Build a Autoregressive and Moving Average Time Series Model Project Overview

Overview

In this beginner-friendly project, we will delve into the world of time series analysis, focusing specifically on forecasting sensor data. Sensor data plays a crucial role in various fields, including IoT, environmental monitoring, manufacturing, and more. Accurately predicting future sensor readings is essential for proactive decision-making and optimizing system performance.

We will begin by gaining a solid understanding of time series analysis and its significance in uncovering patterns and trends in temporal data. We will explore different types of time series, including continuous and discrete, and delve into the components that make up a time series, such as trend, seasonality, and irregularity.

To ensure reliable analysis, we will discuss the concept of stationarity and learn how to test for it using techniques like the Augmented Dickey-Fuller (ADF) and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) tests. Additionally, we will utilize autocorrelation and partial autocorrelation functions (ACF and PACF) to identify dependencies and correlations in the sensor data.

Building upon this foundation, we will develop and apply moving average (MA) and autoregressive (AR) models specifically tailored for forecasting sensor data. These models will enable us to make accurate predictions and anticipate future readings based on historical patterns.

By the end of this project, you will have gained valuable skills in analyzing sensor data using time series analysis techniques and forecasting future values. This knowledge will empower you to make informed decisions, optimize system performance, and harness the power of predictive analytics in the context of sensor data.

Aim

The aim of this project is to provide beginners with a comprehensive introduction to time series analysis and forecasting, specifically focused on sensor data. By exploring basic concepts, developing MA and autoregressive models, and applying them to sensor data, participants will gain practical skills in analyzing and predicting future readings, enabling informed decision-making and optimization in various domains.

Data Description
The dataset used in this project consists of IoT sensor data from a chiller. It contains the following columns:

- Time: This column represents the timestamp at which the sensor reading was taken. It provides the temporal information for each data point.

- IOT_Sensor_Reading: This column represents the reading obtained from the primary sensor at the corresponding timestamp. It captures the specific measurement recorded by the sensor.

- Error_Present: This column indicates whether an error was present while taking the sensor reading. It is a binary variable that can have a value of 1 if an error is present, or 0 if no error is detected.

- Sensor 2: This column represents the reading obtained from a subordinate sensor at the same timestamp. It provides an additional measurement taken by a different sensor, potentially capturing complementary information.

- Sensor_Value: This column represents the final value to be predicted. It may be derived from the primary sensor reading, considering factors such as error correction, calibration, or additional computations.

Tech Stack
- ➔ Language: Python
- ➔ Libraries: pandas, numpy, matplotlib, scipy.stats, statsmodels, seaborn

Approach
- Read the data.
- Preprocess the data.
- Perform Exploratory Data Analysis (EDA).
- Check for stationarity in the data.
- Analyze ACF and PACF plots.
- Build the following models:
    - Moving average (MA).
    - First order autoregressive (AR).
    - Second/general order autoregressive (AR).

- Third order autoregressive (AR).
- Fourth order autoregressive (AR).

- Evaluate the models' performance.

Modular code overview:

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

```
├─ input
│  └─ Data-Chillers.csv
├─ lib
│  ├─ reference
│  │  └─ ppt_autoregressor.pptx
│  └─ TS workbook.ipynb
├─ output
├─ Readme.md
├─ requirements.txt
└─ src
   ├─ Engine.py
   └─ ML_Pipeline
      ├─ acf.py
      ├─ autocovariance.py
      ├─ Autoregressor.py
      ├─ EDA.py
      ├─ MA_model.py
      ├─ pacf.py
      ├─ Preprocess.py
      ├─ RandomWalk.py
      ├─ Stationarity.py
      ├─ WhiteNoise.py
```

- input: This folder contains the input data file "Data-Chillers.csv" which is used as the dataset for analysis.

- lib: This folder includes the following subfolders and files:

    - reference: Contains a PowerPoint presentation file "ppt_autoregressor.pptx" serving as a reference material for autoregressive modeling.
    - TS workbook.ipynb: A Jupyter Notebook file that serves as a workbook for the time series analysis project, containing code and documentation.

- output: This folder is intended to store any output files or results generated during the project. It may include prediction outputs, visualizations, or other relevant files.

- Readme.md: A Markdown file that provides instructions, explanations, or important information about the project and its components.

- requirements.txt: A text file specifying the required dependencies and packages needed to run the project code.

- src: This folder contains the source code files for the project, organized in the following subfolders:
    - Engine.py: The main engine file responsible for executing the time series analysis pipeline and coordinating different components.
    - ML_Pipeline: A subfolder that houses various modules for different steps in the machine learning pipeline.

Project Takeaways

1. Understanding the importance of time series analysis in analyzing and forecasting data collected over time.
2. Knowledge of different types of time series data, such as continuous and discrete.
3. Awareness of the components of a time series, including trend, seasonality, and irregularity.
4. Ability to assess stationarity in time series data using statistical tests like ADF and KPSS.
5. Interpretation of ACF and PACF plots to identify autoregressive and moving average patterns.
6. Familiarity with the concept of a random walk model as a baseline for time series forecasting.

7. Hands-on experience in preprocessing time series data, handling missing values, outliers, and inconsistencies.
8. Exploratory Data Analysis (EDA) skills to gain insights into the distribution, trends, and anomalies in the data.
9. Building moving average (MA) models for forecasting time series data.
10. Developing autoregressive (AR) models of different orders for time series forecasting.
11. Evaluating model performance using metrics like Root Mean Squared Error (RMSE).