

COMP 1844 - Week 10 Report

**Student Name:**

**Student ID:**

**Detailed Analysis and Visualization of Population Data Using Python**

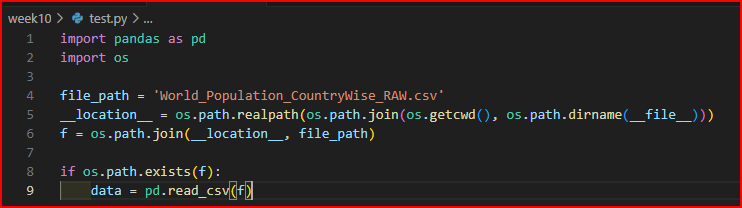
Data analysis and visualization are indispensable in understanding large datasets, especially in fields like demographics. This report delves into two Python scripts, ‘group\_work.py’ and ‘group\_work\_2.py,’ which perform data preprocessing, statistical analysis, and visualization on a CSV file containing country-wise population data from 1960 to 2023. By examining these scripts, we gain insights into their methodology, functionality, and the significance of the results they produce.

**Introduction to the Scripts**

Both group\_work.py and group\_work\_2.py are designed to analyze a CSV file named World\_Population\_CountryWise\_RAW.csv. They employ several popular Python libraries, including pandas for data manipulation, matplotlib and seaborn for plotting, and os for file path operations. These scripts illustrate a comprehensive approach to handling and visualizing large datasets.

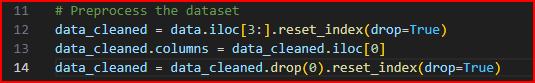
**Data Loading and Preprocessing**

The first step in both scripts is loading the data from the CSV file. This is accomplished using the pandas library, which is adept at handling large datasets. The scripts construct the file path dynamically to ensure compatibility across different environments:

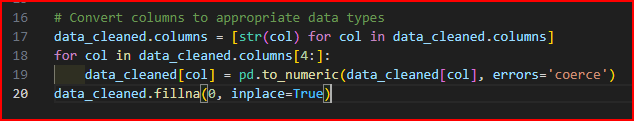


**Data Preprocessing**

The dataset requires several preprocessing steps to ensure it is suitable for analysis. Initially, the scripts skip the first three rows, reset the index, and rename the columns appropriately. This process removes unnecessary metadata and sets up the dataset for further manipulation:



Next, the scripts convert population data columns (from the fifth column onward) to numeric values. Any non-numeric entries are coerced to NaN, which are subsequently filled with zeroes to handle missing data:

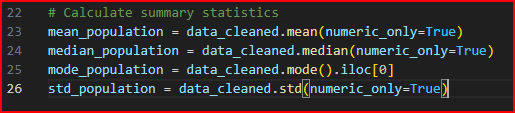


These preprocessing steps are crucial for ensuring that the data is clean and consistent, allowing for accurate analysis and visualization.

**Summary Statistics Calculation**

After preprocessing, both scripts proceed to calculate various summary statistics. These statistics include the mean, median, mode, and standard deviation of population figures for each year. Calculating these statistics provides a deeper understanding of the central tendencies and variability of the population data.

**Code for Summary Statistics Calculation**



**Mean Population**

The mean provides the average population for each year, offering a central value around which the data points are distributed.

**Median Population**

The median represents the middle value of the population data, which is less affected by outliers and skewed data than the mean.

**Mode Population**

The mode is the most frequently occurring population value for each year. It can provide insights into the most common population sizes.

**Standard Deviation**

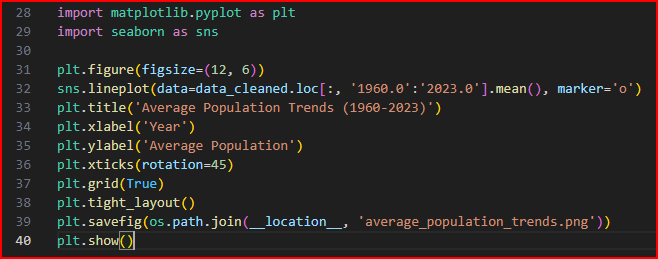
The standard deviation measures the dispersion of the population data from the mean. A higher standard deviation indicates greater variability in the data.

**Visualization**

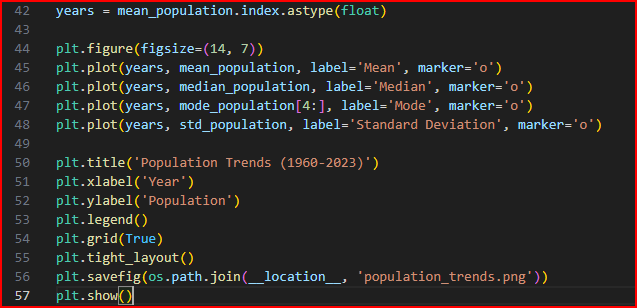
Visualization is a powerful tool for interpreting and communicating data. Both scripts utilize matplotlib and seaborn to create informative plots that illustrate population trends and distributions.

**Population Trends Over the Years**

group\_work.py generates a line plot to show the average population trends from 1960 to 2023. This plot highlights how the average population has changed over time:

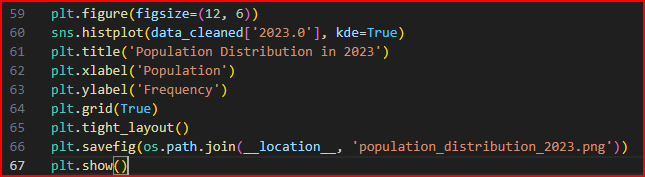


‘group\_work\_2.py’ expands on this by plotting not only the mean but also the median, mode, and standard deviation. This comprehensive plot provides a multi-faceted view of population trends:



**Population Distribution for 2023**

Both scripts also generate a histogram to show the distribution of population sizes for the year 2023. This histogram provides insights into the spread and concentration of population sizes across different countries:



**Saving the Preprocessed Dataset**

group\_work\_2.py includes an additional step to save the cleaned and preprocessed dataset to a new CSV file. This ensures that the processed data is preserved for future use without the need for repeated preprocessing:



**Conclusion**

The Python scripts group\_work.py and group\_work\_2.py effectively demonstrate the process of data analysis and visualization using pandas, matplotlib, and seaborn. They highlight the importance of data cleaning and preprocessing to ensure accurate analysis. By calculating summary statistics, the scripts provide valuable insights into population trends and distributions. The visualizations created by these scripts make complex data more comprehensible and accessible, aiding in the interpretation and communication of demographic trends.

These scripts can serve as a foundation for more advanced data exploration and analysis. They underscore the power of Python in handling large datasets and producing meaningful insights through statistical analysis and visualization. Whether for research, policy-making, or educational purposes, the techniques demonstrated in these scripts are essential for anyone working with demographic data.