

# cost-of-living-model

September 21, 2023

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

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

```
[2]: df = pd.read_csv('cost_of_living.csv')
```

```
[3]: df.head()
```

```
[3]:      Countries  Cost of living, 2017  Global rank Available data
0      Bermuda          225.86           1      2017 - 2017
1      Iceland          209.10           2      2017 - 2017
2  Switzerland          197.89           3      2017 - 2017
3       Norway          186.52           4      2017 - 2017
4      Denmark          171.78           5      2017 - 2017
```

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 167 entries, 0 to 166
Data columns (total 4 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Countries              167 non-null   object
1   Cost of living, 2017    167 non-null   float64
2   Global rank            167 non-null   int64
3   Available data         167 non-null   object
dtypes: float64(1), int64(1), object(2)
memory usage: 5.3+ KB
```

```
[5]: df = df.sort_values("Cost of living, 2017")
```

```
[6]: df
```

```
[6]:      Countries  Cost of living, 2017  Global rank Available data
166      Egypt          27.37           167      2017 - 2017
165     Ukraine          37.76           166      2017 - 2017
```

164	Sudan	38.29	165	2017 - 2017
163	Kyrgyzstan	40.38	164	2017 - 2017
162	Azerbaijan	40.56	163	2017 - 2017
..	...	...	...	...
4	Denmark	171.78	5	2017 - 2017
3	Norway	186.52	4	2017 - 2017
2	Switzerland	197.89	3	2017 - 2017
1	Iceland	209.10	2	2017 - 2017
0	Bermuda	225.86	1	2017 - 2017

[167 rows x 4 columns]

```
[7]: df['Available data'].unique()
```

```
[7]: array(['2017 - 2017'], dtype=object)
```

```
[8]: df = df.drop('Available data',axis=1) # no real use of this column
```

```
[9]: df = df.set_index("Countries")
```

```
[10]: df
```

```
[10]:
```

	Cost of living, 2017	Global rank
Countries		
Egypt	27.37	167
Ukraine	37.76	166
Sudan	38.29	165
Kyrgyzstan	40.38	164
Azerbaijan	40.56	163
...	...	...
Denmark	171.78	5
Norway	186.52	4
Switzerland	197.89	3
Iceland	209.10	2
Bermuda	225.86	1

[167 rows x 2 columns]

```
[11]: max_cost = df['Cost of living, 2017'].max()
```

```
[12]: df['cheaper_by'] = max_cost - df['Cost of living, 2017']
```

```
[13]: df
```

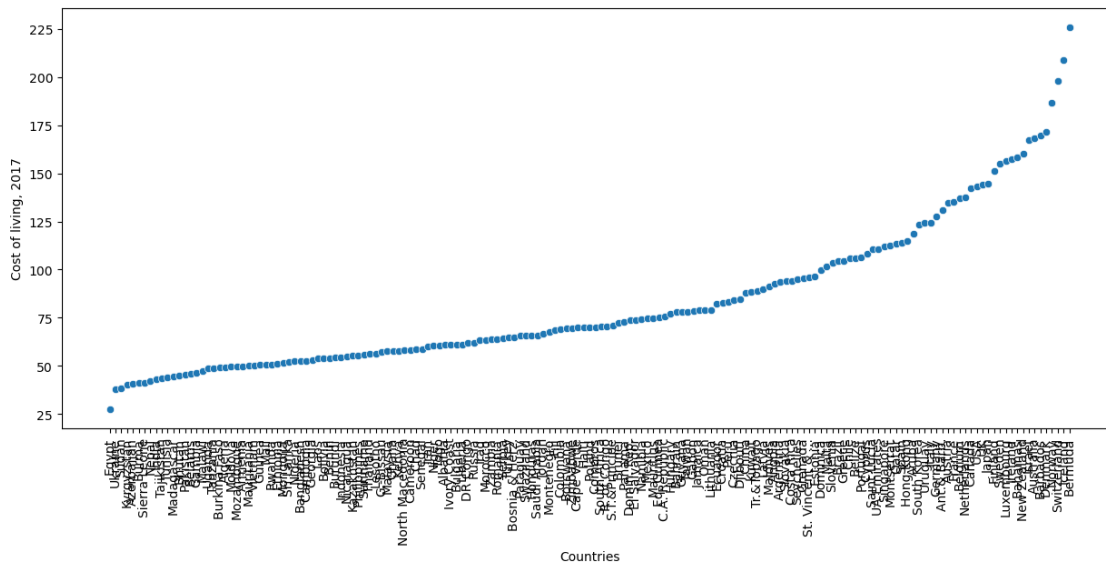
```
[13]:
```

	Cost of living, 2017	Global rank	cheaper_by
Countries			
Egypt	27.37	167	198.49

Ukraine	37.76	166	188.10
Sudan	38.29	165	187.57
Kyrgyzstan	40.38	164	185.48
Azerbaijan	40.56	163	185.30
...	...	...	...
Denmark	171.78	5	54.08
Norway	186.52	4	39.34
Switzerland	197.89	3	27.97
Iceland	209.10	2	16.76
Bermuda	225.86	1	0.00

[167 rows x 3 columns]

```
[14]: plt.figure(figsize=(15,6))
sns.scatterplot(df,y='Cost of living, 2017',x='Countries')
plt.xticks(rotation=90);
```



```
[15]: #cheapest countries
df.nsmallest(10,'Cost of living, 2017')
```

Countries	Cost of living, 2017	Global rank	cheaper_by
Egypt	27.37	167	198.49
Ukraine	37.76	166	188.10
Sudan	38.29	165	187.57
Kyrgyzstan	40.38	164	185.48
Azerbaijan	40.56	163	185.30
Burma	41.08	162	184.78

Sierra Leone	41.32	161	184.54
Nepal	41.85	160	184.01
India	42.86	159	183.00
Tajikistan	43.51	158	182.35

```
[16]: #costliest countries
df.nlargest(10, 'Cost of living, 2017')
```

```
[16]:
```

	Cost of living, 2017	Global rank	cheaper_by
Countries			
Bermuda	225.86	1	0.00
Iceland	209.10	2	16.76
Switzerland	197.89	3	27.97
Norway	186.52	4	39.34
Denmark	171.78	5	54.08
Barbados	169.90	6	55.96
Australia	168.02	7	57.84
Israel	167.52	8	58.34
New Zealand	160.18	9	65.68
Bahamas	158.09	10	67.77

```
[17]: df.index.unique()
```

```
[17]: Index(['Egypt', 'Ukraine', 'Sudan', 'Kyrgyzstan', 'Azerbaijan', 'Burma',
        'Sierra Leone', 'Nepal', 'India', 'Tajikistan',
        ...,
        'Bahamas', 'New Zealand', 'Israel', 'Australia', 'Barbados', 'Denmark',
        'Norway', 'Switzerland', 'Iceland', 'Bermuda'],
        dtype='object', name='Countries', length=167)
```

```
[18]: df.loc['India']
```

```
[18]: Cost of living, 2017    42.86
Global rank              159.00
cheaper_by              183.00
Name: India, dtype: float64
```

```
[19]: # turns out our beloved India was quite economical back in 2017
```

```
[20]: # CHEERS!!
```