Time Series Moving ARIMA

Business Objective

A time series is simply a series of data points ordered in time. In a time-series, time is often the independent variable, and the goal is usually to make a forecast for the future.

Time series data can be helpful for many applications in day-to-day activities like:

- Tracking daily, hourly, or weekly weather data
- Monitoring changes in application performance
- Medical devices to visualize vitals in real-time

Auto-Regressive Integrated Moving Average (ARIMA) model is one of the more popular and widely used statistical methods for time-series forecasting. ARIMA is an acronym that stands for Auto-Regressive Integrated Moving Average. It is a class of statistical algorithms that captures the standard temporal dependencies unique to time-series data.

The model is used to understand past data or predict future data in a series. It's used when a metric is recorded in regular intervals, from fractions of a second to daily, weekly or monthly periods.

ARIMAX (Auto-Regressive Integrated Moving Average Exogenous) is an extension of the ARIMA model, and similarly, SARIMAX (Seasonal Auto-Regressive Integrated Moving Average with Exogenous factors) is also an updated version of the ARIMA model. We will see how to implement these two models as well.

We have already covered the concepts of Autoregression modelling (Time Series Project to Build an Autoregressive Model in Python) and Moving Average Smoothing techniques (Build a Moving Average Time Series Forecasting Model in Python)

In this project, we will be implementing the ARIMA model on the given dataset.

Data Description

The dataset is "Call centers" data. This data is at month level wherein the calls are segregated at domain level as the call centre operates for various domains. There are also external regressors like no of channels and no of phone lines which essentially indicate the traffic prediction of the inhouse analyst and the resources available.
The total number of rows are 132 and number of columns are 8:

- Month, healthcare, telecom, banking, technology, insurance, no of phonelines and no of channels.

Aim

This project aims to build an ARIMA model on the given dataset.

Tech stack

- Language - Python
- Libraries - pandas, numpy, matplotlib, seaborn, statsmodels, scipy

Approach

1. Import the required libraries and read the dataset
2. Perform descriptive analysis
3. Exploratory Data Analysis (EDA) -
    - Data Visualization
4. Check for white noise
5. Check for Random Walk
6. Perform Stationarity tests
    - Augmented Dickey-Fuller test
    - KPSS test
7. Seasonal decomposition plot
8. Holt Winter Exponential Smoothing
    - Create and fit the model
    - Make predictions on the model
    - Plot the results
9. ARIMA model
    - Create models with varying lag values
    - Compare these models using log-likelihood and AIC values
    - Check with the LLR test
    - ACF Plots of residuals
10. ARIMAX model
    - Create a model
    - ACF plots of residuals
11. SARIMAX model
    - Create a model
    - ACF plots of residuals

Modular code overview

```
input
  |_CallCenterData.xlsx


src
  |_Engine.py
  |_ML_Pipeline
            |_ARIMA.py
            |_RandomWalk.py
            |_Seasonality.py
            |_Stationarity.py
            |_WhiteNoise.py
            |_WinterHolt.py

lib
  |_ARIMA_.ipynb


output
  |_Visualization plots(.png)
```

Once you unzip the modular_code.zip file, you can find the following folders within it.

1. input
2. src
3. output
4. lib

1. Input folder **-** It contains all the data that we have for analysis. The following csv is used.
    - CallCenterData.xlsx

2. Src folder - This is the most important folder of the project. This folder contains all the modularized code for all the above steps in a modularized manner. This folder consists of:
    - Engine.py
    - ML_Pipeline
      The ML_pipeline is a folder that contains all the functions put into different python files which are appropriately named. These python functions are then called inside the engine.py file.

3. Output folder **-** The output folder contains all the visualization graphs. There are around 15 different plots.

4. Lib folder **-** This is a reference folder. It contains the original ipython notebook that we saw in the videos. The ppt used during the videos is also present here.

Project Takeaways

1. Introduction to Time series
2. Understand the basics of time series
3. Importing the dataset and required libraries
4. Exploratory Data Analysis (EDA)
5. White Noise detection
6. Random Walk detection
7. Stationarity test
8. Seasonality plot
9. Holt Winter Exponential Smoothing model
10. ARIMA model
11. ACF plots
12. Log-likelihood and AIC test
13. ARIMAX model
14. SARIMAX model