GROUP R

WORK BREAKDOWN STRUCTURE REPORT

DATASET 1

BY

x21226181 - Benjamin Ayodele Kelani

1. **NAME OF DATASET: "**Visits to Physician Offices, Hospital Outpatient Departments and Hospital Emergency Departments by Age, Sex and Race: United State."

2. **DESCRIPTION INFROMATION:** The Us Department of Health and Human Services made available the data on visit to physician offices, hospital emergency and outpatient departments by age sex and race for a selected population for recent years till date.

3. **SOURCE LOCATION:** DATA.GOV**:https://catalog.data.gov/dataset/visits-to-physician-offices-hospital-outpatient-departments-and-hospital-emergency-departm-6ef16.**

4. **RETRIEVAL METHOD:** Extensible Markup Language (XML) - Download: https: https://data.cdc.gov/api/views/xt86-xqxz/rows.xml?accessType=DOWNLOAD.

5. **ATTRIBUTES OF DATASETS:** It contains **3571** numbered rows and **16** columns.

6. **DATA PROCESSING PIPELINE**
    - Downloaded the XML file.
    - Converted XML files to dictionaries.
    - Loaded XML file to MongoDB server – Using Luigi.
    - Extracted the file from MongoDB server as a valid JSON string – Using Luigi.
    - Established a Postgres connection to the database.
    - Displayed Attributes of the data frame.
    - Perform Cleaning and Transformation.
    - Loaded the data frame to Postgres Database – Using Luigi.
    - Performed Visualizations on the data Element frame.

7. **MAIN COMPONETS**
    - Installation of Programming Environment
        - Python IDE – Jupyter Notebook
    - Installation of Databases
        - MongoDB Compass – NoSQL Database Server

    - Postgres Database Admin4 – SQL Database Server
- Installation of Necessary Commands
    - Pip Pymongo
    - Pip Luigi
- Importation of Necessary Packages and Libraries
    - Json – for handing Json file.
    - ElementTree – For handling XML file.
    - Luigi – For automation pipeline in python.
    - Pymongo, mongoclient – To allow interaction with the MongoDB database.
    - Pandas as pd – For manipulation and analysis
    - Matplotlib. pyplot as plt – To create visualizations.
    - Seaborn as sns – To create statistical data visualizations.

8. **CONTRIBUTION IN DETAIL**

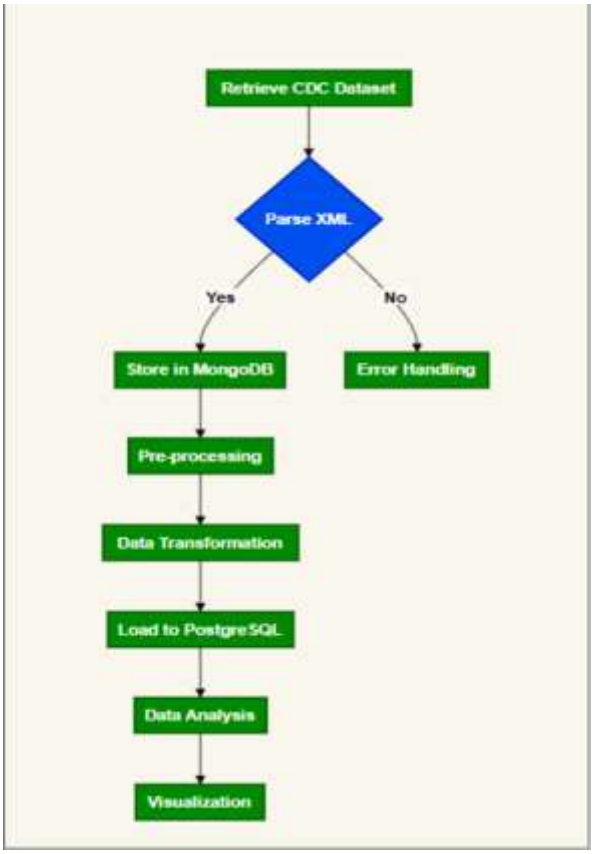
With the responsibility to work on the dataset dealing with the visits to physician offices, hospital outpatient departments and hospital emergency department by age, sex, and race in United State. A series of tasks were carried out to achieve a final quality- analyzed and visualized findings that were helpful to the project's main objective and goal.

Breakdown Contributions.

- Firstly, downloaded the dataset in XML format.
- Converted the XML elements into dictionary.
- Connected the dataset to MongoDB and saved each row in the MongoDB.
- Connected the dataset to the PostgreSQL database.
- Created a Table in the database.
- Extract the dataset from MongoDB and convert it into a valid JSON string.
- In addition to converted Json file, it was also converted to dictionary form.
- The simi-structured Json file using the Luigi automation was then inserted into the MongoDB, to store it for later retrieval when needed.
- Pandas' data frame enabled the storage of format of the Json file to be to be converted back to csv by using the Luigi automation process.
- The attributes of the death rate for suicide data frame were also displayed, to give a proper information of what the data is all about.
- Created a gender column to separate the gender as well as encoding it to 0's and 1's for proper analysis to be made.
- To ensure the data integrity and consistency, outliers and missing values were thoroughly checked and addressed.
- Saved the cleared data preprocessed data into PostgreSQL as the second database used in the analysis.
- PostgreSQL stored the cleaned structured dataset to be ready for retrieval when needed.
- Queries were performed in the database, to better understand and perform analysis.
- A pie chart was created. This showed the distribution proportion of male and female who died from suicide in the us.
- Having a bar chart that showed the estimate of gender also made more impact to the investigation.

- Another bar chart aggregated with the estimated death rate by age group, displayed the highest to the lowest of deaths age rankings with their total estimates.
- The success of the aggregated bar chat of the estimated death rate by the age groups and gender helped uncovered the fatalities of the range in gender and age groups for suicide deaths in the United States.
- Lastly, the final visualization contributed to the findings of the death rates for suicide over time. Showing from 1950 to 2018, and excided it provide year range to predict up to 2020.

9. FLOWCHART DIAGRAM OF PROCESSING DATASET USING PYTHON.



DATASET 2

BY

x23218274 - Hilal Ozcelik.

1. **NAME OF DATASET: "**Death Rates, for Suicide by Drug Type, Sex, Age, Race, And Hispanic Origin: United State."

2. **DESCRIPTION INFROMATION:** The Us Department of Health and Human Services made available the data on death rate for suicide by drug type and selected populations from the year 1990 to 2018.

3. **SOURCE LOCATION:** DATA.GOV**: https://catalog.data.gov/dataset/death-rates-for-suicide-by-sex-race-hispanic-origin-and-age-united-states-020c1/resource/3e3345d3-5759-445c-aa2f-9bfc6891bd6b**

4. **RETRIEVAL METHOD:** Comma Separated Value File (CSV) - Download: https: https://data.cdc.gov/api/views/9j2v-jamp/rows.csv?accessType=DOWNLOAD

5. **ATTRIBUTES OF DATASETS:** It contains **6390** numbered rows and **15** columns.

6. **DATA PROCESSING PIPELINE**
   - Downloaded of CSV file.
   - Converted CSV files into dictionaries.
   - Loaded CSV file to MongoDB server – Using Luigi.
   - Extracted CSV file from MongoDB server as Panda's data frame – Using Luigi.
   - Displayed Attributes of the data frame.
   - Perform Cleaning, Transformation and Future Engineering.
   - Loaded the cleaned data frame to Postgres Database.
   - Performed Visualizations on the date frame.

7. **MAIN COMPONETS**
   - Installation of Programming Environment
     - Python IDE – Jupyter Notebook.
   - Installation of Databases
     - MongoDB Compass – NoSQL Database Server.
     - Postgres Database Admin4 – SQL Database Server.
   - Installation of Necessary Commands
     - Pip Pymongo.
     - Pip Luigi.
   - Importation of Necessary Packages and Libraries
     - Json – for handing Json file.
     - Luigi – For automation pipeline in python
     - Pymongo, mongoclient – To allow interaction with the MongoDB database.

- Pandas as pd – For data manipulation and analysis.
- Matplotlib. pyplot as plt – To create visualizations.
- Seaborn as sns – To create statistical data visualizations.

8. **CONTRIBUTION IN DETAIL**

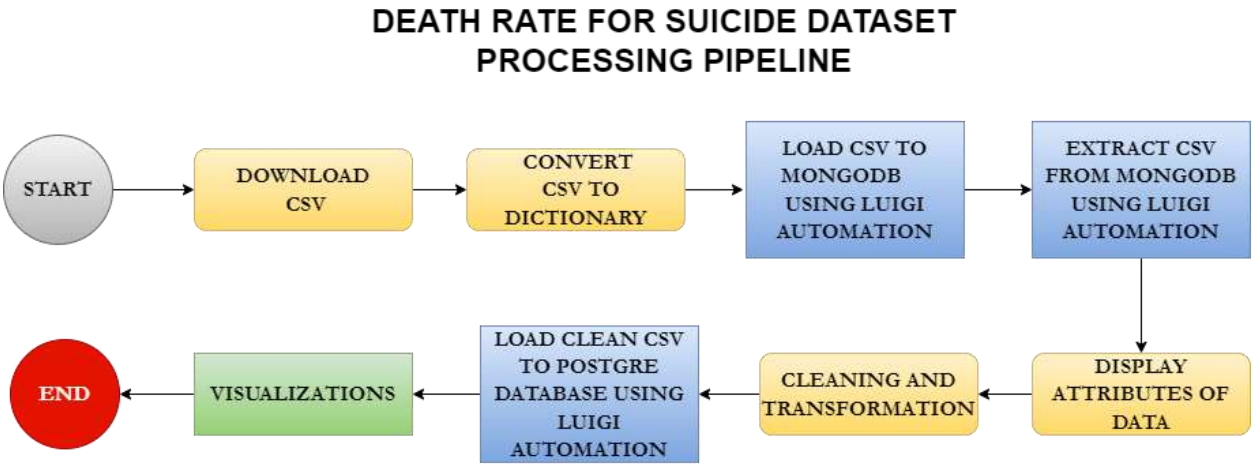
With the responsibility to work on the dataset for death rates for suicide, by drug type, sex, age, race, and Hispanic origin in the United States from 1990 to 2018. A series of tasks were carried out to achieve a final quality- analyzed and visualized findings that were helpful to the project's main objective and goal.

Breakdown Contributions.

- Firstly, downloaded the drug rate for suicide dataset.
- Firstly, downloaded the death rate for suicide dataset.
- Then loaded it into a python environment.
- Then the CSV file was converted into a Json file, ensuring it fits the database's structure intended for it to store in.
- In addition to converted Json file, it was also converted to dictionary form.
- Afterwards, created a database and connection collection, where death rate for suicide was the database and collection.
- A connection made possible to a MongoDB, for the storage of documents, or files.
- The semi-structured Json file using the Luigi automation was then inserted into MongoDB, to store it for later retrieval when needed.
- Pandas' data frame enabled the storage of format of the Json file to be to be converted back to csv by using the Luigi automation process.
- The attributes of the death rate for suicide data frame were also displayed, to give a proper information of what the data is all about.
- Created a gender column to separate the gender as well as encoding it to 0's and 1's for proper analysis to be made.
- To ensure the data integrity and consistency, outliers and missing values were thoroughly checked and addressed.
- Saved the cleared data preprocessed data into PostgreSQL as the second database used in the analysis.
- PostgreSQL stored the cleaned structured dataset to be ready for retrieval when needed.
- Queries were performed in the database, to better understand and perform analysis.
- A pie chart was created. This showed the distribution proportion of male and female who died from suicide in the US over the period of the time.
- Having a bar char that showed the estimate death rate of genders also made more impact to the investigation.
- Another bar chart aggregated with the estimated death rate by age group, displayed the highest to the lowest of deaths age rankings with their total estimates.
- The success of the aggregated bar chat of the estimated death rate by the age groups and gender helped uncovered the fatalities of the range in gender and age groups for suicide deaths in the United States.

Lastly, the final visualization contributed to the findings of the death rates for suicide over time.
Showing from 1950 to 2018, and excided it provide year range to predict up to 2020.

9. FLOWCHART DIAGRAM OF PROCESSING DATASET USING PYTHON.



DATASET 3

BY

x23104201 - Adeola Deborah Adeniji.

1. **NAME OF DATASET: "**Drug Overdose Death Rates, by Drug Type, Sex, Age, Race, And Hispanic Origin: United State."

2. **DESCRIPTION INFROMATION:** The Us Department of Health and Human Services made available the data on drug overdose death rates by drug type and selected populations from the year 1999 to 2018.

3. **SOURCE LOCATION:** DATA.GOV**: https://catalog.data.gov/dataset/drug-overdose-death-rates-by-drug-type-sex-age-race-and-hispanic-origin-united-states-3f72f/resource/48eb6490-5709-43f3-ae4a-3c7d3a4b0c2c**

4. **RETRIEVAL METHOD:** Comma Separated Value File (CSV) - Through an UR: https://data.cdc.gov/api/views/95ax-ymtc/rows.csv?accessType=DOWNLOAD

5. **ATTRIBUTES OF DATASETS:** It contains **6229** numbered rows and **15** columns.

6. **DATA PROCESSING PIPELINE**
   - Get Request URL of CSV file – Using Luigi.
   - Converted CSV to JSON file – Using Luigi.
   - Converted JSON file to dictionaries – Using Luigi.
   - Loaded JSON file to MongoDB server – Using Luigi.
   - Extracted JSON file from MongoDB database.
   - Converted JSON file to CSV file as Panda's data frame – Using Luigi.
   - Displayed Attributes of the data frame – Using Luigi.
   - Perform Cleaning, Transformation and Features Engineering Categorization on the data frame.
   - Performed Visualizations on the date frame.
   - Created a Random Forest Machine Learning Model for the data frame.
   - Loaded the data frame to Postgres Database – Using Luigi.
   - Performed SQL Queries on the data.

7. **MAIN COMPONETS**
   - Installation of Programming Environment
     - Python IDE – Jupyter Notebook.
   - Installation of Databases
     - MongoDB Compass – NoSQL Database Server.
     - Postgres Database Admin4 – SQL Database Server.
   - Installation of Necessary Commands
     - Pip Requests
     - Pip Pymongo
     - Pip Luigi

- Pip Install Seaborn.
- Pip Install scikit-learn.
- Pip Install SQL alchemy.

- Importation of Necessary Packages and Libraries
  - Requests – To make HTTP requests.
  - Json – for handing Json file.
  - Luigi – For automation pipeline in python.
  - Pymongo – To allow interaction with the MongoDB database.
  - Pandas as pd – for manipulation and analysis
  - Ipythone.display - For better display printout of results.
  - Matplotlib. pyplot as plt – To create visualizations.
  - Seaborn as sns – To create statistical data visualizations.
  - Sklean. preprocessing label LabelEncoder – To encoding categorical variables.
  - Sklean. Preprocessing FunctionTransformer – To inserting custom transformations.
  - Sklean. Pipeline, pipeline – To build machine learning pipeline.
  - Sklean. Compose, columnTransformer – To apply different transformation to different columns.
  - Sklean. Model selection, tranin_test_split – For splitting data into training, and testing.
  - Sklean. Ensemble, RandomForestRegressor – To build random forest regression model.
  - Sklean. Model selection, GridSearchCV – For hyperparameter turning search.
  - Sklean. Metrics, mean_standard_error – To evaluate model performance using MSR.
  - Sklean. Preprocessing, OneHotEncoder, StandardScaler – To encode categorical variables and scale numerical variables.
  - Sqllachemy, create_engine – To interact with SQL database.

8. **CONTRIBUTION IN DETAIL**

Having the responsibility to work on the Dataset of drug overdose death rates, by drug type, sex, age, race, and Hispanic origin in the United States from 1990 to 2018. A series of activities were carried out to achieve final quality-analyzed and visualized findings that were helpful to the project's main objective and goal.

Breakdown Contributions.

- Firstly, a request was made to get the URL CSV file of the drug overdose dataset using the Luigi automation pipeline.
- Then the CSV file was converted into a JSON file, ensuring it fit the database's structure intended for it to be stored in.
- In addition to the converted JSON file, it was also converted to dictionary form.
- A connection made possible to a MongoDB, for the storage of documents, or files.
- The simi-structured Json file was then inserted into the MongoDB, to store it for later retrieval when needed.
- For proper analysis, to be carried out on the drug overdose dataset, the JSON file is then extracted and converted back into a CSV format using the panda's data frame. This is to ensure that certain manipulations and analyses can be carried out.

- A lot of preprocessing and transformation was carried out, as the data was not fit to perform reasonable and accurate analysis.
- Certain columns considered redundant were removed from the data frame.
- Missing valves indicated were filled with their estimated mean value.
- Columns with date were converted to specified datetime format and encoded other columns to be accurately represented.
- Outlines were checked and removed, to ensure the prevention of anomalies when presenting the analysis of the overdose death rate.
- Due to the complexity structure of the overdose death rate data, feature engineering categorization was applied. Two data frames were created with the addition of two more columns “Race” and “Gender”.
- The original data frame generalized the “Stub_Name” and Stub_Label” columns which included the Race and Gender of the death caused by drug overdose.
- Proper analysis can be done as the separation gives more light to the hidden features of the data.
- A pie chart was created. This showed the distribution proportion of male and female who died from drug overdose in the us.
- Another pie chart was also created that showed the distribution of race. The proportion of each race (Americans Indian, Alaska Native, White, Blacks, Hispanic or Latin, etc.) providing the impacts of various racial demographics that was affected by the drug overdose.
- A bar chart distribution of age group also brought more insights into the analysis of the drug related fatalities among various ages ranging from 15 years to 85 years and over.
- The total number of deaths caused from 1999 to 2018 by drug overdose recorded each year was also illustrated through and bar chart.
- A stacked bar chart visualized the disposition of death cause by drug overdose across different age groups. Each bar represented an age group with a division indicating the number s of deaths for males and females in the United State.
- Having a line plot that showed the trend of deaths across different racial groups gave more insight into the dynamics of the mortality rate.
- Paring the distribution of years and age with the use of a of deaths caused by drug overdoses provides a Stacked bar plot visual understanding of how deaths vary across age demographics over time.
- Another stacked bar plot was applied to illustrate the deaths over the years across racial groups over time.
- Lastly, for the last visualization, a line plot that showed the estimated deaths of drug overdose over time.
- A machine-learning model called Random Forest was created to predict age and year over-estimate. For the first model, a mean square error of approximately 5.86 was achieved, while for the second model achieved a mean square error of 5.76.
- Finally, there dataset that was prepared and organized earlier was stored into PostgreSQL database. Data. Queries were also performed to gain more insight into the data.

10. FLOWCHART DIAGRAM OF PROCESSING DATASET USING PYTHON.