**COVID Vaccines Analysis**

Phase 3 Submission Document

**Project Name: Covid Vaccines Analysis**

732121205001:Abinav.T

Phase 3 : Development Part 1

**Introduction**

* In the covid vaccine analysis project, in the face of the unprecedented global health crisis posed by the COVID-19 pandemic, the rapid development and deployment of vaccines have emerged as a beacon of hope and a monumental achievement in the field of public health.
* The vaccination efforts, spanning across continents and cultures, represent a collective stride towards curbing the spread of the virus and ultimately, saving lives. As the world grapples with the challenges of vaccinating entire populations, there arises a critical need to analyze the vast and intricate data surrounding these vaccination campaigns.
* This data analytics project delves into the heart of this complex scenario, aiming to dissect the wealth of information available pertaining to COVID-19 vaccines. By employing advanced data analytics techniques, this project seeks to unravel patterns, trends, and insights within the data.
* Through rigorous analysis, we intend to shed light on various aspects of the COVID-19 vaccination process, such as efficacy rates, distribution strategies, public sentiment, and the impact of vaccinations on mitigating the disease's spread.
* The analysis encompasses diverse datasets, including vaccination rates, demographic information, regional disparities, public perception data from social media platforms, and more. By examining this multifaceted data, our objective is to contribute valuable insights that can inform public health policies, optimize vaccine distribution strategies, and aid healthcare professionals, policymakers, and researchers in making informed decisions.
* Through the lens of data analytics, this project serves as a torchbearer, illuminating the path toward a more comprehensive understanding of COVID-19 vaccinations. The findings generated herein are not merely statistical interpretations; they are actionable insights that can foster better decision-making, improve vaccination strategies, and ultimately contribute to the global effort in overcoming the pandemic's challenges.

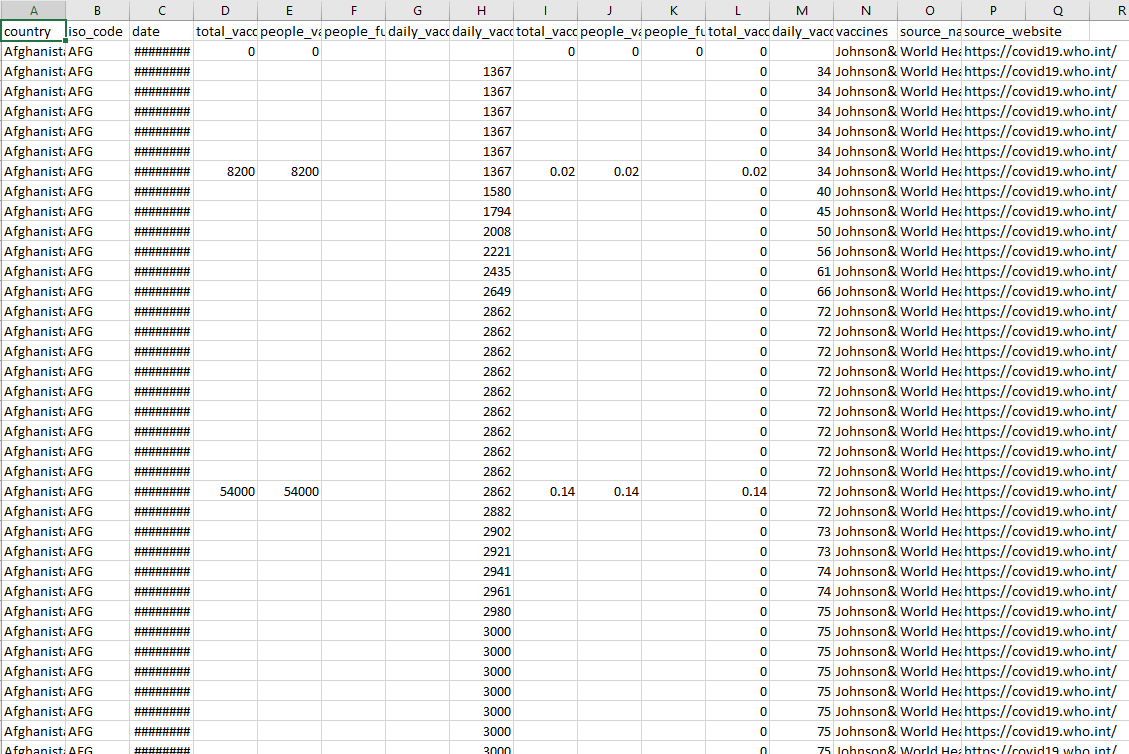
**Content for Project Phase 3**

Begin conducting the Covid-19 vaccine analysis by collecting and preprocessing the data. Collect and preprocess the COVID-19 vaccine data for analysis.

**Data Source**

A good data source for covid vaccine analysis using machine learning should be Accurate , Complete , Covering the geographic area of intrest , Accessible.

**Dataset Link:** ( <https://www.kaggle.com/datasets/gpreda/covid-world-vaccination-progress> )



**Importance of loading and processing dataset:**

Loading and preprocessing the dataset is an important first step inbuilding any machine learning model. However, it is especiallyimportant for house price prediction models, as house price datasets areoften complex and noisy.

By loading and preprocessing the dataset, we can ensure that themachine learning algorithm is able to learn from the data effectively andaccurately.

**Challenges involved in loading and preprocessing a covid vaccine analysis dataset:**

Loading and preprocessing a COVID vaccine analysis dataset can be particularly challenging due to the complexity and variability of the data. Here are some specific challenges associated with such datasets:

**1. Data Source and Formats:**

* Multiple Sources: COVID vaccine data often comes from various sources such as government reports, research studies, hospitals, and public health organizations. Integrating data from different sources while maintaining consistency can be difficult.
* Data Formats: Data might be in different formats like CSV, JSON, or API responses. Ensuring compatibility and consistency in these formats is necessary.

**2. Data Integrity and Quality:**

* Missing Data: Vaccine datasets might have missing values due to various reasons, making it challenging to perform analysis without addressing these gaps.
* Data Accuracy: Ensuring that the data is accurate and reliable is crucial. Inaccurate reporting can lead to misleading analysis and conclusions.

**3. Data Complexity:**

* High Dimensionality: Vaccine datasets can have numerous variables, including demographic information, vaccine types, administration dates, adverse reactions, and more. Managing and understanding these variables is complex.
* Temporal Aspects: COVID vaccine data often involves time-series information, and handling temporal aspects such as vaccination rates over time is vital.

**4. Data Privacy and Security:**

* Privacy Concerns: Vaccine datasets may contain sensitive information, requiring compliance with privacy regulations. Anonymizing and securing the data is crucial to protect individuals' privacy.
* Security: Protecting the dataset from unauthorized access and ensuring secure data transfers is essential.

**5. Data Preprocessing:**

* Categorical Data: Variables such as vaccine types or regions are categorical and need to be appropriately encoded for analysis.
* Imbalanced Data: Certain demographic groups might be vaccinated more or less frequently, leading to imbalanced datasets. Addressing this imbalance is critical for unbiased analysis.

**6. Feature Engineering:**

* Creating Relevant Features: Generating meaningful features from raw data can enhance the analysis. For example, deriving vaccination rates, trends, or efficacy indicators from the raw dataset requires domain knowledge.

**7. Domain Knowledge:**

* Understanding Vaccine Specifics: Interpreting vaccine-specific variables, like dosage, efficacy, and adverse reactions, requires domain expertise in immunology and epidemiology to ensure accurate analysis and interpretation.

**8. Scaling and Performance:**

* Large Datasets: COVID vaccine datasets can be massive, requiring efficient data processing techniques and potentially necessitating the use of big data tools for analysis.
* Computational Resources: Analyzing large datasets demands significant computational resources, including memory and processing power.

**9. Versioning and Data Changes:**

* Changing Data: Datasets may evolve over time due to updates or corrections. Managing different versions of the dataset and handling changes is important for consistency in analysis.

**How to overcome the challenges of loading and preprocessing a covid vaccine analysis dataset:**

Overcoming the challenges of loading and preprocessing a COVID vaccine analysis dataset requires careful planning, attention to detail, and the use of appropriate tools and techniques. Here are some strategies to overcome these challenges effectively:

**1. Understand the Data:**

* Gain a deep understanding of the dataset and its variables. Collaborate with domain experts to comprehend the significance of each data point and its relevance to the analysis.

**2. Data Cleaning and Handling Missing Values:**

* Use techniques like imputation to handle missing data points. Imputation methods fill missing values with estimated values based on the rest of the data, preserving the dataset's integrity.
* Remove duplicate records to ensure data accuracy.

**3. Data Integration:**

* Integrate data from various sources using tools like ETL (Extract, Transform, Load) processes or integration platforms. Ensure consistency and compatibility across different data formats.

**4. Data Security and Privacy:**

* Anonymize and pseudonymize sensitive data to protect individuals' privacy.
* Adhere to data protection laws and regulations such as GDPR. Collaborate with legal experts to ensure compliance.

**5. Data Formatting and Encoding:**

* Convert data into a consistent format. Libraries like Pandas in Python offer functions to handle different data formats.
* Encode categorical variables using techniques like one-hot encoding or label encoding to make them suitable for analysis.

**6. Feature Engineering:**

* Create new features derived from existing ones to provide more meaningful insights. For instance, calculate vaccination rates, age groups, or regional trends from raw data.

**7. Handling Imbalanced Data:**

* Use techniques such as oversampling, undersampling, or generating synthetic samples (SMOTE) to address class imbalances in the dataset.

**8. Temporal Aspects:**

* Manage time-series data effectively. Use appropriate time-series analysis techniques to identify trends, seasonality, and other temporal patterns.

**9. Scaling and Performance:**

* Utilize distributed computing and big data technologies like Apache Spark if dealing with large datasets. These technologies can efficiently handle big data processing tasks.

Optimize code and algorithms to make the most of available computational resources.

**10. Data Versioning and Documentation:**

* Maintain a versioning system for the dataset, ensuring proper documentation of changes made over time. Tools like Git can help in version control.

Document all preprocessing steps thoroughly for reproducibility and transparency.

**11. Collaboration and Expert Involvement:**

* Collaborate with domain experts, statisticians, and data scientists to ensure accurate interpretation and analysis of the dataset.
* Seek feedback and validation from experts in the field to refine preprocessing steps.

**12. Testing and Validation:**

* Split the dataset into training and testing subsets for model validation.
* Validate preprocessing steps rigorously to ensure they do not introduce biases or errors into the dataset.

**13. Continuous Monitoring:**

* Establish a process for continuous monitoring of the dataset. Regularly check for new data, updates, and changes in the data source.

**14. Use of Libraries and Tools:**

* Leverage data preprocessing libraries like Pandas, NumPy, and scikit-learn in Python, which offer a wide array of functions for data manipulation and analysis.
* Utilize visualization tools like Matplotlib and Seaborn to gain insights into the data distribution and relationships between variables.

By applying these strategies and being mindful of the specific challenges posed by COVID vaccine analysis datasets, data scientists and researchers can enhance the quality, accuracy, and reliability of their analyses, leading to more meaningful insights and informed decision-making.

**Loading the dataset:**

Loading the dataset using machine learning is the process of bringingthe data into the machine learning environment so that it can be usedto train and evaluate a model.

The specific steps involved in loading the dataset will vary dependingon the machine learning library or framework that is being used. However, there are some general steps that are common to most machine learning frameworks:

**a. Identify the dataset:**

The first step is to identify the dataset that you want to load. Thisdataset may be stored in a local file, in a database, or in a cloud storageservice.

**b. Load the dataset:**

Once you have identified the dataset, you need to load it into themachine learning environment. This may involve using a built-infunction in the machine learning library, or it may involve writing yourown code.

**c. Preprocess the dataset:**

Once the dataset is loaded into the machine learning environment,

you may need to preprocess it before you can start training andevaluating your model. This may involve cleaning the data, transforming

**Program:**

**In 1:**

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

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

from sklearn.preprocessing import StandardScaler

P a g e| 10from sklearn.metrics import r2\_score,

mean\_absolute\_error,mean\_squared\_error

from sklearn.linear\_model import LinearRegression

from sklearn.linear\_model import Lasso

from sklearn.ensemble import RandomForestRegressor

from sklearn.svm import SVR

import xgboost as xg

%matplotlib inline

import warnings

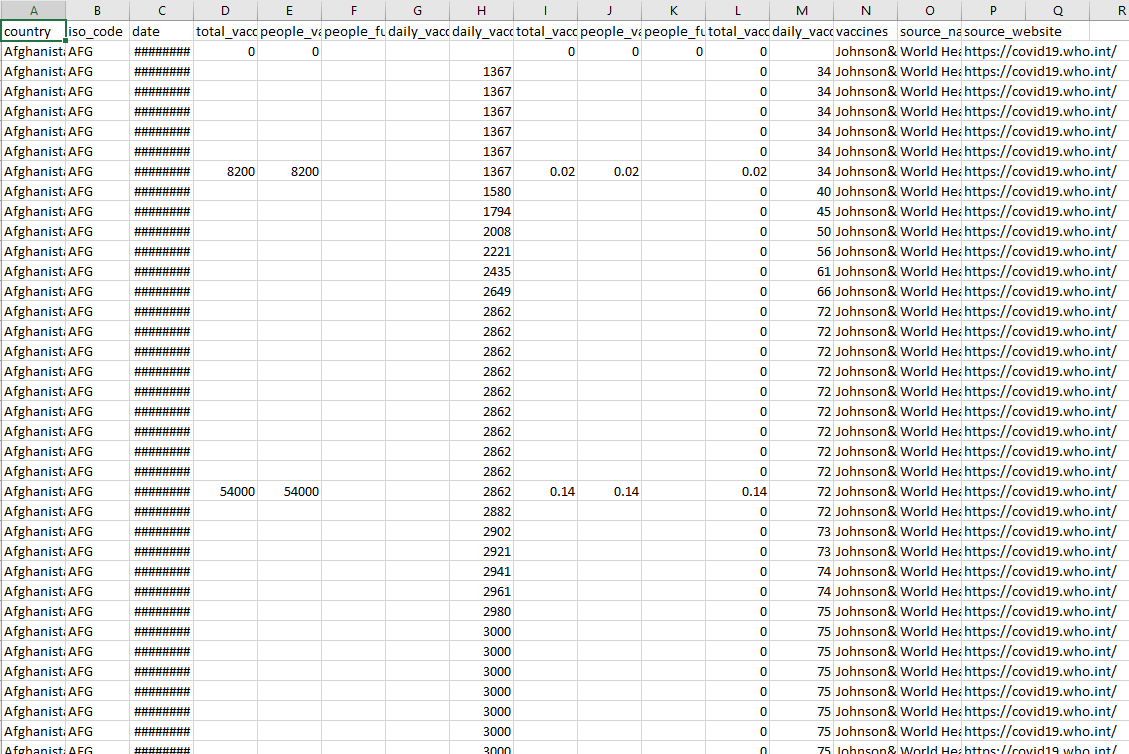
**Loading Dataset:**

**In 2:**

dataset = pd.read\_csv('E:/ country\_vaccinations.csv')

**Data Exploration:**

Dataset:

**Out 2:**

**Preprocessing the Dataset:**

**Step 1 Handling Missing Values:**

* Fills missing values in a specific column with the mean value.

**Step 2 Encoding Categorical Variables:**

* Uses one-hot encoding to convert categorical variables into numerical format.

**Step 3 Feature Scaling:**

* Standardizes numerical features to have zero mean and unit variance.

**Step 4 Column Transformation:**

* Uses Column Transformer to apply different preprocessing techniques to specific columns while passing other columns through as-is.

**Visualisation and Pre-Processing of Data:**

**In 3:**

def get\_multi\_line\_title(title:str, subtitle:str):

return f"{title}<br><sub>{subtitle}</sub>"

def visualize\_column(data: pd.DataFrame, xcolumn: str, ycolumn:str, title:str, colors:str, ylabel="Count", n=None):

hovertemplate ='<br><b>%{x}</b>'+f'<br><b>{ylabel}: </b>'+'%{y}<br><extra></extra>'

data = data.sort\_values(ycolumn, ascending=False).dropna(subset=[ycolumn])

if n is not None:

data = data.iloc[:n]

else:

n = ""

fig = go.Figure(go.Bar( hoverinfo='skip', x=data[xcolumn], y=data[ycolumn], hovertemplate = hovertemplate,

marker=dict(color = data[ycolumn],colorscale=colors ,

),

),

)

fig.update\_layout(

title=title,

xaxis\_title=f"Top {n} {xcolumn.title()}",

yaxis\_title=ylabel,

plot\_bgcolor='rgba(0,0,0,0)',

hovermode="x"

)

fig.show()

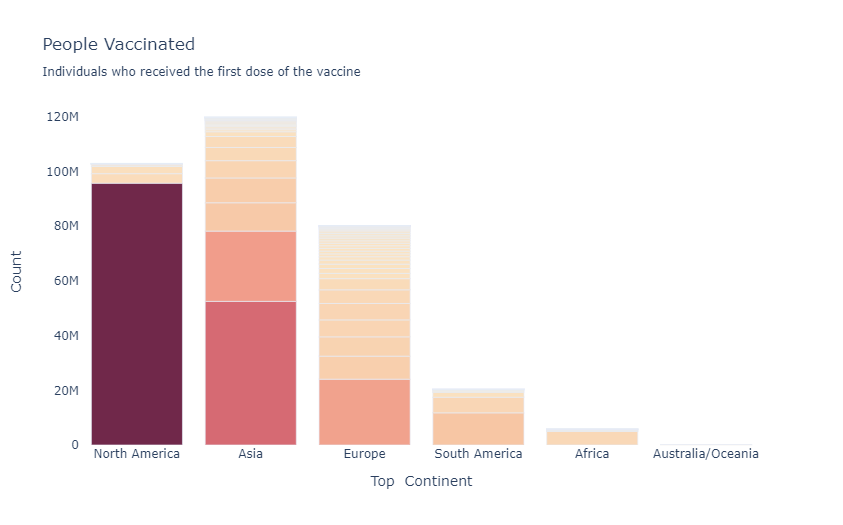
**People Vaccinated-Continent & Country**

**In 4:**

title = get\_multi\_line\_title("People Vaccinated", "Individuals who received the first dose of the vaccine")

visualize\_column(summary.reset\_index(), 'continent', "total\_vaccinations", title, "burgyl")

**Out 4:**

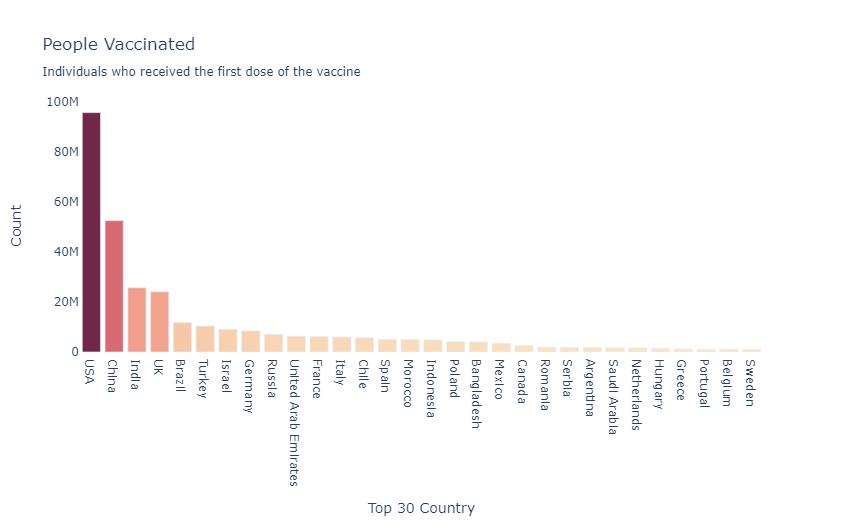


**In 5:**

title = get\_multi\_line\_title("People Vaccinated", "Individuals who received the first dose of the vaccine")

visualize\_column(summary.reset\_index(), 'country', "total\_vaccinations", title, "burgyl", n=30 )

**Out 5:**



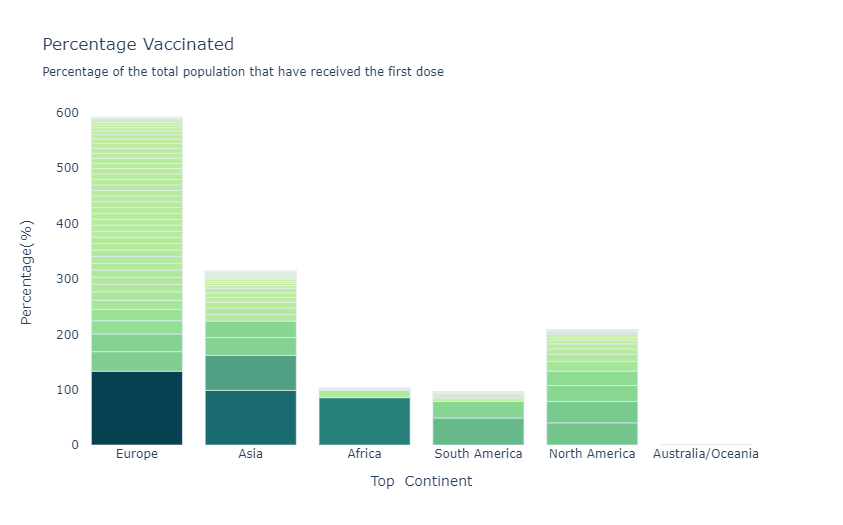
**Percentage Vaccinated-Continent & Country**

**In 6:**

title = get\_multi\_line\_title("Percentage Vaccinated", "Percentage of the total population that have received the first dose")

visualize\_column(summary.reset\_index(), 'continent', "percentage\_vaccinated", title, "emrld", "Percentage(%)")

**Out 6:**

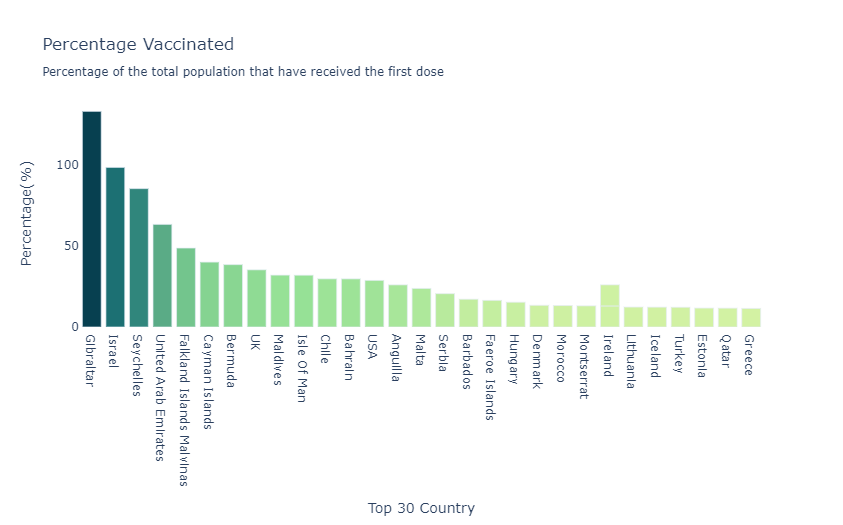


**In 7:**

title = get\_multi\_line\_title("Percentage Vaccinated", "Percentage of the total population that have received the first dose")

visualize\_column(summary.reset\_index(), 'country', "percentage\_vaccinated", title, "emrld", "Percentage(%)", n=30)

**Out 7:**



**Visualising Correlation:**

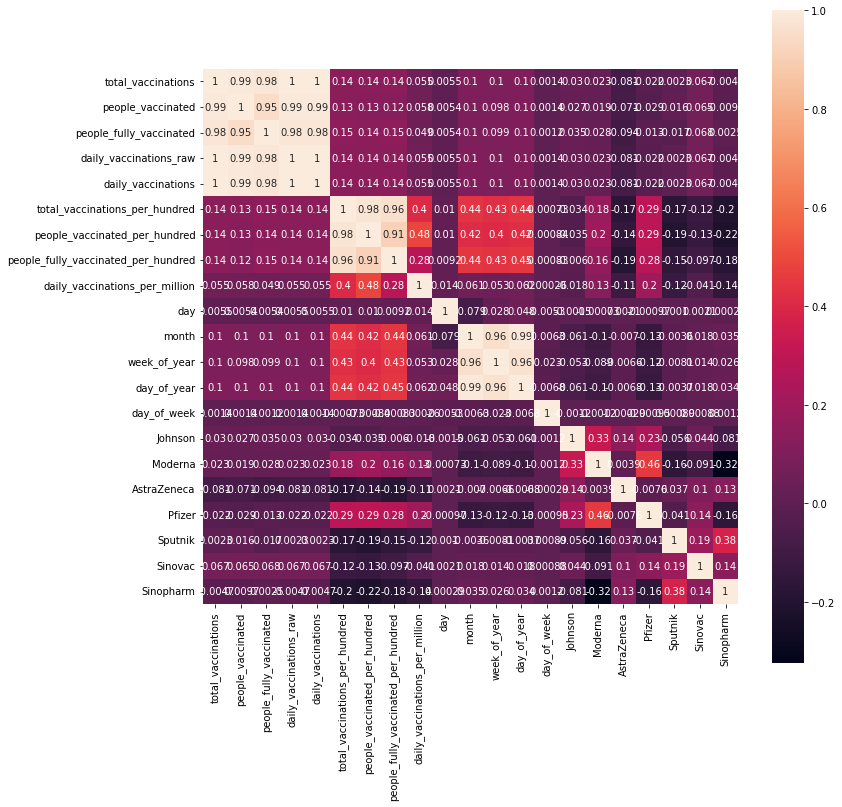
**In 8:**

plt.subplots(figsize=(12, 12))

sns.heatmap(df.corr(), annot=True, square=True)

plt.show()

**Out 8:**



**We will visualize number of records citing each vaccin type**

**In 9:**

d1= pd.DataFrame(df.groupby('Moderna').size())

d1["type"]="Moderna"

d2= pd.DataFrame(df.groupby('Pfizer').size())

d2["type"]="Pfizer"

d3= pd.DataFrame(df.groupby('Johnson').size())

d3["type"]="Johnson"

d4= pd.DataFrame(df.groupby('AstraZeneca').size())

d4["type"]="AstraZeneca"

d5= pd.DataFrame(df.groupby('Sputnik').size())

d5["type"]="Sputnik"

d6= pd.DataFrame(df.groupby('Sinovac').size())

d6["type"]="Sinovac"

d7= pd.DataFrame(df.groupby('Sinopharm').size())

d7["type"]="Sinopharm"

frames = [d1, d2,d3,d4,d5,d6,d7]

df\_vacc = pd.concat(frames)

df\_vacc

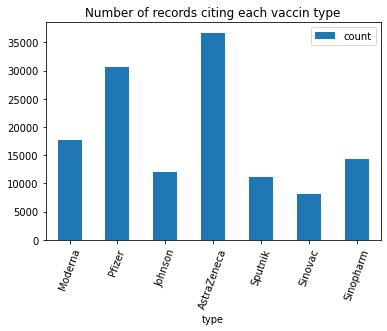
**Out 9:**

| 0 | type |
| --- | --- |
| 0 | 26801 | Moderna |
| 1 | 17695 | Moderna |
| 0 | 13891 | Pfizer |
| 1 | 30605 | Pfizer |
| 0 | 32471 | Johnson |
| 1 | 12025 | Johnson |
| 0 | 7766 | AstraZeneca |
| 1 | 36730 | AstraZeneca |
| 0 | 33413 | Sputnik |
| 1 | 11083 | Sputnik |
| 0 | 36321 | Sinovac |
| 1 | 8175 | Sinovac |
| 0 | 30095 | Sinopharm |
| 1 | 14401 | Sinopharm |

**In 10:**

dfvacc.plot.bar(x="type", y="count", rot=70, title="Number of records citing each vaccin type");

**Out 10:**



**Some common data preprocessing tasks include:**

**Data cleaning:**

This involves identifying and correcting errors andinconsistencies in the data. For example, this may involveremoving duplicate records, correcting typos, and filling in missingvalues.

**Data transformation:**

This involves converting the data into aformat that is suitable for the analysis task. For example, this mayinvolve converting categorical data to numerical data, or scalingthe data to a suitable range.

**Feature engineering:**

This involves creating new features fromthe existing data. For example, this may involve creating featuresthat represent interactions between variables, or features that

represent summary statistics of the data.

**Data integration:**

This involves combining data from multiplesources into a single dataset. This may involve resolvinginconsistencies in the data, such as different data formats or

different variable names.

Data preprocessing is an essential step in many datascience projects. By carefully preprocessing the data, data scientists canimprove the accuracy and reliability of their results.

**Program:**

**daily vaccination per million comparison**

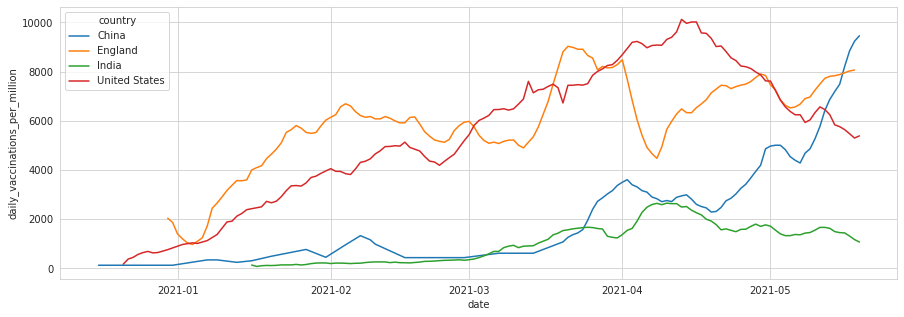
**In 11:**

plt.figure(figsize= (15,5))

sns.lineplot(x= "date",y= "daily\_vaccinations\_per\_million" ,data= x,hue= "country")

plt.show()

**Out 11:**



**People fully vaccinated**

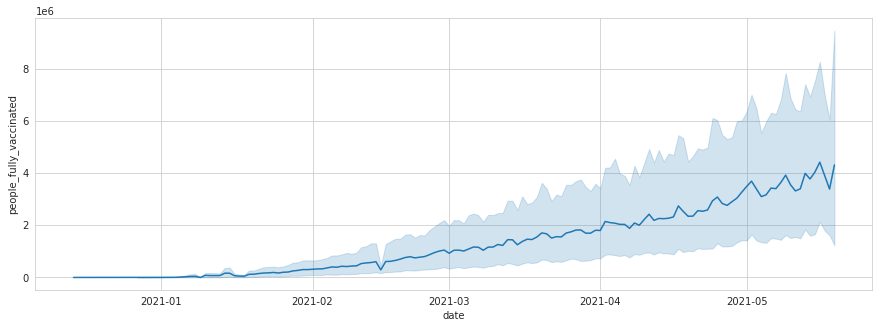
**In 12:**

plt.figure(figsize= (15,5))

sns.lineplot(x= "date",y= "people\_fully\_vaccinated",data= df)

plt.show()

**Out 12:**



total vaccinations

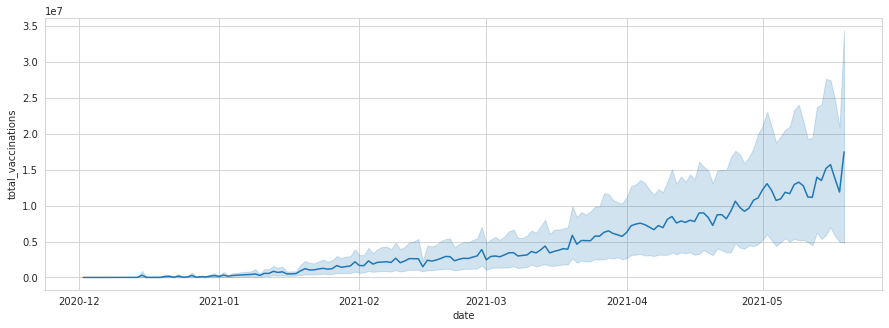
**In 13:**

plt.figure(figsize= (15,5))

sns.lineplot(x= "date",y= "total\_vaccinations",data= df)

plt.show()

**Out 13:**



**Conclusion**

In conclusion, loading and preprocessing a COVID vaccine analysis dataset is a critical and complex task that involves addressing various challenges to ensure accurate and meaningful insights. Through the process of loading and preprocessing the dataset, several key conclusions can be drawn:

1. Data Understanding

2. Data Quality

3. Feature Engineering

4. Categorical Data and Encoding

5. Data Privacy and Security

6. Exploratory Data Analysis (EDA)

7. Data Splitting

8. Collaboration and Expertise

9. Continuous Improvement

successful loading and preprocessing of a COVID vaccine analysis dataset require a combination of technical skills, domain knowledge, and collaboration. By overcoming the challenges associated with data quality, integrity, privacy, and complexity, meaningful insights can be extracted. The preprocessed dataset serves as a solid foundation for building accurate machine learning models and deriving valuable conclusions to inform public health decisions and strategies.