MULTIPLE LINEAR REGRESSION

1) Regression Coefficients : β

Example Suicide rate is associated with divorce rate and population size. Find a multiple linear regression model on the following data.

In [19]:

```
import pandas as pd

dict1 = {'State' : ['Akron, OH', 'Anaheim, CA', 'Buffalo, NY', 'Austin, TX', 'Chicag
        'Population(1.000)' : [679,1420,1349,296,6975,323,4200,633],
         'Divorce Rate'  : [30.4,34.1,17.2,26.8,29.1,18.7,32.6,32.5],
         'Suicide Rate'  : [11.6,16.1,9.3,9.1,8.4,7.7,11.3,8.4]}

df1 = pd.DataFrame(dict1)
df1
```

Out[19]:

	State	Population(1.000)	Divorce Rate	Suicide Rate
0	Akron, OH	679	30.4	11.6
1	Anaheim, CA	1420	34.1	16.1
2	Buffalo, NY	1349	17.2	9.3
3	Austin, TX	296	26.8	9.1
4	Chicago, IL	6975	29.1	8.4
5	Columbia, SC	323	18.7	7.7
6	Detroit, MI	4200	32.6	11.3
7	Gary, IN	633	32.5	8.4

Model : $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$

In [20]:

```
from MultipleLinearRegression.MultipleLinReg import Multiple_Regression

coef_beta = Multiple_Regression(data=df1, x_cols_name=['Population(1.000)', 'Divorce
coef_beta.transpose() # beta_0  beta_1 beta_2
```

Out[20]:

```
array([[ 3.50735336e+00, -2.47709904e-04,  2.60946558e-01]])
```

Y = 3.5073 - 0.0002x_1 + 0.2609x_2

2) Sum of Squares of Residuals : SS_R

In [21]:

```
from MultipleLinearRegression.MultipleLinReg import Multiple_SSr

ssr = Multiple_SSr(data=df1, x_cols_name=['Population(1.000)', 'Divorce Rate'],y_col
ssr
```

Out[21]:

```
34.12123329568158
```

3) Prediction of σ^2

In [22]:

```
from MultipleLinearRegression.MultipleLinReg import Variance_Estimator

variance = Variance_Estimator(data=df1, x_cols_name=['Population(1.000)', 'Divorce R
variance
```

Out[22]:

```
6.824246659136316
```

4) Hypothesis testing for $\beta_i = k$

Example Depending on the age of a tree, the diameter of the tree depends on the amount of precipitation where it is located, the height of the place where it is located and its specific weight. Accordingly, test the $\beta_2 = 0$ hypothesis that the height does not affect the diameter of the tree.

In [23]:

```
dict2 = {'Age' : [44,33,33,32,34,31,33,30,34,34,33,36,33,34,37],
         'Height(1.000 ft)' : [1.3,2.2,2.2,2.6,2,1.8,2.2,3.6,1.6,1.5,2.2,1.7,2.2,1.3
         'Rainfall(inch)' : [250,115,75,85,100,75,85,75,225,250,255,175,75,85,90],
         'Weight' : [0.63,0.59,0.56,0.55,0.54,0.59,0.56,0.46,0.63,0.6,0.63,0.58,0.55
         'Diameter(inch)' : [18.1,19.6,16.6,16.4,16.9,17,20,16.6,16.2,18.5,18.7,19.4

df2 = pd.DataFrame(dict2)
df2
```

Out[23]:

	Age	Height(1.000 ft)	Rainfall(inch)	Weight	Diameter(inch)
0	44	1.3	250	0.63	18.1
1	33	2.2	115	0.59	19.6
2	33	2.2	75	0.56	16.6
3	32	2.6	85	0.55	16.4
4	34	2.0	100	0.54	16.9
5	31	1.8	75	0.59	17.0
6	33	2.2	85	0.56	20.0

	Age	Height(1.000 ft)	Rainfall(inch)	Weight	Diameter(inch)
7	30	3.6	75	0.46	16.6
8	34	1.6	225	0.63	16.2
9	34	1.5	250	0.60	18.5
10	33	2.2	255	0.63	18.7
11	36	1.7	175	0.58	19.4
12	33	2.2	75	0.55	17.6
13	34	1.3	85	0.57	18.3
14	37	2.6	90	0.62	18.8

In [24]:

```
from MultipleLinearRegression.MultipleLinReg import Coef_Hypot

coef_hypot = Coef_Hypot(data=df2,alpha=0.05,beta=0,variable_name='Height(1.000 ft)',

coef_hypot
```

```
test-stat=0.08694500407991285 t-table=-2.2281388519649385
## H_0 hypothesis can be accepted ##
```

Out[24]:

```
(0.08694500407991285, -2.2281388519649385)
```

5) Multiple Determination Coefficient R^2

In [27]:

```
from MultipleLinearRegression.MultipleLinReg import Multiple_R2

R2 = Multiple_R2(data=df2,x_cols_name=['Age','Height(1.000 ft)','Rainfall(inch)','We
R2
```

Out[27]:

```
0.11998618786964044
```

6) Predicting Future Responses

Example A steel company is planning to manufacture a reduced cold steel plate containing 15% copper at an annealing temperature of 1.15 degrees Fahrenheit and is interested in estimating the average strength of the plate. Below are data from 10 different steel plate samples with different copper content and different annealing temperatures. Estimate the mean stiffness and determine the confidence interval for the mean with 95% confidence.

In [29]:

```
dict3 = {'Durability': [79.2, 64, 55.7, 56.3, 58.6, 84.3, 70.4, 61.3, 51.3, 49.8],
         'Plate': [0.02, 0.03, 0.03, 0.04, 0.1, 0.15, 0.15, 0.09, 0.13, 0.09],
         'Heat(1.000 Fah)': [1.05, 1.2, 1.25, 1.3, 1.3, 1, 1.1, 1.2, 1.4, 1.4]}
```

```
df3 = pd.DataFrame(dict3)
df3
```

Out[29]:

	Durability	Plate	Heat(1.000 Fah)
0	79.2	0.02	1.05
1	64.0	0.03	1.20
2	55.7	0.03	1.25
3	56.3	0.04	1.30
4	58.6	0.10	1.30
5	84.3	0.15	1.00
6	70.4	0.15	1.10
7	61.3	0.09	1.20
8	51.3	0.13	1.40
9	49.8	0.09	1.40

In [31]:

```
from MultipleLinearRegression.IntervalPrediction import Interval

interval = Interval(data=df3,alpha=0.05,x_cols_name=['Heat(1.000 Fah)','Plate'],y_cc
interval
```

Out[31]:

```
(63.28197113160484, 71.4466956885688)
```

$$E[Y|x] = \sum_{i=0}^{k} x_i\beta_i \in (63.282, 71.447)$$

In [32]:

```
parameter = Multiple_Regression(data=df3, x_cols_name=['Heat(1.000 Fah)','Plate'], y
x_val = [1, 1.15, 0.15]

b = parameter.transpose()

Y=0

for i in range(len(x_val)):
    Y += b[0,i]*x_val[i]
Y
```

Out[32]:

```
69.86226110983333
```

According to the given x variables, the model output value is 69.862 and the model output value is within the 95% confidence interval produced by the interval estimator.