Introduction

This study is to analyze mortality in the US based on causes across states. This study aims to answer some of questions such as are American facing increasing, decreasing of steady trend of death, what are the four leading causes of death, do individual states show the same 4 leading causes of death and if the year-by-year changes in the 4 leading causes of death nationwide.

There are 2 datasets used in this study and imported into the database. This study will be analyzed by Python programming.

Data Source

Publicly available files will be used.

- The first contains data on causes of death: NCHS_-_Leading_Causes_of_Death__United_States.csv
- The second contains population data: nst-est2018-01.xlsx

Both files have state-level information for multiple years.

In [1]:

```
import pandas as pd
import numpy as np

path = "/Users/qmacstore/Downloads/[BANA 680]/_Assignment/"
file = 'NCHS_-_Leading_Causes_of_Death__United_States.csv'
df = pd.read_csv(path+file)
df.head()
```

Out[1]:

	Year	113 Cause Name	Cause Name	State	Deaths	Age-adjusted Death Rate
0	2012	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Vermont	21	2.6
1	2016	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Vermont	30	3.7
2	2013	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Vermont	30	3.8
3	2000	Intentional self-harm (suicide) (*U03,X60-X84,...	Suicide	District of Columbia	23	3.8
4	2014	Nephritis, nephrotic syndrome and nephrosis (N...	Kidney disease	Arizona	325	4.1

Getting to understand the table

In [2]:

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10296 entries, 0 to 10295
Data columns (total 6 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Year                     10296 non-null  int64
 1   113 Cause Name           10296 non-null  object
 2   Cause Name               10296 non-null  object
 3   State                    10296 non-null  object
 4   Deaths                   10296 non-null  int64
 5   Age-adjusted Death Rate  10296 non-null  float64
dtypes: float64(1), int64(2), object(3)
memory usage: 482.8+ KB
None
```

In [3]:

```
# check for duplicates
df.duplicated().sum()
```

Out[3]: 0

But here Column 113 Cause Name is not required in the project and hence will be dropped.

In [4]:

```
df=df.drop('113 Cause Name', axis=1)
df.head()
```

Out[4]:

	Year	Cause Name	State	Deaths	Age-adjusted Death Rate
0	2012	Kidney disease	Vermont	21	2.6
1	2016	Kidney disease	Vermont	30	3.7
2	2013	Kidney disease	Vermont	30	3.8
3	2000	Suicide	District of Columbia	23	3.8
4	2014	Kidney disease	Arizona	325	4.1

1. Are Americans facing increasing, decreasing, or steady likelihood of death?

Total of deaths by year from 1999 to 2016

In [5]:

```
# not by All causes:
df1 = df[(df['State']=='United States') & (df['Cause Name'] != 'All causes')]

df1a = df1.groupby(['Year'])['Deaths'].sum().reset_index()
df1a
```

Out[5]:

	Year	Deaths
0	1999	1905826
1	2000	1902194
2	2001	1899358
3	2002	1918873
4	2003	1912115
5	2004	1864133
6	2005	1889981
7	2006	1854676
8	2007	1846301
9	2008	1872981
10	2009	1838501
11	2010	1852349
12	2011	1869321
13	2012	1876588
14	2013	1910311
15	2014	1938408
16	2015	2013017
17	2016	2034119

Analyis:

- Americans are facing increasing trend likelihood of deaths (exclude All causes)

In [6]:

```
# by All causes:
df1b = df[(df['State']=='United States') & (df['Cause Name'] == 'All causes')]

df1b_all = df1b.groupby(['Year'])['Deaths'].sum().reset_index()
df1b_all
```

Out[6]:

	Year	Deaths
0	1999	2391399
1	2000	2403351
2	2001	2416425
3	2002	2443387
4	2003	2448288
5	2004	2397615
6	2005	2448017
7	2006	2426264
8	2007	2423712
9	2008	2471984
10	2009	2437163
11	2010	2468435
12	2011	2515458
13	2012	2543279
14	2013	2596993
15	2014	2626418
16	2015	2712630
17	2016	2744248

Analysis:

- Americans are facing increasing trend of deaths (only All causes)

2. What are the 4 leading causes of death for Americans?

Top 4 causes of deaths from 1999 to 2016

In [7]:

```
df2 = df[(df['State']=='United States') & (df['Cause Name'] != 'All causes')]

df2b = df2.groupby(['Cause Name'])['Deaths'].sum()

sorted_grouped = df2b.sort_values(ascending=False)

top_4 = sorted_grouped.nlargest(4)
top_4
```

Out[7]:

```
Cause Name
Heart disease    11575183
Cancer           10244536
Stroke            2580140
CLRD              2434726
Name: Deaths, dtype: int64
```

Analysis:

- Top 4 causes of death for Americans are: Heart disease, Cancer, Stroke and CLRD

3. Do individual states show the same four leading causes of death?

In [8]:

```
us_deaths_causes_df1 = df[(df['State'] != 'United States') & (df['Cause Name'] != 'All causes')]

rows = []

for state in us_deaths_causes_df1['State'].unique():
    state_cause_df = us_deaths_causes_df1[us_deaths_causes_df1['State'] == state].groupby(['Cause Name'])['Deaths'].sum()
    state_cause_df.sort_values(ascending=False, inplace=True)

    # Get the top 4 causes
    cause_list = list(state_cause_df.keys())[:4] + [None] * (4 - len(state_cause_df))

    rows.append({'State': state, '1st cause': cause_list[0], '2nd cause': cause_list[1],
                 '3rd cause': cause_list[2], '4th cause': cause_list[3]})

df3 = pd.DataFrame(rows)

df3.sort_values(by='State', ascending=True, inplace=True)

display(df3)
```

	State	1st cause	2nd cause	3rd cause	4th cause
45	Alabama	Heart disease	Cancer	Stroke	CLRD
12	Alaska	Cancer	Heart disease	Unintentional injuries	Stroke
2	Arizona	Heart disease	Cancer	Unintentional injuries	CLRD
50	Arkansas	Heart disease	Cancer	Stroke	CLRD
9	California	Heart disease	Cancer	Stroke	CLRD
24	Colorado	Cancer	Heart disease	Unintentional injuries	CLRD
16	Connecticut	Heart disease	Cancer	Stroke	CLRD
27	Delaware	Heart disease	Cancer	CLRD	Stroke
1	District of Columbia	Heart disease	Cancer	Stroke	Unintentional injuries
23	Florida	Heart disease	Cancer	CLRD	Stroke
36	Georgia	Heart disease	Cancer	Stroke	Unintentional injuries
21	Hawaii	Heart disease	Cancer	Stroke	Unintentional injuries
15	Idaho	Heart disease	Cancer	CLRD	Stroke
18	Illinois	Heart disease	Cancer	Stroke	CLRD
38	Indiana	Heart disease	Cancer	CLRD	Stroke
5	Iowa	Heart disease	Cancer	Stroke	CLRD
42	Kansas	Heart disease	Cancer	Stroke	CLRD
46	Kentucky	Heart disease	Cancer	CLRD	Unintentional injuries
37	Louisiana	Heart disease	Cancer	Unintentional injuries	Stroke
32	Maine	Cancer	Heart disease	CLRD	Stroke
22	Maryland	Heart disease	Cancer	Stroke	CLRD
7	Massachusetts	Cancer	Heart disease	Stroke	CLRD
31	Michigan	Heart disease	Cancer	Stroke	CLRD
17	Minnesota	Cancer	Heart disease	Stroke	Unintentional injuries
39	Mississippi	Heart disease	Cancer	Unintentional injuries	Stroke
47	Missouri	Heart disease	Cancer	Stroke	CLRD
19	Montana	Heart disease	Cancer	CLRD	Unintentional injuries
29	Nebraska	Heart disease	Cancer	CLRD	Stroke
25	Nevada	Heart disease	Cancer	CLRD	Unintentional injuries
20	New Hampshire	Cancer	Heart disease	CLRD	Stroke
8	New Jersey	Heart disease	Cancer	Stroke	CLRD
28	New Mexico	Heart disease	Cancer	Unintentional injuries	CLRD
6	New York	Heart disease	Cancer	CLRD	Stroke
44	North Carolina	Heart disease	Cancer	Stroke	CLRD
10	North Dakota	Heart disease	Cancer	Stroke	Alzheimer's disease
30	Ohio	Heart disease	Cancer	CLRD	Stroke
48	Oklahoma	Heart disease	Cancer	CLRD	Stroke
13	Oregon	Cancer	Heart disease	Stroke	CLRD
34	Pennsylvania	Heart disease	Cancer	Stroke	CLRD
11	Rhode Island	Heart disease	Cancer	CLRD	Stroke
40	South Carolina	Heart disease	Cancer	Stroke	Unintentional injuries
3	South Dakota	Heart disease	Cancer	Stroke	CLRD
33	Tennessee	Heart disease	Cancer	Stroke	CLRD
35	Texas	Heart disease	Cancer	Stroke	Unintentional injuries
26	Utah	Heart disease	Cancer	Unintentional injuries	Stroke
0	Vermont	Cancer	Heart disease	CLRD	Unintentional injuries
41	Virginia	Heart disease	Cancer	Stroke	CLRD
4	Washington	Cancer	Heart disease	Stroke	CLRD
49	West Virginia	Heart disease	Cancer	CLRD	Unintentional injuries
43	Wisconsin	Heart disease	Cancer	Stroke	Unintentional injuries
14	Wyoming	Heart disease	Cancer	CLRD	Unintentional injuries

Analysis:

- It can be seen that the common pattern of 4 leading causes of individual states are mainly: Heart disease, Cancer, Stroke and CLRD

4. Are there year-by-year changes in the four leading causes of death nationwide?

In [9]:

```
us_deaths_causes_dfyr = df[(df['State'] == 'United States') & (df['Cause Name'] != 'All causes')]

rows = []

# Iterate through the years to find the top four causes of death for each year
for yr in us_deaths_causes_dfyr['Year'].unique():
    state_cause_yrdf = us_deaths_causes_dfyr[us_deaths_causes_dfyr['Year'] == yr].groupby(['Cause Name'])['Deaths'].sum()
    state_cause_yrdf.sort_values(ascending=False, inplace=True)

    # Get the top 4 causes
    cause_list = list(state_cause_yrdf.keys())[:4] + [None] * (4 - len(state_cause_yrdf))

    # Append the row to the list
    rows.append({'Year': yr, '1st cause': cause_list[0], '2nd cause': cause_list[1],
                 '3rd cause': cause_list[2], '4th cause': cause_list[3]})

df4 = pd.DataFrame(rows)

df4.sort_values(by='Year', ascending=True, inplace=True)

display(df4)
```

	Year	1st cause	2nd cause	3rd cause	4th cause
1	1999	Heart disease	Cancer	Stroke	CLRD
0	2000	Heart disease	Cancer	Stroke	CLRD
2	2001	Heart disease	Cancer	Stroke	CLRD
7	2002	Heart disease	Cancer	Stroke	CLRD
3	2003	Heart disease	Cancer	Stroke	CLRD
6	2004	Heart disease	Cancer	Stroke	CLRD
4	2005	Heart disease	Cancer	Stroke	CLRD
5	2006	Heart disease	Cancer	Stroke	CLRD
8	2007	Heart disease	Cancer	Stroke	CLRD
9	2008	Heart disease	Cancer	CLRD	Stroke
10	2009	Heart disease	Cancer	CLRD	Stroke
11	2010	Heart disease	Cancer	CLRD	Stroke
12	2011	Heart disease	Cancer	CLRD	Stroke
14	2012	Heart disease	Cancer	CLRD	Stroke
13	2013	Heart disease	Cancer	CLRD	Unintentional injuries
15	2014	Heart disease	Cancer	CLRD	Unintentional injuries
17	2015	Heart disease	Cancer	CLRD	Unintentional injuries
16	2016	Heart disease	Cancer	Unintentional injuries	CLRD

Analysis:

- Through out the year (1999-2016), Heart disease and Cancer remain top 1 and 2, respectively, while 3rd and 4th causes are normally Stroke, CLRD and Unintentional injuries.

Population

Import and explore Population data

In [10]:

```
file2 = 'nst-est2018-01.xlsx'
df_pop = pd.read_excel(path+file2, skiprows=3)
df_pop.head()
```

Out[10]:

	Unnamed: 0	Census	Estimates Base	2010	2011	2012	2013	2014	2015	2016	2017	2018
0	United States	308745538.0	308758105.0	309326085.0	311580009.0	313874218.0	316057727.0	318386421.0	320742673.0	323071342.0	325147121.0	327167434.0
1	Northeast	55317240.0	55318430.0	55380645.0	55600532.0	55776729.0	55907823.0	56015864.0	56047587.0	56058789.0	56072676.0	56111079.0
2	Midwest	66927001.0	66929743.0	66974749.0	67152631.0	67336937.0	67564135.0	67752238.0	67869139.0	67996917.0	68156035.0	68308744.0
3	South	114555744.0	114563045.0	114867066.0	116039399.0	117271075.0	118393244.0	119657737.0	121037542.0	122401186.0	123598424.0	124753948.0
4	West	71945553.0	71946887.0	72103625.0	72787447.0	73489477.0	74192525.0	74960582.0	75788405.0	76614450.0	77319986.0	77993663.0

Cleaning the data

In [11]:

```
# Rename column
df_pop.rename(columns={'Unnamed: 0': 'State'}, inplace=True)

# Cleaning
df_pop['State'] = df_pop['State'].str.replace('^\.', '', regex=True)

# Drop unused columns
df_pop = df_pop.drop(['Census','Estimates Base'], axis=1)

# Select States only:
df_pop1 = df_pop.iloc[5:56]
df_pop1
```

Out[11]:

	State	2010	2011	2012	2013	2014	2015	2016	2017	2018
5	Alabama	4785448.0	4798834.0	4815564.0	4830460.0	4842481.0	4853160.0	4864745.0	4875120.0	4887871.0
6	Alaska	713906.0	722038.0	730399.0	737045.0	736307.0	737547.0	741504.0	739786.0	737438.0
7	Arizona	6407774.0	6473497.0	6556629.0	6634999.0	6733840.0	6833596.0	6945452.0	7048876.0	7171646.0
8	Arkansas	2921978.0	2940407.0	2952109.0	2959549.0	2967726.0	2978407.0	2990410.0	3002997.0	3013825.0
9	California	37320903.0	37641823.0	37960782.0	38280824.0	38625139.0	38953142.0	39209127.0	39399349.0	39557045.0
10	Colorado	5048281.0	5121771.0	5193721.0	5270482.0	5351218.0	5452107.0	5540921.0	5615902.0	5695564.0
11	Connecticut	3579125.0	3588023.0	3594395.0	3594915.0	3594783.0	3587509.0	3578674.0	3573880.0	3572665.0
12	Delaware	899595.0	907316.0	915188.0	923638.0	932596.0	941413.0	949216.0	957078.0	967171.0
13	District of Columbia	605085.0	619602.0	634725.0	650431.0	662513.0	675254.0	686575.0	695691.0	702455.0
14	Florida	18845785.0	19093352.0	19326230.0	19563166.0	19860330.0	20224249.0	20629982.0	20976812.0	21299325.0
15	Georgia	9711810.0	9801578.0	9901496.0	9973326.0	10069001.0	10181111.0	10304763.0	10413055.0	10519475.0
16	Hawaii	1363963.0	1379252.0	1394905.0	1408453.0	1414862.0	1422484.0	1428105.0	1424203.0	1420491.0
17	Idaho	1570773.0	1583828.0	1595441.0	1611530.0	1631479.0	1651523.0	1682930.0	1718904.0	1754208.0
18	Illinois	12840762.0	12867291.0	12884119.0	12898269.0	12888962.0	12864342.0	12826895.0	12786196.0	12741080.0
19	Indiana	6490436.0	6516045.0	6537640.0	6568367.0	6593533.0	6608296.0	6633344.0	6660082.0	6691878.0
20	Iowa	3050767.0	3066054.0	3076097.0	3093078.0	3109504.0	3121460.0	3131785.0	3143637.0	3156145.0
21	Kansas	2858213.0	2869035.0	2885361.0	2893510.0	2900896.0	2909502.0	2911263.0	2910689.0	2911505.0
22	Kentucky	4348200.0	4369488.0	4386381.0	4404817.0	4414483.0	4425999.0	4438229.0	4453874.0	4468402.0
23	Louisiana	4544532.0	4575184.0	4600814.0	4624577.0	4644204.0	4664851.0	4678215.0	4670818.0	4659978.0
24	Maine	1327632.0	1328150.0	1327691.0	1328196.0	1330760.0	1328484.0	1331370.0	1335063.0	1338404.0
25	Maryland	5788642.0	5838991.0	5887072.0	5923704.0	5958165.0	5986717.0	6004692.0	6024891.0	6042718.0
26	Massachusetts	6566431.0	6613149.0	6663158.0	6713944.0	6763652.0	6795891.0	6826022.0	6863246.0	6902149.0
27	Michigan	9877535.0	9881521.0	9896930.0	9913349.0	9930589.0	9932573.0	9951890.0	9976447.0	9995915.0
28	Minnesota	5310843.0	5345668.0	5376550.0	5413693.0	5451522.0	5482503.0	5523409.0	5568155.0	5611179.0
29	Mississippi	2970536.0	2978470.0	2983767.0	2988797.0	2990623.0	2988693.0	2988298.0	2989663.0	2986530.0
30	Missouri	5995976.0	6009641.0	6024081.0	6040658.0	6056293.0	6071745.0	6087203.0	6108612.0	6126452.0
31	Montana	990722.0	997221.0	1003754.0	1013564.0	1021891.0	1030503.0	1040863.0	1053090.0	1062305.0
32	Nebraska	1829536.0	1840538.0	1853323.0	1865414.0	1879522.0	1891507.0	1905924.0	1917575.0	1929268.0
33	Nevada	2702464.0	2712799.0	2744566.0	2776972.0	2819012.0	2868666.0	2919772.0	2972405.0	3034392.0
34	New Hampshire	1316777.0	1319815.0	1323962.0	1326408.0	1333223.0	1336294.0	1342373.0	1349767.0	1356458.0
35	New Jersey	8799624.0	8827783.0	8845483.0	8858362.0	8866780.0	8870869.0	8874516.0	8888543.0	8908520.0
36	New Mexico	2064588.0	2080395.0	2087549.0	2092792.0	2090342.0	2090211.0	2092789.0	2093395.0	2095428.0
37	New York	19400080.0	19498514.0	19574549.0	19628043.0	19656330.0	19661411.0	19641589.0	19590719.0	19542209.0
38	North Carolina	9574293.0	9656754.0	9749123.0	9843599.0	9933944.0	10033079.0	10156679.0	10270800.0	10383620.0
39	North Dakota	674710.0	685136.0	701116.0	721999.0	737382.0	754022.0	754353.0	755176.0	760077.0
40	Ohio	11539327.0	11543463.0	11548369.0	11576576.0	11602973.0	11617850.0	11635003.0	11664129.0	11689442.0
41	Oklahoma	3759632.0	3787821.0	3818600.0	3853205.0	3878367.0	3909831.0	3926769.0	3932640.0	3943079.0
42	Oregon	3837532.0	3871728.0	3899118.0	3922908.0	3964106.0	4016918.0	4091404.0	4146592.0	4190713.0
43	Pennsylvania	12711158.0	12744583.0	12766827.0	12776621.0	12789101.0	12785759.0	12783538.0	12790447.0	12807060.0
44	Rhode Island	1053938.0	1053536.0	1054601.0	1055122.0	1056017.0	1056173.0	1057063.0	1056486.0	1057315.0
45	South Carolina	4635656.0	4671422.0	4717112.0	4764153.0	4823793.0	4892253.0	4958235.0	5021219.0	5084127.0
46	South Dakota	816165.0	823484.0	833496.0	842270.0	849088.0	853933.0	862890.0	873286.0	882235.0
47	Tennessee	6355301.0	6397410.0	6451281.0	6493432.0	6540826.0	6590808.0	6645011.0	6708794.0	6770010.0
48	Texas	25242679.0	25646227.0	26089620.0	26489464.0	26977142.0	27486814.0	27937492.0	28322717.0	28701845.0
49	Utah	2775334.0	2814216.0	2853467.0	2897927.0	2937399.0	2982497.0	3042613.0	3103118.0	3161105.0
50	Vermont	625880.0	626979.0	626063.0	626212.0	625218.0	625197.0	623644.0	624525.0	626299.0
51	Virginia	8023680.0	8100469.0	8185229.0	8253053.0	8312076.0	8362907.0	8410946.0	8465207.0	8517685.0
52	Washington	6742902.0	6821655.0	6892876.0	6962906.0	7052439.0	7163543.0	7294680.0	7425432.0	7535591.0
53	West Virginia	1854214.0	1856074.0	1856764.0	1853873.0	1849467.0	1841996.0	1830929.0	1817048.0	1805832.0
54	Wisconsin	5690479.0	5704755.0	5719855.0	5736952.0	5751974.0	5761406.0	5772958.0	5792051.0	5813568.0
55	Wyoming	564483.0	567224.0	576270.0	582123.0	582548.0	585668.0	584290.0	578934.0	577737.0

Melt the DataFrame to create a year column

In [12]:

```
df_melted = df_pop1.melt(id_vars=["State"],
                    var_name="Year",
                    value_name="Population")

# Convert the Year column to integer type
df_melted['Year'] = df_melted['Year'].astype(int)
df_melted.head()
```

Out[12]:

	State	Year	Population
0	Alabama	2010	4785448.0
1	Alaska	2010	713906.0
2	Arizona	2010	6407774.0
3	Arkansas	2010	2921978.0
4	California	2010	37320903.0

In [13]:

```
df['Year'].unique()
```

Out[13]:

```
array([2012, 2016, 2013, 2000, 2014, 2009, 2011, 2015, 2001, 1999, 2006,
       2002, 2003, 2008, 2005, 2004, 2007, 2010])
```

In [14]:

```
df_melted['Year'].unique()
```

Out[14]:

```
array([2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018])
```

In [15]:

```
# filter years (2010-2016)
df0 = df[~(df.Year.isin([1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007,
                          2008, 2009, 2017, 2018]))]
df0.head()
```

Out[15]:

	Year	Cause Name	State	Deaths	Age-adjusted Death Rate
0	2012	Kidney disease	Vermont	21	2.6
1	2016	Kidney disease	Vermont	30	3.7
2	2013	Kidney disease	Vermont	30	3.8
4	2014	Kidney disease	Arizona	325	4.1
6	2011	Kidney disease	South Dakota	49	4.5

Merge Death data and Population data

In [16]:

```
# Merge 2 datasets:

merged_df = pd.merge(df0, df_melted, on=['State', 'Year'], how='left')
merged_df.head()
```

Out[16]:

	Year	Cause Name	State	Deaths	Age-adjusted Death Rate	Population
0	2012	Kidney disease	Vermont	21	2.6	626063.0
1	2016	Kidney disease	Vermont	30	3.7	623644.0
2	2013	Kidney disease	Vermont	30	3.8	626212.0
3	2014	Kidney disease	Arizona	325	4.1	6733840.0
4	2011	Kidney disease	South Dakota	49	4.5	823484.0

Calculate Death Rate

In [17]:

```
merged_df['Death Rate'] = (merged_df['Deaths'] / merged_df['Population']) * 100000
merged_df.head()
```

Out[17]:

	Year	Cause Name	State	Deaths	Age-adjusted Death Rate	Population	Death Rate
0	2012	Kidney disease	Vermont	21	2.6	626063.0	3.354295
1	2016	Kidney disease	Vermont	30	3.7	623644.0	4.810437
2	2013	Kidney disease	Vermont	30	3.8	626212.0	4.790710
3	2014	Kidney disease	Arizona	325	4.1	6733840.0	4.826370
4	2011	Kidney disease	South Dakota	49	4.5	823484.0	5.950328

Standardize the Death Rates

In [18]:

```
# Standardize the Death Rates
mean_rate = merged_df['Death Rate'].mean()
std_dev_rate = merged_df['Death Rate'].std()
merged_df['Standardized Rate'] = (merged_df['Death Rate'] - mean_rate) / std_dev_rate

std = merged_df[['Year', 'State', 'Cause Name', 'Deaths', 'Population', 'Death Rate', 'Standardized Rate']]
std
```

Out[18]:

	Year	State	Cause Name	Deaths	Population	Death Rate	Standardized Rate
0	2012	Vermont	Kidney disease	21	626063.0	3.354295	-0.548988
1	2016	Vermont	Kidney disease	30	623644.0	4.810437	-0.542917
2	2013	Vermont	Kidney disease	30	626212.0	4.790710	-0.542999
3	2014	Arizona	Kidney disease	325	6733840.0	4.826370	-0.542850
4	2011	South Dakota	Kidney disease	49	823484.0	5.950328	-0.538164
...	...	...	...	...	...	...	...
3999	2011	West Virginia	All causes	21867	1856074.0	1178.131906	4.349228
4000	2011	Mississippi	All causes	29278	2978470.0	982.987910	3.535579
4001	2013	Mississippi	All causes	30703	2988797.0	1027.269500	3.720210
4002	2010	Mississippi	All causes	28965	2970536.0	975.076552	3.502592
4003	2015	Mississippi	All causes	31783	2988693.0	1063.441444	3.871028

4004 rows × 7 columns

Interpretation of the Standardized Rate:

- A standardized rate greater than 1 indicates that the original rate is above the average.
- A standardized rate less than -1 suggests it is below average.
- Rates that are significantly above or below 1 or -1 may be considered outliers.

Normalize Death Rate and Population using Min-Max Normalization

In [19]:

```
# Normalize Deaths and Population using Min-Max Normalization

merged_df['Normalized Deaths'] = (merged_df['Deaths'] - merged_df['Deaths'].min()) / (merged_df['Deaths'].max() - merged_df['Deaths'].min())
merged_df['Normalized Population'] = (merged_df['Population'] - merged_df['Population'].min()) / (merged_df['Population'].max() - merged_df['Population'].mi

normal = merged_df[['Year', 'State', 'Cause Name', 'Deaths', 'Population',
                    'Death Rate', 'Normalized Deaths', 'Normalized Population']]
normal
```

Out[19]:

	Year	State	Cause Name	Deaths	Population	Death Rate	Normalized Deaths	Normalized Population
0	2012	Vermont	Kidney disease	21	626063.0	3.354295	0.000000	0.001593
1	2016	Vermont	Kidney disease	30	623644.0	4.810437	0.000003	0.001531
2	2013	Vermont	Kidney disease	30	626212.0	4.790710	0.000003	0.001597
3	2014	Arizona	Kidney disease	325	6733840.0	4.826370	0.000111	0.159643
4	2011	South Dakota	Kidney disease	49	823484.0	5.950328	0.000010	0.006702
...	...	...	...	...	...	...	...	...
3999	2011	West Virginia	All causes	21867	1856074.0	1178.131906	0.007961	0.033422
4000	2011	Mississippi	All causes	29278	2978470.0	982.987910	0.010661	0.062466
4001	2013	Mississippi	All causes	30703	2988797.0	1027.269500	0.011181	0.062734
4002	2010	Mississippi	All causes	28965	2970536.0	975.076552	0.010547	0.062261
4003	2015	Mississippi	All causes	31783	2988693.0	1063.441444	0.011574	0.062731

4004 rows × 8 columns

Interpretation of Normalized Values:

- Normalized values indicate how each value relates to the minimum and maximum values within the dataset.
- A normalized death count of 0.8 means that the number of deaths is closer to the maximum value than the minimum.
- A normalized population of 0.1 indicates that the population is closer to the minimum value than the maximum.

Conclusions

- There is an increasing trend of deaths among Americans.
- Top 4 causes of death are Heart diseases, Cancer, Stroke and CLRD.

Future Work

- Create interactive charts with updated data

In [ ]: