PyDS PROBLEM STATEMENTS | HUBBLEMIND

Dataset Description:

The dataset used for this project is focused on energy consumption in smart homes. It includes hourly data for various smart homes, with features such as:

- ➢ **Energy_Consumption_kWh**: The total energy consumed (in kilowatt-hours).
- ➢ **Temperature_C**: The temperature in degrees Celsius.
- ➢ **Humidity_%**: The relative humidity percentage.
- ➢ **HVAC_Usage_kWh**: The energy consumption of the HVAC (Heating, Ventilation, and Air Conditioning) system.
- ➢ **Kitchen_Usage_kWh**: The energy consumption in the kitchen.
- ➢ **Electronics_Usage_kWh**: The energy consumption of electronic devices.
- ➢ **Occupancy**: The number of occupants in the home.
- ➢ **Weather_Conditions**: Categorical data describing the weather (e.g., Sunny, Rainy).
- ➢ **City**: The name of the Indian city where the home is located.

The dataset provides a rich set of features to analyze patterns in energy consumption, understand the impact of environmental factors, and build predictive models.

Dataset Link : Download

Week 1: Data Cleaning

Focus: Understanding and preparing the dataset for analysis.

Task 1: Data Importing and Initial Exploration

- ➢ Import the dataset using Pandas.
- ➢ Explore the dataset by checking the structure, data types, and basic statistics.
- ➢ Identify any missing values, outliers, or inconsistencies.

Task 2: Handling Missing Data

- Implement strategies to handle missing data (e.g., imputation, removal).
- Document the decisions made during the data cleaning process.

Task 3: Outlier Detection and Handling

- Identify outliers in the energy consumption and temperature columns.
- Apply techniques to handle outliers (e.g., capping, transformation).

Task 4: Time-Series Consistency

- Check for any inconsistencies in the time-series data (e.g., duplicate timestamps).
- Correct any issues found to ensure smooth time-series analysis.

Task 5: Data Normalization

- Normalize or standardize relevant features (e.g., energy consumption, appliance usage) to prepare them for machine learning.

Week 2: Exploratory Data Analysis (EDA) & Visualization

Focus: Gaining insights and understanding relationships within the data.

Task 1: Univariate Analysis

- Perform univariate analysis on key features like energy consumption, temperature, and appliance usage.
- Visualize the distribution of these features using histograms and box plots.

Task 2: Bivariate and Multivariate Analysis

- Explore relationships between energy consumption and other variables (e.g., occupancy, weather conditions).
- Use scatter plots, pair plots, and correlation heatmaps for visualization.

Task 3: Time-Series Analysis

- Analyze energy consumption trends over time (e.g., daily, weekly patterns).
- Identify any seasonality or trends using line plots and moving averages.

Task 4: Feature Engineering

- Create new features based on existing data (e.g., energy consumption per occupant, temperature difference).
- Evaluate the importance of these features using correlation analysis and visualization.

Task 5: Advanced Visualizations

- Create more advanced visualizations like joint plots, violin plots, and interactive plots (using libraries like Plotly).
- Summarize the findings from EDA in a well-documented format.

Week 3: Machine Learning

Focus: Building and evaluating a predictive model.

Task 1: Data Splitting

- Split the dataset into training and testing sets (e.g., 80/20 split).
- Ensure that the time-series nature of the data is respected during splitting.

Task 2: Model Selection and Training

- Implement a simple regression model (e.g., Linear Regression) to predict energy consumption.
- Train the model on the training data and evaluate initial performance.

Task 3: Model Evaluation

- Evaluate the model using appropriate metrics (e.g., Mean Absolute Error, R-squared).

- Analyze residuals and check for any patterns or biases.

Task 4: Feature Importance and Interpretation

- Identify the most important features contributing to the model's predictions.
- Use techniques like coefficient analysis or feature importance plots to explain the model's behavior.

Task 5: Predictive System and Testing

- Build a simple predictive system that uses the trained model to make predictions on the test set.
- Compare predictions with actual values and visualize the results using scatter plots or residual plots.

Week 4 : Documentation and Submission

Focus: Summarizing the project, documenting code, and submitting your work.

- **Write a Project Summary:** Cover the introduction, challenges, solutions, roadblocks, and conclusion.
- **Compile Code:** Gather all code from Weeks 1 to 3 into a single Jupyter Notebook, ensuring explanations are provided for each code block.
- **Upload to GitHub:** Upload the Jupyter Notebook to a GitHub repository and include a `README.md` file that summarizes the project.
- **Create a Google Doc:** Include the project summary and the GitHub repository link.
- **Submit the Google Doc:** Submit the document via the provided **Google Form**.