In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [2]:

```
data_suicide = pd.read_csv("suicide.csv")
```

In [3]:

```
data_suicide.head()
```

Out[3]:

	Unnamed: 0	Unnamed: 1	Probability (%) of dying between age 30 and exact age 70 from any of cardiovascular disease, cancer, diabetes, or chronic respiratory disease	Probability (%) of dying between age 30 and exact age 70 from any of cardiovascular disease, cancer, diabetes, or chronic respiratory disease.1	Probability (%) of dying between age 30 and exact age 70 from any of cardiovascular disease, cancer, diabetes, or chronic respiratory disease.2	Crude suicide rates (per 100 000 population)	Crud suicid rates (pe 100 00 population).
0	Country	Year	Both sexes	Male	Female	Both sexes	Mal
1	Afghanistan	2019	35.3 [21.8-49.8]	34.4 [21.9-47.9]	36.2 [21.7-51.6]	4.1 [2.3-7.0]	4.6 [2.6-7.9
2	Afghanistan	2018	35.4 [21.7-50.1]	34.7 [22.0-48.5]	36.0 [21.4-51.6]	4.1 [2.3-6.8]	4.6 [2.6-7.8
3	Afghanistan	2017	35.5 [21.7-50.4]	35.0 [22.0-49.1]	35.9 [21.3-51.7]	4.1 [2.3-6.8]	4.6 [2.7-7.9
4	Afghanistan	2016	35.6 [21.6-50.7]	35.1 [22.0-49.4]	36.0 [21.2-51.9]	4.0 [2.3-6.7]	4.4 [2.6-7.4

In [4]:

```
data_suicide.tail()
```

Out[4]:

	Unnamed: 0	Unnamed: 1	Probability (%) of dying between age 30 and exact age 70 from any of cardiovascular disease, cancer, diabetes, or chronic respiratory disease	Probability (%) of dying between age 30 and exact age 70 from any of cardiovascular disease, cancer, diabetes, or chronic respiratory disease.1	Probability (%) of dying between age 30 and exact age 70 from any of cardiovascular disease, cancer, diabetes, or chronic respiratory disease.2	Crude suicide rates (per 100 000 population)	Cr sui rates 100 populatio
3656	Zimbabwe	2004	24.5 [15.7-35.3]	24.6 [16.7-34.7]	24.5 [15.0-35.8]	14.2 [7.5-22.8]	15.7 [2
3657	Zimbabwe	2003	23.1 [14.7-33.4]	23.5 [15.9-33.3]	22.7 [13.7-33.5]	13.2 [6.9-21.1]	15.1 [2
3658	Zimbabwe	2002	22.7 [14.4-32.9]	24.1 [16.3-33.9]	21.5 [13.0-32.0]	13.4 [7.0-21.6]	15.5 [2
3659	Zimbabwe	2001	21.7 [13.8-31.6]	23.6 [16.0-33.2]	20.2 [12.1-30.4]	12.8 [6.6-20.9]	14.4 [2
3660	Zimbabwe	2000	21.4 [13.6-31.2]	23.3 [15.9-32.8]	19.8 [11.7-29.9]	13.3 [6.7-21.7]	14.1 [2

In [5]:

```
data_suicide.shape
```

Out[5]:

```
(3661, 8)
```

In [6]:

```
data_suicide.columns
```

Out[6]:

```
Index(['Unnamed: 0', 'Unnamed: 1',
       'Probability (%) of dying between age 30 and exact age 70 from any
of cardiovascular disease, cancer, diabetes, or chronic respiratory diseas
e',
       'Probability (%) of dying between age 30 and exact age 70 from any
of cardiovascular disease, cancer, diabetes, or chronic respiratory diseas
e.1',
       'Probability (%) of dying between age 30 and exact age 70 from any
of cardiovascular disease, cancer, diabetes, or chronic respiratory diseas
e.2',
       'Crude suicide rates (per 100 000 population)',
       'Crude suicide rates (per 100 000 population).1',
       'Crude suicide rates (per 100 000 population).2'],
      dtype='object')
```

In [9]:

```
columns = ['Country', 'Year',
           'ProbDyingBoth','ProbDyingMale','ProbDyingFemale',
           'SuicideBoth','SuicideMale','SuicideFemale']

values = data_suicide.iloc[1:,:].values
data_suicide = pd.DataFrame(values, columns=columns)

for col in columns[2:]:
    data_suicide[col] = data_suicide[col].map(lambda x: x.split('[')[0]).astype('float')
```

In [13]:

```
data_suicide['ProbDying_male_minus_female'] = data_suicide['ProbDyingMale'] - data_suici
data_suicide['Suicide_male_minus_female'] = data_suicide['SuicideMale'] - data_suicide[
```

In [14]:

```
data_suicide.head()
```

Out[14]:

	Country	Year	ProbDyingBoth	ProbDyingMale	ProbDyingFemale	SuicideBoth	SuicideMale
0	Afghanistan	2019	35.3	34.4	36.2	4.1	4.6
1	Afghanistan	2018	35.4	34.7	36.0	4.1	4.6
2	Afghanistan	2017	35.5	35.0	35.9	4.1	4.6
3	Afghanistan	2016	35.6	35.1	36.0	4.0	4.4
4	Afghanistan	2015	35.6	35.4	35.7	4.0	4.3

In [15]:

```
data_suicide.tail()
```

Out[15]:

	Country	Year	ProbDyingBoth	ProbDyingMale	ProbDyingFemale	SuicideBoth	SuicideMal
3655	Zimbabwe	2004	24.5	24.6	24.5	14.2	15.
3656	Zimbabwe	2003	23.1	23.5	22.7	13.2	15.
3657	Zimbabwe	2002	22.7	24.1	21.5	13.4	15.
3658	Zimbabwe	2001	21.7	23.6	20.2	12.8	14.
3659	Zimbabwe	2000	21.4	23.3	19.8	13.3	14.

In [16]:

```
data_suicide.shape
```

Out[16]:

```
(3660, 10)
```

In [17]:

```
data_suicide.columns
```

Out[17]:

```
Index(['Country', 'Year', 'ProbDyingBoth', 'ProbDyingMale', 'ProbDyingFema
le',
       'SuicideBoth', 'SuicideMale', 'SuicideFemale',
       'ProbDying_male_minus_female', 'Suicide_male_minus_female'],
      dtype='object')
```

In [18]:

```
data_suicide.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3660 entries, 0 to 3659
Data columns (total 10 columns):
 #   Column                        Non-Null Count  Dtype  
---  ------                        --------------  -----  
 0   Country                       3660 non-null   object 
 1   Year                          3660 non-null   object 
 2   ProbDyingBoth                 3660 non-null   float64
 3   ProbDyingMale                 3660 non-null   float64
 4   ProbDyingFemale               3660 non-null   float64
 5   SuicideBoth                   3660 non-null   float64
 6   SuicideMale                   3660 non-null   float64
 7   SuicideFemale                 3660 non-null   float64
 8   ProbDying_male_minus_female   3660 non-null   float64
 9   Suicide_male_minus_female     3660 non-null   float64
dtypes: float64(8), object(2)
memory usage: 286.1+ KB
```

In [19]:

```
data_suicide.describe()
```

Out[19]:

	ProbDyingBoth	ProbDyingMale	ProbDyingFemale	SuicideBoth	SuicideMale	SuicideFemal
count	3660.000000	3660.000000	3660.000000	3660.000000	3660.000000	3660.00000
mean	22.058415	25.639590	18.670464	10.629180	16.370546	5.05584
std	8.170809	9.727201	7.530725	8.986046	14.950656	4.03554
min	7.300000	9.600000	4.400000	0.000000	0.000000	0.00000
25%	16.100000	18.000000	13.100000	4.900000	7.000000	2.30000
50%	21.700000	24.500000	18.300000	8.100000	12.000000	4.00000
75%	26.600000	31.500000	23.300000	13.200000	20.200000	6.60000
max	56.000000	64.100000	47.800000	92.600000	147.800000	39.50000

In [20]:

```
data_suicide.isnull().sum()
```

Out[20]:

```
Country                         0
Year                            0
ProbDyingBoth                   0
ProbDyingMale                   0
ProbDyingFemale                 0
SuicideBoth                     0
SuicideMale                     0
SuicideFemale                   0
ProbDying_male_minus_female     0
Suicide_male_minus_female       0
dtype: int64
```

In [35]:

```
overall_suicides = data_suicide.groupby('Country').sum()
overall_suicides.drop(['ProbDyingBoth','ProbDyingMale','ProbDyingFemale',
                       'SuicideMale','SuicideFemale','ProbDying_male_minus_female',
                       'Suicide_male_minus_female'], axis = 1, inplace = True)
print('Overall Sucides = ' , overall_suicides['SuicideBoth'].sum())
overall_suicides.sort_values(by = 'SuicideBoth', ascending = False).head()
```

```
Overall Sucides =  38902.8
```

Out[35]:

Country	SuicideBoth
Lesotho	1191.4
Eswatini	820.0
Russian Federation	794.4
Lithuania	759.6
Guyana	691.5

In [36]:

```
overall_suicides = data_suicide.groupby('Country').sum()
overall_suicides.drop(['ProbDyingBoth','ProbDyingMale','ProbDyingFemale',
                       'SuicideMale','SuicideFemale','ProbDying_male_minus_female',
                       'Suicide_male_minus_female'], axis = 1, inplace = True)
print('Overall Sucides = ' , overall_suicides['SuicideBoth'].sum())
overall_suicides.sort_values(by = 'SuicideBoth', ascending = False).tail()
```

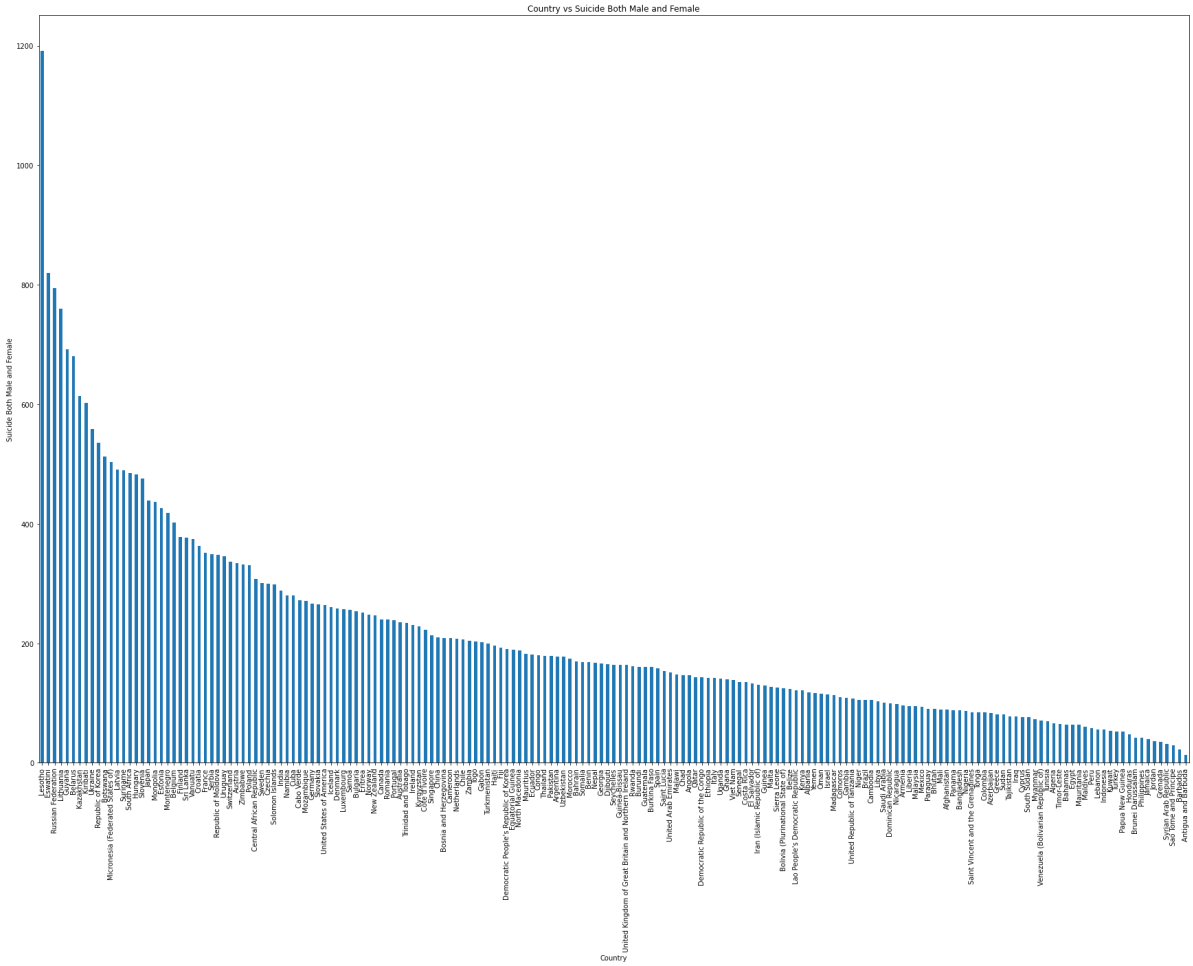
```
Overall Sucides =  38902.8
```

Out[36]:

Country	SuicideBoth
Grenada	35.2
Syrian Arab Republic	31.2
Sao Tome and Principe	29.7
Barbados	22.6
Antigua and Barbuda	13.1

In [41]:

```
plt.subplots(figsize = (30, 20))
cr = overall_suicides['SuicideBoth'].sort_values(ascending = False)
ax = cr.plot.bar()
ax.set_xlabel('Country')
ax.set_ylabel('Suicide Both Male and Female')
ax.set_title('Country vs Suicide Both Male and Female')
plt.show()
print(cr)
```




```
Country
Lesotho                  1191.4
Eswatini                  820.0
Russian Federation        794.4
Lithuania                 759.6
Guyana                    691.5
                          ...
Grenada                    35.2
Syrian Arab Republic       31.2
Sao Tome and Principe      29.7
Barbados                   22.6
Antigua and Barbuda        13.1
Name: SuicideBoth, Length: 183, dtype: float64
```

In [42]:

```
india_suicides = data_suicide[data_suicide['Country'] == 'India']
```

In [43]:

```
india_suicides.head()
```

Out[43]:

	Country	Year	ProbDyingBoth	ProbDyingMale	ProbDyingFemale	SuicideBoth	SuicideMale
1500	India	2019	21.9	24.6	19.0	12.7	14.1
1501	India	2018	22.0	24.7	19.1	12.6	14.0
1502	India	2017	22.2	25.0	19.3	12.0	13.4
1503	India	2016	22.6	25.4	19.7	12.1	13.4
1504	India	2015	22.6	25.4	19.8	12.3	13.7

In [44]:

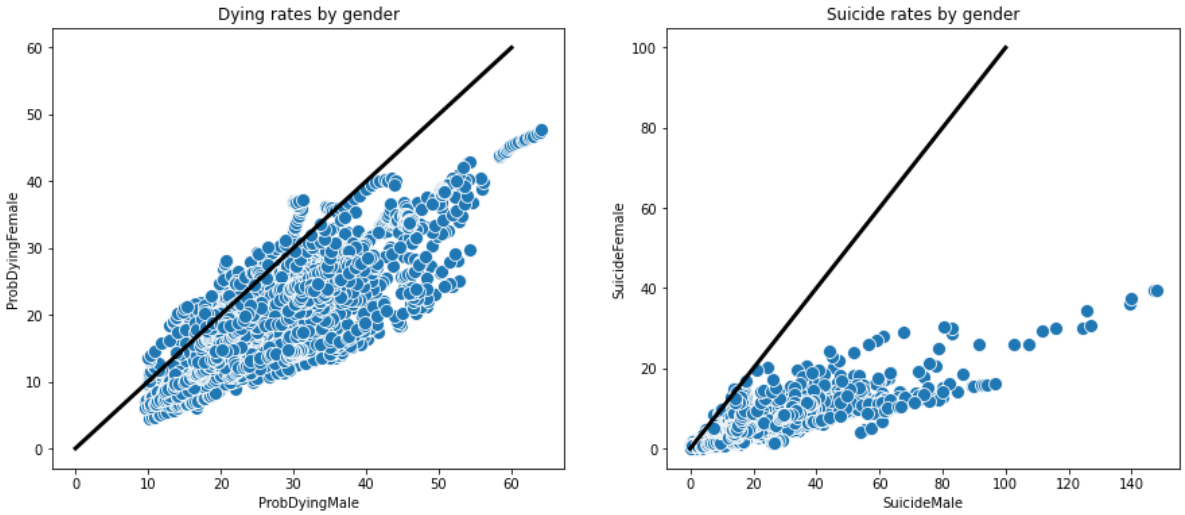
```
fig, ax = plt.subplots(nrows=1, ncols=2, figsize=(15,6))

ax[0].plot([0,60],[0,60], linewidth=3, color='black')
sns.scatterplot(data=data_suicide, x='ProbDyingMale', y='ProbDyingFemale', s=100, ax=ax[
ax[0].set_title('Dying rates by gender')

ax[1].plot([0,100],[0,100], linewidth=3, color='black')
sns.scatterplot(data=data_suicide, x='SuicideMale', y='SuicideFemale', s=100, ax=ax[1])
ax[1].set_title('Suicide rates by gender')
```

Out[44]:

```
Text(0.5, 1.0, 'Suicide rates by gender')
```



In [46]:

```
data_suicide.sort_values(by='ProbDying_male_minus_female').head()
```

Out[46]:

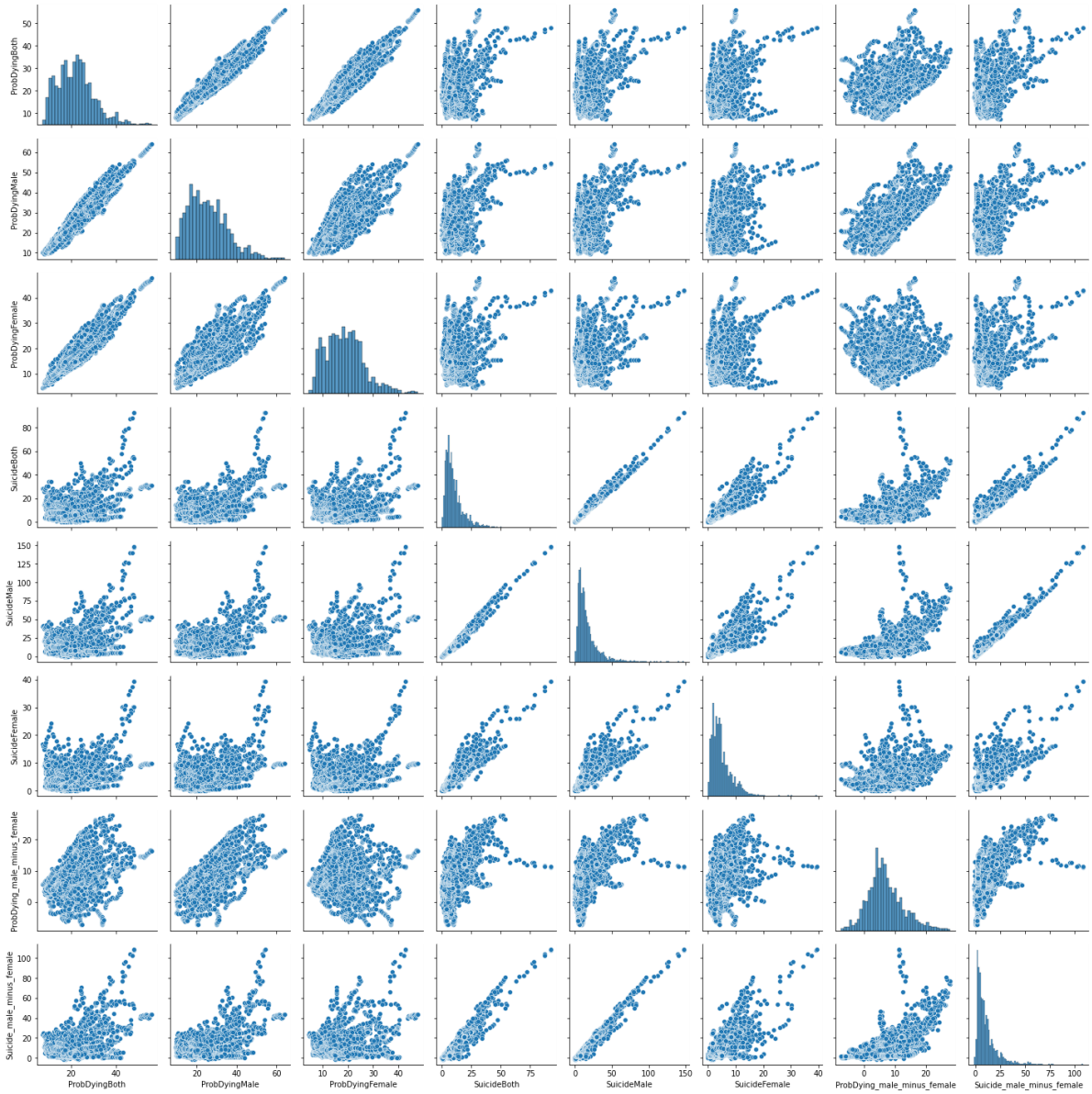
	Country	Year	ProbDyingBoth	ProbDyingMale	ProbDyingFemale	SuicideBoth	SuicideMale
2059	Mali	2000	25.0	20.8	28.2	4.8	5.5
2058	Mali	2001	24.6	20.5	27.8	4.7	5.5
1434	Haiti	2005	33.8	29.9	37.1	9.4	9.4
2057	Mali	2002	24.4	20.5	27.5	4.7	5.4
1435	Haiti	2004	33.7	29.9	36.9	9.0	9.2

In [48]:

```
sns.pairplot(data_suicide)
```

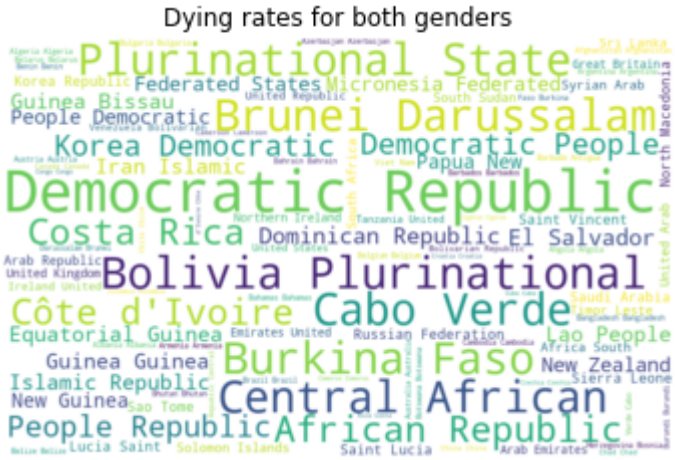
Out[48]:

```
<seaborn.axisgrid.PairGrid at 0x649297ec70>
```



In [50]:

```
from wordcloud import WordCloud
txt = ' '.join(text for text in data_suicide['Country'])
wordcloud = WordCloud(
            background_color = 'white',
            max_font_size = 100,
            max_words = 100,
            width = 800,
            height = 500
            ).generate(txt)
plt.imshow(wordcloud,interpolation = 'bilinear')
plt.axis('off')
plt.title('Dying rates for both genders')
plt.show()
```



In [54]:

```
top_deaths = data_suicide.sort_values('ProbDyingBoth', ascending=False).head(10)
top_deaths
```

Out[54]:

	Country	Year	ProbDyingBoth	ProbDyingMale	ProbDyingFemale	SuicideBoth	SuicideMale
1759	Kiribati	2000	56.0	64.1	47.8	31.2	53.1
1758	Kiribati	2001	55.7	63.9	47.4	31.1	53.1
1757	Kiribati	2002	55.4	63.6	47.1	30.9	52.6
1756	Kiribati	2003	54.9	63.0	46.8	29.4	49.8
1755	Kiribati	2004	54.6	62.6	46.7	29.3	49.6
1754	Kiribati	2005	54.5	62.4	46.5	30.1	51.2
1753	Kiribati	2006	54.0	61.8	46.3	29.4	49.7
1752	Kiribati	2007	53.9	61.6	46.2	30.9	52.7
1751	Kiribati	2008	53.7	61.4	46.2	31.5	53.8
1750	Kiribati	2009	53.4	60.9	46.0	30.9	52.8

In [55]:

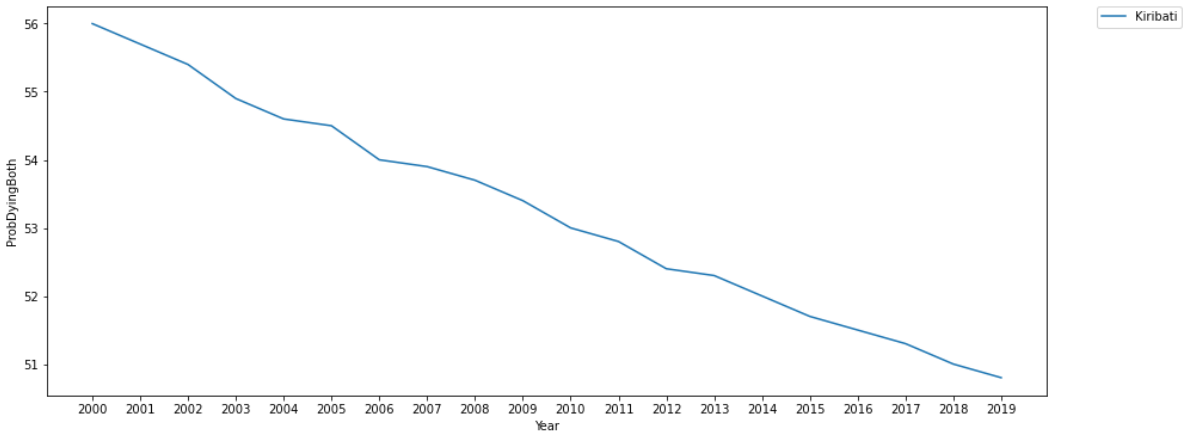
```
top_suicide = data_suicide.sort_values('SuicideBoth', ascending=False).head(10)
top_suicide
```

Out[55]:

	Country	Year	ProbDyingBoth	ProbDyingMale	ProbDyingFemale	SuicideBoth	SuicideMale
1865	Lesotho	2014	48.1	54.3	42.9	92.6	147.8
1864	Lesotho	2015	47.9	54.0	42.8	92.4	147.3
1866	Lesotho	2013	47.3	53.4	42.1	87.6	139.9
1863	Lesotho	2016	46.3	52.7	41.1	87.0	139.6
1867	Lesotho	2012	45.7	51.9	40.4	79.1	125.9
1862	Lesotho	2017	44.2	51.0	38.5	78.3	127.2
1861	Lesotho	2018	43.4	50.4	37.8	76.6	124.6
1860	Lesotho	2019	42.7	49.4	37.5	72.4	116.0
1868	Lesotho	2011	43.9	50.6	38.1	69.5	111.8
1869	Lesotho	2010	43.2	50.4	36.8	65.4	107.2

In [56]:

```
data_top_10_death = data_suicide[data_suicide.Country.isin(top_deaths['Country'])].iloc[
plt.figure(figsize=(15,6))
sns.lineplot(data=data_top_10_death, x='Year', y='ProbDyingBoth', hue='Country')
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
plt.show()
```



In [57]:

```
data_top_10_suicide = data_suicide[data_suicide.Country.isin(top_suicide['Country'])].i
plt.figure(figsize=(15,6))
sns.lineplot(data=data_top_10_suicide, x='Year', y='SuicideBoth', hue='Country')
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
plt.show()
```