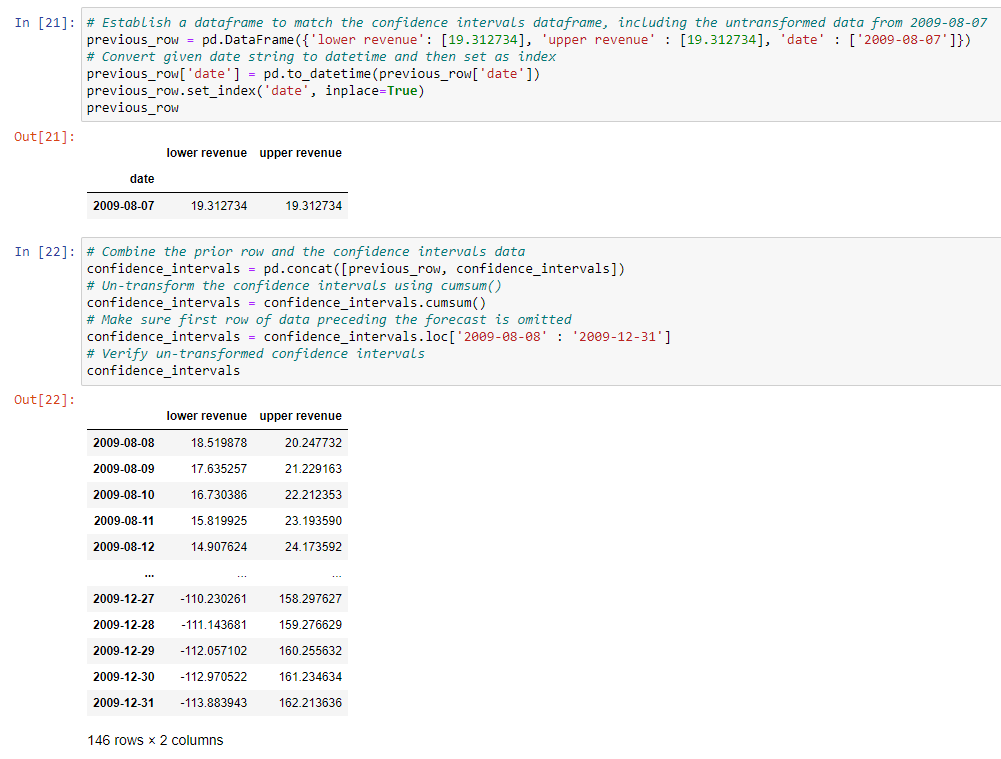
The testing dataset comprises 146 rows, allowing for forecasting attempts of these specific values. This establishes a basis for comparison, enabling the verification of predicted values against the actual values present in the test set.

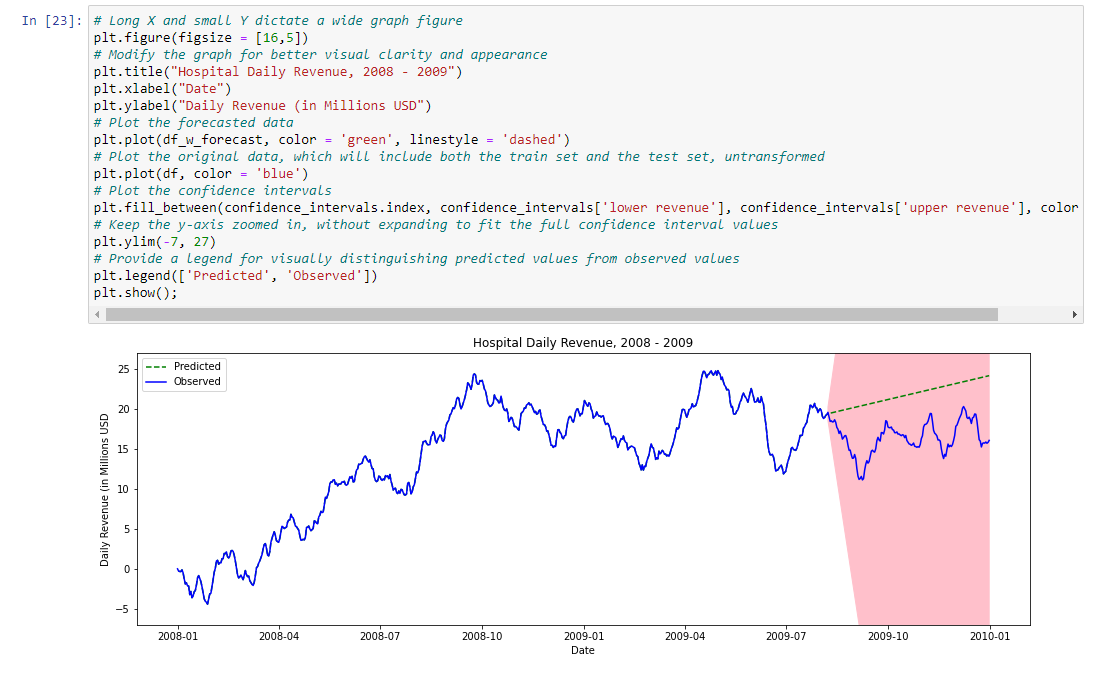


The analysis reveals the forecast's limitations, as shown by its gradual decline and stabilization at the value 0.032791. This figure doesn't directly predict the daily revenue for the Hospital, but rather the forecasted a prediction of the daily difference in daily revenue. Thus, the ARIMA model isn't predicting a revenue of 0.032791 million dollars; rather, it anticipates a daily increase by a factor of 0.032791 million dollars compared to the previous day. With this in mind, we can proceed by inverting the earlier transformation and create a plot comparing the predicted data with the observed data in the test set. Additionally, we will calculate the confidence intervals for the forecast and include them in the plot.

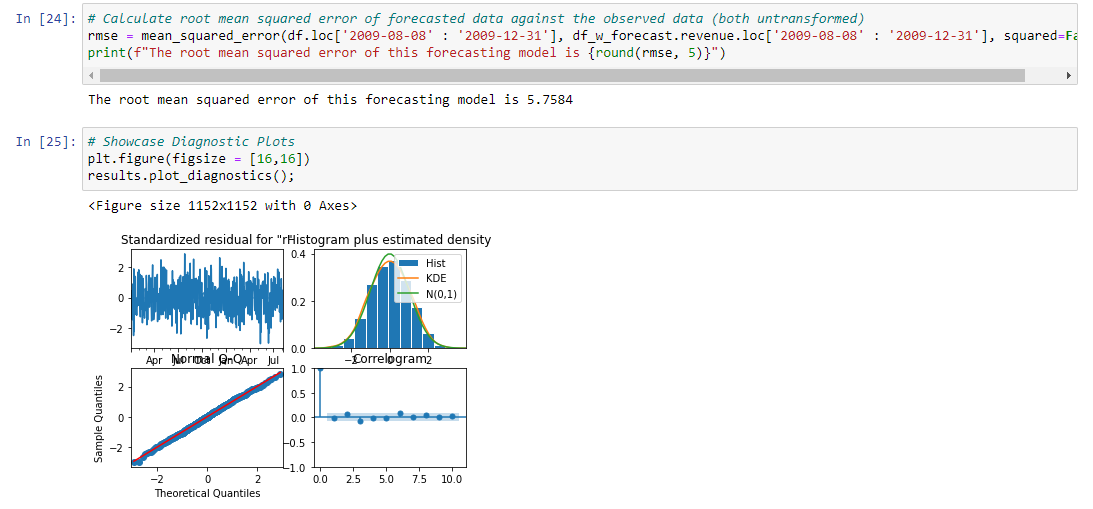








The above plot incorporates the forecast for the final 20% of the period (from Aug 08, 2009, to Dec 31, 2009) derived from the initial 80% of the period, along with the corresponding actual values for that duration, which reveals a model characterized by a broad confidence interval. When this model is extended over a substantial forecasting period of 146 days ahead, the uncertainty amplifies, leading to a wide confidence interval. Having conducted the forecast, various elements can be generated for model assessment. This assessment encompasses the calculation of the root mean square error and the examination of diagnostic plots.

**D4. Analysis Output & Calculations**

All calculations performed and the resulting outputs for this assessment are showcased above and are also provided in the Jupyter notebook accompanying this task submission.

**D5. Arima Model Code**

The code used for generating the Arima model is provided above and in the Jupyter notebook. In addition, a txt document with the full code is provided with this task submission.

**Part V: Data Summary and Implications**

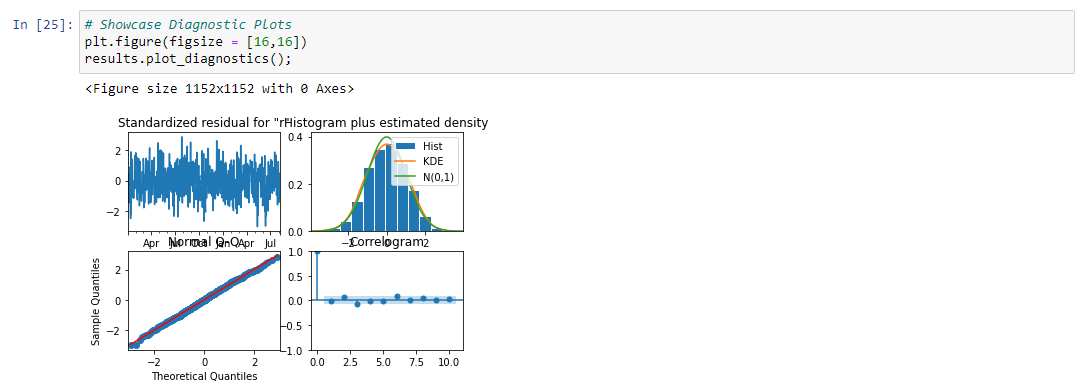
**E1. Results of Analysis**

Selection of an Arima Model: The autocorrelation and partial autocorrelation functions were used to determine if this data is best suited to an AR (autoregression) or MA (moving average) model. With the observed behavior in section D1 of the ACF tailing off at 2 and the PACF cutting off at 1, this indicated that the model is best suited for a model of AR(1). The resulting ARIMA model had an overall order of (1, 0, 0). Fitting the ARIMA model to the training data produced the following equation for this dataset:

*Xt* = (0.4079(*X(t-1)*) + 0.0194 + at

The prediction interval of the forecast extends 146 steps beyond the conclusion of the training data, equivalent to 20% of the complete dataset. Considering the substantial forecasting duration, the decision was made to allocate the minimum value of 20% of the data to the test set. This translates to predicting around 5 months' worth of revenue based on only 19 months of available data. There is a concern that a significant portion of the 19 months of training data might not accurately represent later data due to the complexities involved in early business observations. This concern seems to have some validity, as the data reveals three distinct "phases" or "eras." In the initial 3-4 months, daily revenue fluctuates around zero, reflecting the challenges of establishing a new and substantial business. Following this phase, there is a period of 4-5 months with consistent revenue growth, eventually stabilizing within the range of 15-25 million dollars per day. If this plateau truly represents the hospital’s performance, then the initial 7-9 months of operation can be considered a startup period and may not accurately predict future performance. Furthermore, more than 25% of the dataset might not contribute significantly to the model's predictive value.

Independently, the model's performance can be evaluated by the output of the diagnostic plots.

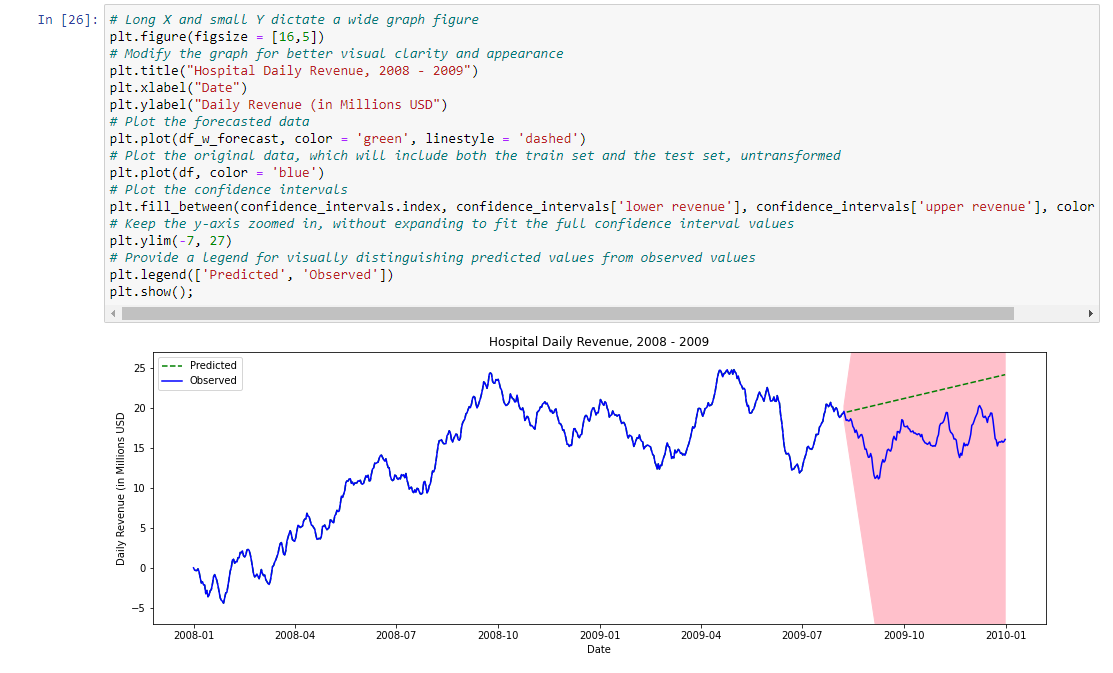


* The top left is the standardized residual plot, which should contain no obvious patterns. This plot does not show any apparent patterns.
* The top right is the histogram with KDE estimate. The KDE curve should be very close to the normal distribution, which ours is.
* The bottom left is the Normal Q-Q plot. Most (preferably all) of the datapoints should occur on the red line, which is the case here.
* The bottom right is the correlogram. All correlations for lag greater than zero should be insignificant, meaning they should be in the shaded area. This model appears to have elements at lag(3) and lag(6) which look to be right on the border of being significant.

The results of the correlogram indicate that the model may have slightly too much correlation for particular lag values. This is likely related to the very slight seasonality by day of the week effect that was examined earlier. Nonetheless, this model doesn't appear to be particularly problematic from a statistical perspective. I do believe that it does lack practical significance, however, based upon the imprecise nature of its forecasting, only accounting for a slight linear upward increase in daily revenues. This is a conclusion of limited value, especially given the confidence range associated with this forecast.

**E2: Forecast Visualization**

The final visualization that plots both the observed data provided by the hospital and the forecast generated by this model is provided below. The confidence interval is also included:



The final forecast of the hospital daily revenues is successful in predicting the general growth rate of revenues. However, it only projects the rates of increasing revenues, and does not account for revenue drops such as during the initial 9 months period. While the expected growth over the next few years is constant, the reality of owning a business means that there will be drops in revenue and other unexpected outcomes and anomalies that need to be considered.

**E3: Recommended course of Action**

One recommended course of action would be to exclude revenue data from the first 9 months, given the volatility of revenue growth and decreases during this time frame. This exclusion would help give a more accurate model for projected revenue growth as the growth rate appears more consistent after these months. Another recommended course of action would be to begin data collection after projected start up periods. This way, a more consistent growth of revenue can be evaluated and used to create forecasts with higher accuracy. Lastly, any foreseeable events with determined dates, such as expanding the hospital, construction updates, government grants, should be accounted for. That way, we can determine what events may cause volatility in revenue growth, and help better contextualize why the forecasted revenue rates might be altered.

**Part VI: Reporting**

**F: Reporting**

The preparation of the dataset was performed in Python using a Jupyter notebook environment. The Jupyter notebook file is attached to the task submission. A pdf copy of the notebook and a txt. file of code used is provided with the task submission as well. A copy of the cleaned dataset is also provided with the task submission. Lastly, the entire code used is also provided at the end of the document.

**G. Sources Cited:**

Statistics Solutions. (n.d.). Time Series Analysis. Retrieved June 6, 2023, from [https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/time-series-analysis**/**](https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/time-series-analysis/)

Tyagi, S. (2021) Introduction to time series forecasting - part 2 (Arima models), Medium. Available at: <https://towardsdatascience.com/introduction-to-time-series-forecasting-part-2-arima-models-9f47bf0f476b>

Monigatti, L. (2022) Interpreting ACF and PACF plots for time series forecasting, Medium. Available at: <https://towardsdatascience.com/interpreting-acf-and-pacf-plots-for-time-series-forecasting-af0d6db4061c>

Calendar for year 2009 (United States) (no date) Year 2009 Calendar – United States. Available at: <https://www.timeanddate.com/calendar/?year=2009&amp;country=1>

**Third party code used:**

<https://www.kaggle.com/competitions/store-sales-time-series-forecasting>

<https://www.kaggle.com/code/kashnitsky/topic-9-part-1-time-series-analysis-in-python>

<https://www.kaggle.com/code/prashant111/complete-guide-on-time-series-analysis-in-python>

<https://www.kaggle.com/code/galibce003/stationarity-and-dickey-fuller-test-with-example>

<https://www.kaggle.com/code/jurk06/time-series-the-various-steps>

<https://stackoverflow.com/questions/61040284/problems-with-acf-plots-and-operands>

<https://www.kaggle.com/code/prakharprasad/time-series-ar-model-stationarity-test>

<https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/>