Airline Passengers Tickets Sales Prediction using Time Series Forecasting

Self Project

1 Introduction

The goal of this project is to explore and implement various time series forecasting methods on a dataset containing airline passengers' ticket sales. The models considered for this analysis include ARIMA, SARIMA, PROPHET, and XGBoost. The dataset used is relatively clean and straightforward, making it suitable for demonstrating the effectiveness of these forecasting techniques.

2 Importing Libraries

To begin, we import the necessary libraries for data manipulation, visualization, and time series forecasting. These include `pandas` and `numpy` for data processing, `statsmodels`, `Prophet`, and `pmdarima` for modeling, and `matplotlib` and `seaborn` for plotting graphs.

3 Loading the Dataset

The dataset is loaded into a pandas DataFrame for analysis. It consists of monthly data on airline passengers or ticket sales.

```
df = pd.read_csv("Airlines_Passenger.csv", nrows=144, names=["month", "passengers"], header=0)
```

4 Stationarity Check using ADF Test

Before applying any time series models, it is essential to check the stationarity of the data. The Augmented Dickey-Fuller (ADF) test is used for this purpose.

Implementing Augmented Dickey–Fuller (ADF) test to check the stationarity of data

Since p-value(0.991512) is greater than the significance level(0.05), so the data is non-stationary. Now, let's difference the series and see how the autocorrelation plot looks like.

Checking Stationarity with ADF Test ADF Test:

The notebook applies the Augmented Dickey-Fuller (ADF) test to check if the dataset is stationary. A stationary series is crucial for effective time series modeling. Visualization:

ADF Test Results: Display results to understand if differencing or transformation is needed.

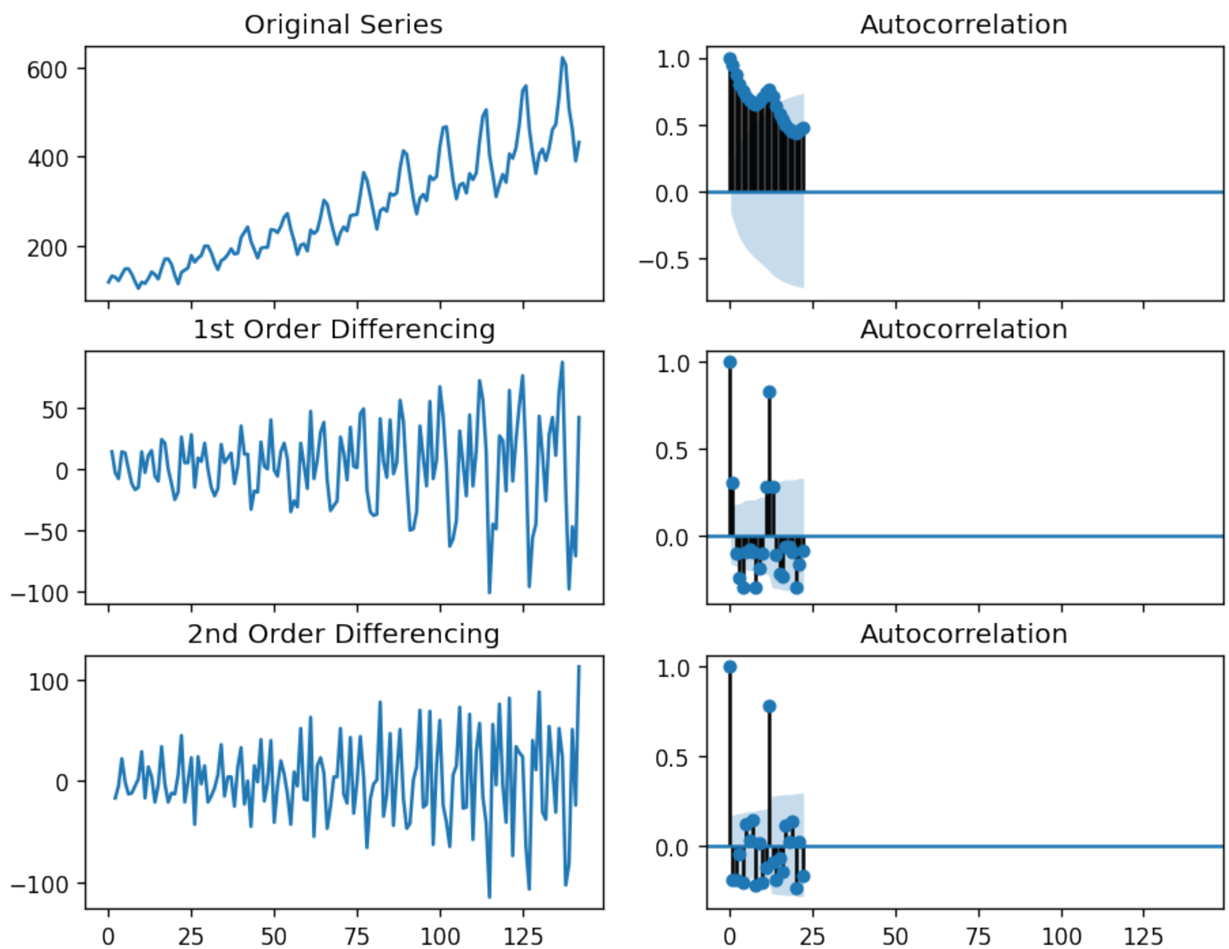


Figure 1: For the above data, we can see that the time series reaches stationarity with two orders of differencing

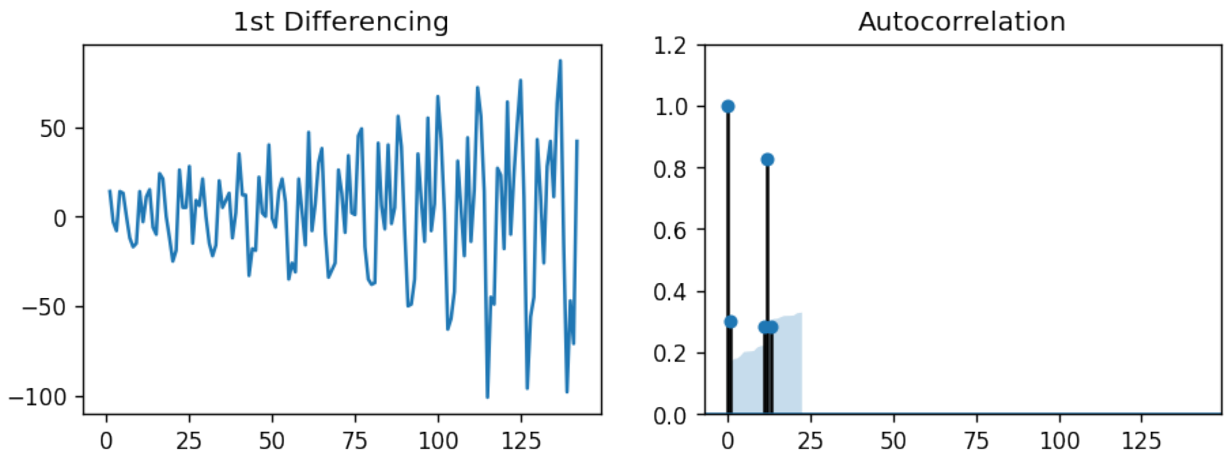


Figure 2: ADF Test Results

5 Time Series Visualization and Differencing

To visualize the time series, we plot the original data, and then apply first and second order differencing to make the series stationary.

Plotting the Series:

Original time series data is plotted. First and second order differencing is applied to make the series stationary, which is essential for models like ARIMA.

Visualization:

Original vs Differenced Series: The notebook generates multiple plots showing the original data, and its first and second differenced versions, along with ACF (Autocorrelation Function) plots to analyze the lags and identify patterns

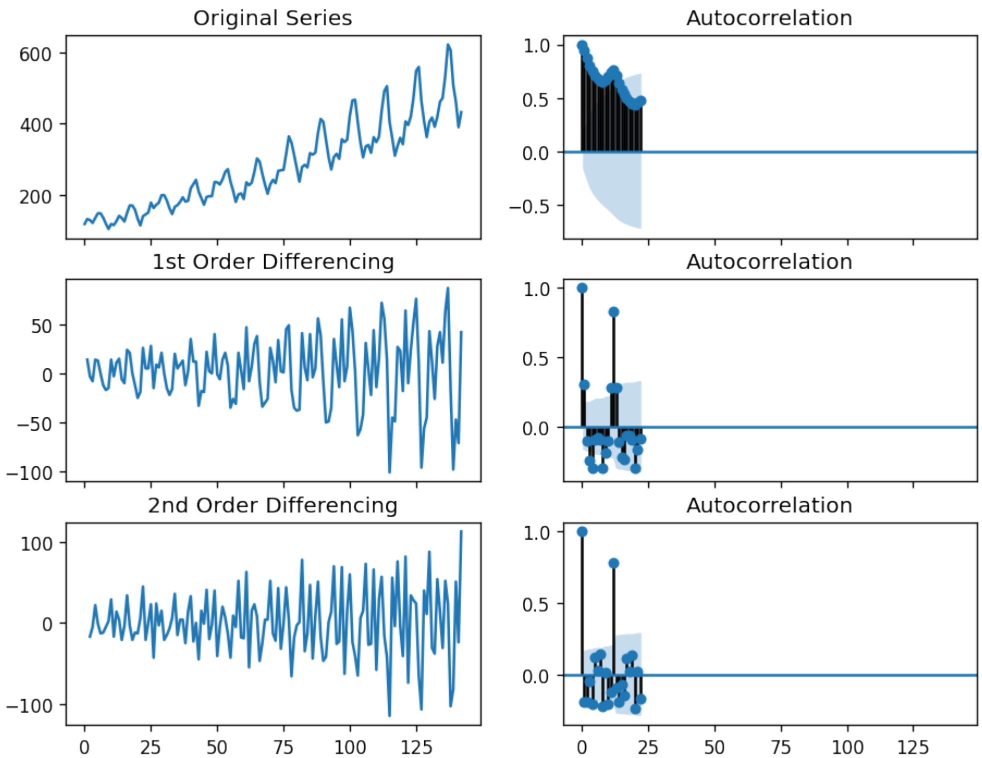


Figure 3: ime Series Visualization and Differencing

6 Modeling Techniques

The dataset is then modeled using various forecasting techniques including ARIMA, SARIMA, PROPHET, and XGBoost. Each model is trained on the data and predictions are generated.

ARIMA, SARIMA, PROPHET, XGBoost:

The notebook likely delves into applying these models sequentially. Each model will require fitting the data, predicting future values, and evaluating the results.

6.1 Visualization:

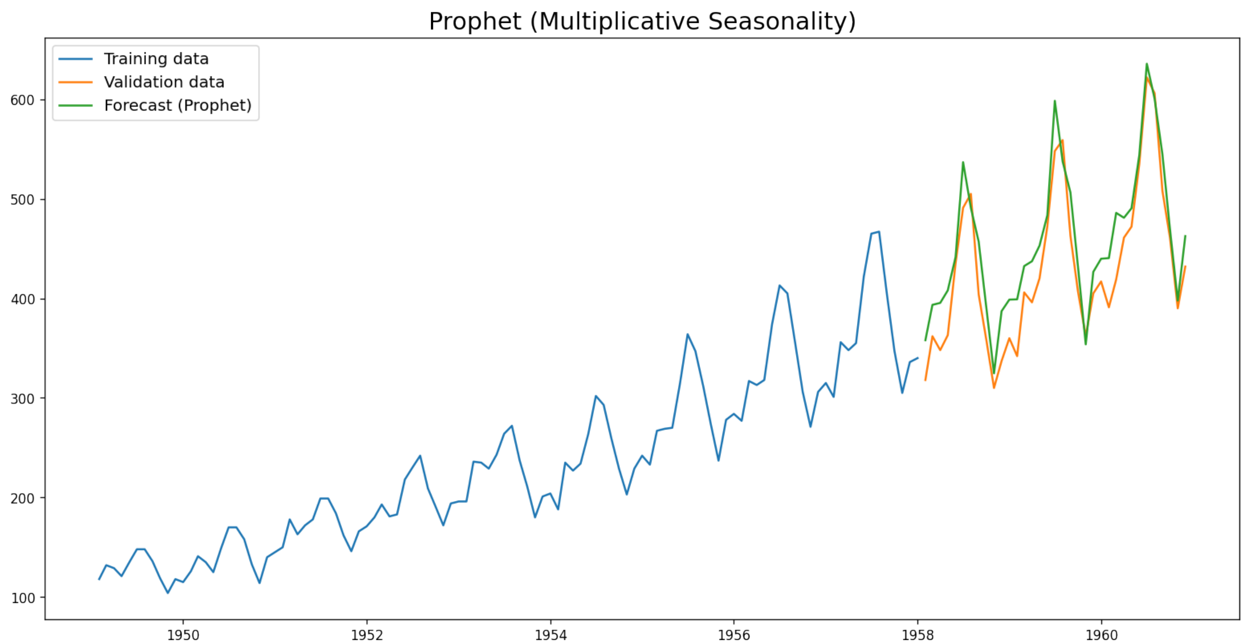
Model Fitting Forecasting: Visualize model predictions versus actual data to assess accuracy.

6.2 Model Comparisons:

Use plots to compare the effectiveness of each model in forecasting future sales.

7 Evaluation

The performance of each model is evaluated using metrics like Root Mean Squared Error (RMSE). The results are compared to determine the best model for forecasting airline passengers' ticket sales.



8 Conclusion

The findings of the various models suggest the most accurate model for forecasting future airline passenger ticket sales. The performance of each model is discussed, along with possible areas for improvement.

Summary: The notebook concludes with a discussion of which model performed best and provides recommendations for future forecasting or model improvements.

Visualization: Final Predictions: A final plot showing the best model's predictions alongside actual data