Time Series Auto Regression model

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 in 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

Autoregression (AR) modelling is one of the techniques used for time-series analysis**.** Autoregression is a time series model that uses observations from previous time steps as input to a regression equation to predict the value at the next time step.

In this project, we will build an autoregression model and also check different parameters such as presence of white noise, stationarity of the data, seasonality, etc.

Data Description

The dataset attached in the following CSV showcases the readings of 3 sensors of a chiller. The file contains data from one chiller, and the sensors give out one value at every hour of the day.

The data has 1895 rows and 5 columns. Following are the variables:

- Time: At what time the reading was taken (timestamp)
- IOT_Sensor_Reading: The reading of the sensor at the above-mentioned timestamp
- Error_Present: The error which may or may not be present while taking the reading
- Sensor 2: The reading from the subordinate sensor
- Sensor_Value: The final value to be predicted

Aim

The aim of this project is to build an auto regression model on the given dataset.

Tech stack

- Language - Python
- Libraries - pandas, numpy, matplotlib, scipy.stats, pylab, statsmodels, seaborn

Approach

1. Import the required libraries and read the dataset.
2. Perform descriptive analysis
3. Exploratory Data Analysis (EDA) -
    - Data Visualization (Q-Q plot)
4. Pre-processing
    - Convert date from string format to date format
    - Set time as the index column
    - Setting desired frequency
    - Handle missing data using forward filling, backward filling, and mean filling.
5. Check for white noise
6. Create a random walk model
7. Perform Stationarity tests
    - Augmented Dickey-Fuller test
    - KPSS test
8. Seasonal decomposition plot
9. Plot an Autocorrelation plot (ACF)
10. Plot a Partial Autocorrelation plot (PACF)
11. Perform Autoregression Modelling (ARMA)
12. Log Likelihood test
13. Calculating Rolling window
14. Calculating Expanding window

Modular code overview

```
input
  |_Data-Chillers.csv


src
  |_Engine.py
  |_ML_Pipeline
            |_AcfAndPacf.py
            |_Autoregressor.py
            |_Preprocess.py
            |_RandomWalk.py
            |_RollingWindow.py
            |_Stationarity.py
            |_WhiteNoise.py


lib
  |_TimeSeries_AutoRegressor_RollingWindow.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.
    - Data-chillers.csv

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 20 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. Performing basic Exploratory Data Analysis (EDA)
5. Plot Q-Q plots
6. Data pre-processing
7. Missing data handling using various techniques
8. White Noise testing
9. Random Walk model
10. Stationarity test
11. Seasonality plot
12. ACF and PACF plots
13. Autoregression model
14. Log likelihood test
15. Rolling window calculation
16. Expanding window calculation