CSE506: Introduction to Data Mining
Assignment 0

Instructions

- The assignment is to be attempted in pairs.
- Programming Language: Python
- For Plagiarism, institute policy will be followed
- You need to submit the readme.pdf, Code file
- You are allowed to use libraries such as pandas, matplotlib, etc.
- Mention methodology, assumptions, and results you may have in Readme.pdf.
    - This assignment has 0 weight but it is mandatory to do.

Requirements:

1. Mention all assumptions if any in the report.
2. Report and code in .py format should be submitted in the classroom in a zip folder with the name 'A0_RollNumber1_RollNumber2.zip'.
3. You can use any library for pre-processing and post-processing.
4. One member should submit on google classroom while other member can mark turn in without the attachement.
5. In case of doubts, please comment on the classroom.
6. You can refer to any source to map states and UTs to the acronyms. Please mention the states and UTs you have taken into account while computing your queries.
7. The data will have inconsistencies and outliers please handle them as per your understanding and mention them in the readme. You can use the "tt" tag which will provide the total counts of states/UTs.

Dataset: https://data.covid19india.org/states_daily.json
Total: 0

Q1. Data Manipulation

1. Count the total number of "Confirmed", "Recovered" and "Deceased" from 14-Mar-2020 to 16-Aug-2021 and report the numbers.
2. Count the total number of "Confirmed", "Recovered" and "Deceased" from 14-Mar-2020 to 16-Aug-2021 for each state: Delhi, Maharashtra, West Bengal and Tamil Nadu,
3. Report the top 10 states with the highest recovery rate and top 10 states with the lowest recovery rate from 14-Mar-2020 to 16-Aug-2021.
4. Report the top 3 highest affected states in terms of "Confirmed", "Recovered" and "Deceased" with the count from 14-Mar-2020 to 16-Aug-2021.
5. Report the top 3 lowest affected states in terms of "Confirmed", "Recovered" and "Deceased" with the count from 14-Mar-2020 to 16-Aug-2021.
6. Find the day and count with the highest spike in a day in the number of cases for each state and UTs for "Confirmed", "Recovered" and "Deceased" between dates 14-Mar-2020 and 16-Aug-2021.

7. Report active cases (Assume active = Confirmed - (Recovered + Deceased)) state wise for all individual states and UTs on date 15-Aug-2021 (This date only) starting from 14-March-2020.

Q2. Plotting

1. Plot the area trend line for total “Confirmed”, “Recovered” and “Deceased” cases from 14-Mar-2020 to 16-Aug-2021.
2. Plot the area trend line for total “Confirmed”, “Recovered” and “Deceased” cases for Delhi (dl) from 14-Mar-2020 to 16-Aug-2021.
3. Plot the area trend line for active cases. Assume active = Confirmed - (Recovered + Deceased) from 14-Mar-2020 to 16-Aug-2021.
4. Plot a bar plot of the number of active cases in Delhi, Tamil Nadu and Gujarat for any date range of your choice.

General FAQs:

1. You can use any library; pandas, NumPy matplotlib, seaborn etc.
2. You can sum counts obtained from the individual states and UTs or directly pick from the total tag “tt”.
3. You can convert JSON to any format CSV, pickle if needed.
4. Please paste the obtained graph in the report.
5. For Q1_7, you will have to cumulate the count to provide the active cases.
6. If a question mentions the count of states then please count states only. For other scenarios count states+UTs.
7. There will be certain count mismatches when using the “tt” tag and while summing up individual entities.
8. You can make your own assumptions for any question and mention them in your report.
9. You have been provided sources as hyperlinks wherever required, You can use the same or any other source as per your ease.
10. For plotting use different shades when plotting the area trend line. Do not stack the graph. Graph produced should be unstacked.
11. You can store the data locally, You do not need to fetch the JSON from the source directly.

Resources:

pandas - Python Data Analysis Library (pydata.org)
NumPy, Matplotlib: Python plotting — Matplotlib 3.4.3 documentation
blog/2020-06-15-hornbill.md at 7cf01da3d5de7b860ae3bfc0ec3638f2e72863a6 · covid19india/blog (github.com)
You can refer to the last tutorial for more information on EDA.