

Set 1

The annual rainfall data (in millimeters) for 6 regions is recorded monthly and stored in a CSV file named `rainfall_data.csv`:

1. Write a Python script to load the data from `rainfall_data.csv` and perform polynomial curve fitting for **Region_C** using polynomial orders 2, 3, and 4.
2. Using the fitted models, predict the rainfall for Region_C for the next three months (January, February, and March of the following year).
3. Create a user interface for the users to select which order they want to display in the graph. Example; order 1 or order 2 or order 3 or order 1,2 and 3.
4. Plot the original data and overlay the polynomial fitting curves on the same graph based on user selection.

Using the best-fit polynomial model from above, compute the expected rainfall for Region_C for the first, second, and third months of the following year. Print the results in a tabular format.

Set 2

A retail chain tracks the monthly profits (in \$1000s) of 4 branches over a year, stored in a CSV file named `profits.csv`:

1. Write a Python script to load the data from `profits.csv` and perform polynomial curve fitting for **Branch_Z** using polynomial orders 1, 3, and 4.
2. Using the fitted models, predict the profits for Branch_Z for the next three months (January, February, and March of the following year).
3. Create a user interface for the users to select which order they want to display in the graph. Example; order 1 or order 2 or order 3 or order 1,2 and 3.
4. Plot the original data and overlay the polynomial fitting curves on the same graph based on user selection.

Using the best-fit polynomial model from above, calculate the expected profits for Branch_Z for the first, second, and third months of the following year.

Set 3

A city measures the average monthly air quality index (AQI) over the past year. The data is stored in a CSV file named `air_quality.csv`:

1. Write a Python script to load the data from `air_quality.csv` and perform polynomial curve fitting for the AQI using polynomial orders 1, 2, and 4.
2. Using the fitted models, predict the AQI for the next three months (January, February, and March of the following year).
3. Create a user interface for the users to select which order they want to display in the graph. Example; order 1 or order 2 or order 3 or order 1,2 and 3.
4. Plot the original data and overlay the polynomial fitting curves on the same graph based on user selection.

Using the best-fit polynomial model from above, calculate and display the AQI for the first three months of the next year in a Python console table.