

# Day 6

## 2015 Dataset

In [1]:

```
import numpy as np
import pandas as pd
```

In [2]:

```
d=pd.read_csv(r"c:\Users\user\Downloads\2015.csv")
d
```

Out[2]:

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)
0	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143
1	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784
2	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464
3	Norway	Western Europe	4	7.522	0.03880	1.45900	1.33095	0.88521
4	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563
...	...	...	...	...	...	...	...	...
153	Rwanda	Sub-Saharan Africa	154	3.465	0.03464	0.22208	0.77370	0.42864
154	Benin	Sub-Saharan Africa	155	3.340	0.03656	0.28665	0.35386	0.31910
155	Syria	Middle East and Northern Africa	156	3.006	0.05015	0.66320	0.47489	0.72193
156	Burundi	Sub-Saharan Africa	157	2.905	0.08658	0.01530	0.41587	0.22396
157	Togo	Sub-Saharan Africa	158	2.839	0.06727	0.20868	0.13995	0.28443

158 rows × 12 columns



# Mean,median,mode,describe

In [3]:

```
data=pd.DataFrame(d[['Happiness Score','Happiness Rank']][0:500])  
data
```

Out[3]:

	Happiness Score	Happiness Rank
0	7.587	1
1	7.561	2
2	7.527	3
3	7.522	4
4	7.427	5
...	...	...
153	3.465	154
154	3.340	155
155	3.006	156
156	2.905	157
157	2.839	158

158 rows × 2 columns

In [4]:

```
print(data.mean())
```

```
Happiness Score    5.375734  
Happiness Rank     79.493671  
dtype: float64
```

In [5]:

```
print(data.median())
```

```
Happiness Score    5.2325  
Happiness Rank     79.5000  
dtype: float64
```

In [6]:

```
data.fillna(value=1)
```

Out[6]:

	Happiness Score	Happiness Rank
0	7.587	1
1	7.561	2
2	7.527	3
3	7.522	4
4	7.427	5
...	...	...
153	3.465	154
154	3.340	155
155	3.006	156
156	2.905	157
157	2.839	158

158 rows × 2 columns

In [7]:

```
print(data.mode())
```

	Happiness Score	Happiness Rank
0	5.192	82

In [8]:

```
print(data.describe())
```

	Happiness Score	Happiness Rank
count	158.000000	158.000000
mean	5.375734	79.493671
std	1.145010	45.754363
min	2.839000	1.000000
25%	4.526000	40.250000
50%	5.232500	79.500000
75%	6.243750	118.750000
max	7.587000	158.000000

# Sum,cumsum,count,min,max

In [9]:

```
print(data.sum())
```

Happiness Score	849.366
Happiness Rank	12560.000
dtype: float64	

In [10]:

```
print(data.cumsum())
```

	Happiness Score	Happiness Rank
0	7.587	1
1	15.148	3
2	22.675	6
3	30.197	10
4	37.624	15
..	...	...
153	837.276	11934
154	840.616	12089
155	843.622	12245
156	846.527	12402
157	849.366	12560

[158 rows x 2 columns]

In [11]:

```
print(data.count())
```

```
Happiness Score    158
Happiness Rank     158
dtype: int64
```

In [12]:

```
print(data.min())
```

```
Happiness Score    2.839
Happiness Rank     1.000
dtype: float64
```

In [13]:

```
print(data.max())
```

```
Happiness Score    7.587
Happiness Rank    158.000
dtype: float64
```

## covariance and correlation (spearman and pearsons)

In [14]:

```
data1=data['Happiness Score'][0:10]  
data1
```

Out[14]:

```
0    7.587  
1    7.561  
2    7.527  
3    7.522  
4    7.427  
5    7.406  
6    7.378  
7    7.364  
8    7.286  
9    7.284
```

Name: Happiness Score, dtype: float64

In [15]:

```
data2=data['Happiness Rank'][0:10]  
data2
```

Out[15]:

```
0     1  
1     2  
2     3  
3     4  
4     5  
5     6  
6     7  
7     8  
8     9  
9    10
```

Name: Happiness Rank, dtype: int64

In [16]:

```
from numpy import cov  
print(cov(data1,data2))
```

```
[[ 0.01213373 -0.32888889]  
 [-0.32888889  9.16666667]]
```

In [17]:

```
from scipy.stats import pearsonr  
print(pearsonr(data1,data2))
```

```
(-0.9861574478166932, 1.57982971907728e-07)
```

In [18]:

```
from scipy.stats import spearmanr  
print(spearmanr(data1,data2))
```

```
SpearmanrResult(correlation=-0.9999999999999999, pvalue=6.646897422032013e  
-64)
```

In [ ]: