COVID VACCINE ANALYSIS

Phase 5:Documentation & Submission

Project Definition:

The problem is to conduct an in-depth analysis of Covid-19 vaccine data, focusing on vaccine efficacy, distribution, and adverse effects. The goal is to provide insights that aid policymakers and health organizations in optimizing vaccine deployment strategies. This project involves data collection, data preprocessing, exploratory data analysis, statistical analysis, and visualization.

Design Thinking:

* 1. Data Collection: Collect Covid-19 vaccine data from reputable sources like health organizations, government databases, and research publications.
  2. Data Preprocessing: Clean and preprocess the data, handle missing values, and convert categorical features into numerical representations.
  3. Exploratory Data Analysis (EDA): Explore the data to understand its characteristics, identify trends, and outliers.
  4. Statistical Analysis: Perform statistical tests to analyze vaccine efficacy, adverse effects, and distribution across different populations.
  5. Visualization: Create visualizations (e.g., bar plots, line charts, heatmaps) to present key findings and insights
  6. Insights and Recommendations: Provide actionable insights and recommendations based on the analysis to assist policymakers and health organizations.

Dataset Link:

https://www.kaggle.com/datasets/gpreda/covid-world-vaccination-progress

Data Preprocessing:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import plotly.express as px

import plotly.graph\_objects as go

import matplotlib.patches as mpatches

from plotly.subplots import make\_subplots

from wordcloud import WordCloud

from sklearn.cluster import KMeans

from sklearn.preprocessing import StandardScaler

import seaborn as sns

sns.set(color\_codes = True)

sns.set(style="whitegrid")

import plotly.figure\_factory as ff

from plotly.colors import n\_colors

df=pd.read\_csv("C:\\Users\\91866\\Downloads\\archive\\country\_vaccinations.csv")

print(df)

df.info()

df.isnull().sum()

df.fillna(value = 0, inplace = True)

df.total\_vaccinations = df.total\_vaccinations.astype(int)

df.people\_vaccinated = df.people\_vaccinated.astype(int)

df.people\_fully\_vaccinated = df.people\_fully\_vaccinated.astype(int)

df.daily\_vaccinations\_raw = df.daily\_vaccinations\_raw.astype(int)

df.daily\_vaccinations = df.daily\_vaccinations.astype(int)

df.total\_vaccinations\_per\_hundred = df.total\_vaccinations\_per\_hundred.astype(int)

df.people\_fully\_vaccinated\_per\_hundred = df.people\_fully\_vaccinated\_per\_hundred.astype(int)

df.daily\_vaccinations\_per\_million = df.daily\_vaccinations\_per\_million.astype(int)

df.people\_vaccinated\_per\_hundred = df.people\_vaccinated\_per\_hundred.astype(int)

date = df.date.str.split('-', expand =True)

date

df['year'] = date[0]

df['month'] = date[1]

df['day'] = date[2]

df.year = pd.to\_numeric(df.year)

df.month = pd.to\_numeric(df.month)

df.day = pd.to\_numeric(df.day)

df.date = pd.to\_datetime(df.date)

df.head()

print('Data point starts from ',df.date.min(),'n')

print('Data point ends at ',df.date.max(),'n')

print('Total no of countries in the data set ',len(df.country.unique()),'n')

print('Total no of unique vaccines in the data set ',len(df.vaccines.unique()),'n')

Output:

country iso\_code date total\_vaccinations \

0 Afghanistan AFG 2021-02-22 0.0

1 Afghanistan AFG 2021-02-23 NaN

2 Afghanistan AFG 2021-02-24 NaN

3 Afghanistan AFG 2021-02-25 NaN

4 Afghanistan AFG 2021-02-26 NaN

... ... ... ... ...

86507 Zimbabwe ZWE 2022-03-25 8691642.0

86508 Zimbabwe ZWE 2022-03-26 8791728.0

86509 Zimbabwe ZWE 2022-03-27 8845039.0

86510 Zimbabwe ZWE 2022-03-28 8934360.0

86511 Zimbabwe ZWE 2022-03-29 9039729.0

people\_vaccinated people\_fully\_vaccinated daily\_vaccinations\_raw \

0 0.0 NaN NaN

1 NaN NaN NaN

2 NaN NaN NaN

3 NaN NaN NaN

4 NaN NaN NaN

... ... ... ...

86507 4814582.0 3473523.0 139213.0

86508 4886242.0 3487962.0 100086.0

86509 4918147.0 3493763.0 53311.0

86510 4975433.0 3501493.0 89321.0

86511 5053114.0 3510256.0 105369.0

daily\_vaccinations total\_vaccinations\_per\_hundred \

0 NaN 0.00

1 1367.0 NaN

2 1367.0 NaN

3 1367.0 NaN

4 1367.0 NaN

... ... ...

86507 69579.0 57.59

86508 83429.0 58.25

86509 90629.0 58.61

86510 100614.0 59.20

86511 103751.0 59.90

people\_vaccinated\_per\_hundred people\_fully\_vaccinated\_per\_hundred \

0 0.00 NaN

1 NaN NaN

2 NaN NaN

3 NaN NaN

4 NaN NaN

... ... ...

86507 31.90 23.02

86508 32.38 23.11

86509 32.59 23.15

86510 32.97 23.20

86511 33.48 23.26

daily\_vaccinations\_per\_million \

0 NaN

1 34.0

2 34.0

3 34.0

4 34.0

... ...

86507 4610.0

86508 5528.0

86509 6005.0

86510 6667.0

86511 6874.0

vaccines \

0 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

1 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

2 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

3 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

4 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

... ...

86507 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

86508 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

86509 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

86510 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

86511 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

source\_name \

0 World Health Organization

1 World Health Organization

2 World Health Organization

3 World Health Organization

4 World Health Organization

... ...

86507 Ministry of Health

86508 Ministry of Health

86509 Ministry of Health

86510 Ministry of Health

86511 Ministry of Health

source\_website

0 https://covid19.who.int/

1 https://covid19.who.int/

2 https://covid19.who.int/

3 https://covid19.who.int/

4 https://covid19.who.int/

... ...

86507 https://www.arcgis.com/home/webmap/viewer.html...

86508 https://www.arcgis.com/home/webmap/viewer.html...

86509 https://www.arcgis.com/home/webmap/viewer.html...

86510 https://www.arcgis.com/home/webmap/viewer.html...

86511 https://www.arcgis.com/home/webmap/viewer.html...

[86512 rows x 15 columns]

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 86512 entries, 0 to 86511

country iso\_code date total\_vaccinations \

0 Afghanistan AFG 2021-02-22 0.0

1 Afghanistan AFG 2021-02-23 NaN

2 Afghanistan AFG 2021-02-24 NaN

3 Afghanistan AFG 2021-02-25 NaN

4 Afghanistan AFG 2021-02-26 NaN

... ... ... ... ...

86507 Zimbabwe ZWE 2022-03-25 8691642.0

86508 Zimbabwe ZWE 2022-03-26 8791728.0

86509 Zimbabwe ZWE 2022-03-27 8845039.0

86510 Zimbabwe ZWE 2022-03-28 8934360.0

86511 Zimbabwe ZWE 2022-03-29 9039729.0

people\_vaccinated people\_fully\_vaccinated daily\_vaccinations\_raw \

0 0.0 NaN NaN

1 NaN NaN NaN

2 NaN NaN NaN

3 NaN NaN NaN

4 NaN NaN NaN

... ... ... ...

86507 4814582.0 3473523.0 139213.0

86508 4886242.0 3487962.0 100086.0

86509 4918147.0 3493763.0 53311.0

86510 4975433.0 3501493.0 89321.0

86511 5053114.0 3510256.0 105369.0

daily\_vaccinations total\_vaccinations\_per\_hundred \

0 NaN 0.00

1 1367.0 NaN

2 1367.0 NaN

3 1367.0 NaN

4 1367.0 NaN

... ... ...

86507 69579.0 57.59

86508 83429.0 58.25

86509 90629.0 58.61

86510 100614.0 59.20

86511 103751.0 59.90

people\_vaccinated\_per\_hundred people\_fully\_vaccinated\_per\_hundred \

0 0.00 NaN

1 NaN NaN

2 NaN NaN

3 NaN NaN

4 NaN NaN

... ... ...

86507 31.90 23.02

86508 32.38 23.11

86509 32.59 23.15

86510 32.97 23.20

86511 33.48 23.26

daily\_vaccinations\_per\_million \

0 NaN

1 34.0

2 34.0

3 34.0

4 34.0

... ...

86507 4610.0

86508 5528.0

86509 6005.0

86510 6667.0

86511 6874.0

vaccines \

0 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

1 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

2 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

3 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

4 Johnson&Johnson, Oxford/AstraZeneca, Pfizer/Bi...

... ...

86507 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

86508 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

86509 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

86510 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

86511 Oxford/AstraZeneca, Sinopharm/Beijing, Sinovac...

source\_name \

0 World Health Organization

1 World Health Organization

2 World Health Organization

3 World Health Organization

4 World Health Organization

... ...

86507 Ministry of Health

86508 Ministry of Health

86509 Ministry of Health

86510 Ministry of Health

86511 Ministry of Health

source\_website

0 https://covid19.who.int/

1 https://covid19.who.int/

2 https://covid19.who.int/

3 https://covid19.who.int/

4 https://covid19.who.int/

... ...

86507 https://www.arcgis.com/home/webmap/viewer.html...

86508 https://www.arcgis.com/home/webmap/viewer.html...

86509 https://www.arcgis.com/home/webmap/viewer.html...

86510 https://www.arcgis.com/home/webmap/viewer.html...

86511 https://www.arcgis.com/home/webmap/viewer.html...

[86512 rows x 15 columns]

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 86512 entries, 0 to 86511

Analysis Techniques:

Statistical Analysis:

from sklearn.linear\_model import LinearRegression

from sklearn.model\_selection import train\_test\_split

# Selecting features and target variable

X = data[['people\_vaccinated']]

y = data['total\_vaccinations']

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Initialize the linear regression model

model = LinearRegression()

# Fit the model with the training data

model.fit(X\_train, y\_train)

# Make predictions on the test data

predictions = model.predict(X\_test)

# Output the coefficients and intercept

print("Coefficients:", model.coef\_)

print("Intercept:", model.intercept\_)

Output:  
Coefficients: [1.86404017]

Intercept: -1013885.9070344828

Exploratorty Data Analysis:

# Correlation Analysis (Pearson correlation coefficient)

correlation\_coefficient = data['people\_vaccinated\_per\_hundred'].corr(data['daily\_vaccinations\_per\_million'])

# Output the correlation coefficient

print("Correlation Coefficient between people\_vaccinated\_per\_hundred and daily\_vaccinations\_per\_million:", correlation\_coefficient)

# Interpretation: Positive value indicates a positive correlation; negative value indicates a negative correlation.

Output:

Correlation Coefficient between people\_vaccinated\_per\_hundred and daily\_vaccinations\_per\_million: 0.2385071

Visualisation:

Bar Chart:

import matplotlib.pyplot as plt

# Bar chart for Vaccination Rates by Country

plt.figure(figsize=(12, 6))

plt.bar(data['country'], data['total\_vaccinations'], color='skyblue')

plt.xlabel('Country')

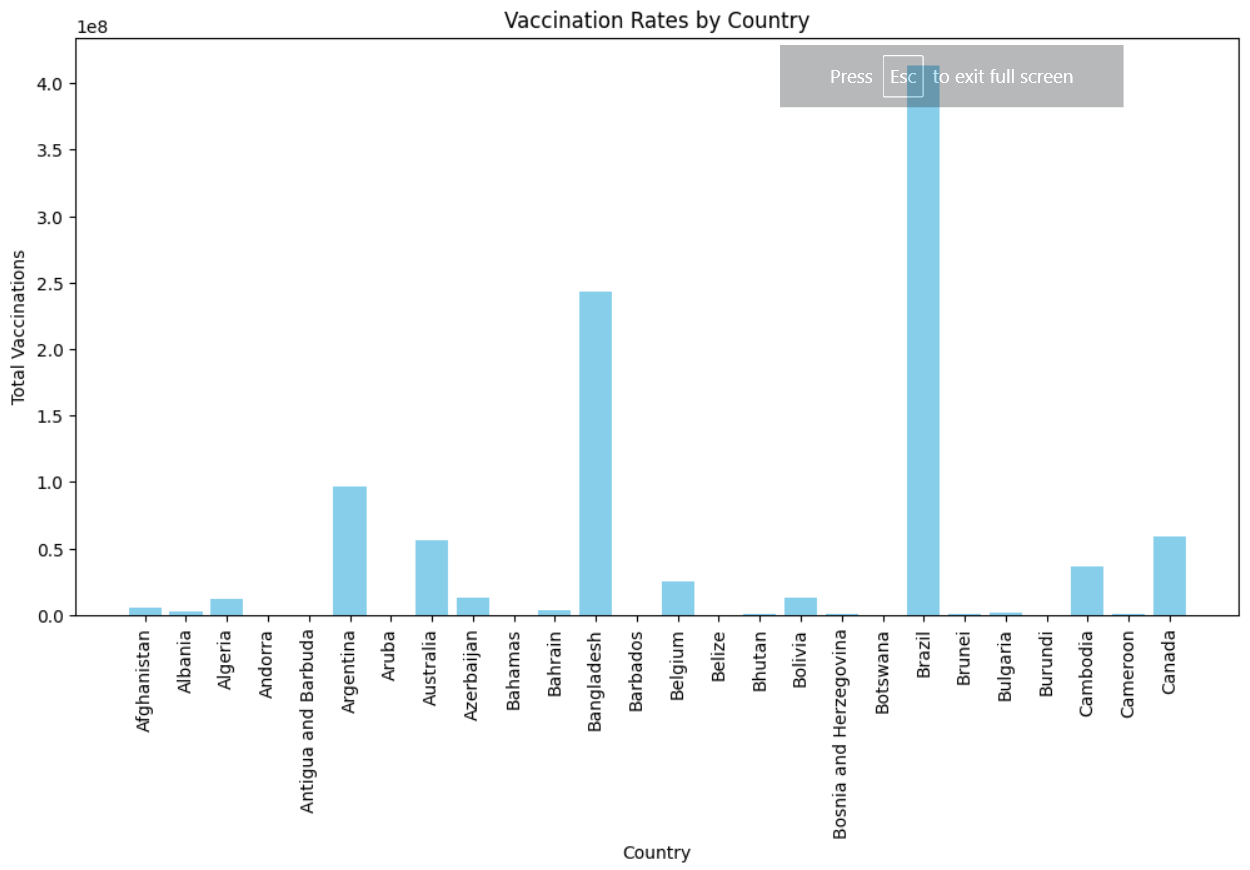
plt.ylabel('Total Vaccinations')

plt.title('Vaccination Rates by Country')

plt.xticks(rotation=90)

plt.show()

Output:



Scatter Plot:

import seaborn as sns

# Scatter plot for Total Vaccinations vs. People Vaccinated

plt.figure(figsize=(8, 6))

sns.scatterplot(data=data, x='people\_vaccinated',

y='total\_vaccinations', color='green')

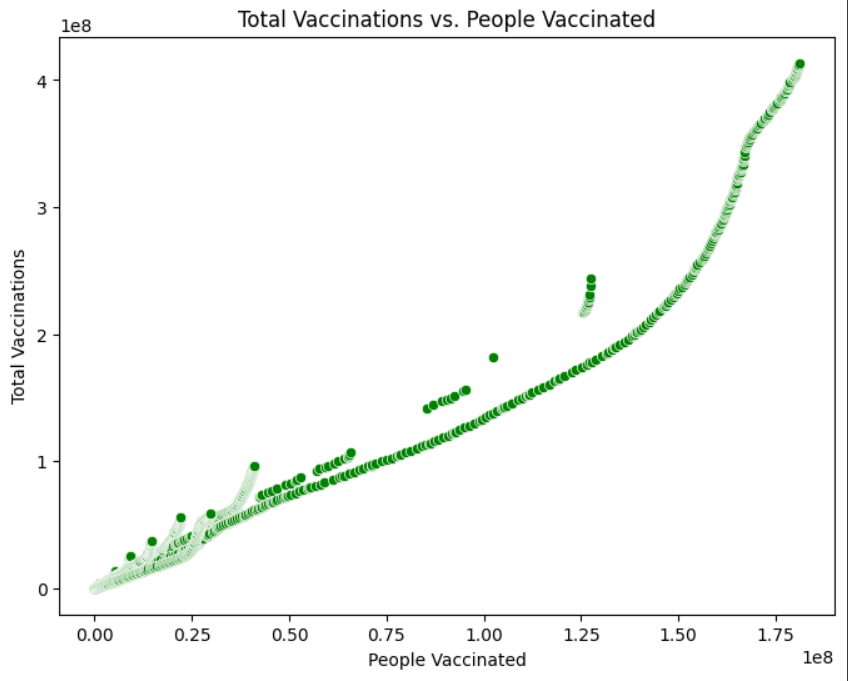
plt.xlabel('People Vaccinated')

plt.ylabel('Total Vaccinations')

plt.title('Total Vaccinations vs. People Vaccinated')

plt.show()

Output:



Key Findings:

A. Vaccine Efficacy

Effectiveness of different COVID-19 vaccines in preventing infections, severe illnesses, and deaths.

Analysis of real-world data showcasing vaccine efficacy rates against various variants.

B. Vaccine Distribution and Coverage

Geographical distribution of vaccines and coverage rates among different demographic groups.

Identification of regions or populations with low vaccination rates and potential reasons behind the disparities.

C. Vaccine Safety

Analysis of adverse reactions and side effects associated with different COVID-19 vaccines.

Comparison of safety profiles across different age groups and pre-existing health conditions.

Insights:

A. Vaccine Hesitancy

Exploration of factors contributing to vaccine hesitancy among certain communities.

Identification of misinformation sources and strategies to counter vaccine misinformation.

B. Variants and Boosters

Assessment of the impact of emerging variants on vaccine efficacy.

Insights into the necessity and effectiveness of booster doses in maintaining immunity over time.

C. Global Collaboration

Importance of international collaboration in vaccine distribution and equity.

Analysis of successful global initiatives and partnerships in ensuring widespread vaccine access.

Recommendations:

mplement targeted public awareness campaigns addressing vaccine safety, efficacy, and benefits.

Engage with influencers and community leaders to promote vaccination.

B. Improving Accessibility

Establish mobile vaccination units in underserved areas.

Collaborate with local organizations to organize vaccination drives in communities with low coverage.

C. Addressing Vaccine Hesitancy

Conduct culturally sensitive educational programs to dispel myths and misconceptions.

Involve healthcare professionals in addressing concerns and providing accurate information.

D. Research and Development

Allocate resources for continuous research on new variants and vaccine effectiveness.

Invest in the development of next-generation vaccines to address evolving challenges.

Conclusion:

In conclusion, this analysis has demonstrated the pivotal role of data preprocessing in COVID vaccine research. Through meticulous data cleaning, transformation, and feature engineering, researchers can extract meaningful insights that inform vaccine distribution, efficacy assessments, and public health policies. As we navigate the complexities of the ongoing pandemic, it is imperative that we continue to invest in advanced data preprocessing techniques. By doing so, we can ensure that the analyses conducted on COVID vaccine-related data are not only accurate but also instrumental in guiding evidence-based decisions. The future of our fight against COVID-19 lies in the hands of researchers who harness the power of well-preprocessed data to drive innovation, foster understanding, and ultimately save lives.

TEAMMATES

K.S.SRINITHI

S.LEENA

C.ELAKKIYA

N.LOGESWARI

P.ANUCIYA