Documentation

This project is a **data visualization and analysis program** that reads wind speed data from multiple CSV files, processes it, and offers a simple command-line interface to the user for visualizing and analyzing the data. The project is divided into two files: `main.py` and `methods.py`.

File 1: `main.py`

This is the primary script where data is loaded, processed, and user interaction is handled. It presents the user with a menu and calls appropriate functions for different tasks.

1. Data Loading:

- Wind speed data from six CSV files (spanning from 2007 to 2012) is read using **Pandas**.
- The rows are concatenated into a single DataFrame using **pd.concat()**.
- A `Datetime` column is generated by combining year, month, day, hour, and minute columns using **pd.to_datetime()**.

2. User Interface:

- The script provides a command-line interface with four options:
    - View a histogram of daily mean wind speed with a Weibull distribution curve.
    - View a regression model for predicting wind speeds over a specified number of years.
    - Display the list of libraries used in the project.
    - Exit the program.

3. Functionality:

- The script handles the user input, calling functions from the `methods.py` file based on the user's selection.
- It resamples the wind speed data into daily averages for histogram plotting.
- For regression modeling, it extracts monthly average wind speeds and fits a linear regression model, allowing the user to predict future values.

File 2: `methods.py`

This file contains the core functions used for data visualization and analysis, including plotting histograms, fitting regression models, and calculating statistical metrics.

1. `plot_histogram(wind_speeds)`

- This function takes wind speed data and generates a histogram.
- It fits a **Weibull distribution** to the wind speed data and overlays the Weibull probability density function (PDF) on the histogram.
- The histogram shows the frequency of wind speeds, while the Weibull curve provides a distribution fit for the data.
- Libraries used: **Matplotlib**, **Scipy**, **NumPy**.

2. `fit_and_plot_regression(X, y)`

- This function fits a **linear regression model** to predict future wind speeds.
- It prompts the user to specify the number of years into the future to predict (2-15 years).
- The function plots the historical wind speeds as scatter points and overlays the linear regression line, which includes predictions for future years.
- It also calculates the **Mean Squared Error (MSE)** and the **R^2 score** for the regression model.
- Libraries used: **Scikit-learn**, **Matplotlib**, **NumPy**.

3. `r2_manualCalc(y_true, y_pred)`

- This function manually calculates the **R^2 score** for the regression model.
- It computes the residual sum of squares and total sum of squares to determine how well the regression line fits the data.

4. `t_squared_test_manual(X, y, y_pred)`

- This function performs a manual **t-squared test** for the regression model.
- It calculates the residual sum of squares, total sum of squares, and the **t-statistic** for the regression slope.
- The **standard error** of the slope is computed, and the function returns the t-squared value.

Total Libraries Used

- **Pandas**: For reading CSV files, handling data frames, and manipulating data.
- **Matplotlib**: For generating visualizations (histograms, regression plots).
- **Scikit-learn**: For building and evaluating the linear regression model.
- **NumPy**: For numerical calculations.
- **Scipy.stats**: For fitting the Weibull distribution and calculating probability density functions.

Program Flow

1. The program begins by loading data from the CSV files.
2. The user is presented with a menu of options.
3. Depending on the user’s choice:
    - A histogram of wind speed data is plotted with a Weibull distribution fit.
    - A regression model is built, and future wind speeds are predicted.
    - A list of libraries used in the project is displayed.
4. The program exits when the user selects the "Exit" option.