

# 8-Life\_Expectancy\_Linear\_Coefficients

January 29, 2023

## 1 Life\_Expectancy\_WHO\_UN\_Analysis\_Modeling

### 1.1 Linear Model Coefficient Investigation

To: [Magnimind](#)

From: Matt Curcio, matt.curcio.ri@gmail.com

Date: 2023-01-29

Re: NOTEBOOK #8

---

```
[1]: # Common Python Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns

# import warnings
import warnings
warnings.filterwarnings("ignore")

# Loading Modeling Libraries
from sklearn.linear_model import LinearRegression
from sklearn import linear_model
from sklearn.model_selection import train_test_split, cross_val_score
```

```
[2]: !ls *.csv
```

```
Clean_LE_Data_FEng_4.csv      Life_Expectancy_Data.csv  y_test.csv
Clean_LE_Data_Post_EDA_3.csv  x_test.csv               y_train.csv
Clean_LE_Data_w_Means_2.csv   x_train.csv
```

```
[3]: df = pd.read_csv('Clean_LE_Data_FEng_4.csv', header=0) # Drop Index Column
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2928 entries, 0 to 2927
```

Data columns (total 18 columns):

#	Column	Non-Null Count	Dtype
0	Country	2928 non-null	object
1	Year	2928 non-null	int64
2	Status	2928 non-null	int64
3	LifeExpectancy	2928 non-null	float64
4	AdultMort	2928 non-null	float64
5	EtOH	2928 non-null	float64
6	PercExpen	2928 non-null	float64
7	Measles	2928 non-null	int64
8	BMI	2928 non-null	float64
9	lt5yD	2928 non-null	int64
10	Polio	2928 non-null	float64
11	TotalExpen	2928 non-null	float64
12	DTP	2928 non-null	float64
13	HIV	2928 non-null	float64
14	Thin1_19y	2928 non-null	float64
15	Income	2928 non-null	float64
16	Education	2928 non-null	float64
17	Region	2928 non-null	int64

dtypes: float64(12), int64(5), object(1)

memory usage: 411.9+ KB

```
[4]: # Data splits

y = df['LifeExpectancy']

x = df[['Income', 'Education', 'Polio', 'DTP', 'AdultMort', 'HIV']]

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.3,
↪random_state = 100)
```

## 1.2 Linear Regression Parameter Investigation

```
[5]: reg_model = linear_model.LinearRegression()

reg_model = LinearRegression().fit(x_train, y_train)

#Printing the model coefficients
print('Intercept: ', reg_model.intercept_)

# pair the feature names with the coefficients
list(zip(x, reg_model.coef_))
```

Intercept: 50.66478178594173

```
[5]: [('Income', 9.113970547007359),  
      ('Education', 0.8860513547432141),  
      ('Polio', 0.037639530276743494),  
      ('DTP', 0.04333747425602766),  
      ('AdultMort', -0.02244631132206054),  
      ('HIV', -0.47927856071777236)]
```

$Life\ Expectancy = 50.7 + 9.1 \cdot Income + 0.90 \cdot Education + 0.04 \cdot Polio + 0.04 \cdot DTP - 0.02 \cdot Adult\ Mortality - 0.48 \cdot HIV$

```
[ ]:
```