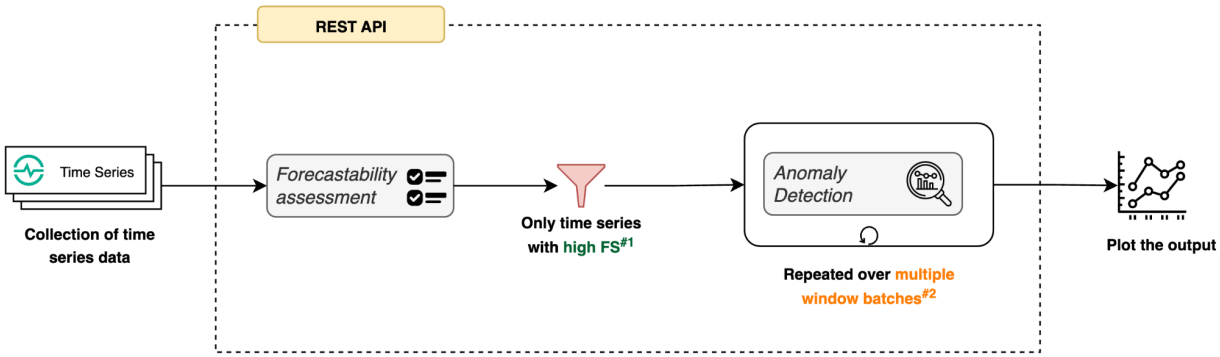
DataGenie

Discover Autonomous Data Intelligence

Akhil SM
21PC03

High Level Overview



During coding, I have followed the above mentioned high level overview.

1. I have done the following steps-
   i) Imported the dataset
   ii) Cleaning the dataset (imputed nan with mean values)
   iii) Visualized necessary plots

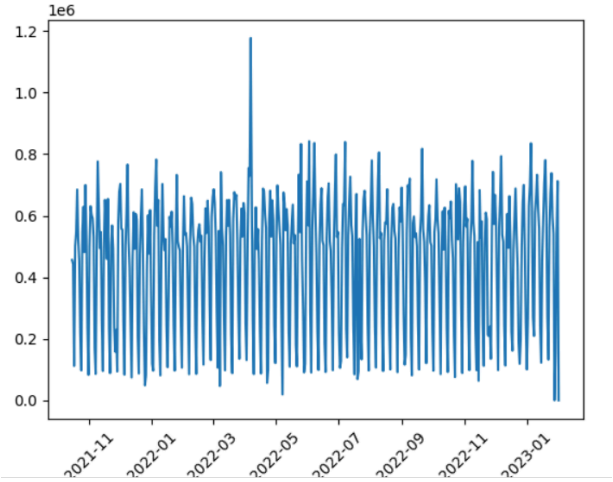
2. Performing the Augmented Dickey-Fuller test for checking the stationarity of the given time series dataset.
3. seasonality is detected
4. Plot Autocorrelation function (ACF) and Partial Autocorrelation function (PACF)
5. With this, I fixed the parameters for the model SARIMA
6. Then prediction is done for next 7 days

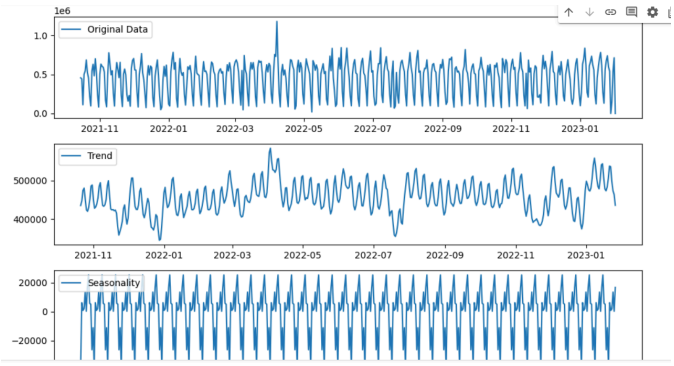
7. Check for significant spikes at seasonal lags
8. Check if seasonality is detected
9. Feature selection
10. Splitting the data into training (70%) and test (30%) sets
11. Calculate the Moving Average
12. Plotting the original data and the moving average
13. Create SARIMAmodel and calculate mape
    i) Load the time series data
    ii) Convert data to pandas Series if it's not already
    iii) Normalize the data
    iv) Split the normalized data into training and test sets
    v) Fit a SARIMA model on the training data
    vi) Predict values on the test data using the trained model
    vii) Inverse transform the predictions to the original scale
    viii) Calculate the Mean Absolute Percentage Error (MAPE)
    ix) used ACI and BCI to evaluate how well the SARIMA model fits the data

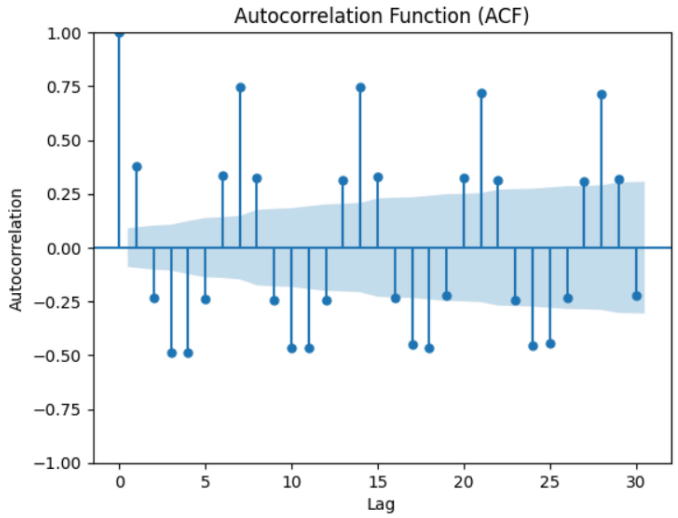
14. train a model with the given dataset
15. make predictions on the test dataset
16. calculate the RMSE
17. calculate the MAPE (Mean Absolute Percentage Error)
18. calculate the forecastability score
19. Split data into overlapping windows of 7 days each
20. Initialize Prophet model
21. Iterate over each window
22. Fit the model to the window of data
23. Make predictions for the window
24. Calculate residuals
25. Define threshold for anomaly detection (e.g., 3 standard deviations from the mean)
26. Detect anomalies
27.
28. determine the optimal threshold for forecastability
29. train the model
30. generate predictions
31. calculate the MAPE
32. return the predictions, forecastability score, and MAPE

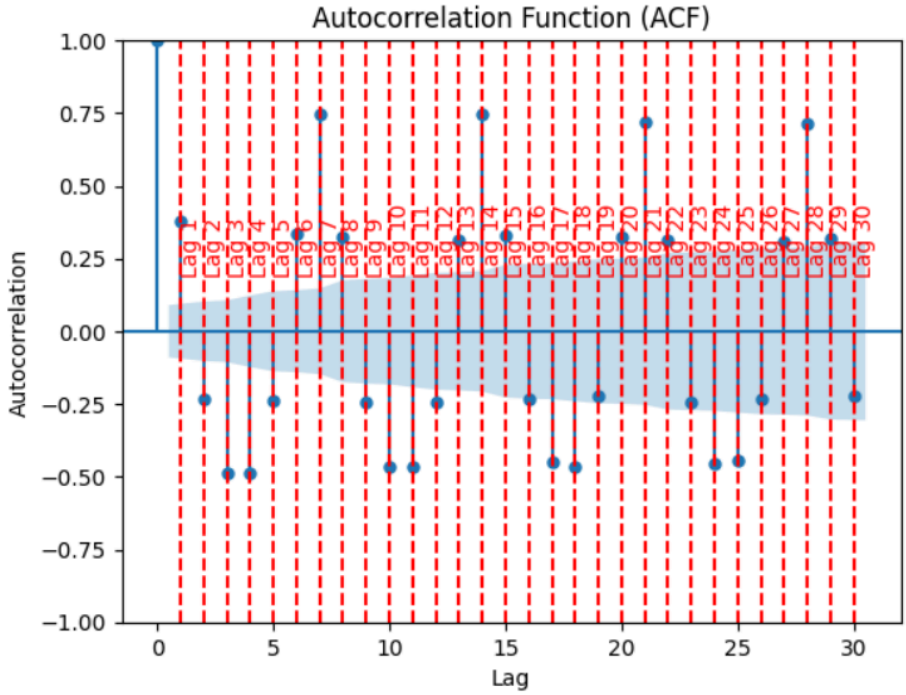
33. using GridSearchCV for hyperparameter tuning
34. The insights from the ACF and PACF graphs are also gained in order to fix the parameters for SARIMA
35. define a list of hyperparameters
36. create a model and initialize grid search
37. searching the best possible parameter
38. detect change points
39. train the model with the adapted window size

40. define the timeout for the sweeping algorithm
41. define the range of parameter tuning
42. Calculate the Augmented Dickey-Fuller test statistic
43. Determine the changepoint by comparing the ADF statistic with the critical value
44. Initialize the minimum window size
45. Determine the optimal window size







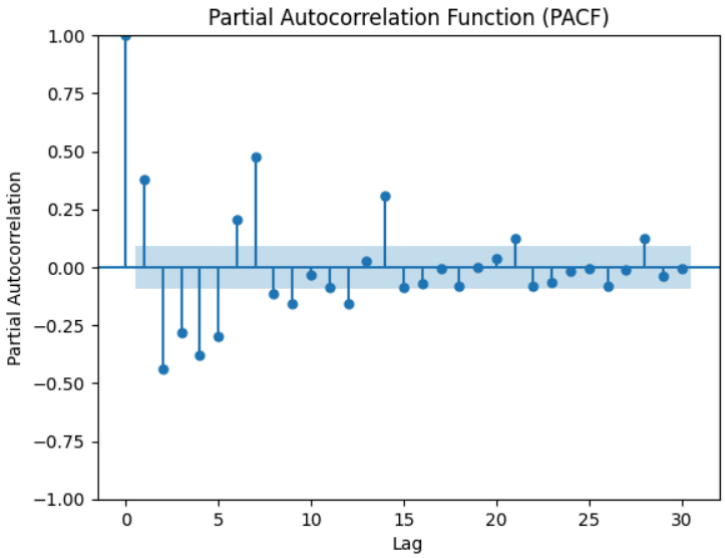


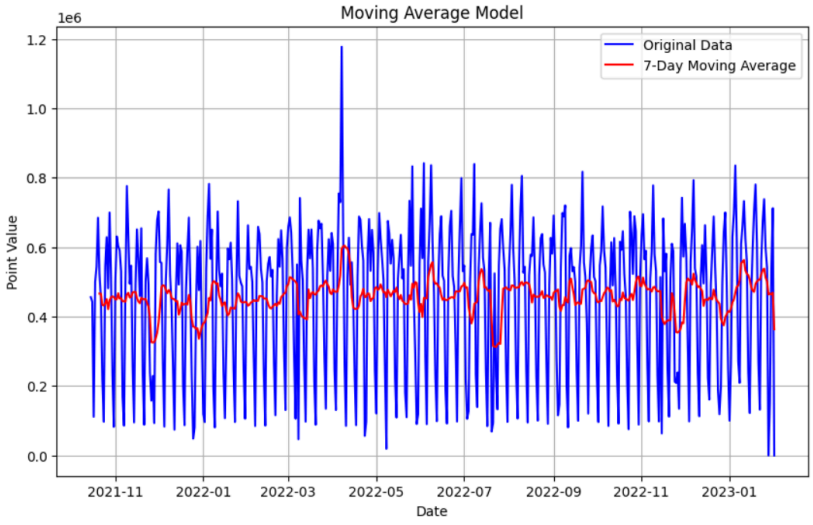

```
Seasonality Detected: True
```

```
Forecast for the next day: [284663.91671664 519954.43131244 697144.18187734 732604.10345634
 617407.80178454 287956.72263744  95888.13286257]
```

For next week

```
Mean Absolute Percentage Error (MAPE) for SARIMA: 0.99%
```






```
ADF Statistic: -5.591258452497603
p-value: 1.3275038715229179e-06
Critical Values:
   1%: -3.4446773373329576
   5%: -2.8678574606780654
   10%: -2.5701349669405404
```

```
Extracted Features:
            mean    median        std_dev  skewness  kurtosis seasonality  \
0  456027.723629  522566.0  216471.561459 -0.459241 -0.793585 -268.213832

           trend
0  457171.167141
```