Assignment: Data Analysis using Calgary Emergency Shelters Dataset

Instructions:

1. Download the dataset provided in the repo.
2. Complete all tasks using Python, Pandas, NumPy, and Matplotlib for analysis and visualization. Submit your code and results as a Jupyter Notebook.

Objective: This assignment is designed to challenge your ability to perform data analysis and visualization on a real-world dataset. You will handle missing data, perform group operations, use joins, and create insightful visualizations to uncover trends and patterns.

Part 1: Data Preparation and Cleaning

1. **Dataset Overview**:
    - Load the dataset and display its shape, columns, and data types.
    - Identify and handle missing data:
        - Calculate the percentage of missing data for each column.
        - Drop columns or rows with more than 40% missing values.
        - Impute missing values in numerical columns using the median and in categorical columns using the mode.
2. **Date and Time Handling**:
    - Convert the `Date` column to a datetime object.
    - Create new columns for the year, month, and day of the week.
    - Add a column indicating whether each date falls on a weekend.

Part 2: Data Transformation and Analysis

1. **Group and Aggregate**:
    - Group the data by `ShelterType` and `MONTH`, and calculate the following for each group:
        - Average `Capacity`.
        - Total `Overnight` occupancy.
        - Standard deviation of `Overnight`.
    - Identify the `ShelterType` with the highest average occupancy for each month.
2. **Joining Data**:
    - Create a summary DataFrame with the total annual `Overnight` occupancy for each shelter.
    - Merge this summary with the main dataset to add a column for `AnnualOccupancy`.
    - Use this merged dataset to calculate each shelter's contribution percentage to its organization's total occupancy.

3. **Handling Outliers**:
    - Use the IQR method to identify outliers in the `Overnight` column.
    - Replace outliers with the median value of their respective `ShelterType` group.

Part 3: Visualization

1. **Time-Series Visualization**:
    - Plot the monthly average `Overnight` occupancy over time for each `ShelterType`.
    - Highlight trends and identify peaks in the occupancy.
2. **Heatmap**:
    - Create a heatmap showing the average occupancy by `ShelterType` and `MONTH`.
3. **Bar Charts**:
    - Plot the total occupancy for each organization as a bar chart.
    - Add annotations to indicate the exact totals for each organization.
4. **Box Plot**:
    - Create a box plot for `Overnight` occupancy grouped by `ShelterType`.
    - Highlight any noticeable patterns or anomalies.

Part 4:

1. **Custom Functions**:
    - Write a custom function to dynamically impute missing values based on grouping criteria (e.g., median capacity by `ShelterType`).
2. **Correlation and Hypothesis Testing**:
    - Calculate the correlation between `Capacity` and `Overnight`.
    - Perform a hypothesis test to determine if the average occupancy differs significantly between two selected shelter types.
3. ** Joins**:
    - Create a DataFrame summarizing average occupancy for each organization.
    - Perform an inner join with the main dataset to analyze trends specific to the top 3 organizations.

Part 5: Reporting and Insights

1. Summarize your findings in a concise report (300-500 words), including:
    - Key trends in occupancy and capacity utilization.
    - Seasonal and organizational patterns.
    - Recommendations for improving resource allocation and addressing underutilization.
2. Include visualizations that support your insights:

- Line plots, heatmaps, and bar charts.
- Ensure all plots have titles, labels, and legends.

Bonus Tasks (Optional)

1. Analyze the impact of weekend vs. weekday occupancy on shelter utilization.
2. Develop a simple regression model using NumPy to predict `Overnight` occupancy based on `Capacity`, `ShelterType`, and `MONTH`.
3. Use Matplotlib to create a subplot grid showcasing trends for each `ShelterType` over the years.

Submission Requirements:

- Jupyter Notebook (.ipynb) file containing your code and outputs.
- PDF or Markdown report summarizing your findings.
- Include detailed comments in your code to explain your logic.

Evaluation Criteria:

- Completeness: Did you address all parts of the assignment?
- Accuracy: Are the results and insights correct?
- Code Quality: Is your code readable, efficient, and well-documented?
- Visualization: Are your plots clear, relevant, and informative?
- Creativity: Did you provide unique insights or go beyond the minimum requirements?

Good luck!