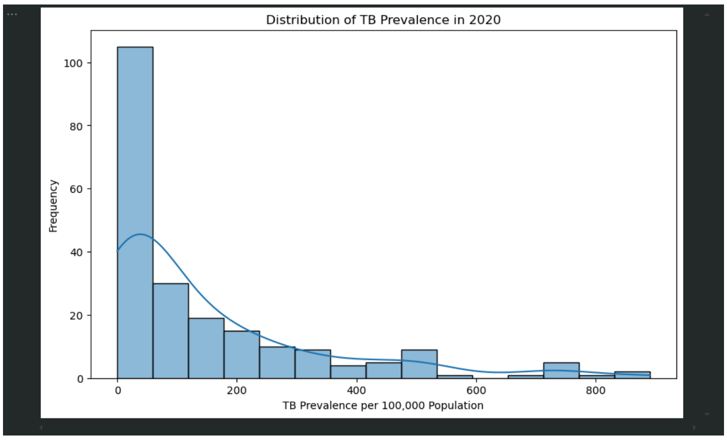
Assignment 2 - Seaborn and matplotlib

1) Report on TB Prevalence Distribution in 2010

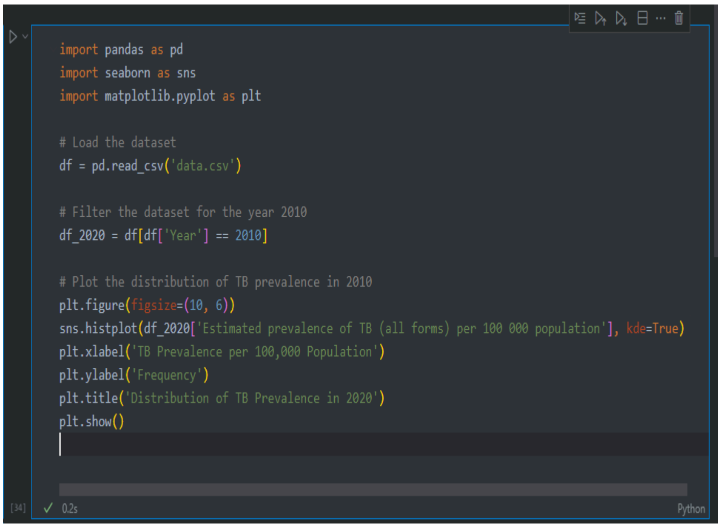
The histogram combined with a Kernel Density Estimate (KDE) curve provides a detailed view of the distribution of TB prevalence rates per 100,000 population for the year 2010. This visualization is crucial for identifying the central tendency and variability of TB prevalence across different regions or countries. The histogram displays the frequency of various TB prevalence rates, allowing us to see how common or rare certain rates are. The KDE curve, which smooths the histogram, helps in understanding the overall distribution shape and highlights any underlying patterns or anomalies.

In this chart, peaks represent prevalence rates that are most common among the data points, while the spread shows how these rates vary. By analyzing this distribution, one can identify whether the TB prevalence is concentrated around specific values or if there is significant variability. This can guide public health strategies and resource allocation by indicating where intervention might be most needed.



Code Explanation

The code snippet begins by loading the dataset from a CSV file into a DataFrame using Pandas. It then filters the data to focus on the year 2010 by selecting rows where the 'Year' column equals 2010. The sns.histplot function from Seaborn is used to create a histogram of the 'Estimated prevalence of TB (all forms) per 100 000 population', augmented with a KDE curve to provide a smooth estimation of the distribution. The plot's labels and title are set using Matplotlib functions to ensure the chart is informative and easy to interpret. Finally, plt.show() renders the plot, presenting a clear visualization of TB prevalence distribution for the year 2010.

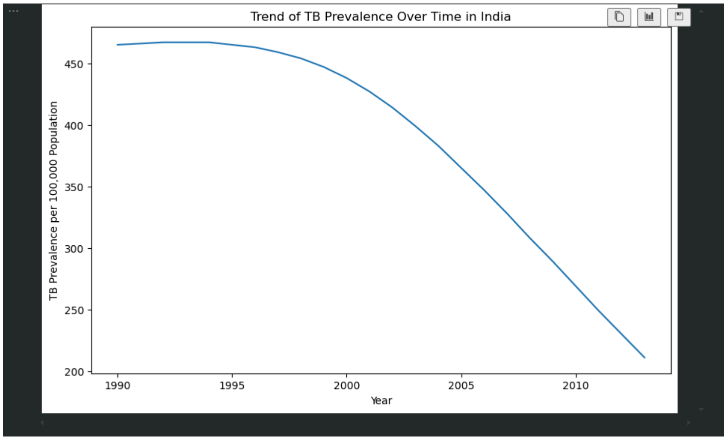


```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the dataset
df = pd.read_csv('data.csv')

# Filter the dataset for the year 2010
df_2020 = df[df['Year'] == 2010]

# Plot the distribution of TB prevalence in 2010
plt.figure(figsize=(10, 6))
sns.histplot(df_2020['Estimated prevalence of TB (all forms) per 100 000 population'], kde=True)
plt.xlabel('TB Prevalence per 100,000 Population')
plt.ylabel('Frequency')
plt.title('Distribution of TB Prevalence in 2020')
plt.show()
```

2) Report on TB Prevalence Trend Over Time in India

The line plot illustrates the trend of TB prevalence rates per 100,000 population in India over time. This visualization provides valuable insights into how the TB prevalence has evolved from year to year. By examining the trend line, one can observe fluctuations in TB prevalence and identify any patterns, such as increasing or decreasing rates. This trend analysis is essential for assessing the effectiveness of public health interventions and understanding how TB control efforts impact prevalence over time.

The plot highlights significant changes in TB prevalence, which can inform policy decisions and resource allocation. For example, if the prevalence shows a decreasing trend, it might indicate successful control measures, while an increasing trend could signal emerging challenges that need to be addressed.



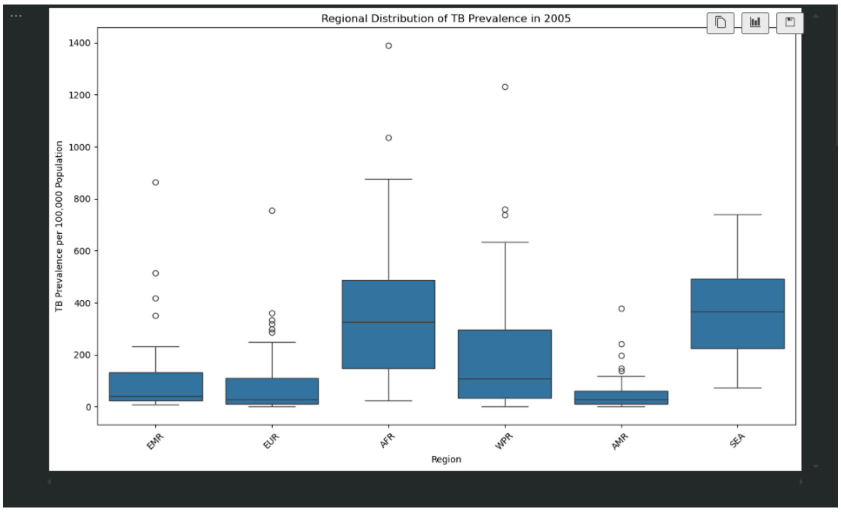
Code Explanation

The code snippet starts by loading the dataset from a CSV file into a DataFrame using Pandas. It then filters the data to focus specifically on India by selecting rows where the 'Country or territory name' column matches 'India'. The sns.lineplot function from Seaborn is used to create a line plot that visualizes the trend of TB prevalence per 100,000 population over the years. The x-axis represents the years, while the y-axis shows the TB prevalence rates. The plot is customized with labels and a title using Matplotlib functions for clarity. Finally, plt.show() renders the plot, providing a clear view of how TB prevalence in India has changed over time.

3) Report on Regional Distribution of TB Prevalence in 2005

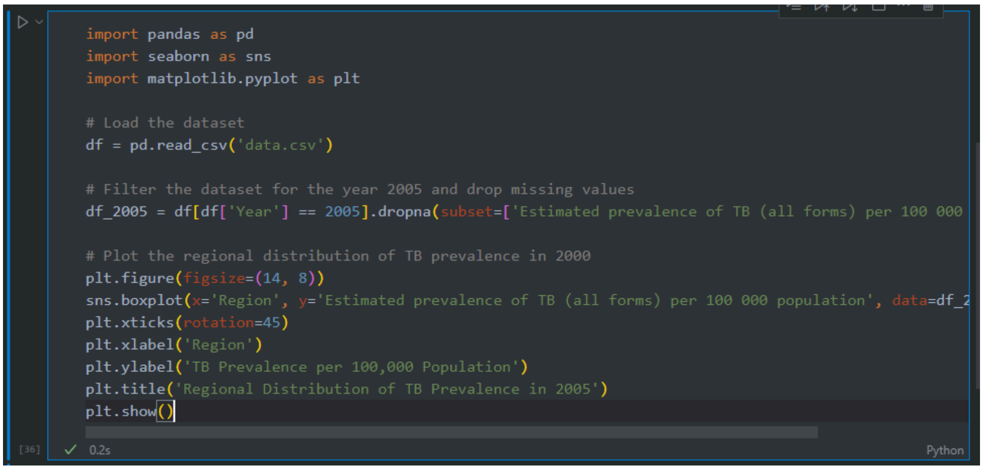
The box plot provides an overview of the regional distribution of TB prevalence rates per 100,000 population for the year 2005. This visualization allows us to compare TB prevalence across different regions, showing the range, median, and variability of prevalence rates within each region. The box plot’s use of quartiles and outliers helps identify regions with higher or lower prevalence rates, as well as regions with significant variability.

By examining the distribution of TB prevalence across regions, public health officials and researchers can identify which regions are experiencing higher TB rates and may require targeted interventions. Additionally, the presence of outliers in some regions may highlight specific areas with unusually high or low prevalence that warrant further investigation.



Code Explanation

The code snippet starts by loading the dataset from a CSV file into a Pandas DataFrame. It then filters the data to include only records from the year 2005 and removes rows with missing values in the 'Estimated prevalence of TB (all forms) per 100 000 population' column. The sns.boxplot function from Seaborn is used to create a box plot that visualizes TB prevalence across different regions. The x-axis represents the regions, and the y-axis shows the TB prevalence rates. The plt.xticks(rotation=45) function rotates the x-axis labels for better readability. The plot is customized with labels and a title for clarity. Finally, plt.show() renders the plot, providing a clear comparison of TB prevalence across regions for the year 2005.

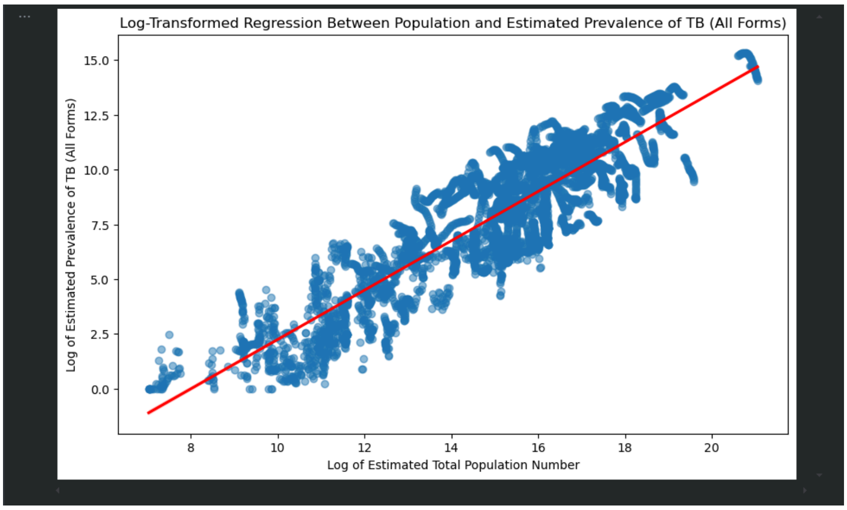


```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the dataset
df = pd.read_csv('data.csv')

# Filter the dataset for the year 2005 and drop missing values
df_2005 = df[df['Year'] == 2005].dropna(subset=['Estimated prevalence of TB (all forms) per 100 000

# Plot the regional distribution of TB prevalence in 2000
plt.figure(figsize=(14, 8))
sns.boxplot(x='Region', y='Estimated prevalence of TB (all forms) per 100 000 population', data=df_2
plt.xticks(rotation=45)
plt.xlabel('Region')
plt.ylabel('TB Prevalence per 100,000 Population')
plt.title('Regional Distribution of TB Prevalence in 2005')
plt.show()
```

4) Report on Log-Transformed Regression Between Population and TB Prevalence

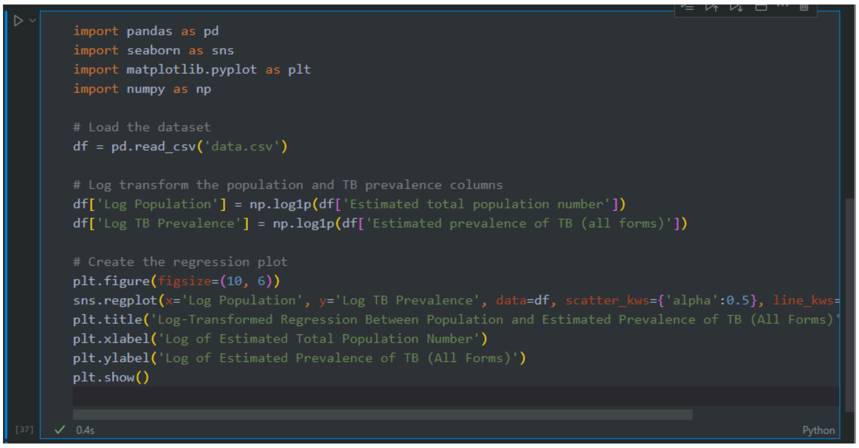
The regression plot illustrates the relationship between the log-transformed values of population and TB prevalence. By applying a log transformation to both the population and TB prevalence columns, this plot addresses potential skewness in the data and highlights the relationship in a more linear fashion. The scatter points, which represent individual data entries, are overlaid with a regression line that shows the trend and strength of the relationship between the two variables.

The log transformation helps in managing wide-ranging values and can reveal patterns that might be obscured in a linear scale. In this plot, a positive or negative slope of the regression line can indicate how changes in population size are associated with changes in TB prevalence, providing insights into how TB burden scales with population.



Code Explanation

The code snippet begins by loading the dataset from a CSV file into a Pandas DataFrame. It then applies a log transformation to the 'Estimated total population number' and 'Estimated prevalence of TB (all forms)' columns using np.log1p(), which calculates the natural logarithm of 1 plus the value. This transformation is used to stabilize variance and make the relationship more linear. The sns.regplot function from Seaborn creates a regression plot with log-transformed variables, showing both the scatter of data points and the regression line. The scatter_kws={'alpha':0.5} parameter makes the scatter points semi-transparent, and line_kws={'color':'red'} sets the color of the regression line. The plot is customized with labels and a title for better interpretation. Finally, plt.show() displays the plot, revealing the relationship between log-transformed population and TB prevalence.

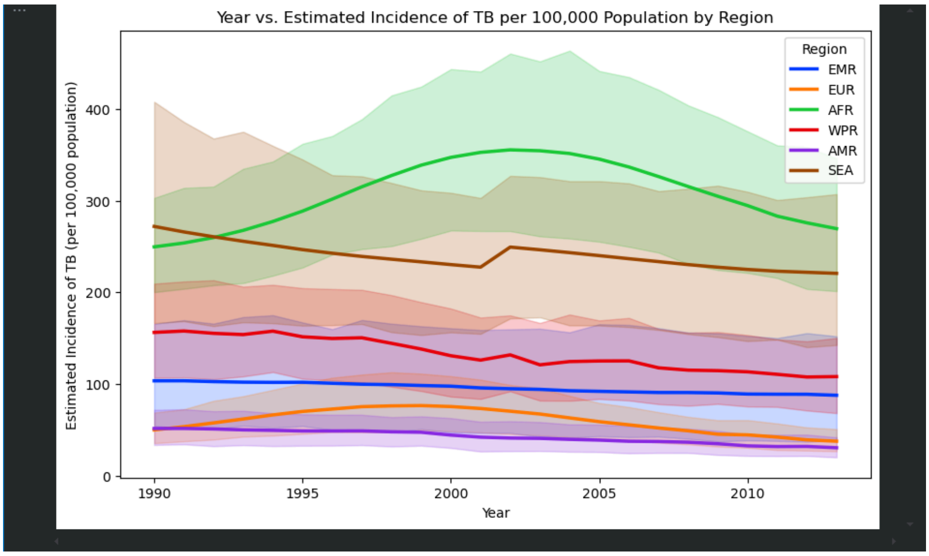


```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

# Load the dataset
df = pd.read_csv('data.csv')

# Log transform the population and TB prevalence columns
df['Log Population'] = np.log1p(df['Estimated total population number'])
df['Log TB Prevalence'] = np.log1p(df['Estimated prevalence of TB (all forms)'])

# Create the regression plot
plt.figure(figsize=(10, 6))
sns.regplot(x='Log Population', y='Log TB Prevalence', data=df, scatter_kws={'alpha':0.5}, line_kws=
plt.title('Log-Transformed Regression Between Population and Estimated Prevalence of TB (All Forms)'
plt.xlabel('Log of Estimated Total Population Number')
plt.ylabel('Log of Estimated Prevalence of TB (All Forms)')
plt.show()
```

5) Report on TB Incidence Over Time by Region

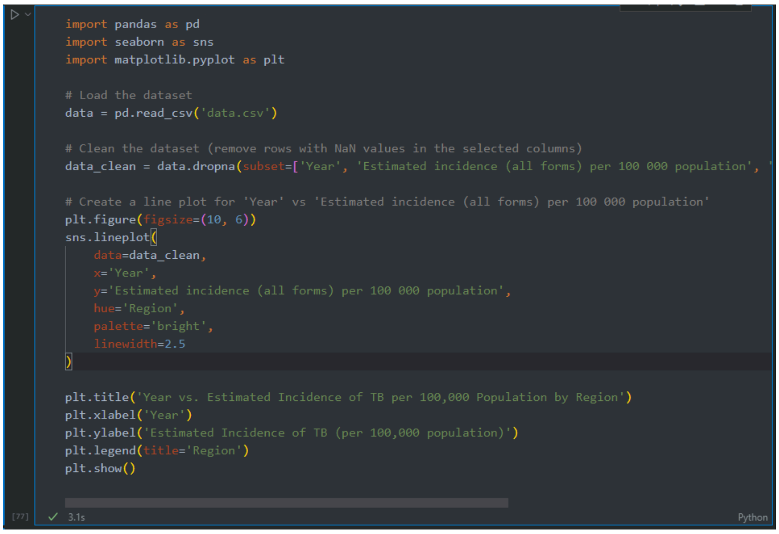
The line plot displays the trend of estimated TB incidence rates per 100,000 population over the years, segmented by region. This visualization allows us to track how TB incidence has changed over time in different regions and compare these trends across regions. By using different lines for each region, the plot effectively highlights regional variations and temporal patterns in TB incidence rates.

The plot helps identify trends, such as whether TB incidence is increasing or decreasing in specific regions. It also facilitates comparison between regions, revealing which regions have higher or lower TB incidence rates and how these rates evolve over time. This information is crucial for public health planning and resource allocation, as it highlights areas that may need targeted interventions.



Code Explanation

The code snippet starts by loading the dataset from a CSV file into a Pandas DataFrame. It then cleans the dataset by removing rows with missing values in critical columns: 'Year', 'Estimated incidence (all forms) per 100 000 population', 'Estimated prevalence of TB (all forms)', and 'Region'. The sns.lineplot function from Seaborn is used to create a line plot, where the x-axis represents the year, and the y-axis shows the estimated TB incidence per 100,000 population. Each line in the plot corresponds to a different region, distinguished by the 'Region' column. The palette='bright' parameter ensures distinct colors for each region, and linewidth=2.5 makes the lines more prominent. The plot is labeled and titled for clarity, and plt.legend(title='Region') provides a legend to identify each region. Finally, plt.show() renders the plot, offering a comprehensive view of TB incidence trends across regions over time.



```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the dataset
data = pd.read_csv('data.csv')

# Clean the dataset (remove rows with NaN values in the selected columns)
data_clean = data.dropna(subset=['Year', 'Estimated incidence (all forms) per 100 000 population', '

# Create a line plot for 'Year' vs 'Estimated incidence (all forms) per 100 000 population'
plt.figure(figsize=(10, 6))
sns.lineplot(
    data=data_clean,
    x='Year',
    y='Estimated incidence (all forms) per 100 000 population',
    hue='Region',
    palette='bright',
    linewidth=2.5
)

plt.title('Year vs. Estimated Incidence of TB per 100,000 Population by Region')
plt.xlabel('Year')
plt.ylabel('Estimated Incidence of TB (per 100,000 population)')
plt.legend(title='Region')
plt.show()
```

Resources:

dataset - TB_Burden_Country.csv.zip

Pandas Development Team. (2024). *Pandas* (Version 2.0.0) [Software]. Available from https://pandas.pydata.org
Waskom, M. (2024). *Seaborn* (Version 0.12.2) [Software]. Available from https://seaborn.pydata.org
Hunter, J. D. (2024). *Matplotlib* (Version 3.7.0) [Software]. Available from https://matplotlib.org