Balaji_Pamidi_Final_project_002644804

April 10, 2024

1 Data Analysis Report: Uncovering Trends in Weekly Provisional Counts of Deaths.

1.1 1. Data collected from the Data Gov.

1.1.1 link : https://catalog.data.gov/dataset/weekly-counts-of-deaths-by-state-and-select-causes-2019-2020

[446]:
```
# Calling the .CSV Dataset file using the pandas Library.

import pandas as pd

# Assuming the dataset is in CSV format and located in the same directory as␣
 ↪your script or notebook
file_path = "/Users/balajipamidi/Desktop/Final Project /
 ↪Weekly_Provisional_Counts_of_Deaths_by_State_and_Select_Causes__2020-2023␣
 ↪(1) copy 2.csv"

# Load the dataset into a Pandas DataFrame
df = pd.read_csv(file_path)

# Display the first few rows of the DataFrame to verify that the data has been␣
 ↪loaded correctly
df
```

[446]:

	Data As Of	Jurisdiction of Occurrence	MMWR Year	MMWR Week	\
0	09/27/2023	United States	2020	1	
1	09/27/2023	United States	2020	2	
2	09/27/2023	United States	2020	3	
3	09/27/2023	United States	2020	4	
4	09/27/2023	United States	2020	5	
…	…	…	…	…	
10471	09/27/2023	Puerto Rico	2023	33	
10472	09/27/2023	Puerto Rico	2023	34	
10473	09/27/2023	Puerto Rico	2023	35	
10474	09/27/2023	Puerto Rico	2023	36	
10475	09/27/2023	Puerto Rico	2023	37	

	Week Ending Date	All Cause	Natural Cause	Septicemia (A40-A41)	\
0	2020-01-04	60179	55010	843.0	
1	2020-01-11	60735	55755	861.0	
2	2020-01-18	59363	54516	829.0	
3	2020-01-25	59162	54401	828.0	
4	2020-02-01	58834	54001	811.0	
…	…	…	…	…	
10471	2023-08-19	612	590	NaN	
10472	2023-08-26	657	624	15.0	
10473	2023-09-02	580	552	16.0	
10474	2023-09-09	533	516	10.0	
10475	2023-09-16	453	453	NaN	

	Malignant neoplasms (C00-C97)	Diabetes mellitus (E10-E14)	…	\
0	11567.0	1829.0	…	
1	11963.0	1942.0	…	
2	11701.0	1819.0	…	
3	11879.0	1864.0	…	
4	11963.0	1828.0	…	
…	…	…	…	
10471	92.0	48.0	…	
10472	110.0	48.0	…	
10473	97.0	70.0	…	
10474	81.0	50.0	…	
10475	85.0	40.0	…	

	flag_alz	flag_inflpn	flag_clrd	flag_otherresp	flag_nephr	\
0	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	
…	…	…	…	…	…	
10471	NaN	NaN	NaN	NaN	NaN	
10472	NaN	NaN	NaN	NaN	NaN	
10473	NaN	NaN	NaN	Suppressed (counts 1-9)	NaN	
10474	NaN	NaN	NaN	NaN	NaN	
10475	NaN	NaN	NaN	NaN	NaN	

	flag_otherunk	flag_hd	flag_stroke	flag_cov19mcod	flag_cov19ucod
0	NaN	NaN	NaN	NaN	NaN
1	NaN	NaN	NaN	NaN	NaN
2	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN
…	…	…	…	…	…
10471	NaN	NaN	NaN	NaN	NaN

```
10472            NaN      NaN          NaN             NaN             NaN
10473            NaN      NaN          NaN             NaN             NaN
10474            NaN      NaN          NaN             NaN             NaN
10475            NaN      NaN          NaN             NaN             NaN

[10476 rows x 35 columns]
```

[447]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10476 entries, 0 to 10475
Data columns (total 35 columns):
 #   Column
Non-Null Count  Dtype
---  ------
--------------  -----
 0   Data As Of
10476 non-null  object
 1   Jurisdiction of Occurrence
10476 non-null  object
 2   MMWR Year
10476 non-null  int64
 3   MMWR Week
10476 non-null  int64
 4   Week Ending Date
10476 non-null  object
 5   All Cause
10476 non-null  int64
 6   Natural Cause
10476 non-null  int64
 7   Septicemia (A40-A41)
6083 non-null   float64
 8   Malignant neoplasms (C00-C97)
10466 non-null  float64
 9   Diabetes mellitus (E10-E14)
8234 non-null   float64
 10  Alzheimer disease (G30)
8732 non-null   float64
 11  Influenza and pneumonia (J09-J18)
6241 non-null   float64
 12  Chronic lower respiratory diseases (J40-J47)
8965 non-null   float64
 13  Other diseases of respiratory system (J00-J06,J30-J39,J67,J70-J98)
6305 non-null   float64
 14  Nephritis, nephrotic syndrome and nephrosis (N00-N07,N17-N19,N25-N27)
6716 non-null   float64
 15  Symptoms, signs and abnormal clinical and laboratory findings, not
elsewhere classified (R00-R99)  5813 non-null   float64
```

```
 16  Diseases of heart (I00-I09,I11,I13,I20-I51)
10464 non-null  float64
 17  Cerebrovascular diseases (I60-I69)
9031 non-null   float64
 18  COVID-19 (U071, Multiple Cause of Death)
8721 non-null   float64
 19  COVID-19 (U071, Underlying Cause of Death)
8180 non-null   float64
 20  flag_allcause
0 non-null      float64
 21  flag_natcause
0 non-null      float64
 22  flag_sept
4393 non-null   object
 23  flag_neopl
10 non-null     object
 24  flag_diab
2242 non-null   object
 25  flag_alz
1744 non-null   object
 26  flag_inflpn
4235 non-null   object
 27  flag_clrd
1511 non-null   object
 28  flag_otherresp
4171 non-null   object
 29  flag_nephr
3760 non-null   object
 30  flag_otherunk
4663 non-null   object
 31  flag_hd
12 non-null     object
 32  flag_stroke
1445 non-null   object
 33  flag_cov19mcod
1755 non-null   object
 34  flag_cov19ucod
2296 non-null   object
dtypes: float64(15), int64(4), object(16)
memory usage: 2.8+ MB
```

1.2 2. Data Cleaning

1.3 First Method is Renaming the columns Headings

[448]:
```
#Columns Headings
df.columns
```

[448]:
```
Index(['Data As Of', 'Jurisdiction of Occurrence', 'MMWR Year', 'MMWR Week',
       'Week Ending Date', 'All Cause', 'Natural Cause',
       'Septicemia (A40-A41)', 'Malignant neoplasms (C00-C97)',
       'Diabetes mellitus (E10-E14)', 'Alzheimer disease (G30)',
       'Influenza and pneumonia (J09-J18)',
       'Chronic lower respiratory diseases (J40-J47)',
       'Other diseases of respiratory system (J00-J06,J30-J39,J67,J70-J98)',
       'Nephritis, nephrotic syndrome and nephrosis (N00-N07,N17-N19,N25-N27)',
       'Symptoms, signs and abnormal clinical and laboratory findings, not
elsewhere classified (R00-R99)',
       'Diseases of heart (I00-I09,I11,I13,I20-I51)',
       'Cerebrovascular diseases (I60-I69)',
       'COVID-19 (U071, Multiple Cause of Death)',
       'COVID-19 (U071, Underlying Cause of Death)', 'flag_allcause',
       'flag_natcause', 'flag_sept', 'flag_neopl', 'flag_diab', 'flag_alz',
       'flag_inflpn', 'flag_clrd', 'flag_otherresp', 'flag_nephr',
       'flag_otherunk', 'flag_hd', 'flag_stroke', 'flag_cov19mcod',
       'flag_cov19ucod'],
      dtype='object')
```

[449]:
```
df.rename(columns={"Septicemia (A40-A41)":"Septicemia",
                   'Malignant neoplasms (C00-C97)':'Malignant neoplasms',
                   'Diabetes mellitus (E10-E14)':'Diabetes mellitus',
                   'Alzheimer disease (G30)':'Alzheimer disease',
                   'Influenza and pneumonia (J09-J18)':'Influenza and␣
 ↪pneumonia',
                   'Chronic lower respiratory diseases (J40-J47)':'Chronic␣
 ↪lower respiratory diseases',
                   'Other diseases of respiratory system␣
 ↪(J00-J06,J30-J39,J67,J70-J98)':'Other diseases of respiratory system',
                   'Nephritis, nephrotic syndrome and nephrosis␣
 ↪(N00-N07,N17-N19,N25-N27)':'Nephritis',
                   'Symptoms, signs and abnormal clinical and laboratory␣
 ↪findings, not elsewhere classified (R00-R99)':'not elsewhere classified',
                   'Diseases of heart (I00-I09,I11,I13,I20-I51)':'Diseases of␣
 ↪heart',
                   'Cerebrovascular diseases (I60-I69)':'Cerebrovascular␣
 ↪diseases',
                   'COVID-19 (U071, Multiple Cause of Death)':
 ↪'COVID-19_Multiple Cause of Death',
```

```
                    'COVID-19 (U071, Underlying Cause of Death)':
 ↪'COVID-19_Underlying Cause of Death'}, inplace=True)

#New columns headings
df.columns
```

[449]:
```
Index(['Data As Of', 'Jurisdiction of Occurrence', 'MMWR Year', 'MMWR Week',
       'Week Ending Date', 'All Cause', 'Natural Cause', 'Septicemia',
       'Malignant neoplasms', 'Diabetes mellitus', 'Alzheimer disease',
       'Influenza and pneumonia', 'Chronic lower respiratory diseases',
       'Other diseases of respiratory system', 'Nephritis',
       'not elsewhere classified', 'Diseases of heart',
       'Cerebrovascular diseases', 'COVID-19_Multiple Cause of Death',
       'COVID-19_Underlying Cause of Death', 'flag_allcause', 'flag_natcause',
       'flag_sept', 'flag_neopl', 'flag_diab', 'flag_alz', 'flag_inflpn',
       'flag_clrd', 'flag_otherresp', 'flag_nephr', 'flag_otherunk', 'flag_hd',
       'flag_stroke', 'flag_cov19mcod', 'flag_cov19ucod'],
      dtype='object')
```

[450]:
```
# After Renaming the columns we see the new columns names with Datatype for␣
 ↪each column.
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10476 entries, 0 to 10475
Data columns (total 35 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Data As Of                            10476 non-null  object
 1   Jurisdiction of Occurrence            10476 non-null  object
 2   MMWR Year                             10476 non-null  int64
 3   MMWR Week                             10476 non-null  int64
 4   Week Ending Date                      10476 non-null  object
 5   All Cause                             10476 non-null  int64
 6   Natural Cause                         10476 non-null  int64
 7   Septicemia                            6083 non-null   float64
 8   Malignant neoplasms                   10466 non-null  float64
 9   Diabetes mellitus                     8234 non-null   float64
 10  Alzheimer disease                     8732 non-null   float64
 11  Influenza and pneumonia               6241 non-null   float64
 12  Chronic lower respiratory diseases    8965 non-null   float64
 13  Other diseases of respiratory system  6305 non-null   float64
 14  Nephritis                             6716 non-null   float64
 15  not elsewhere classified              5813 non-null   float64
 16  Diseases of heart                     10464 non-null  float64
 17  Cerebrovascular diseases              9031 non-null   float64
 18  COVID-19_Multiple Cause of Death      8721 non-null   float64
 19  COVID-19_Underlying Cause of Death    8180 non-null   float64
```

```
 20  flag_allcause                          0 non-null      float64
 21  flag_natcause                          0 non-null      float64
 22  flag_sept                              4393 non-null   object
 23  flag_neopl                             10 non-null     object
 24  flag_diab                              2242 non-null   object
 25  flag_alz                               1744 non-null   object
 26  flag_inflpn                            4235 non-null   object
 27  flag_clrd                              1511 non-null   object
 28  flag_otherresp                         4171 non-null   object
 29  flag_nephr                             3760 non-null   object
 30  flag_otherunk                          4663 non-null   object
 31  flag_hd                                12 non-null     object
 32  flag_stroke                            1445 non-null   object
 33  flag_cov19mcod                         1755 non-null   object
 34  flag_cov19ucod                         2296 non-null   object
dtypes: float64(15), int64(4), object(16)
memory usage: 2.8+ MB
```

1.4 Second Method is Finding the missing values.

[451]:
```
# To check for Missing values
df.isnull().sum()
```

[451]:
```
Data As Of                                0
Jurisdiction of Occurrence                0
MMWR Year                                 0
MMWR Week                                 0
Week Ending Date                          0
All Cause                                 0
Natural Cause                             0
Septicemia                             4393
Malignant neoplasms                      10
Diabetes mellitus                      2242
Alzheimer disease                      1744
Influenza and pneumonia                4235
Chronic lower respiratory diseases     1511
Other diseases of respiratory system   4171
Nephritis                              3760
not elsewhere classified               4663
Diseases of heart                        12
Cerebrovascular diseases               1445
COVID-19_Multiple Cause of Death       1755
COVID-19_Underlying Cause of Death     2296
flag_allcause                         10476
flag_natcause                         10476
flag_sept                              6083
flag_neopl                            10466
```

```
flag_diab                          8234
flag_alz                           8732
flag_inflpn                        6241
flag_clrd                          8965
flag_otherresp                     6305
flag_nephr                         6716
flag_otherunk                      5813
flag_hd                           10464
flag_stroke                        9031
flag_cov19mcod                     8721
flag_cov19ucod                     8180
dtype: int64
```

1.4.1 Dropping some columns due more missing values than threshold value is 20%.

[452]:
```
# Calculate the threshold number of missing values(Here i have consider that if␣
 ↪a column have more than 20% of missing values then the columns id dropped) )
threshold_columns = int(0.2 * df.shape[0])

# Drop columns with more than the threshold number of missing values
df.dropna(axis=1, thresh=df.shape[0] - threshold_columns, inplace=True)

# Now columns with more than 20% missing values have been dropped from the␣
 ↪DataFrame
#df = df.drop(df.columns[df.apply(lambda col: col.isna().sum() > 2000)], axis=1)
df.shape
```

[452]:
```
(10476, 13)
```

[453]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10476 entries, 0 to 10475
Data columns (total 13 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Data As Of                            10476 non-null  object
 1   Jurisdiction of Occurrence            10476 non-null  object
 2   MMWR Year                             10476 non-null  int64
 3   MMWR Week                             10476 non-null  int64
 4   Week Ending Date                      10476 non-null  object
 5   All Cause                             10476 non-null  int64
 6   Natural Cause                         10476 non-null  int64
 7   Malignant neoplasms                   10466 non-null  float64
 8   Alzheimer disease                     8732 non-null   float64
 9   Chronic lower respiratory diseases    8965 non-null   float64
 10  Diseases of heart                     10464 non-null  float64
 11  Cerebrovascular diseases              9031 non-null   float64
```

```
 12  COVID-19_Multiple Cause of Death    8721 non-null   float64
dtypes: float64(6), int64(4), object(3)
memory usage: 1.0+ MB
```

1.5 Third Method is Imputation using to fill the missing values.

[454]:
```
#Here we have used the Mean value of each column and filled the missing values␣
 ↪of that paticular column.
# Calculate the mean for each specified column
mean_malignant_neoplasms = df['Malignant neoplasms'].mean()
mean_alzheimer_disease = df['Alzheimer disease'].mean()
mean_chronic_respiratory_diseases = df['Chronic lower respiratory diseases'].
 ↪mean()
mean_diseases_of_heart = df['Diseases of heart'].mean()
mean_cerebrovascular_diseases = df['Cerebrovascular diseases'].mean()
mean_covid_multiple_cause_of_death = df['COVID-19_Multiple Cause of Death'].
 ↪mean()


# Fill missing values in each specified column with the respective mean
df['Malignant neoplasms'].fillna(mean_malignant_neoplasms, inplace=True)
df['Alzheimer disease'].fillna(mean_alzheimer_disease, inplace=True)
df['Chronic lower respiratory diseases'].
 ↪fillna(mean_chronic_respiratory_diseases, inplace=True)
df['Diseases of heart'].fillna(mean_diseases_of_heart, inplace=True)
df['Cerebrovascular diseases'].fillna(mean_cerebrovascular_diseases,␣
 ↪inplace=True)
df['COVID-19_Multiple Cause of Death'].
 ↪fillna(mean_covid_multiple_cause_of_death, inplace=True)


# Now the DataFrame 'df' will have missing values in these columns filled with␣
 ↪their respective means

df.isnull().sum()
```

[454]:
```
Data As Of                          0
Jurisdiction of Occurrence          0
MMWR Year                           0
MMWR Week                           0
Week Ending Date                    0
All Cause                           0
Natural Cause                       0
Malignant neoplasms                 0
Alzheimer disease                   0
Chronic lower respiratory diseases  0
Diseases of heart                   0
```

```
Cerebrovascular diseases                0
COVID-19_Multiple Cause of Death        0
dtype: int64
```

1.6 Fourth Method is Finding and deleting the Duplicate values.

[455]:
```
# Find duplicate values in both rows and columns
duplicate_values = df[df.duplicated(keep=False)]

# Display duplicate values if any are found
if not duplicate_values.empty:
    print("Duplicate Values in Both Rows and Columns:")
    print(duplicate_values)
else:
    print("No duplicate values found in both rows and columns.")

df
```

```
No duplicate values found in both rows and columns.
```

[455]:

	Data As Of	Jurisdiction of Occurrence	MMWR Year	MMWR Week	\
0	09/27/2023	United States	2020	1	
1	09/27/2023	United States	2020	2	
2	09/27/2023	United States	2020	3	
3	09/27/2023	United States	2020	4	
4	09/27/2023	United States	2020	5	
…	…	…	…	…	
10471	09/27/2023	Puerto Rico	2023	33	
10472	09/27/2023	Puerto Rico	2023	34	
10473	09/27/2023	Puerto Rico	2023	35	
10474	09/27/2023	Puerto Rico	2023	36	
10475	09/27/2023	Puerto Rico	2023	37	

	Week Ending Date	All Cause	Natural Cause	Malignant neoplasms	\
0	2020-01-04	60179	55010	11567.0	
1	2020-01-11	60735	55755	11963.0	
2	2020-01-18	59363	54516	11701.0	
3	2020-01-25	59162	54401	11879.0	
4	2020-02-01	58834	54001	11963.0	
…	…	…	…	…	
10471	2023-08-19	612	590	92.0	
10472	2023-08-26	657	624	110.0	
10473	2023-09-02	580	552	97.0	
10474	2023-09-09	533	516	81.0	
10475	2023-09-16	453	453	85.0	

	Alzheimer disease	Chronic lower respiratory diseases	\

```
0                 2537.0                                3501.0
1                 2566.0                                3708.0
2                 2491.0                                3526.0
3                 2517.0                                3403.0
4                 2480.0                                3313.0
…                  …                                    …
10471               48.0                                  15.0
10472               64.0                                  15.0
10473               62.0                                  14.0
10474               47.0                                  19.0
10475               25.0                                  13.0

       Diseases of heart  Cerebrovascular diseases  \
0                14204.0                    3110.0
1                13911.0                    3189.0
2                13593.0                    3256.0
3                13612.0                    3185.0
4                13465.0                    3084.0
…                  …                        …
10471              112.0                      27.0
10472              120.0                      30.0
10473               98.0                      21.0
10474               99.0                      18.0
10475               79.0                      21.0

       COVID-19_Multiple Cause of Death
0                                   0.0
1                                   1.0
2                                   2.0
3                                   3.0
4                                   0.0
…                                   …
10471                              15.0
10472                              21.0
10473                              16.0
10474                              16.0
10475                              11.0

[10476 rows x 13 columns]
```

1.7 Fifth Method : Finding the unique values

[456]:
```
#inding the unquie values in column the "Jurisdiction of Occurrence"

unique_values = df['Jurisdiction of Occurrence'].unique()

# Print the unique values
```

```
print(unique_values)
```

```
['United States' 'Alabama' 'Alaska' 'Arizona' 'Arkansas' 'California'
 'Colorado' 'Connecticut' 'Delaware' 'District of Columbia' 'Florida'
 'Georgia' 'Hawaii' 'Idaho' 'Illinois' 'Indiana' 'Iowa' 'Kansas'
 'Kentucky' 'Louisiana' 'Maine' 'Maryland' 'Massachusetts' 'Michigan'
 'Minnesota' 'Mississippi' 'Missouri' 'Montana' 'Nebraska' 'Nevada'
 'New Hampshire' 'New Jersey' 'New Mexico' 'New York' 'New York City'
 'North Carolina' 'North Dakota' 'Ohio' 'Oklahoma' 'Oregon' 'Pennsylvania'
 'Rhode Island' 'South Carolina' 'South Dakota' 'Tennessee' 'Texas' 'Utah'
 'Vermont' 'Virginia' 'Washington' 'West Virginia' 'Wisconsin' 'Wyoming'
 'Puerto Rico']
```

1.7.1 We can see that the columns Contains all States of USA. Except ('New York' 'New York City'). We are going to replace the New York City values to New York.

[457]:
```
# Assuming df is your DataFrame with the 'Jurisdiction of Occurrence' column
# Replace df with your actual DataFrame name

# Replace 'New York City' with 'New York'
df['Jurisdiction of Occurrence'].replace('New York City', 'New York',␣
 ↪inplace=True)
#new unique values
unique_values = df['Jurisdiction of Occurrence'].unique()
# Count the occurrences of each state
state_counts = df['Jurisdiction of Occurrence'].value_counts()

# Print the unique values and their counts
print("Counts of each state:")
# Print the unique values
print(unique_values)
print(state_counts)
```

```
Counts of each state:
['United States' 'Alabama' 'Alaska' 'Arizona' 'Arkansas' 'California'
 'Colorado' 'Connecticut' 'Delaware' 'District of Columbia' 'Florida'
 'Georgia' 'Hawaii' 'Idaho' 'Illinois' 'Indiana' 'Iowa' 'Kansas'
 'Kentucky' 'Louisiana' 'Maine' 'Maryland' 'Massachusetts' 'Michigan'
 'Minnesota' 'Mississippi' 'Missouri' 'Montana' 'Nebraska' 'Nevada'
 'New Hampshire' 'New Jersey' 'New Mexico' 'New York' 'North Carolina'
 'North Dakota' 'Ohio' 'Oklahoma' 'Oregon' 'Pennsylvania' 'Rhode Island'
 'South Carolina' 'South Dakota' 'Tennessee' 'Texas' 'Utah' 'Vermont'
 'Virginia' 'Washington' 'West Virginia' 'Wisconsin' 'Wyoming'
 'Puerto Rico']
Jurisdiction of Occurrence
New York                388
United States           194
```

Montana	194
Nevada	194
New Hampshire	194
New Jersey	194
New Mexico	194
North Carolina	194
North Dakota	194
Ohio	194
Oklahoma	194
Oregon	194
Pennsylvania	194
Rhode Island	194
South Carolina	194
South Dakota	194
Tennessee	194
Texas	194
Utah	194
Vermont	194
Virginia	194
Washington	194
West Virginia	194
Wisconsin	194
Wyoming	194
Nebraska	194
Missouri	194
Alabama	194
Mississippi	194
Alaska	194
Arizona	194
Arkansas	194
California	194
Colorado	194
Connecticut	194
Delaware	194
District of Columbia	194
Florida	194
Georgia	194
Hawaii	194
Idaho	194
Illinois	194
Indiana	194
Iowa	194
Kansas	194
Kentucky	194
Louisiana	194
Maine	194
Maryland	194
Massachusetts	194

```
Michigan                194
Minnesota               194
Puerto Rico             194
Name: count, dtype: int64
```

1.8 Sixth Method is changing the Format of the Date column

1.9 Here we have changed the date format into standard format d/m/y

[458]:
```
# Assuming df is your DataFrame with columns 'Data As Of' and 'Week Ending Date'
# Replace df with your actual DataFrame name

# Convert 'Data As Of' column to desired format
df['Data As Of'] = pd.to_datetime(df['Data As Of'], errors='coerce').dt.
 ↪strftime('%d/%m/%Y')

# Convert 'Week Ending Date' column to desired format
df['Week Ending Date'] = pd.to_datetime(df['Week Ending Date'],␣
 ↪errors='coerce').dt.strftime('%d/%m/%Y')

# Now both columns are in the "day/month/year" format
df
```

[458]:

	Data As Of	Jurisdiction of Occurrence	MMWR Year	MMWR Week	\
0	27/09/2023	United States	2020	1	
1	27/09/2023	United States	2020	2	
2	27/09/2023	United States	2020	3	
3	27/09/2023	United States	2020	4	
4	27/09/2023	United States	2020	5	
…	…	…	…	…	
10471	27/09/2023	Puerto Rico	2023	33	
10472	27/09/2023	Puerto Rico	2023	34	
10473	27/09/2023	Puerto Rico	2023	35	
10474	27/09/2023	Puerto Rico	2023	36	
10475	27/09/2023	Puerto Rico	2023	37	

	Week Ending Date	All Cause	Natural Cause	Malignant neoplasms	\
0	04/01/2020	60179	55010	11567.0	
1	11/01/2020	60735	55755	11963.0	
2	18/01/2020	59363	54516	11701.0	
3	25/01/2020	59162	54401	11879.0	
4	01/02/2020	58834	54001	11963.0	
…	…	…	…	…	
10471	19/08/2023	612	590	92.0	
10472	26/08/2023	657	624	110.0	
10473	02/09/2023	580	552	97.0	
10474	09/09/2023	533	516	81.0	
10475	16/09/2023	453	453	85.0	

```
       Alzheimer disease  Chronic lower respiratory diseases  \
0                 2537.0                               3501.0
1                 2566.0                               3708.0
2                 2491.0                               3526.0
3                 2517.0                               3403.0
4                 2480.0                               3313.0
…                    …                                  …
10471               48.0                                  15.0
10472               64.0                                  15.0
10473               62.0                                  14.0
10474               47.0                                  19.0
10475               25.0                                  13.0

       Diseases of heart  Cerebrovascular diseases  \
0                14204.0                    3110.0
1                13911.0                    3189.0
2                13593.0                    3256.0
3                13612.0                    3185.0
4                13465.0                    3084.0
…                   …                       …
10471              112.0                      27.0
10472              120.0                      30.0
10473               98.0                      21.0
10474               99.0                      18.0
10475               79.0                      21.0

       COVID-19_Multiple Cause of Death
0                                   0.0
1                                   1.0
2                                   2.0
3                                   3.0
4                                   0.0
…                                   …
10471                              15.0
10472                              21.0
10473                              16.0
10474                              16.0
10475                              11.0

[10476 rows x 13 columns]
```

1.10 Seventh Method is changing the datatype for numeric values

1.10.1 from float64 to int64

[459]:
```
import pandas as pd

# Assuming df is your DataFrame containing the data
# Replace df with the name of your DataFrame

# Columns to convert from float64 to int64
columns_to_convert = [
    'Malignant neoplasms',
    'Alzheimer disease',
    'Chronic lower respiratory diseases',
    'Diseases of heart',
    'Cerebrovascular diseases',
    'COVID-19_Multiple Cause of Death'
]

# Convert each column to int64
df[columns_to_convert] = df[columns_to_convert].astype('int64')

# Verify the changes
print(df.dtypes)
```

```
Data As Of                            object
Jurisdiction of Occurrence            object
MMWR Year                              int64
MMWR Week                              int64
Week Ending Date                      object
All Cause                              int64
Natural Cause                          int64
Malignant neoplasms                    int64
Alzheimer disease                      int64
Chronic lower respiratory diseases     int64
Diseases of heart                      int64
Cerebrovascular diseases               int64
COVID-19_Multiple Cause of Death       int64
dtype: object
```

1.11 Eight Method is to remove leading and trailing spaces from values in all columns

1.11.1 removing at column names and columns values

[460]:
```
import pandas as pd

# Example DataFrame
data = df
```

```
df = pd.DataFrame(data)

# Remove leading and trailing spaces from column names
df.columns = df.columns.str.strip()

# Remove leading and trailing spaces from values in all columns
df = df.apply(lambda x: x.str.strip() if x.dtype == "object" else x)

# Print the resulting DataFrame
print(df)
```

```
       Data As Of Jurisdiction of Occurrence  MMWR Year  MMWR Week  \
0      27/09/2023              United States       2020          1
1      27/09/2023              United States       2020          2
2      27/09/2023              United States       2020          3
3      27/09/2023              United States       2020          4
4      27/09/2023              United States       2020          5
…           …                         …         …         …
10471  27/09/2023                Puerto Rico       2023         33
10472  27/09/2023                Puerto Rico       2023         34
10473  27/09/2023                Puerto Rico       2023         35
10474  27/09/2023                Puerto Rico       2023         36
10475  27/09/2023                Puerto Rico       2023         37

      Week Ending Date  All Cause  Natural Cause  Malignant neoplasms  \
0           04/01/2020      60179          55010                11567
1           11/01/2020      60735          55755                11963
2           18/01/2020      59363          54516                11701
3           25/01/2020      59162          54401                11879
4           01/02/2020      58834          54001                11963
…                  …          …              …                    …
10471       19/08/2023        612            590                   92
10472       26/08/2023        657            624                  110
10473       02/09/2023        580            552                   97
10474       09/09/2023        533            516                   81
10475       16/09/2023        453            453                   85

       Alzheimer disease  Chronic lower respiratory diseases  \
0                   2537                                3501
1                   2566                                3708
2                   2491                                3526
3                   2517                                3403
4                   2480                                3313
…                    …                                   …
10471                 48                                  15
10472                 64                                  15
10473                 62                                  14
```

```
10474                47                                    19
10475                25                                    13

       Diseases of heart  Cerebrovascular diseases  \
0                  14204                      3110
1                  13911                      3189
2                  13593                      3256
3                  13612                      3185
4                  13465                      3084
…                  …                          …
10471                112                        27
10472                120                        30
10473                 98                        21
10474                 99                        18
10475                 79                        21

       COVID-19_Multiple Cause of Death
0                                     0
1                                     1
2                                     2
3                                     3
4                                     0
…                                     …
10471                                15
10472                                21
10473                                16
10474                                16
10475                                11

[10476 rows x 13 columns]
```

[461]:
```
df
```

[461]:
```
        Data As Of Jurisdiction of Occurrence  MMWR Year  MMWR Week  \
0      27/09/2023                United States       2020          1
1      27/09/2023                United States       2020          2
2      27/09/2023                United States       2020          3
3      27/09/2023                United States       2020          4
4      27/09/2023                United States       2020          5
…           …                          …          …          …
10471  27/09/2023                  Puerto Rico       2023         33
10472  27/09/2023                  Puerto Rico       2023         34
10473  27/09/2023                  Puerto Rico       2023         35
10474  27/09/2023                  Puerto Rico       2023         36
10475  27/09/2023                  Puerto Rico       2023         37

       Week Ending Date  All Cause  Natural Cause  Malignant neoplasms  \
```

```
0           04/01/2020      60179          55010                11567
1           11/01/2020      60735          55755                11963
2           18/01/2020      59363          54516                11701
3           25/01/2020      59162          54401                11879
4           01/02/2020      58834          54001                11963
…                 …         …              …                    …
10471       19/08/2023        612            590                   92
10472       26/08/2023        657            624                  110
10473       02/09/2023        580            552                   97
10474       09/09/2023        533            516                   81
10475       16/09/2023        453            453                   85

       Alzheimer disease  Chronic lower respiratory diseases  \
0                   2537                                3501
1                   2566                                3708
2                   2491                                3526
3                   2517                                3403
4                   2480                                3313
…                    …                                   …
10471                 48                                  15
10472                 64                                  15
10473                 62                                  14
10474                 47                                  19
10475                 25                                  13

       Diseases of heart  Cerebrovascular diseases  \
0                  14204                      3110
1                  13911                      3189
2                  13593                      3256
3                  13612                      3185
4                  13465                      3084
…                    …                         …
10471                112                        27
10472                120                        30
10473                 98                        21
10474                 99                        18
10475                 79                        21

       COVID-19_Multiple Cause of Death
0                                     0
1                                     1
2                                     2
3                                     3
4                                     0
…                                   …
10471                                15
10472                                21
```

```
10473                                16
10474                                16
10475                                11

[10476 rows x 13 columns]
```

[462]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10476 entries, 0 to 10475
Data columns (total 13 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Data As Of                            10476 non-null  object
 1   Jurisdiction of Occurrence            10476 non-null  object
 2   MMWR Year                             10476 non-null  int64
 3   MMWR Week                             10476 non-null  int64
 4   Week Ending Date                      10476 non-null  object
 5   All Cause                             10476 non-null  int64
 6   Natural Cause                         10476 non-null  int64
 7   Malignant neoplasms                   10476 non-null  int64
 8   Alzheimer disease                     10476 non-null  int64
 9   Chronic lower respiratory diseases    10476 non-null  int64
 10  Diseases of heart                     10476 non-null  int64
 11  Cerebrovascular diseases              10476 non-null  int64
 12  COVID-19_Multiple Cause of Death      10476 non-null  int64
dtypes: int64(10), object(3)
memory usage: 1.0+ MB
```

2 3. Data Manipulation

2.0.1 Performed data manipulation and feature engineering tasks on the DataFrame.

2.0.2 The code preprocesses and performs feature engineering on a DataFrame, including converting a date column to datetime format, extracting year and week information, calculating days since the last week, computing total deaths and cause-specific mortality rates, and calculating ratios, all while dropping unnecessary columns.

[463]:
```
import pandas as pd

# Assuming cleaned_df is your DataFrame with the provided columns
df = df

# Convert 'Week Ending Date' to datetime with the appropriate format
df['Week Ending Date'] = pd.to_datetime(df['Week Ending Date'], format='%d/%m/
 ↪%Y')
```

```
# Feature Engineering
df['Year'] = df['Week Ending Date'].dt.year
df['Week'] = df['Week Ending Date'].dt.isocalendar().week

# Days since last week (assuming data is sorted by date)
df['Days_Since_Last_Week'] = df['Week Ending Date'].diff().dt.days.fillna(0)

# Total Deaths
# Sum up all the columns related to causes of death
df['Total_Deaths'] = df[['All Cause', 'Natural Cause', 'Malignant neoplasms',␣
 ↪'Alzheimer disease', 'Chronic lower respiratory diseases', 'Diseases of␣
 ↪heart', 'Cerebrovascular diseases', 'COVID-19_Multiple Cause of Death']].
 ↪sum(axis=1)

# Cause-specific Mortality Rates (examples)
df['Malignancy_Mortality_Rate'] = df['Malignant neoplasms'] /␣
 ↪df['Total_Deaths'] * 100
df['Heart_Disease_Mortality_Rate'] = df['Diseases of heart'] /␣
 ↪df['Total_Deaths'] * 100

# Ratio (example)
df['Natural_Cause_Prop'] = cleaned_df['Natural Cause'] / df['Total_Deaths']

df['Month'] = df['Week Ending Date'].dt.month
# Assuming 'Season' is derived from 'Month' column
df['Season'] = df['Month'].apply(lambda x: (x%12 + 3)//3)
# Identifying and encoding seasonal patterns
season_map = {1: 'Winter', 2: 'Spring', 3: 'Summer', 4: 'Autumn'}
df['Seasonal Pattern'] = df['Season'].map(season_map)

print(df)
```

```
       Data As Of Jurisdiction of Occurrence  MMWR Year  MMWR Week  \
0      27/09/2023              United States       2020          1
1      27/09/2023              United States       2020          2
2      27/09/2023              United States       2020          3
3      27/09/2023              United States       2020          4
4      27/09/2023              United States       2020          5
…           …                         …          …          …
10471  27/09/2023                Puerto Rico       2023         33
10472  27/09/2023                Puerto Rico       2023         34
10473  27/09/2023                Puerto Rico       2023         35
10474  27/09/2023                Puerto Rico       2023         36
10475  27/09/2023                Puerto Rico       2023         37

      Week Ending Date  All Cause  Natural Cause  Malignant neoplasms  \
0           2020-01-04      60179          55010                11567
```

```
1           2020-01-11      60735          55755                11963
2           2020-01-18      59363          54516                11701
3           2020-01-25      59162          54401                11879
4           2020-02-01      58834          54001                11963
…                 …         …              …                    …
10471       2023-08-19        612            590                   92
10472       2023-08-26        657            624                  110
10473       2023-09-02        580            552                   97
10474       2023-09-09        533            516                   81
10475       2023-09-16        453            453                   85

       Alzheimer disease  Chronic lower respiratory diseases  …  Year  Week  \
0                   2537                                3501  …  2020     1
1                   2566                                3708  …  2020     2
2                   2491                                3526  …  2020     3
3                   2517                                3403  …  2020     4
4                   2480                                3313  …  2020     5
…                    …                                   …  …   …    …
10471                 48                                  15  …  2023    33
10472                 64                                  15  …  2023    34
10473                 62                                  14  …  2023    35
10474                 47                                  19  …  2023    36
10475                 25                                  13  …  2023    37

       Days_Since_Last_Week  Total_Deaths  Malignancy_Mortality_Rate  \
0                       0.0        150108                   7.705785
1                       7.0        151828                   7.879311
2                       7.0        148448                   7.882221
3                       7.0        148162                   8.017575
4                       7.0        147140                   8.130352
…                        …             …                          …
10471                   7.0          1511                   6.088683
10472                   7.0          1641                   6.703230
10473                   7.0          1440                   6.736111
10474                   7.0          1329                   6.094808
10475                   7.0          1140                   7.456140

       Heart_Disease_Mortality_Rate  Natural_Cause_Prop  Month  Season  \
0                          9.462520            0.366469      1       1
1                          9.162342            0.367225      1       1
2                          9.156742            0.367240      1       1
3                          9.187241            0.367172      1       1
4                          9.151149            0.367004      2       1
…                             …                  …    …     …
10471                      7.412310            0.390470      8       3
10472                      7.312614            0.380256      8       3
10473                      6.805556            0.383333      9       4
10474                      7.449210            0.388262      9       4
```

```
10475                    6.929825            0.397368      9       4

       Seasonal Pattern
0                Winter
1                Winter
2                Winter
3                Winter
4                Winter
…                 …
10471            Summer
10472            Summer
10473            Autumn
10474            Autumn
10475            Autumn

[10476 rows x 23 columns]
```

[464]:
```
df
```

[464]:
```
       Data As Of Jurisdiction of Occurrence  MMWR Year  MMWR Week  \
0      27/09/2023              United States       2020          1
1      27/09/2023              United States       2020          2
2      27/09/2023              United States       2020          3
3      27/09/2023              United States       2020          4
4      27/09/2023              United States       2020          5
…           …                         …         …        …
10471  27/09/2023                Puerto Rico       2023         33
10472  27/09/2023                Puerto Rico       2023         34
10473  27/09/2023                Puerto Rico       2023         35
10474  27/09/2023                Puerto Rico       2023         36
10475  27/09/2023                Puerto Rico       2023         37

      Week Ending Date  All Cause  Natural Cause  Malignant neoplasms  \
0           2020-01-04      60179          55010                11567
1           2020-01-11      60735          55755                11963
2           2020-01-18      59363          54516                11701
3           2020-01-25      59162          54401                11879
4           2020-02-01      58834          54001                11963
…                 …         …              …                    …
10471       2023-08-19        612            590                   92
10472       2023-08-26        657            624                  110
10473       2023-09-02        580            552                   97
10474       2023-09-09        533            516                   81
10475       2023-09-16        453            453                   85

       Alzheimer disease  Chronic lower respiratory diseases  …  Year  Week  \
0                   2537                                3501  …  2020     1
```

```
1                    2566                                3708  …  2020     2
2                    2491                                3526  …  2020     3
3                    2517                                3403  …  2020     4
4                    2480                                3313  …  2020     5
…                     …                                   …   …   …    …
10471                  48                                  15  …  2023    33
10472                  64                                  15  …  2023    34
10473                  62                                  14  …  2023    35
10474                  47                                  19  …  2023    36
10475                  25                                  13  …  2023    37

       Days_Since_Last_Week  Total_Deaths  Malignancy_Mortality_Rate  \
0                       0.0        150108                   7.705785
1                       7.0        151828                   7.879311
2                       7.0        148448                   7.882221
3                       7.0        148162                   8.017575
4                       7.0        147140                   8.130352
…                       …             …                          …
10471                   7.0          1511                   6.088683
10472                   7.0          1641                   6.703230
10473                   7.0          1440                   6.736111
10474                   7.0          1329                   6.094808
10475                   7.0          1140                   7.456140

       Heart_Disease_Mortality_Rate  Natural_Cause_Prop  Month  Season  \
0                          9.462520            0.366469      1       1
1                          9.162342            0.367225      1       1
2                          9.156742            0.367240      1       1
3                          9.187241            0.367172      1       1
4                          9.151149            0.367004      2       1
…                             …                 …    …     …
10471                      7.412310            0.390470      8       3
10472                      7.312614            0.380256      8       3
10473                      6.805556            0.383333      9       4
10474                      7.449210            0.388262      9       4
10475                      6.929825            0.397368      9       4

       Seasonal Pattern
0                Winter
1                Winter
2                Winter
3                Winter
4                Winter
…                 …
10471            Summer
10472            Summer
10473            Autumn
```

```
10474            Autumn
10475            Autumn

[10476 rows x 23 columns]
```

[465]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10476 entries, 0 to 10475
Data columns (total 23 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Data As Of                            10476 non-null  object
 1   Jurisdiction of Occurrence            10476 non-null  object
 2   MMWR Year                             10476 non-null  int64
 3   MMWR Week                             10476 non-null  int64
 4   Week Ending Date                      10476 non-null  datetime64[ns]
 5   All Cause                             10476 non-null  int64
 6   Natural Cause                         10476 non-null  int64
 7   Malignant neoplasms                   10476 non-null  int64
 8   Alzheimer disease                     10476 non-null  int64
 9   Chronic lower respiratory diseases    10476 non-null  int64
 10  Diseases of heart                     10476 non-null  int64
 11  Cerebrovascular diseases              10476 non-null  int64
 12  COVID-19_Multiple Cause of Death      10476 non-null  int64
 13  Year                                  10476 non-null  int32
 14  Week                                  10476 non-null  UInt32
 15  Days_Since_Last_Week                  10476 non-null  float64
 16  Total_Deaths                          10476 non-null  int64
 17  Malignancy_Mortality_Rate             10476 non-null  float64
 18  Heart_Disease_Mortality_Rate          10476 non-null  float64
 19  Natural_Cause_Prop                    10476 non-null  float64
 20  Month                                 10476 non-null  int32
 21  Season                                10476 non-null  int64
 22  Seasonal Pattern                      10476 non-null  object
dtypes: UInt32(1), datetime64[ns](1), float64(4), int32(2), int64(12), object(3)
memory usage: 1.7+ MB
```

3 Data Analysis

3.1 Descriptive Statistics:

3.1.1 Calculate mean, median, mode, standard deviation and variance

[487]:
```
import pandas as pd
```

```
# Assuming your DataFrame is named 'df'http://localhost:8888/notebooks/
 ↪Balaji_Pamidi_Final_project_002644804.
 ↪ipynb#Calculate-mean,-median,-mode,-standard-deviation-and-variance

# Select numeric columns
numeric_columns = df.select_dtypes(include=['int64', 'float64'])

# Calculate descriptive statistics
descriptive_stats = numeric_columns.describe()

# Calculate mode
#mode = numeric_columns.mode()

# Loop through each numeric column and calculate mode separately
for column in numeric_columns:
    mode = numeric_columns[column].mode()
    print(f"Mode for {column}: {mode.tolist()}")

# Calculate variance
variance = numeric_columns.var()


# Display the results
print("Descriptive Statistics:")
print(descriptive_stats)

#print("\nMode:")
#print(mode)

print("\nVariance:")
print(variance)
```

```
Mode for MMWR Year: [2020]
Mode for MMWR Week: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17,
18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37]
Mode for All Cause: [125]
Mode for Natural Cause: [106]
Mode for Malignant neoplasms: [22]
Mode for Alzheimer disease: [103]
Mode for Chronic lower respiratory diseases: [121]
Mode for Diseases of heart: [30]
Mode for Cerebrovascular diseases: [133]
Mode for COVID-19_Multiple Cause of Death: [262]
Mode for Days_Since_Last_Week: [7.0]
Mode for Total_Deaths: [842]
Mode for Malignancy_Mortality_Rate: [6.25, 7.142857142857142,
7.6923076923076925]
Mode for Heart_Disease_Mortality_Rate: [7.142857142857142, 9.090909090909092]
```

```
Mode for Natural_Cause_Prop: [0.36363636363636365]
Mode for Season: [2, 3]
Descriptive Statistics:
          MMWR Year     MMWR Week     All Cause  Natural Cause  \
count  10476.000000  10476.000000  10476.000000   10476.000000
mean    2021.376289     25.206186   2362.293910    2151.945208
std        1.078259     14.682156   8552.044924    7799.519725
min     2020.000000      1.000000     12.000000      12.000000
25%     2020.000000     13.000000    366.750000     331.000000
50%     2021.000000     25.000000    931.500000     837.000000
75%     2022.000000     37.000000   1548.000000    1405.000000
max     2023.000000     53.000000  87415.000000   81622.000000

       Malignant neoplasms  Alzheimer disease  \
count         10476.000000       10476.000000
mean            431.874380         103.788183
std            1550.675559         313.861633
min              10.000000           0.000000
25%              66.000000          25.000000
50%             171.000000          47.000000
75%             282.000000         103.000000
max           12284.000000        3075.000000

       Chronic lower respiratory diseases  Diseases of heart  \
count                        10476.000000       10476.000000
mean                           121.146334         493.532837
std                            375.488612        1777.161012
min                              0.000000          10.000000
25%                             31.000000          72.000000
50%                             56.000000         191.000000
75%                            114.000000         329.000000
max                           3708.000000       16538.000000

       Cerebrovascular diseases  COVID-19_Multiple Cause of Death  \
count              10476.000000                       10476.000000
mean                 133.053169                         262.532932
std                  415.103496                        1160.062676
min                    0.000000                           0.000000
25%                   29.000000                          27.000000
50%                   56.000000                          75.500000
75%                  126.000000                         262.000000
max                 3833.000000                       26028.000000

       Days_Since_Last_Week   Total_Deaths  Malignancy_Mortality_Rate  \
count          10476.000000   10476.000000               10476.000000
mean               0.128961    6060.166953                   6.634836
std               96.351661   21635.364667                   1.984468
min            -1351.000000     340.000000                   1.002506
```

```
25%                 7.000000    1056.000000                  5.640041
50%                 7.000000    2348.500000                  7.042898
75%                 7.000000    3903.500000                  7.877299
max                 7.000000  231114.000000                 46.746204

       Heart_Disease_Mortality_Rate  Natural_Cause_Prop        Season
count                  10476.000000        10476.000000  10476.000000
mean                       7.506880            0.324006      2.453608
std                        2.423096            0.072995      1.084341
min                        0.945830            0.011101      1.000000
25%                        6.518929            0.324171      2.000000
50%                        7.904952            0.357918      2.000000
75%                        8.808975            0.363222      3.000000
max                       68.854749            0.397368      4.000000

Variance:
MMWR Year                                 1.162642e+00
MMWR Week                                 2.155657e+02
All Cause                                 7.313747e+07
Natural Cause                             6.083251e+07
Malignant neoplasms                       2.404595e+06
Alzheimer disease                         9.850912e+04
Chronic lower respiratory diseases        1.409917e+05
Diseases of heart                         3.158301e+06
Cerebrovascular diseases                  1.723109e+05
COVID-19_Multiple Cause of Death          1.345745e+06
Days_Since_Last_Week                      9.283643e+03
Total_Deaths                              4.680890e+08
Malignancy_Mortality_Rate                 3.938114e+00
Heart_Disease_Mortality_Rate              5.871395e+00
Natural_Cause_Prop                        5.328327e-03
Season                                    1.175795e+00
dtype: float64
```

4 Correlation

[466]:
```python
# Select only numeric columns
numeric_columns = df.select_dtypes(include=['int64', 'float64','UInt32'])

# Calculate correlation
correlation_matrix = numeric_columns.corr()

# Display correlation matrix
print(correlation_matrix)


# Finding the strongest correlation
```

```
strongest_corr = correlation_matrix.unstack().sort_values(ascending=False).
  ↪drop_duplicates()
print("\nStrongest Correlation:")
print(strongest_corr[:30])  # Displaying the top 5 strongest correlations

# Finding the moderate correlation (between 0.3 and 0.7)
moderate_corr = strongest_corr[(strongest_corr < 0.7) & (strongest_corr >= 0.3)]
print("\nModerate Correlation:")
print(moderate_corr)

# Finding the weakest correlation
weakest_corr = correlation_matrix.unstack().sort_values().drop_duplicates()
print("\nWeakest Correlation:")
print(weakest_corr[:5])  # Displaying the top 5 weakest correlations
```

```
                                MMWR Year  MMWR Week  All Cause  \
MMWR Year                        1.000000  -0.158600  -0.010860
MMWR Week                       -0.158600   1.000000   0.000712
All Cause                       -0.010860   0.000712   1.000000
Natural Cause                   -0.011232   0.000617   0.999878
Malignant neoplasms              0.000507   0.000387   0.992298
Alzheimer disease               -0.014961  -0.001048   0.988653
Chronic lower respiratory diseases -0.004665  -0.006989   0.985634
Diseases of heart               -0.003930  -0.003451   0.996925
Cerebrovascular diseases         0.000523  -0.002440   0.989995
COVID-19_Multiple Cause of Death -0.056272   0.012200   0.756529
Week                            -0.158600   1.000000   0.000712
Days_Since_Last_Week             0.091031   0.117582   0.010096
Total_Deaths                    -0.011934   0.000719   0.999752
Malignancy_Mortality_Rate        0.064687  -0.024684   0.076491
Heart_Disease_Mortality_Rate     0.051316  -0.044226   0.079460
Natural_Cause_Prop              -0.053027   0.002755   0.127399
Season                          -0.044591   0.577298  -0.012889

                                Natural Cause  Malignant neoplasms  \
MMWR Year                           -0.011232             0.000507
MMWR Week                            0.000617             0.000387
All Cause                            0.999878             0.992298
Natural Cause                        1.000000             0.991056
Malignant neoplasms                  0.991056             1.000000
Alzheimer disease                    0.988531             0.984426
Chronic lower respiratory diseases   0.984874             0.988655
Diseases of heart                    0.996135             0.997139
Cerebrovascular diseases             0.989117             0.992359
COVID-19_Multiple Cause of Death     0.762467             0.674407
Week                                 0.000617             0.000387
Days_Since_Last_Week                 0.010050             0.009696
Total_Deaths                         0.999881             0.989730
```

Malignancy_Mortality_Rate	0.075319	0.100612
Heart_Disease_Mortality_Rate	0.078819	0.088311
Natural_Cause_Prop	0.127509	0.128154
Season	-0.014022	-0.001309

	Alzheimer disease \
MMWR Year	-0.014961
MMWR Week	-0.001048
All Cause	0.988653
Natural Cause	0.988531
Malignant neoplasms	0.984426
Alzheimer disease	1.000000
Chronic lower respiratory diseases	0.987334
Diseases of heart	0.988065
Cerebrovascular diseases	0.989455
COVID-19_Multiple Cause of Death	0.731211
Week	-0.001048
Days_Since_Last_Week	0.009591
Total_Deaths	0.988711
Malignancy_Mortality_Rate	0.020538
Heart_Disease_Mortality_Rate	0.018241
Natural_Cause_Prop	0.037757
Season	-0.011721

	Chronic lower respiratory diseases \
MMWR Year	-0.004665
MMWR Week	-0.006989
All Cause	0.985634
Natural Cause	0.984874
Malignant neoplasms	0.988655
Alzheimer disease	0.987334
Chronic lower respiratory diseases	1.000000
Diseases of heart	0.991100
Cerebrovascular diseases	0.991540
COVID-19_Multiple Cause of Death	0.681292
Week	-0.006989
Days_Since_Last_Week	0.008275
Total_Deaths	0.984151
Malignancy_Mortality_Rate	0.027884
Heart_Disease_Mortality_Rate	0.029545
Natural_Cause_Prop	0.039669
Season	-0.015135

	Diseases of heart \
MMWR Year	-0.003930
MMWR Week	-0.003451
All Cause	0.996925
Natural Cause	0.996135

Malignant neoplasms	0.997139
Alzheimer disease	0.988065
Chronic lower respiratory diseases	0.991100
Diseases of heart	1.000000
Cerebrovascular diseases	0.992986
COVID-19_Multiple Cause of Death	0.707251
Week	-0.003451
Days_Since_Last_Week	0.008912
Total_Deaths	0.995289
Malignancy_Mortality_Rate	0.084877
Heart_Disease_Mortality_Rate	0.095955
Natural_Cause_Prop	0.127441
Season	-0.012800

	Cerebrovascular diseases \
MMWR Year	0.000523
MMWR Week	-0.002440
All Cause	0.989995
Natural Cause	0.989117
Malignant neoplasms	0.992359
Alzheimer disease	0.989455
Chronic lower respiratory diseases	0.991540
Diseases of heart	0.992986
Cerebrovascular diseases	1.000000
COVID-19_Multiple Cause of Death	0.698365
Week	-0.002440
Days_Since_Last_Week	0.009729
Total_Deaths	0.988786
Malignancy_Mortality_Rate	0.031458
Heart_Disease_Mortality_Rate	0.031310
Natural_Cause_Prop	0.042136
Season	-0.009075

	COVID-19_Multiple Cause of Death \
MMWR Year	-0.056272
MMWR Week	0.012200
All Cause	0.756529
Natural Cause	0.762467
Malignant neoplasms	0.674407
Alzheimer disease	0.731211
Chronic lower respiratory diseases	0.681292
Diseases of heart	0.707251
Cerebrovascular diseases	0.698365
COVID-19_Multiple Cause of Death	1.000000
Week	0.012200
Days_Since_Last_Week	0.016140
Total_Deaths	0.769791
Malignancy_Mortality_Rate	-0.066400

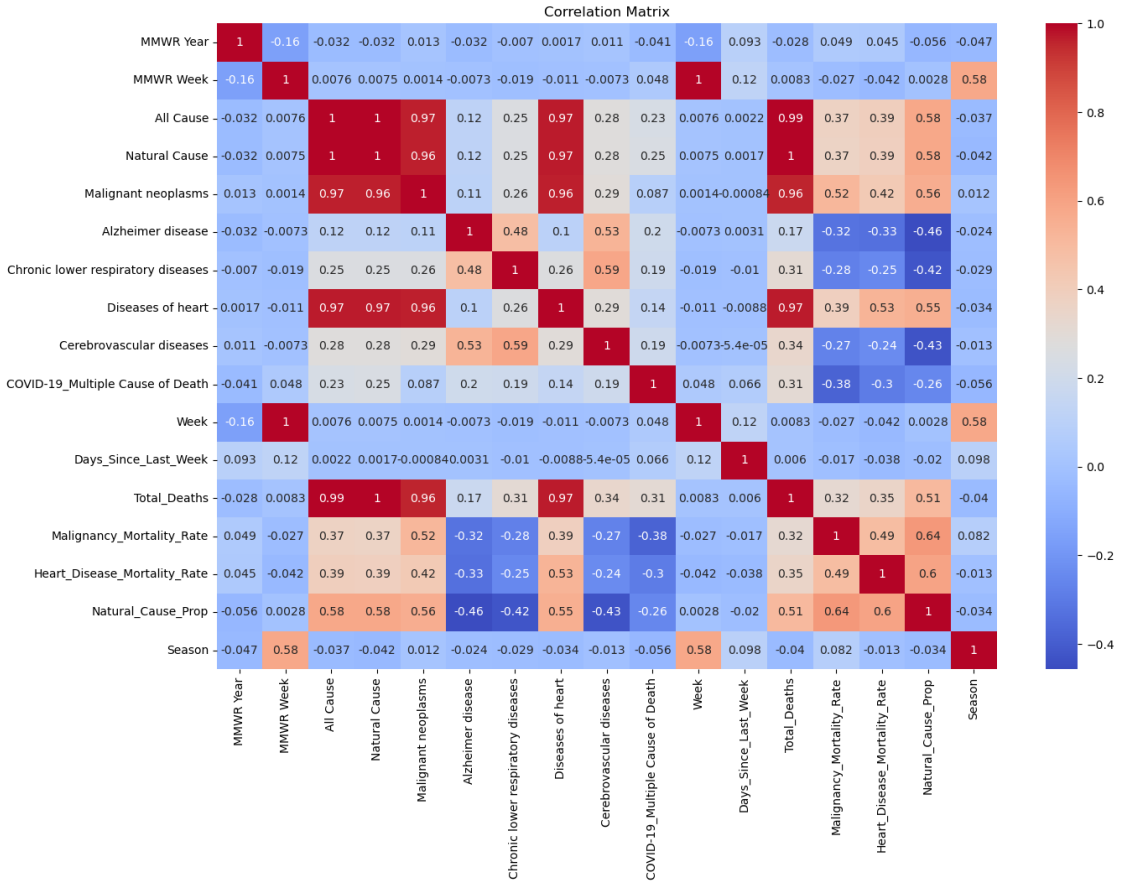
```
Heart_Disease_Mortality_Rate                                -0.033000
Natural_Cause_Prop                                           0.024065
Season                                                      -0.036286

                                        Week  Days_Since_Last_Week  \
MMWR Year                          -0.158600              0.091031
MMWR Week                           1.000000              0.117582
All Cause                           0.000712              0.010096
Natural Cause                       0.000617              0.010050
Malignant neoplasms                 0.000387              0.009696
Alzheimer disease                  -0.001048              0.009591
Chronic lower respiratory diseases -0.006989              0.008275
Diseases of heart                  -0.003451              0.008912
Cerebrovascular diseases           -0.002440              0.009729
COVID-19_Multiple Cause of Death    0.012200              0.016140
Week                                1.000000              0.117582
Days_Since_Last_Week                0.117582              1.000000
Total_Deaths                        0.000719              0.010376
Malignancy_Mortality_Rate          -0.024684             -0.017301
Heart_Disease_Mortality_Rate       -0.044226             -0.037892
Natural_Cause_Prop                  0.002755             -0.018175
Season                              0.577298              0.095606

                                    Total_Deaths  Malignancy_Mortality_Rate  \
MMWR Year                          -0.011934                     0.064687
MMWR Week                           0.000719                    -0.024684
All Cause                           0.999752                     0.076491
Natural Cause                       0.999881                     0.075319
Malignant neoplasms                 0.989730                     0.100612
Alzheimer disease                   0.988711                     0.020538
Chronic lower respiratory diseases  0.984151                     0.027884
Diseases of heart                   0.995289                     0.084877
Cerebrovascular diseases            0.988786                     0.031458
COVID-19_Multiple Cause of Death    0.769791                    -0.066400
Week                                0.000719                    -0.024684
Days_Since_Last_Week                0.010376                    -0.017301
Total_Deaths                        1.000000                     0.069396
Malignancy_Mortality_Rate           0.069396                     1.000000
Heart_Disease_Mortality_Rate        0.073643                     0.494980
Natural_Cause_Prop                  0.119314                     0.630406
Season                             -0.013847                     0.090407

                                    Heart_Disease_Mortality_Rate  \
MMWR Year                                           0.051316
MMWR Week                                          -0.044226
All Cause                                           0.079460
Natural Cause                                       0.078819
Malignant neoplasms                                 0.088311
```

```
Alzheimer disease                                 0.018241
Chronic lower respiratory diseases                0.029545
Diseases of heart                                 0.095955
Cerebrovascular diseases                          0.031310
COVID-19_Multiple Cause of Death                 -0.033000
Week                                             -0.044226
Days_Since_Last_Week                             -0.037892
Total_Deaths                                      0.073643
Malignancy_Mortality_Rate                         0.494980
Heart_Disease_Mortality_Rate                      1.000000
Natural_Cause_Prop                                0.601936
Season                                           -0.013210

                                  Natural_Cause_Prop    Season
MMWR Year                                -0.053027 -0.044591
MMWR Week                                 0.002755  0.577298
All Cause                                 0.127399 -0.012889
Natural Cause                             0.127509 -0.014022
Malignant neoplasms                       0.128154 -0.001309
Alzheimer disease                         0.037757 -0.011721
Chronic lower respiratory diseases        0.039669 -0.015135
Diseases of heart                         0.127441 -0.012800
Cerebrovascular diseases                  0.042136 -0.009075
COVID-19_Multiple Cause of Death          0.024065 -0.036286
Week                                      0.002755  0.577298
Days_Since_Last_Week                     -0.018175  0.095606
Total_Deaths                              0.119314 -0.013847
Malignancy_Mortality_Rate                 0.630406  0.090407
Heart_Disease_Mortality_Rate              0.601936 -0.013210
Natural_Cause_Prop                        1.000000 -0.035179
Season                                   -0.035179  1.000000

Strongest Correlation:
MMWR Year                        MMWR Year
1.000000
Total_Deaths                     Natural Cause
0.999881
All Cause                        Natural Cause
0.999878
Total_Deaths                     All Cause
0.999752
Diseases of heart                Malignant neoplasms
0.997139
All Cause                        Diseases of heart
0.996925
Natural Cause                    Diseases of heart
0.996135
Total_Deaths                     Diseases of heart
```

```
0.995289
Diseases of heart                   Cerebrovascular diseases
0.992986
Malignant neoplasms                 Cerebrovascular diseases
0.992359
All Cause                           Malignant neoplasms
0.992298
Chronic lower respiratory diseases  Cerebrovascular diseases
0.991540
Diseases of heart                   Chronic lower respiratory diseases
0.991100
Malignant neoplasms                 Natural Cause
0.991056
All Cause                           Cerebrovascular diseases
0.989995
Malignant neoplasms                 Total_Deaths
0.989730
Cerebrovascular diseases            Alzheimer disease
0.989455
                                    Natural Cause
0.989117
Total_Deaths                        Cerebrovascular diseases
0.988786
                                    Alzheimer disease
0.988711
Malignant neoplasms                 Chronic lower respiratory diseases
0.988655
Alzheimer disease                   All Cause
0.988653
Natural Cause                       Alzheimer disease
0.988531
Alzheimer disease                   Diseases of heart
0.988065
Chronic lower respiratory diseases  Alzheimer disease
0.987334
All Cause                           Chronic lower respiratory diseases
0.985634
Chronic lower respiratory diseases  Natural Cause
0.984874
Alzheimer disease                   Malignant neoplasms
0.984426
Chronic lower respiratory diseases  Total_Deaths
0.984151
COVID-19_Multiple Cause of Death    Total_Deaths
0.769791
dtype: float64

Moderate Correlation:
```

```
Cerebrovascular diseases          COVID-19_Multiple Cause of Death      0.698365
COVID-19_Multiple Cause of Death  Chronic lower respiratory diseases    0.681292
Malignant neoplasms               COVID-19_Multiple Cause of Death      0.674407
Malignancy_Mortality_Rate         Natural_Cause_Prop                    0.630406
Natural_Cause_Prop                Heart_Disease_Mortality_Rate          0.601936
MMWR Week                         Season                                0.577298
Malignancy_Mortality_Rate         Heart_Disease_Mortality_Rate          0.494980
dtype: float64

Weakest Correlation:
MMWR Year                         Week                        -0.158600
COVID-19_Multiple Cause of Death  Malignancy_Mortality_Rate   -0.066400
                                  MMWR Year                   -0.056272
Natural_Cause_Prop                MMWR Year                   -0.053027
Season                            MMWR Year                   -0.044591
dtype: float64
```

[490]:
```
# Heatmap for correlation matrix
plt.figure(figsize=(15, 10))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title("Correlation Matrix")
plt.show()
```



5 Multiple Linear Regression Analysis

5.0.1 Multiple Linear Regression using sklearn and using statsmodels

[467]:
```
import pandas as pd
import statsmodels.api as sm

# Assuming cleaned_df is your DataFrame with the provided columns
# Select predictor variables and outcome variable
predictor_cols = ['Cerebrovascular diseases','Alzheimer disease','Chronic lower␣
 ↪respiratory diseases']

outcome_col = ['Total_Deaths']

# Create a DataFrame with predictor variables
X = df[predictor_cols]

# Add a constant term to the predictor variables (for intercept)
X = sm.add_constant(X)

# Create a Series with the outcome variable
y = df[outcome_col]

# Concatenate X and y along the columns axis
df_copy = pd.concat([X, y], axis=1)

# Print the concatenated DataFrame
print(df_copy.head(5))

# Fit the multiple linear regression model
model = sm.OLS(y, X).fit()

# Print the model summary
print(model.summary())
```

```
   const  Cerebrovascular diseases  Alzheimer disease  \
0    1.0                      3110               2537
1    1.0                      3189               2566
2    1.0                      3256               2491
3    1.0                      3185               2517
4    1.0                      3084               2480

   Chronic lower respiratory diseases  Total_Deaths
0                                3501        150108
1                                3708        151828
```

```
2                                3526        148448
3                                3403        148162
4                                3313        147140
                            OLS Regression Results
==============================================================================
Dep. Variable:           Total_Deaths   R-squared:                       0.983
Model:                            OLS   Adj. R-squared:                  0.983
Method:                 Least Squares   F-statistic:                 1.999e+05
Date:                Wed, 10 Apr 2024   Prob (F-statistic):               0.00
Time:                        14:06:02   Log-Likelihood:                -98145.
No. Observations:               10476   AIC:                         1.963e+05
Df Residuals:                   10472   BIC:                         1.963e+05
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
======================
                                       coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------------
----------------------
const                               -942.0375     29.203    -32.259      0.000
-999.280    -884.795
Cerebrovascular diseases              24.5240      0.597     41.073      0.000
23.354      25.694
Alzheimer disease                     33.1267      0.646     51.273      0.000
31.860      34.393
Chronic lower respiratory diseases     2.4850      0.603      4.124      0.000
1.304       3.666
==============================================================================
Omnibus:                     5801.986   Durbin-Watson:                   0.224
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           477771.487
Skew:                           1.795   Prob(JB):                         0.00
Kurtosis:                      35.889   Cond. No.                         709.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

5.1 Model Building

5.1.1 Training Models

[493]:
```
#split the dataset in training set and test set

X_train, X_test, y_train, y_test = train_test_split(df[predictor_cols],␣
 ↪df[outcome_col], test_size=0.3, random_state=0)
```

```
# Convert y_train and y_test to DataFrame with a single column
y_train = pd.DataFrame(y_train, columns=outcome_col)
y_test = pd.DataFrame(y_test, columns=outcome_col)

print("X_train shape:", X_train.shape)
print("X_test shape:", X_test.shape)
print("y_train shape:", y_train.shape)
print("y_test shape:", y_test.shape)
```

```
X_train shape: (7333, 3)
X_test shape: (3143, 3)
y_train shape: (7333, 1)
y_test shape: (3143, 1)
```

5.2 Using the sklearn we are going to implement the Multiple linear regression

[494]:
```
from sklearn.linear_model import LinearRegression

# Create an instance of the LinearRegression class
lin_reg = LinearRegression()

# Fit the model using the training data
lin_reg.fit(X_train, y_train)

# After fitting, you can access the coefficients and intercept of the model
coefficients = lin_reg.coef_
intercept = lin_reg.intercept_

# Print the coefficients and intercept
print("Coefficients:", coefficients)
print("Intercept:", intercept)
```

```
Coefficients: [[ 2.74878287e+01  3.27094228e+01 -3.39925188e-03]]
Intercept: [-970.51379839]
```

5.2.1 Making Predictions

[495]:
```
# Make predictions on the testing data
y_pred = lin_reg.predict(X_test)

# Print the first few predictions
print("y_pred:", y_pred[:5])
```

```
y_pred: [[ 2967.00376817]
 [11613.61714689]
 [  458.43647989]
 [ 2838.30761808]
 [ 2956.61836722]]
```

5.2.2 Evaluating Model Predict

[496]:
```
# Use model to predict

lin_reg.predict([[3110,2537,3501]])
```

[496]:
```
array([[167488.53813431]])
```

5.3 R2_Score

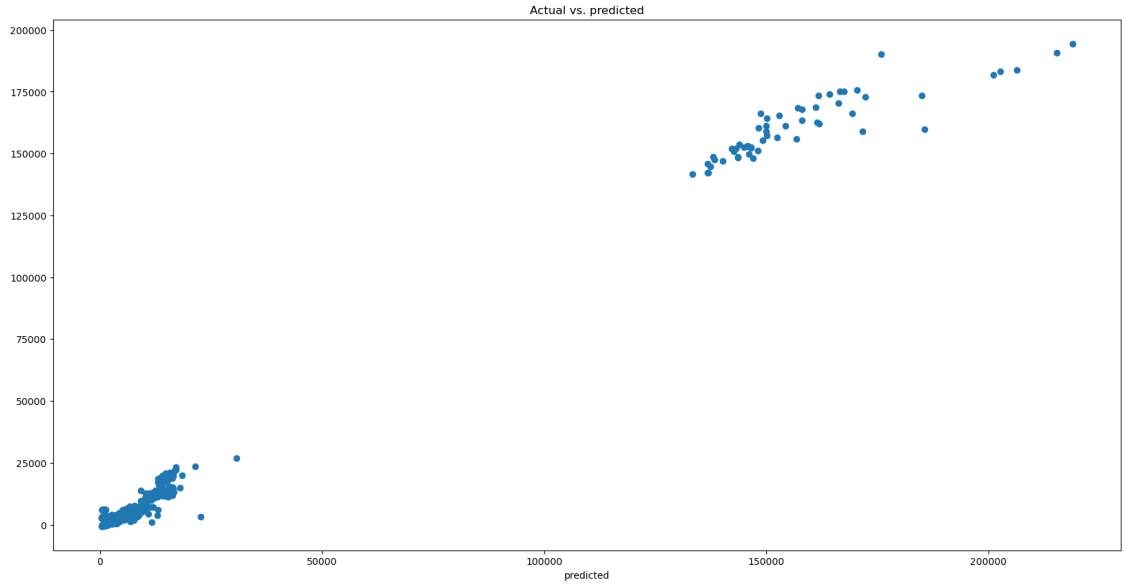
[497]:
```
from sklearn.metrics import r2_score
r2_score(y_test,y_pred)
```

[497]:
```
0.9837079490658794
```

5.4 Plot the results using matplotlib

[498]:
```
#plot the results
import matplotlib.pyplot as plt
plt.figure(figsize=(20,10))
plt.scatter(y_test,y_pred)
plt.xlabel('Actual')
plt.xlabel('predicted')
plt.title('Actual vs. predicted')
```

[498]:
```
Text(0.5, 1.0, 'Actual vs. predicted')
```



5.4.1 Differnce between the test and predicted values

[474]:
```
# Convert y_test and y_pred to one-dimensional arrays
y_test_1d = y_test.values.flatten()
y_pred_1d = y_pred.flatten()

# Create DataFrame with one-dimensional arrays
pred_y_df = pd.DataFrame({'Actual Value': y_test_1d, 'Predicted Value':␣
 ↪y_pred_1d, 'Difference': y_test_1d - y_pred_1d})
pred_y_df.head(20)
```

[474]:

	Actual Value	Predicted Value	Difference
0	5119	2967.003768	2151.996232
1	11407	11613.617147	-206.617147
2	1553	458.436480	1094.563520
3	937	2838.307618	-1901.307618
4	3822	2956.618367	865.381633
5	1480	360.315010	1119.684990
6	1387	-110.775623	1497.775623
7	2883	1711.046753	1171.953247
8	1872	824.878974	1047.121026
9	992	6054.373373	-5062.373373
10	1057	6054.026650	-4997.026650
11	2351	1290.734416	1060.265584
12	734	-67.623012	801.623012
13	916	168.980933	747.019067
14	18082	15014.939513	3067.060487
15	5153	4928.459921	224.540079
16	718	6054.026650	-5336.026650
17	3776	529.131308	3246.868692
18	2933	2548.217457	384.782543
19	4772	2490.766325	2281.233675

5.5 Removing the Outliers

[475]:
```
import pandas as pd
import numpy as np

# Load your dataset (replace 'your_dataset.csv' with the actual path to your␣
 ↪dataset)
df_uncleaned = df

# Make a copy of the original DataFrame to work with
df_uncleaned = df.copy()

# Specify the numerical columns in your dataset
numeric_columns = [
```

```
   'Cerebrovascular diseases',
    'Alzheimer disease',
    'Chronic lower respiratory diseases',
    'Total_Deaths'
]

# Function to detect outliers using IQR method
def detect_outliers_iqr(data):
    Q1 = np.percentile(data, 25)
    Q3 = np.percentile(data, 75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    outliers = (data < lower_bound) | (data > upper_bound)
    return outliers

# Detect outliers in each numerical column
outliers = {}
for col in numeric_columns:
    outliers[col] = detect_outliers_iqr(df_uncleaned[col])

# Print count of outliers for each column
for col, is_outlier in outliers.items():
    num_outliers = sum(is_outlier)
    print(f"Outliers detected in '{col}': {num_outliers}")

# Create a copy of the DataFrame for trimming outliers
df_trimmed = df_uncleaned.copy()

# Filter out rows containing outliers
for col, is_outlier in outliers.items():
    df_trimmed = df_trimmed.loc[~is_outlier]

# Reset the index of the trimmed DataFrame
df_trimmed.reset_index(drop=True, inplace=True)

# Print count of rows after removing outliers
print(f"Count of rows after removing outliers: {len(df_trimmed)}")

# Print count of values in each column after trimming outliers
for col in df_trimmed.columns:
    print(f"Count of values in '{col}': {df_trimmed[col].count()}")
```

```
Outliers detected in 'Cerebrovascular diseases': 581
Outliers detected in 'Alzheimer disease': 438
Outliers detected in 'Chronic lower respiratory diseases': 314
Outliers detected in 'Total_Deaths': 861
```

```
Count of rows after removing outliers: 9615
Count of values in 'Data As Of': 9615
Count of values in 'Jurisdiction of Occurrence': 9615
Count of values in 'MMWR Year': 9615
Count of values in 'MMWR Week': 9615
Count of values in 'Week Ending Date': 9615
Count of values in 'All Cause': 9615
Count of values in 'Natural Cause': 9615
Count of values in 'Malignant neoplasms': 9615
Count of values in 'Alzheimer disease': 9615
Count of values in 'Chronic lower respiratory diseases': 9615
Count of values in 'Diseases of heart': 9615
Count of values in 'Cerebrovascular diseases': 9615
Count of values in 'COVID-19_Multiple Cause of Death': 9615
Count of values in 'Year': 9615
Count of values in 'Week': 9615
Count of values in 'Days_Since_Last_Week': 9615
Count of values in 'Total_Deaths': 9615
Count of values in 'Malignancy_Mortality_Rate': 9615
Count of values in 'Heart_Disease_Mortality_Rate': 9615
Count of values in 'Natural_Cause_Prop': 9615
Count of values in 'Month': 9615
Count of values in 'Season': 9615
Count of values in 'Seasonal Pattern': 9615
```

5.6 Correlation after removing the outliers

[476]:
```
# Select only numeric columns
numeric_columns = df_trimmed.select_dtypes(include=['int64',␣
 ↪'float64','UInt32'])

# Calculate correlation
correlation_matrix = numeric_columns.corr()

# Display correlation matrix
print(correlation_matrix)


# Finding the strongest correlation
strongest_corr = correlation_matrix.unstack().sort_values(ascending=False).
 ↪drop_duplicates()
print("\nStrongest Correlation:")
print(strongest_corr[:30])  # Displaying the top 5 strongest correlations

# Finding the moderate correlation (between 0.3 and 0.7)
moderate_corr = strongest_corr[(strongest_corr < 0.7) & (strongest_corr >= 0.3)]
print("\nModerate Correlation:")
```

```
print(moderate_corr)

# Finding the weakest correlation
weakest_corr = correlation_matrix.unstack().sort_values().drop_duplicates()
print("\nWeakest Correlation:")
print(weakest_corr[:5])  # Displaying the top 5 weakest correlations
```

```
                                MMWR Year  MMWR Week  All Cause  \
MMWR Year                        1.000000  -0.156452  -0.032034
MMWR Week                       -0.156452   1.000000   0.007553
All Cause                       -0.032034   0.007553   1.000000
Natural Cause                   -0.032181   0.007483   0.999343
Malignant neoplasms              0.012838   0.001386   0.965457
Alzheimer disease               -0.032217  -0.007281   0.116138
Chronic lower respiratory diseases -0.007016 -0.018771   0.254713
Diseases of heart                0.001717  -0.011150   0.970740
Cerebrovascular diseases         0.011405  -0.007267   0.278655
COVID-19_Multiple Cause of Death -0.041055   0.047655   0.233667
Week                            -0.156452   1.000000   0.007553
Days_Since_Last_Week             0.092664   0.119697   0.002227
Total_Deaths                    -0.028358   0.008273   0.994881
Malignancy_Mortality_Rate        0.049489  -0.027132   0.371876
Heart_Disease_Mortality_Rate     0.045311  -0.041957   0.394220
Natural_Cause_Prop              -0.055960   0.002801   0.581320
Season                          -0.046604   0.583348  -0.037395

                                Natural Cause  Malignant neoplasms  \
MMWR Year                           -0.032181             0.012838
MMWR Week                            0.007483             0.001386
All Cause                            0.999343             0.965457
Natural Cause                        1.000000             0.962886
Malignant neoplasms                  0.962886             1.000000
Alzheimer disease                    0.117136             0.109277
Chronic lower respiratory diseases   0.254662             0.258360
Diseases of heart                    0.969820             0.960234
Cerebrovascular diseases             0.278413             0.289633
COVID-19_Multiple Cause of Death     0.245628             0.086697
Week                                 0.007483             0.001386
Days_Since_Last_Week                 0.001735            -0.000836
Total_Deaths                         0.995575             0.956414
Malignancy_Mortality_Rate            0.366184             0.519428
Heart_Disease_Mortality_Rate         0.390895             0.420624
Natural_Cause_Prop                   0.579714             0.561321
Season                              -0.041922             0.012371

                                Alzheimer disease  \
MMWR Year                               -0.032217
MMWR Week                               -0.007281
```

All Cause	0.116138
Natural Cause	0.117136
Malignant neoplasms	0.109277
Alzheimer disease	1.000000
Chronic lower respiratory diseases	0.482919
Diseases of heart	0.102748
Cerebrovascular diseases	0.528086
COVID-19_Multiple Cause of Death	0.195941
Week	-0.007281
Days_Since_Last_Week	0.003073
Total_Deaths	0.170954
Malignancy_Mortality_Rate	-0.321867
Heart_Disease_Mortality_Rate	-0.332906
Natural_Cause_Prop	-0.457149
Season	-0.023546

	Chronic lower respiratory diseases \
MMWR Year	-0.007016
MMWR Week	-0.018771
All Cause	0.254713
Natural Cause	0.254662
Malignant neoplasms	0.258360
Alzheimer disease	0.482919
Chronic lower respiratory diseases	1.000000
Diseases of heart	0.264781
Cerebrovascular diseases	0.587461
COVID-19_Multiple Cause of Death	0.189567
Week	-0.018771
Days_Since_Last_Week	-0.010226
Total_Deaths	0.309996
Malignancy_Mortality_Rate	-0.279188
Heart_Disease_Mortality_Rate	-0.246924
Natural_Cause_Prop	-0.424611
Season	-0.028994

	Diseases of heart \
MMWR Year	0.001717
MMWR Week	-0.011150
All Cause	0.970740
Natural Cause	0.969820
Malignant neoplasms	0.960234
Alzheimer disease	0.102748
Chronic lower respiratory diseases	0.264781
Diseases of heart	1.000000
Cerebrovascular diseases	0.288689
COVID-19_Multiple Cause of Death	0.144967
Week	-0.011150
Days_Since_Last_Week	-0.008793

Total_Deaths	0.966584
Malignancy_Mortality_Rate	0.390779
Heart_Disease_Mortality_Rate	0.528176
Natural_Cause_Prop	0.550739
Season	-0.034471

	Cerebrovascular diseases \
MMWR Year	0.011405
MMWR Week	-0.007267
All Cause	0.278655
Natural Cause	0.278413
Malignant neoplasms	0.289633
Alzheimer disease	0.528086
Chronic lower respiratory diseases	0.587461
Diseases of heart	0.288689
Cerebrovascular diseases	1.000000
COVID-19_Multiple Cause of Death	0.189077
Week	-0.007267
Days_Since_Last_Week	-0.000054
Total_Deaths	0.335578
Malignancy_Mortality_Rate	-0.272973
Heart_Disease_Mortality_Rate	-0.235500
Natural_Cause_Prop	-0.433810
Season	-0.012552

	COVID-19_Multiple Cause of Death \
MMWR Year	-0.041055
MMWR Week	0.047655
All Cause	0.233667
Natural Cause	0.245628
Malignant neoplasms	0.086697
Alzheimer disease	0.195941
Chronic lower respiratory diseases	0.189567
Diseases of heart	0.144967
Cerebrovascular diseases	0.189077
COVID-19_Multiple Cause of Death	1.000000
Week	0.047655
Days_Since_Last_Week	0.065516
Total_Deaths	0.306288
Malignancy_Mortality_Rate	-0.382908
Heart_Disease_Mortality_Rate	-0.295936
Natural_Cause_Prop	-0.256067
Season	-0.056187

	Week	Days_Since_Last_Week \
MMWR Year	-0.156452	0.092664
MMWR Week	1.000000	0.119697
All Cause	0.007553	0.002227

Natural Cause	0.007483	0.001735
Malignant neoplasms	0.001386	-0.000836
Alzheimer disease	-0.007281	0.003073
Chronic lower respiratory diseases	-0.018771	-0.010226
Diseases of heart	-0.011150	-0.008793
Cerebrovascular diseases	-0.007267	-0.000054
COVID-19_Multiple Cause of Death	0.047655	0.065516
Week	1.000000	0.119697
Days_Since_Last_Week	0.119697	1.000000
Total_Deaths	0.008273	0.005956
Malignancy_Mortality_Rate	-0.027132	-0.017254
Heart_Disease_Mortality_Rate	-0.041957	-0.037672
Natural_Cause_Prop	0.002801	-0.019633
Season	0.583348	0.097585

	Total_Deaths	Malignancy_Mortality_Rate	\
MMWR Year	-0.028358	0.049489	
MMWR Week	0.008273	-0.027132	
All Cause	0.994881	0.371876	
Natural Cause	0.995575	0.366184	
Malignant neoplasms	0.956414	0.519428	
Alzheimer disease	0.170954	-0.321867	
Chronic lower respiratory diseases	0.309996	-0.279188	
Diseases of heart	0.966584	0.390779	
Cerebrovascular diseases	0.335578	-0.272973	
COVID-19_Multiple Cause of Death	0.306288	-0.382908	
Week	0.008273	-0.027132	
Days_Since_Last_Week	0.005956	-0.017254	
Total_Deaths	1.000000	0.319043	
Malignancy_Mortality_Rate	0.319043	1.000000	
Heart_Disease_Mortality_Rate	0.351223	0.487904	
Natural_Cause_Prop	0.505903	0.636020	
Season	-0.039810	0.081567	

	Heart_Disease_Mortality_Rate	\
MMWR Year	0.045311	
MMWR Week	-0.041957	
All Cause	0.394220	
Natural Cause	0.390895	
Malignant neoplasms	0.420624	
Alzheimer disease	-0.332906	
Chronic lower respiratory diseases	-0.246924	
Diseases of heart	0.528176	
Cerebrovascular diseases	-0.235500	
COVID-19_Multiple Cause of Death	-0.295936	
Week	-0.041957	
Days_Since_Last_Week	-0.037672	
Total_Deaths	0.351223	

```
Malignancy_Mortality_Rate                          0.487904
Heart_Disease_Mortality_Rate                       1.000000
Natural_Cause_Prop                                 0.601368
Season                                            -0.012889

                                   Natural_Cause_Prop    Season
MMWR Year                                   -0.055960 -0.046604
MMWR Week                                    0.002801  0.583348
All Cause                                    0.581320 -0.037395
Natural Cause                                0.579714 -0.041922
Malignant neoplasms                          0.561321  0.012371
Alzheimer disease                           -0.457149 -0.023546
Chronic lower respiratory diseases          -0.424611 -0.028994
Diseases of heart                            0.550739 -0.034471
Cerebrovascular diseases                    -0.433810 -0.012552
COVID-19_Multiple Cause of Death            -0.256067 -0.056187
Week                                         0.002801  0.583348
Days_Since_Last_Week                        -0.019633  0.097585
Total_Deaths                                 0.505903 -0.039810
Malignancy_Mortality_Rate                    0.636020  0.081567
Heart_Disease_Mortality_Rate                 0.601368 -0.012889
Natural_Cause_Prop                           1.000000 -0.034203
Season                                      -0.034203  1.000000

Strongest Correlation:
MMWR Year                          MMWR Year
1.000000
Natural Cause                      All Cause
0.999343
                                   Total_Deaths
0.995575
All Cause                          Total_Deaths
0.994881
Diseases of heart                  All Cause
0.970740
                                   Natural Cause
0.969820
Total_Deaths                       Diseases of heart
0.966584
Malignant neoplasms                All Cause
0.965457
Natural Cause                      Malignant neoplasms
0.962886
Diseases of heart                  Malignant neoplasms
0.960234
Total_Deaths                       Malignant neoplasms
0.956414
Natural_Cause_Prop                 Malignancy_Mortality_Rate
```

```
0.636020
Heart_Disease_Mortality_Rate        Natural_Cause_Prop
0.601368
Cerebrovascular diseases            Chronic lower respiratory diseases
0.587461
Week                                Season
0.583348
All Cause                           Natural_Cause_Prop
0.581320
Natural_Cause_Prop                  Natural Cause
0.579714
                                    Malignant neoplasms
0.561321
                                    Diseases of heart
0.550739
Heart_Disease_Mortality_Rate        Diseases of heart
0.528176
Cerebrovascular diseases            Alzheimer disease
0.528086
Malignant neoplasms                 Malignancy_Mortality_Rate
0.519428
Natural_Cause_Prop                  Total_Deaths
0.505903
Heart_Disease_Mortality_Rate        Malignancy_Mortality_Rate
0.487904
Chronic lower respiratory diseases  Alzheimer disease
0.482919
Heart_Disease_Mortality_Rate        Malignant neoplasms
0.420624
                                    All Cause
0.394220
Natural Cause                       Heart_Disease_Mortality_Rate
0.390895
Malignancy_Mortality_Rate           Diseases of heart
0.390779
All Cause                           Malignancy_Mortality_Rate
0.371876
dtype: float64

Moderate Correlation:
Natural_Cause_Prop                  Malignancy_Mortality_Rate
0.636020
Heart_Disease_Mortality_Rate        Natural_Cause_Prop
0.601368
Cerebrovascular diseases            Chronic lower respiratory diseases
0.587461
Week                                Season
0.583348
```

```
All Cause                            Natural_Cause_Prop
0.581320
Natural_Cause_Prop                   Natural Cause
0.579714
                                     Malignant neoplasms
0.561321
                                     Diseases of heart
0.550739
Heart_Disease_Mortality_Rate         Diseases of heart
0.528176
Cerebrovascular diseases             Alzheimer disease
0.528086
Malignant neoplasms                  Malignancy_Mortality_Rate
0.519428
Natural_Cause_Prop                   Total_Deaths
0.505903
Heart_Disease_Mortality_Rate         Malignancy_Mortality_Rate
0.487904
Chronic lower respiratory diseases   Alzheimer disease
0.482919
Heart_Disease_Mortality_Rate         Malignant neoplasms
0.420624
                                     All Cause
0.394220
Natural Cause                        Heart_Disease_Mortality_Rate
0.390895
Malignancy_Mortality_Rate            Diseases of heart
0.390779
All Cause                            Malignancy_Mortality_Rate
0.371876
Natural Cause                        Malignancy_Mortality_Rate
0.366184
Total_Deaths                         Heart_Disease_Mortality_Rate
0.351223
Cerebrovascular diseases             Total_Deaths
0.335578
Malignancy_Mortality_Rate            Total_Deaths
0.319043
Total_Deaths                         Chronic lower respiratory diseases
0.309996
COVID-19_Multiple Cause of Death     Total_Deaths
0.306288
dtype: float64

Weakest Correlation:
Natural_Cause_Prop                   Alzheimer disease              -0.457149
                                     Cerebrovascular diseases       -0.433810
Chronic lower respiratory diseases   Natural_Cause_Prop             -0.424611
```

```
COVID-19_Multiple Cause of Death    Malignancy_Mortality_Rate      -0.382908
Alzheimer disease                   Heart_Disease_Mortality_Rate   -0.332906
dtype: float64
```

6 Multiple linear regression after removing the outliers

[477]:
```
import pandas as pd
import statsmodels.api as sm

# Assuming cleaned_df is your DataFrame with the provided columns
# Select predictor variables and outcome variable
predictor_cols = ['Cerebrovascular diseases','Alzheimer disease','Chronic lower␣
 ↪respiratory diseases']

outcome_col = ['Total_Deaths']

# Create a DataFrame with predictor variables
X = df_trimmed[predictor_cols]

# Add a constant term to the predictor variables (for intercept)
X = sm.add_constant(X)

# Create a Series with the outcome variable
y = df_trimmed[outcome_col]

# Concatenate X and y along the columns axis
df_copy = pd.concat([X, y], axis=1)

# Print the concatenated DataFrame
print(df_copy.head(5))

# Fit the multiple linear regression model
model = sm.OLS(y, X).fit()

# Print the model summary
print(model.summary())
```

```
   const  Cerebrovascular diseases  Alzheimer disease  \
0    1.0                        81                 54
1    1.0                        68                 41
2    1.0                        45                 53
3    1.0                        70                 54
4    1.0                        55                 58

   Chronic lower respiratory diseases  Total_Deaths
0                                  86          2776
1                                  72          2839
```

```
2                                   73          2626
3                                   61          2959
4                                   76          2586
                            OLS Regression Results
==============================================================================
Dep. Variable:           Total_Deaths   R-squared:                       0.134
Model:                            OLS   Adj. R-squared:                  0.134
Method:                 Least Squares   F-statistic:                     495.6
Date:                Wed, 10 Apr 2024   Prob (F-statistic):          1.75e-299
Time:                        14:06:02   Log-Likelihood:                -84347.
No. Observations:                9615   AIC:                         1.687e+05
Df Residuals:                    9611   BIC:                         1.687e+05
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
======================
                                        coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------------
----------------------
const                                1488.6473     33.943     43.857      0.000
1422.112    1555.182
Cerebrovascular diseases                9.9878      0.491     20.345      0.000
9.025      10.950
Alzheimer disease                      -2.7276      0.591     -4.616      0.000
-3.886      -1.569
Chronic lower respiratory diseases      8.1970      0.533     15.380      0.000
7.152       9.242
==============================================================================
Omnibus:                       21.256   Durbin-Watson:                   0.080
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               21.376
Skew:                           0.113   Prob(JB):                     2.28e-05
Kurtosis:                       2.952   Cond. No.                         247.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

[478]:
```
#split the dataset in training set and test set

X_train, X_test, y_train, y_test = train_test_split(df_trimmed[predictor_cols],␣
 ↪df_trimmed[outcome_col], test_size=0.3, random_state=0)

# Convert y_train and y_test to DataFrame with a single column
y_train = pd.DataFrame(y_train, columns=outcome_col)
y_test = pd.DataFrame(y_test, columns=outcome_col)
```

```
print("X_train shape:", X_train.shape)
print("X_test shape:", X_test.shape)
print("y_train shape:", y_train.shape)
print("y_test shape:", y_test.shape)
```

```
X_train shape: (6730, 3)
X_test shape: (2885, 3)
y_train shape: (6730, 1)
y_test shape: (2885, 1)
```

[479]:
```
from sklearn.linear_model import LinearRegression

# Create an instance of the LinearRegression class
lin_reg = LinearRegression()

# Fit the model using the training data
lin_reg.fit(X_train, y_train)

# After fitting, you can access the coefficients and intercept of the model
coefficients = lin_reg.coef_
intercept = lin_reg.intercept_

# Print the coefficients and intercept
print("Coefficients:", coefficients)
print("Intercept:", intercept)
```

```
Coefficients: [[ 9.93637803 -2.28435416  8.06153015]]
Intercept: [1478.89134253]
```

[480]:
```
# Make predictions on the testing data
y_pred = lin_reg.predict(X_test)

# Print the first few predictions
print("y_pred:", y_pred[:5])
```

```
y_pred: [[3454.83185337]
 [1888.8248982 ]
 [3462.16286402]
 [2840.01239434]
 [1764.44922648]]
```

[481]:
```
# # use model to predict
#2776
lin_reg.predict([[81,54,86]])
```

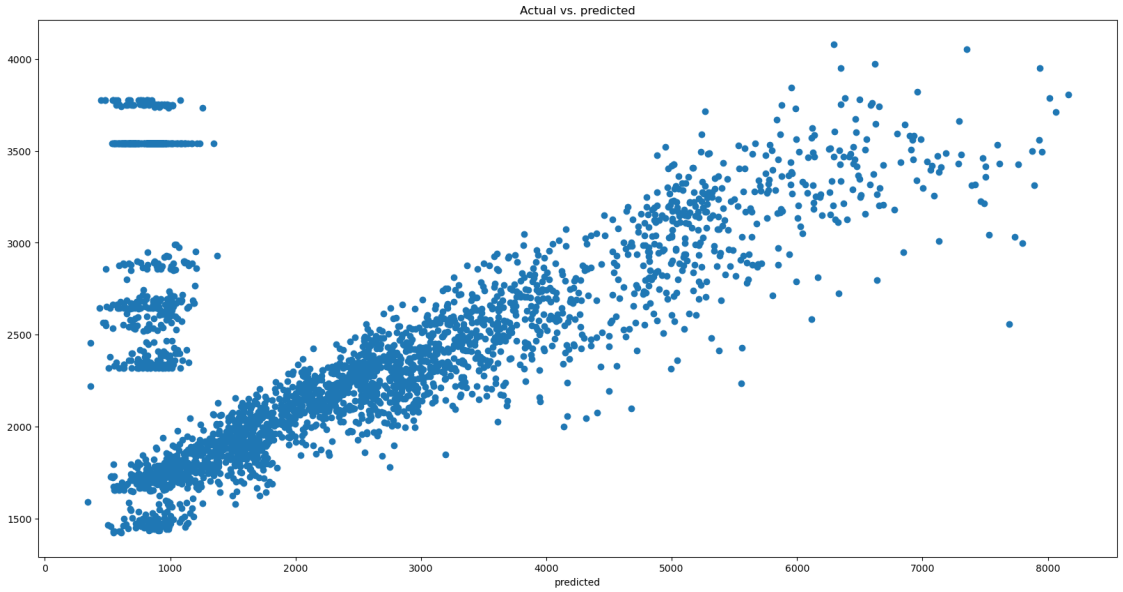
[481]:
```
array([[2853.6744312]])
```

[482]:
```
from sklearn.metrics import r2_score
r2_score(y_test,y_pred)
```

[482]:
```
0.1326009203447095
```

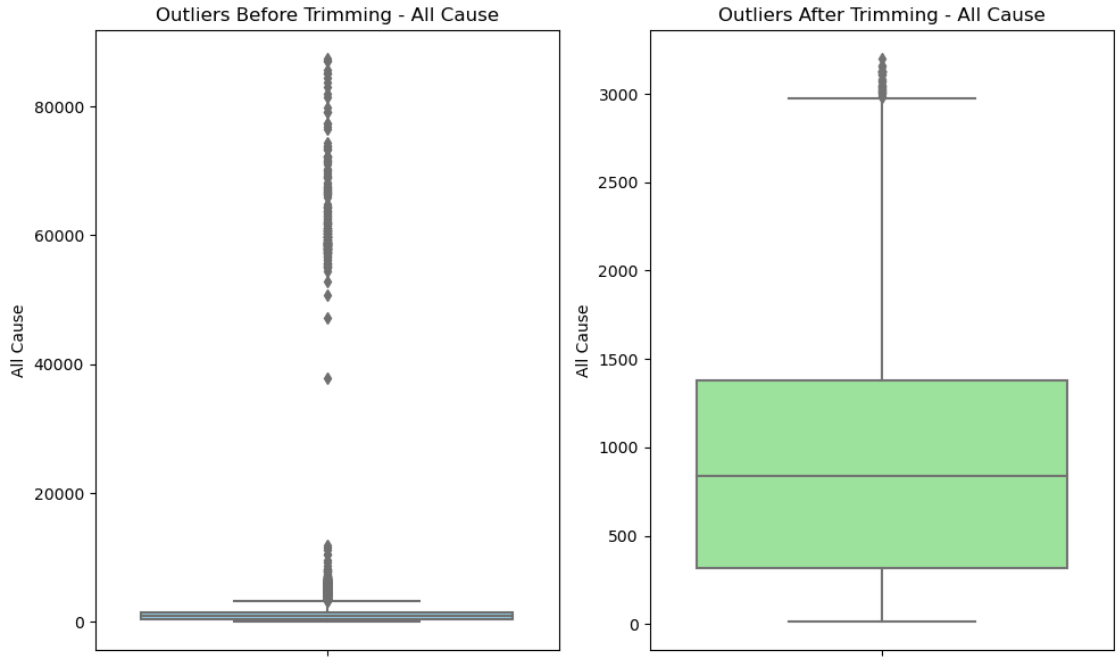
[483]:
```
#plot the results

import matplotlib.pyplot as plt
plt.figure(figsize=(20,10))
plt.scatter(y_test,y_pred)
plt.xlabel('Actual')
plt.xlabel('predicted')
plt.title('Actual vs. predicted')
```

[483]:
```
Text(0.5, 1.0, 'Actual vs. predicted')
```



[484]:
```
import seaborn as sns
import matplotlib.pyplot as plt

# Specify the column for which you want to compare outliers
column_to_check = 'All Cause'

# Detect outliers for the specified column using the IQR method
outliers_before = detect_outliers_iqr(df_uncleaned[column_to_check])

# Filter out rows containing outliers for the specified column
df_trimmed = df_trimmed.loc[~outliers_before]
```

```
# Create a box plot using Seaborn to visualize outliers before and after␣
 ↪trimming
plt.figure(figsize=(10, 6))

# Box plot before trimming outliers
plt.subplot(1, 2, 1)
sns.boxplot(y=df_uncleaned[column_to_check], color='skyblue')
plt.title(f'Outliers Before Trimming - {column_to_check}')

# Box plot after trimming outliers
plt.subplot(1, 2, 2)
sns.boxplot(y=df_trimmed[column_to_check], color='lightgreen')
plt.title(f'Outliers After Trimming - {column_to_check}')

plt.tight_layout()
plt.show()
```



[ ]: