BCHI

April 1, 2019

BCHI Data Analysis
Following are the steps we followed for data analysis

1. Import the libraries

```
In [1]: #Import the Libraries
        import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt

        %matplotlib inline

        import seaborn as sns
        import warnings
        warnings.filterwarnings('ignore')
```

2.Load the dataset.

```
In [2]: dataset=pd.read_csv('data.csv')

In [3]: dataset.head(3)

Out[3]:                   Indicator Category  \
        0  Behavioral Health/Substance Abuse
        1  Behavioral Health/Substance Abuse
        2  Behavioral Health/Substance Abuse

                                                 Indicator  Year   Sex  \
        0  Opioid-Related Unintentional Drug Overdose Mor...  2010  Both
        1  Opioid-Related Unintentional Drug Overdose Mor...  2010  Both
        2  Opioid-Related Unintentional Drug Overdose Mor...  2010  Both

          Race/Ethnicity  Value                           Place  \
        0            All    1.7                  Washington, DC
        1            All    2.2  Fort Worth (Tarrant County), TX
        2            All    2.3     Oakland (Alameda County), CA

                                BCHC Requested Methodology  \
```

```
0  Age-Adjusted rate of opioid-related mortality ...
1  Age-adjusted rate of opioid-related mortality ...
2  Age-adjusted rate of opioid-related mortality ...

                                                Source  \
0  D.C. Department of Health, Center for Policy, ...
1              National Center for Health Statistics
2                                         CDC Wonder

                                               Methods  \
0                                                  NaN
1                                                  NaN
2  Age-adjusted rate of opioid-related mortality ...

                                                 Notes  \
0  This indicator is not exclusive of other drugs...
1  This indicator is not exclusive of other drugs...
2  Data is for Alameda County. This indicator is ...

   90% Confidence Level - Low  90% Confidence Level - High  \
0                         NaN                          NaN
1                         NaN                          NaN
2                         NaN                          NaN

   95% Confidence Level - Low  95% Confidence Level - High
0                         NaN                          NaN
1                         1.5                          3.0
2                         1.6                          3.2
```

Above we saw the column names and we might need to fix the spaces in the column names. In order to change that we need to first know what are the actual names of the columns.

We do that using the pandas function columns to list all the columns

```
In [4]: dataset.columns
```

```
Out[4]: Index(['Indicator Category', 'Indicator', 'Year', 'Sex', 'Race/Ethnicity',
               'Value', 'Place', 'BCHC Requested Methodology', 'Source', 'Methods',
               'Notes', '90% Confidence Level - Low', '90% Confidence Level - High',
               '95% Confidence Level - Low', '95% Confidence Level - High'],
              dtype='object')
```

Now we rename the columns

```
In [5]: dataset.rename(columns={'Indicator Category':'indicator_category','Indicator':'indicato
               'Value':'Value', 'Place':'Place', 'BCHC Requested Methodology':'BCHC_req_meth',
               'Notes':'Notes', '90% Confidence Level - Low':'90pc_con_lvl-low', '90% Confidenc
               '95% Confidence Level - Low':'95pc_con_lvl-low','95% Confidence Level - High':'9
```

3.Now we need to filter the data according to the indicator category. We use one of the values "Cancer".

```
In [6]: cancer_ds = dataset.loc[dataset["indicator_category"] == "Cancer"]
```

4.And then we remove empty columns and unnecessary columns

```
In [7]: cancer_ds.drop(['indicator_category','BCHC_req_meth','Source','Methods','Notes','90pc_
                        axis = 1, inplace= True)
```

5. Now we remove all the rows which has NaN or NA values

```
In [8]: cancer_ds.dropna(axis=0, how='any',inplace= True)

In [9]: cancer_ds.to_csv("Social_and_economic_Factors.csv")

In [10]: cancer_ds.head(3)
```

Out[10]:

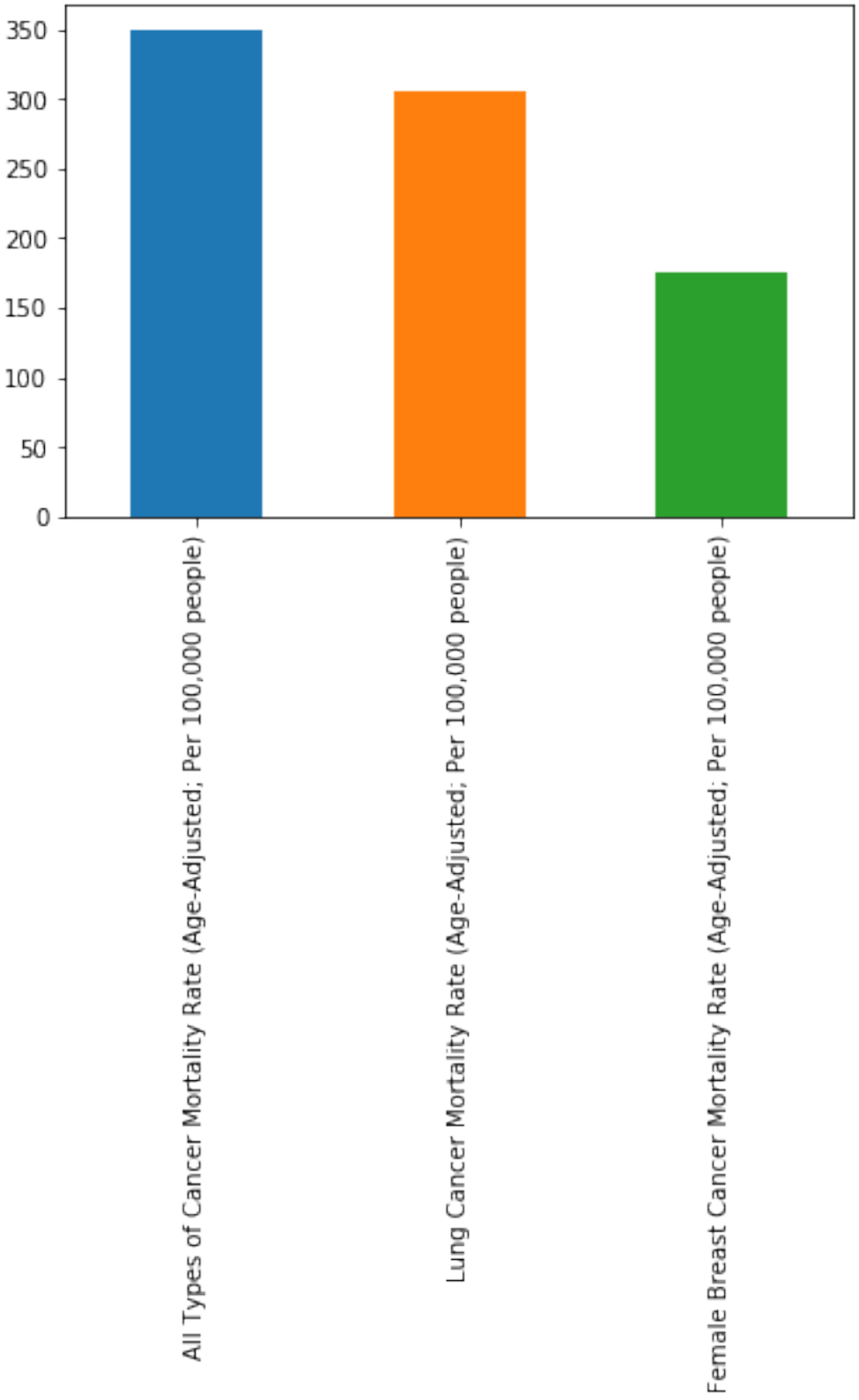
	indicator	Year	Sex	Race/Ethnicity	Value	Place	95pc_con_lvl-low	95pc_con_lvl-high
1468	All Types of Cancer Mortality Rate (Age-Adjust...	2010	Both	All	149.9	San Francisco, CA	145.3	154.6
1469	All Types of Cancer Mortality Rate (Age-Adjust...	2010	Both	All	156.7	Seattle, WA	146.5	167.6
1470	All Types of Cancer Mortality Rate (Age-Adjust...	2010	Both	All	159.9	San Antonio, TX	153.5	166.4

```
In [55]: sorted_cancer = cancer_ds['indicator'].value_counts()
         sorted_cancer
```

```
Out[55]: All Types of Cancer Mortality Rate (Age-Adjusted; Per 100,000 people)     350
         Lung Cancer Mortality Rate (Age-Adjusted; Per 100,000 people)             306
         Female Breast Cancer Mortality Rate (Age-Adjusted; Per 100,000 people)    175
         Name: indicator, dtype: int64
```

```
In [56]: sorted_cancer = cancer_ds['indicator'].value_counts().plot(kind='bar')
         sorted_cancer
```

```
Out[56]: <matplotlib.axes._subplots.AxesSubplot at 0x1a196dde10>
```


350
300
250
200
150
100
50
0
All Types of Cancer Mortality Rate (Age-Adjusted; Per 100,000 people)
Lung Cancer Mortality Rate (Age-Adjusted; Per 100,000 people)
Female Breast Cancer Mortality Rate (Age-Adjusted; Per 100,000 people)

In [ ]:

In [ ]:

In [ ]: