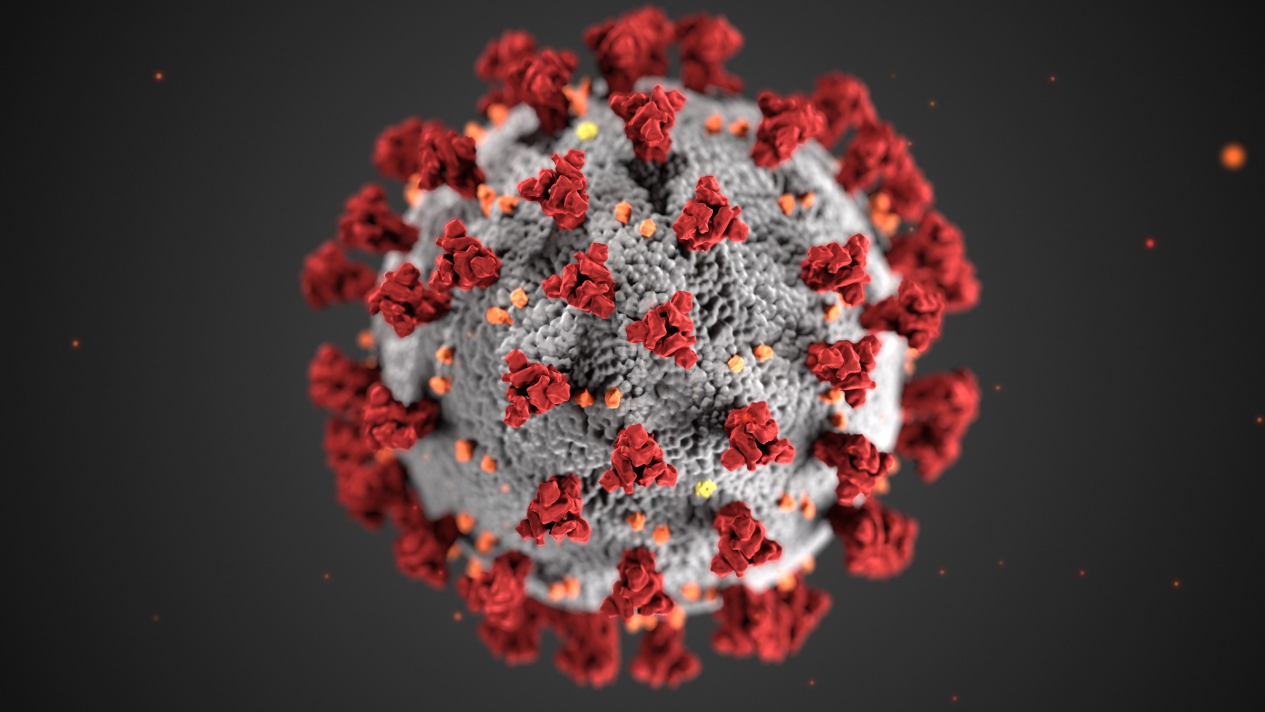
**COVID-19 Vaccine Analysis - Phase 2: Innovative Problem Solving**

|  |  |  |
| --- | --- | --- |
| **NAME** | **REG NO.** | **EMAIL ID/NM** |
| **Jeyasurya S** | **312821243010** | **21aids11@act.edu.in** |



**Introduction:**

In the introduction, you should provide a concise overview of the document's purpose. Mention that this phase of the analysis aims to delve into specific issues related to COVID-19 vaccination data and present innovative solutions to tackle these problems. Highlight the importance of data-driven decision-making in managing the pandemic.

**Problem Statement:**

In this section, elaborate on the specific problem you intend to address. For instance, you might focus on the challenge of vaccine distribution, vaccine hesitancy, or vaccine supply shortages in certain regions. Explain why this problem is significant in the context of COVID-19 vaccination efforts. Provide statistics or examples to support your claims.

**Data Source**

Explain that the data for this analysis is sourced from **Kaggle.com**, a reputable platform for data sharing and analysis. Clarify how the dataset was obtained, including any data collection methods or sources used. Highlight the credibility and relevance of Kaggle's data in the context of your analysis.

**Data Details**

Provide more specific details about the dataset, including the following:

- A list of countries included in the dataset.

- Explanation of the ISO codes and their significance.

- Breakdown of the data on the number of people vaccinated and not vaccinated in each country.

- Any other relevant data points, such as vaccine types, vaccination rates, or population statistics.

**Items in the dataset:**

* **Countries**
* **ISO\_code**
* **Dates**
* **Vaccines**
* **Total Vaccinations**
* **Source Name**
* **Source Websites**
* **Daily Vaccinations**

**Desired data to find:**

* **Most commonly used vaccines in countries**
* **Average daily vaccination count in countries**
* **Number of countries where vaccines are used**
* **Choropleth map of the most used vaccine**

**Methodology:**

In this section, provide a detailed overview of the methods and algorithms used for the analysis. If linear algorithms were employed, explain why they were chosen and how they work. Additionally, mention any other methods or techniques used for data preprocessing or analysis, such as data cleaning, feature engineering, or machine learning models.

**Library:**

* Numpy
* Seaborn
* OS
* Pandas
* plotly
* Folium

**Training and Testing:**

1. Data Collection

2. Data Preprocessing

3. Feature Selection

1. Model Selection

5. Training and Testing Split

6. Model Training

7. Model Evaluation

8. Hyperparameter Tuning

9. Validation and Cross-Validation

1. Interpretability and Visualization

**Innovative Solutions:**

Detail the innovative solutions or approaches proposed to address the problem identified in the problem statement. Provide a step-by-step explanation of how each solution works and how it leverages the analysis and data to achieve the desired outcomes. If there are multiple solutions, present them individually and discuss their potential impact on the problem.

**Metrics for Accuracy Check:**

Explain the metrics used to evaluate the accuracy and effectiveness of the proposed solutions. Common metrics in data analysis include accuracy, precision, recall, F1-score, and ROC curves. Describe why these metrics were chosen and how they are relevant to the specific problem you are addressing. Provide examples of how the metrics are calculated and interpreted in the context of your analysis.

**References:**

Include a list of references or sources of information used in the document. This helps validate the credibility of your analysis and allows readers to explore further if they are interested.

By expanding on these sections with more detailed explanations and possibly including charts, graphs, or data visualizations, your document will provide a comprehensive and informative analysis of COVID-19 vaccine data and innovative solutions.

**Code/Program:**

import numpy as np

import pandas as pd

import os

for dirname, \_, filenames in os.walk('/kaggle/input'):

for filename in filenames:

print(os.path.join(dirname, filename))

**Loading dataset**

import pandas as pd

import plotly.express as px

import plotly.graph\_objects as go

from folium.features import Choropleth

import folium

from folium.features import Tooltip

import seaborn as sns

df = pd.read\_csv("/kaggle/input/covid-world-vaccination-progress/country\_vaccinations\_by\_manufacturer.csv")

df.head(10)

df["location"].nunique()

df.isnull().sum()

df.dtypes

df['date'] = pd.to\_datetime(df['date'])

data=pd.DataFrame(columns=['Country', 'Vaccine', 'Total\_vaccine'])

for country in df["location"].unique():

for vaccine in df["vaccine"].unique():

filtered\_data = df[(df['location'] == country) & (df['vaccine'] == vaccine)]

total\_count = filtered\_data['total\_vaccinations'].max()

data = pd.concat([data, pd.DataFrame({'Country': [country], 'Vaccine': [vaccine], 'Total\_vaccine': [total\_count]})], ignore\_index=True)

data.head(10)

data.dropna(axis=0,inplace=True)

data.head(20)

data\_2=pd.DataFrame(columns=['Country', 'Vaccine'])

data["Total\_vaccine"] = pd.to\_numeric(data["Total\_vaccine"], errors="coerce")

for country in data["Country"].unique():

new\_data = data[data["Country"] == country]

max\_vaccine = new\_data.loc[new\_data["Total\_vaccine"].idxmax(), "Vaccine"]

data\_2 = pd.concat([data\_2, pd.DataFrame({'Country': [country], 'Vaccine': [max\_vaccine]})], ignore\_index=True)

data\_2.head()

data\_2["Vaccine"].value\_counts().plot(kind="bar",

color=["Red","Gray","Gray","Gray"])

number\_of\_days = (df["date"].max() -df["date"].min() ).days

dtfrm=data[data["Vaccine"]=="Pfizer/BioNTech"]

dtfrm = dtfrm.drop(dtfrm[dtfrm['Country'] == 'European Union'].index)

dtfrm.head(10)

dtfrm["average\_vaccination\_count"] = dtfrm["Total\_vaccine"] / number\_of\_days

dtfrm["average\_vaccination\_count"] =dtfrm["average\_vaccination\_count"].astype(int)

dtfrm.head(15)

dtfrm.set\_index("Country",inplace=True)

color=["Lightblue","Purple","Green","Orange","darkgoldenrod","tan","Gray","Blue","Pink","Lightgreen"]

dtfrm["average\_vaccination\_count"].sort\_values(ascending=False).head(10).plot(kind="bar",color=color)

number\_of\_vaccines = data.groupby('Vaccine')['Country'].nunique()

number\_of\_vaccines.sort\_values(ascending=False).plot(kind="bar",color="r")

fig = px.choropleth(data\_frame=dtfrm,

locations=dtfrm.index,

locationmode='country names',

color='Total\_vaccine',

color\_continuous\_scale='YlOrRd',

title='Ülkelerde Yapılan Biontech Aşıları')

fig.update\_layout(title\_x=0.5)

m = folium.Map(location=[0, 0], zoom\_start=2)

Choropleth(

geo\_data='https://raw.githubusercontent.com/johan/world.geo.json/master/countries.geo.json',

name='choropleth',

data=dtfrm,

columns=[dtfrm.index, 'Total\_vaccine'],

key\_on='feature.properties.name',

fill\_color='YlOrRd',

fill\_opacity=0.7,

line\_opacity=0.2,

legend\_name='Aşı Sayısı',

).add\_to(m)

m