Week 2 Tasks
Deadline 24-Jan-2025

NumPy Tasks:

1. Create a NumPy array of zeros with shape (3, 4).
2. Generate an array of 20 random numbers as integers ranging between 1 and 10. Reshape the array into a 4x5 matrix.
3. Create a 5x5 identity matrix and extract the diagonal elements into a 1D array.
4. Create matrix and access rows and columns
    - create a 4x5 array of even numbers: 10, 12, 14, …
    - extract third column
    - set the fourth row to 1,2,3,4,5
5. consider two vectors

    names = np.array(["Roxana", "Statira", "Roxana", "Statira", "Roxana"])

    score = np.array([126, 115, 130, 141, 132])

    Do the following using a single one-line vectorized operation.

    - Extract all test scores that are smaller than 130
    - Extract all test scores by Statira
    - Add 10 points to Roxana's scores. (You need to extract it first.)

Pandas Tasks:

Dataset link: https://github.com/jay-D-Deshmukh/ML-Linear-Regression/blob/main/CarPrice_project.csv

1. Read the dataset into a pandas DataFrame. Display the first 5 rows of the DataFrame.
2. Slice the DataFrame to create a new DataFrame containing only the columns CarName, fueltype, carbody, and price. Display the first 10 rows of the new DataFrame.
3. Filter the DataFrame to include only cars with fueltype as 'gas'. Display the first 20 rows of the filtered DataFrame.
4. Slice the DataFrame to include only cars with carlength greater than 180 and fueltype as 'diesel'. Display the first 5 rows of the sliced DataFrame.
5. Count the number of cars for each fueltype. Display the result.
6. Create a new column car_type which represents the size of the car based on the carlength.

    Hint: Use the .apply() method.

    - If carlength is below 150, consider it as short_length_cars.
    - If it is between 150 to 170, consider it as mid_length_cars.
    - If it is greater than 170, consider it as long_length_cars.

    - Apply this function to the DataFrame to create the column car_type. Display the count of each car_type.
7. Remove the column symboling from the DataFrame as an inplace operation. Display the first 5 rows of the DataFrame after removal.
8. Group by carbody and calculate the mean price for each group.
9. Create a series of 4 capital cities where the index is the name of corresponding country.
10. Create a dataframe of (at least 4) countries, with 2 variables: population and capital. Country name should be the index.

Matplotlib and Seaborn Tasks:

1. Plot y=sin(x) and y=cos(x) on the same graph for x in [0,2π].
2. Create a heatmap to show the correlation matrix of numerical columns in the CarPrice_project.csv dataset.
3. Create a combination plot:
    - Use a bar plot to show the average price for each carbody.
    - Overlay a line plot to show the count of cars for each carbody
4. Visualize the relationship between **horsepower**, **enginesize**, and **price**.
5. Which column has the highest variance? Why might this be?
6. What insights can you draw by comparing the max and min values of the columns?
7. How does the variance of price compare to that of enginesize and horsepower?
8. Create a bar plot showing the maximum and minimum values for each numerical column in the dataset.
9. Create a pie chart showing the proportion of each carbody and drivewheel type.
    - Which carbody type dominates the dataset?
    - Which drivewheel type is the least common?