Pattern Recognition & Machine Learning

Lab-6 :- *“Linear Regression”*

Objectives

Here were the following aims and objectives , regarding Lab-6 :-

1. Load the Dataset and understand its characteristics through Distribution plots, Correlation Matrices & Categorical assessment.

2. Build a Linear regression model using Normal Equation as well as Sklearn’s implementation. Compare their Performance through Mean Squared Errors and Parameters.

3. Plot the Graphs between predicted and actual values, in order to determine the relationship between dependent and independent variables, for both the models.

Datasets

1. Medical Insurance Dataset :- dataset
   Consist of 6 features, with a target variable, as following :-

   1. Age ----> Discrete Valued

   2. Sex ----> Categorical

   3. BMI ----> Continuous Valued

   4. Children ----> Discrete Valued

   5. Smoker ----> Categorical

   6. Region ----> Categorical

   Charges ----> Target Variable

Dependencies

Here were the following dependencies for fulfilling Lab-6 :-

1. Numpy
2. Pandas
3. Sklearn
4. Scipy
5. Seaborn
6. Matplotlib

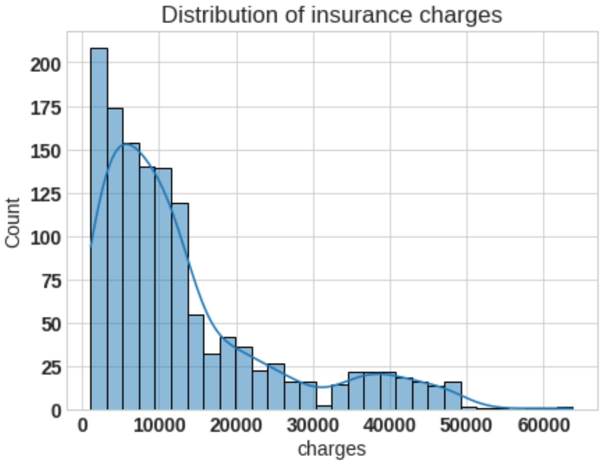
Preprocessing Methods

The Preprocessing Techniques applied, were as following :-

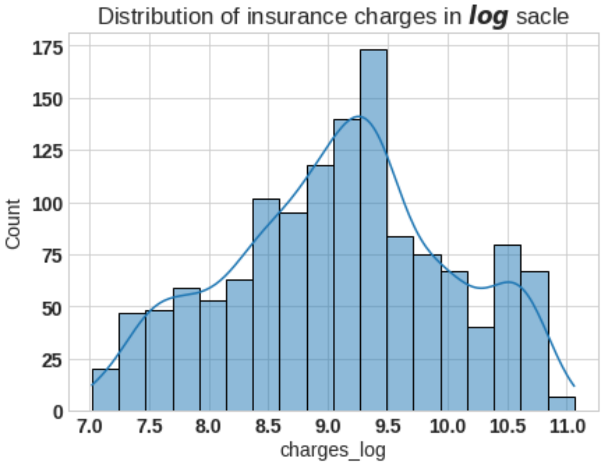
1. For Categorical Features like Smoker,Region and Sex ; Label Encoder was used to convert them into Discrete Values. This Time, it was more preferred than Ordinal Encoder because the former makes discrete values for 1st to (n-1)th classes, whereas later focuses on making discrete values according to Alphabetical order and can extend this procedure non-finitely.

2. For ease of handling the target values, Target Variable (Charges) were log scaled with natural base. The changes can be visualised as following :-

Original Plot was right skewed



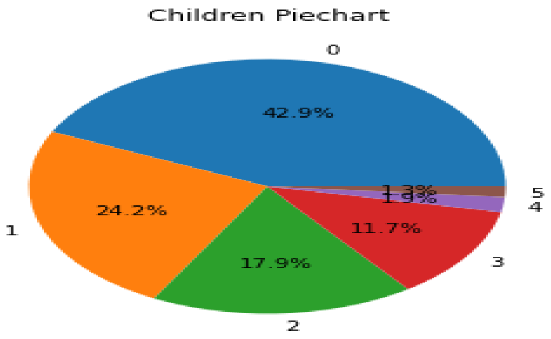
Newer Plot seemed to be more central

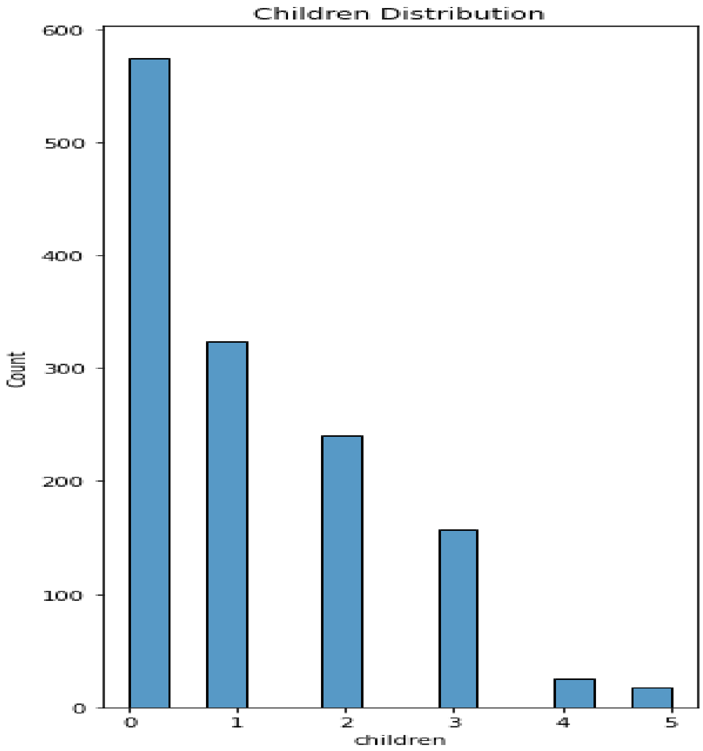


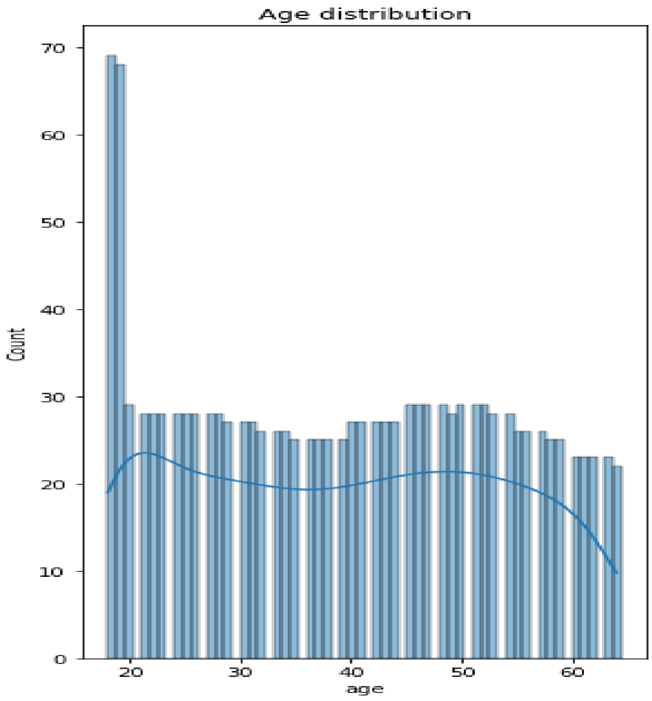
3. The Dataset was shuffled and split into Train and Test Data Portions, in the ratio 7:3, using Sklearn's Train_Test_Split technique. Later on, in one of the implementations, x0=1 was concatenated.

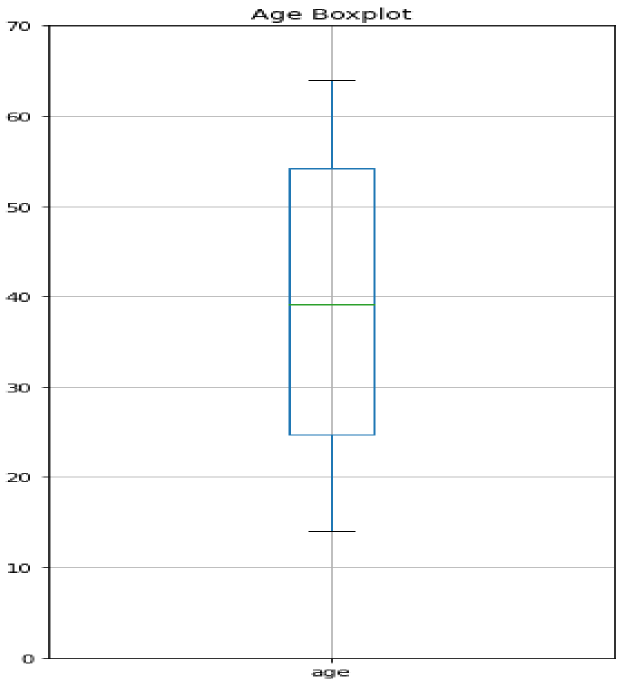
Feature Distribution Plots

1. For Discrete Features like “Children” and “Age” :-

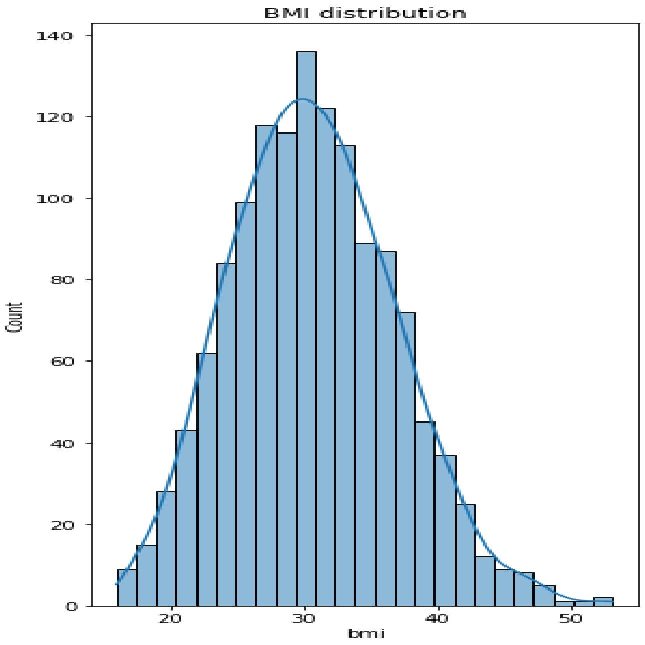


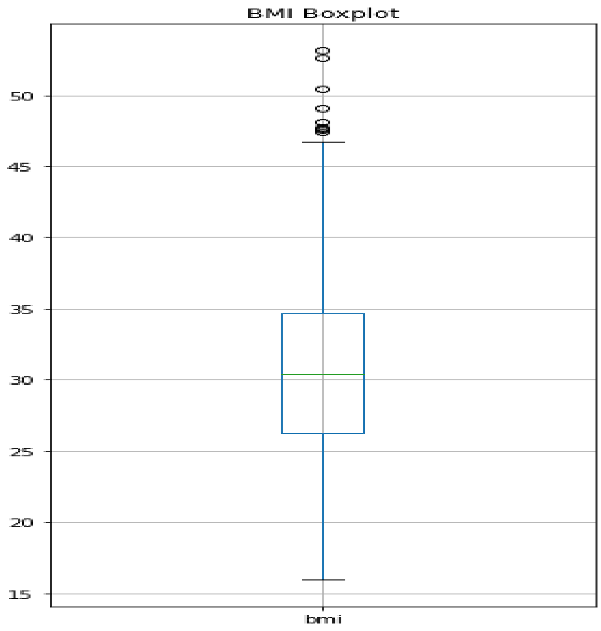




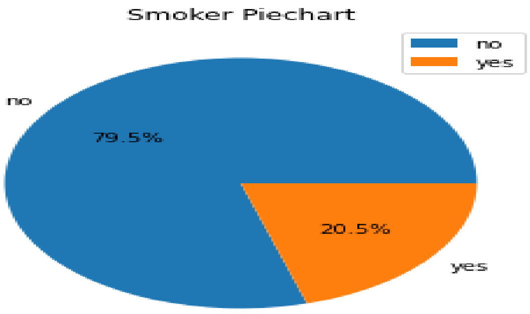


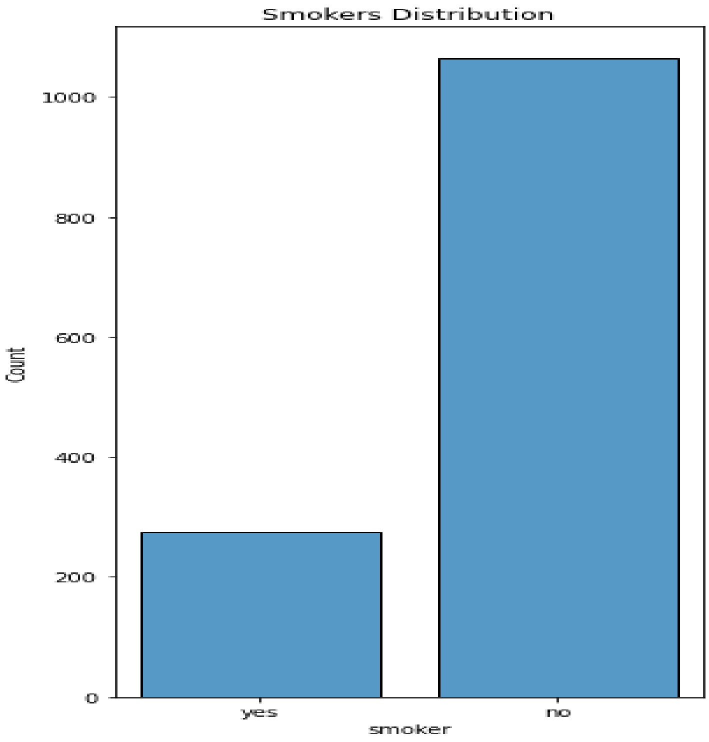
2. For Continuous Feature like “BMI” :-

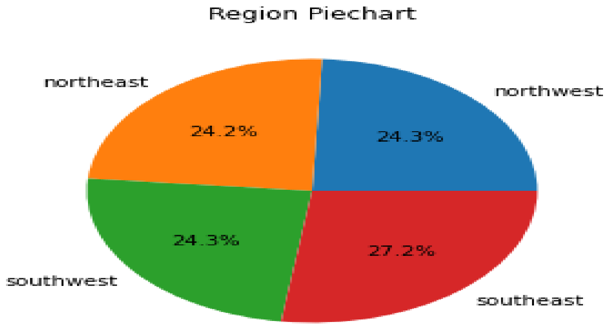


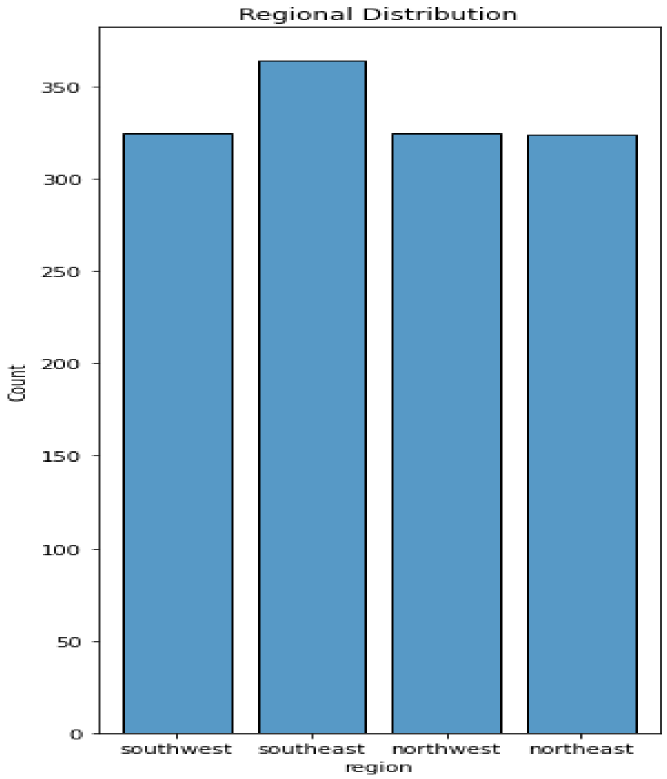


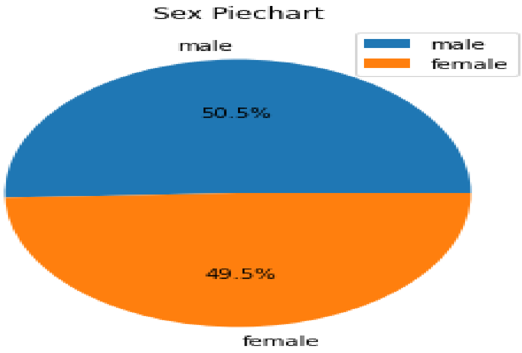
3. For Categorical Features like “Smoker”,”Region” and “Sex :-

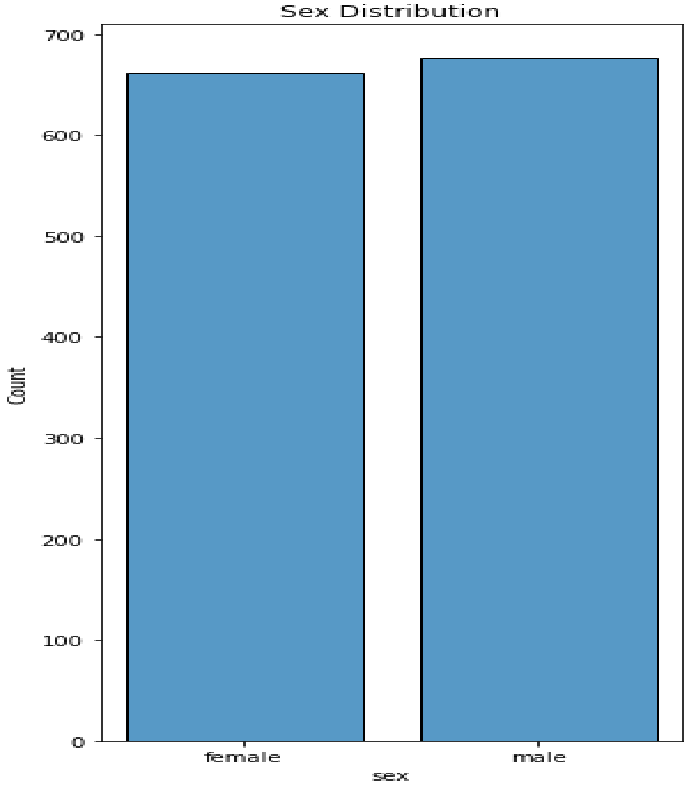





Region Piechart
northeast
24.2%
northwest
24.3%
24.3%
27.2%
southwest
southeast

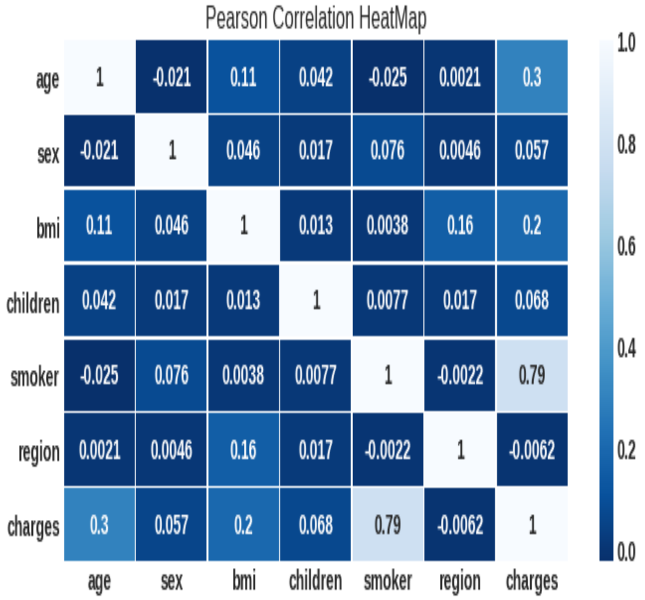

Regional Distribution
Count
350
300
250
200
150
100
50
0
southwest
southeast
northwest
northeast
region

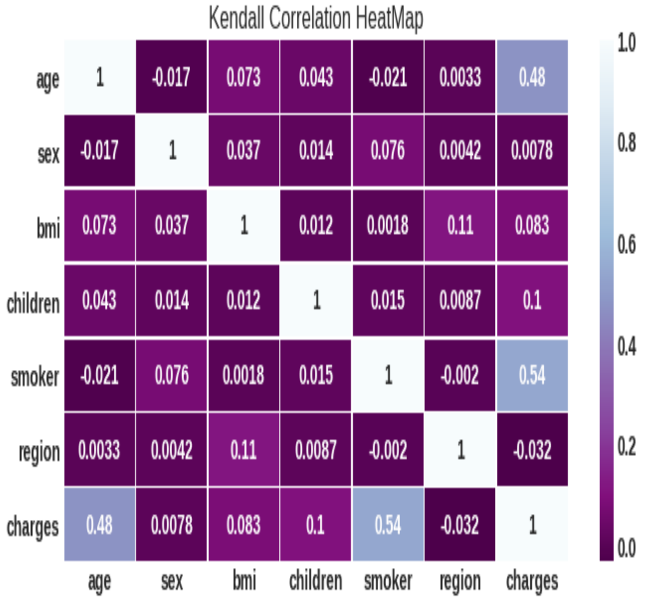

Sex Piechart
male
male
female
50.5%
49.5%
female

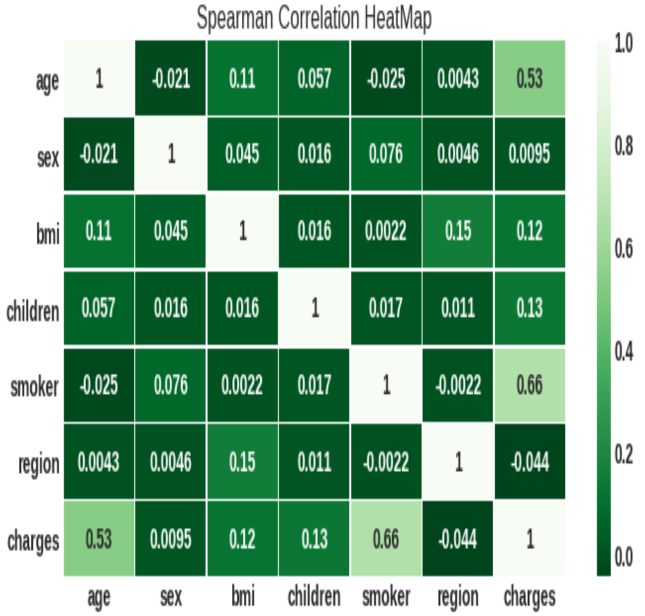

Sex Distribution
Count
700
600
500
400
300
200
100
0
female
male
sex

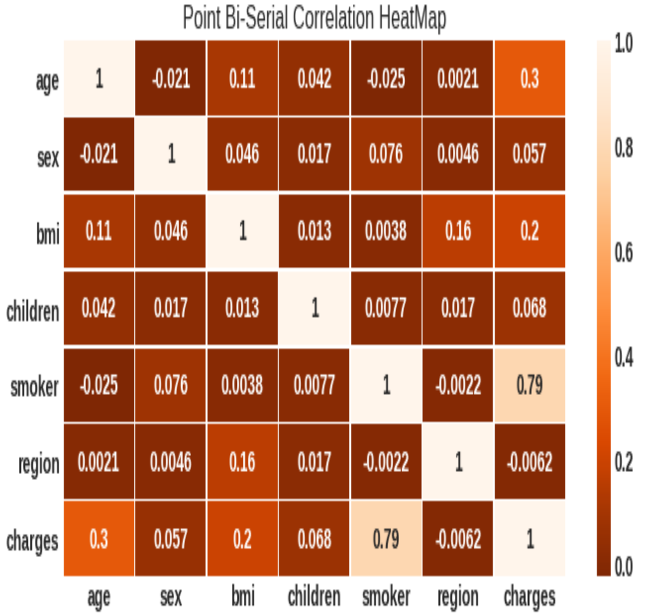
Types of Correlations

For 6 Features and a Target Variable, Correlation Matrices were generated , using 4 Different Techniques as following :-









Linear Regression and Gradient Descent

For 6 Features and a Target Variable, the hypothesis function was defined, as following :-

$$\mathbf{h_\theta(x_i) = \theta_0 + \theta_1 age + \theta_2 sex + \theta_3 bmi + \theta_4 children + \theta_5 smoker + \theta_6 region}$$

Where :-

$\boldsymbol{\theta_i}$ --> Weight / Parameter for the feature

h --> Hypothesis Function / Prediction for the input X

X --> Vectorized Featured Data-Points

The Equation for Gradient Descent, went as following :-

$$\boldsymbol{\theta_i} = \boldsymbol{\theta_i} - \lambda * (d(\mathbf{h_\theta}) / d(\boldsymbol{\theta_i}))$$

Where :-

λ --> Learning Rate / Jump Factor for Loss v/s Weight Curve

For the Scenario of Normal Equation, Gradient Descent is not in requirement as such, as following :-

$$\boldsymbol{\theta} = (x, T \ * \ x)^{-1} * (x.T * y)$$

Where :-

x --> Vectorized Featured Data-Points

x.T --> Transpose of x

$(x, T \ * \ x)^{-1}$ --> Inverse of $(x, T \ * \ x)$ vector

y --> Target Variable Data Points in vector form

Comparison b/w both Models

Parameters,that came from Normal Equation with/without taking Natural Log,were as following:-

Features	Weights from Normal Equation (with natural log)	Weight from Normal Equation (without natural log)
Age	0.034487	274.288995
Sex	-0.083259	-272.997632
BMI	0.013198	361.857772
Children	0.096467	498.453002
Smoker	1.539780	23676.540913
Region	-0.047934	-340.505802
Constant	7.036637	-13250.525844

Parameters, that came from Sklearn’s Model Implementation with/without taking Natural Log , were as following :-

Features	Weights from Sklearn’s Model (with natural log)	Weight from Sklearn’s Model (without natural log)
Age	0.033610	254.124648
Sex	-0.079802	-248.319163
BMI	0.012516	357.540192
Children	0.112047	506.953512
Smoker	1.561585	23959.912013
Region	-0.047081	-512.007785
Children	7.080098	-12296.016885

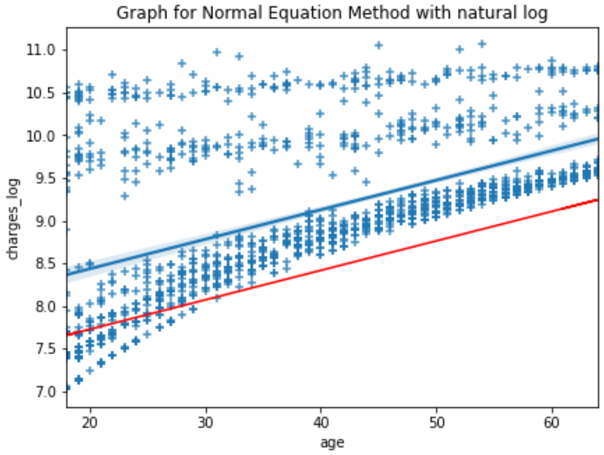
Mean Squared Loss , in both the cases were as following :-

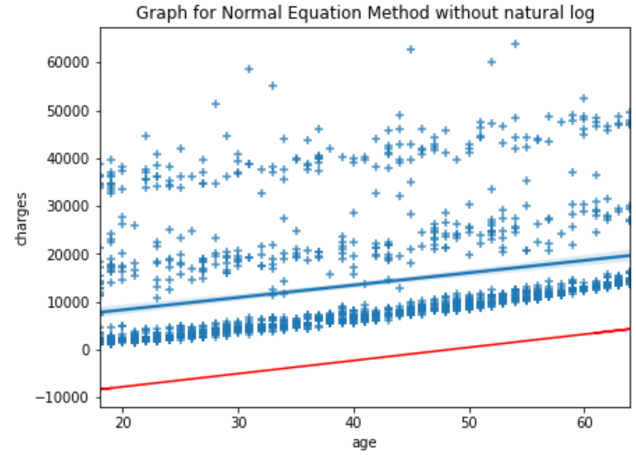
Model Type	MSE (with natural log)	MSE (without natural log)
Normal Equation	0.18830646855802602	33803464.35208089
Sklearn’s Model	0.1999741097872681	36497803.516599864

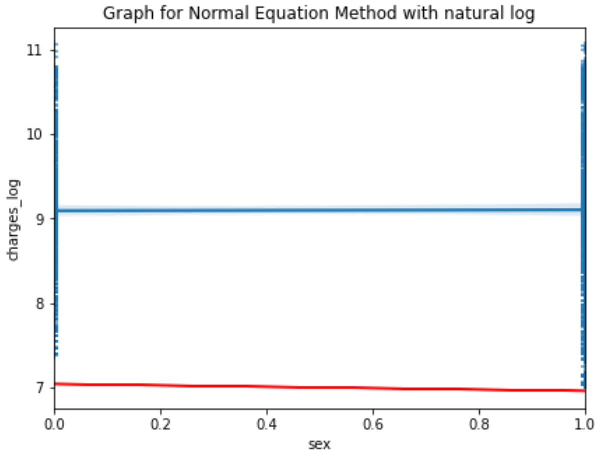
Feature-Wise Plots

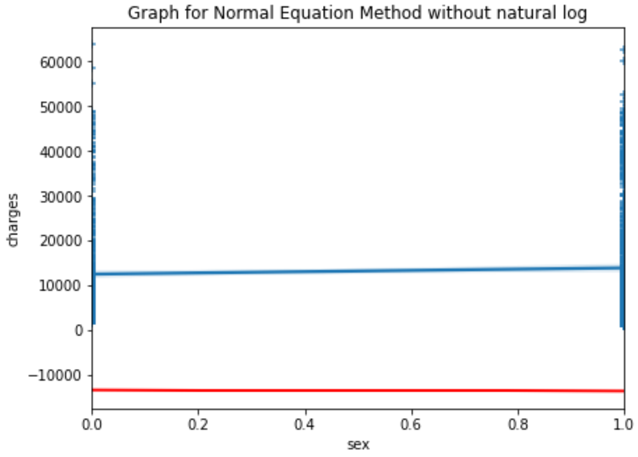
The Following Graphs (Red Line - Hypothesis Function , Blue Line - Feature-Wise Regression Line), were feature-wise plots for Normal Equation as well as Sklearn's Linear Regression Models, with and without taking natural log :-

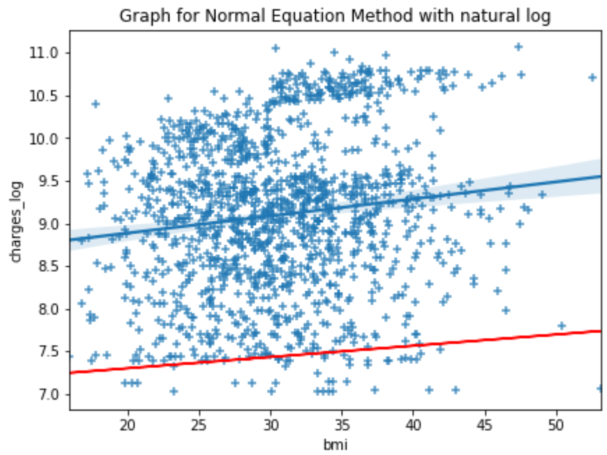
1. Normal Equation , with and without Natural Log

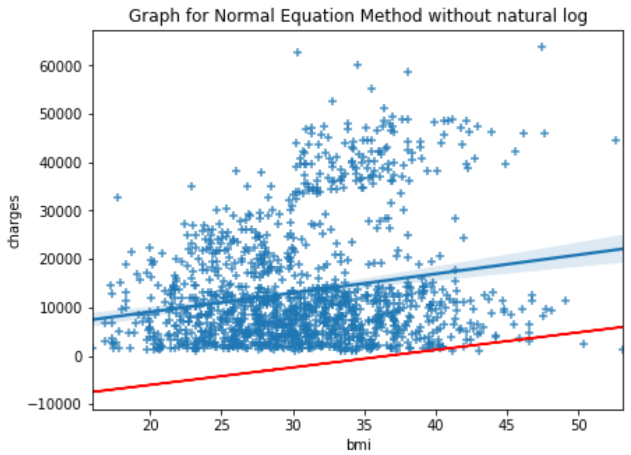

Graph for Normal Equation Method with natural log

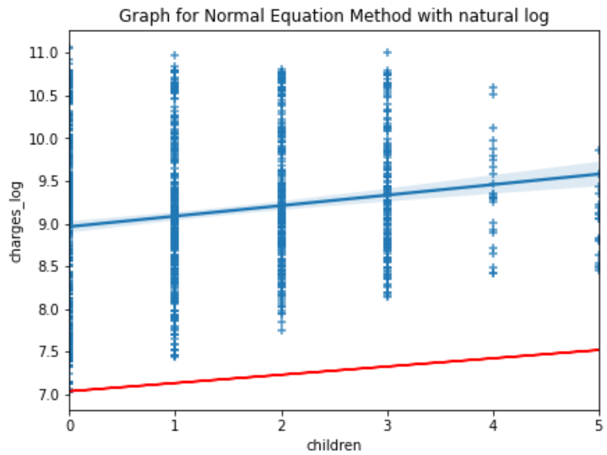

Graph for Normal Equation Method without natural log

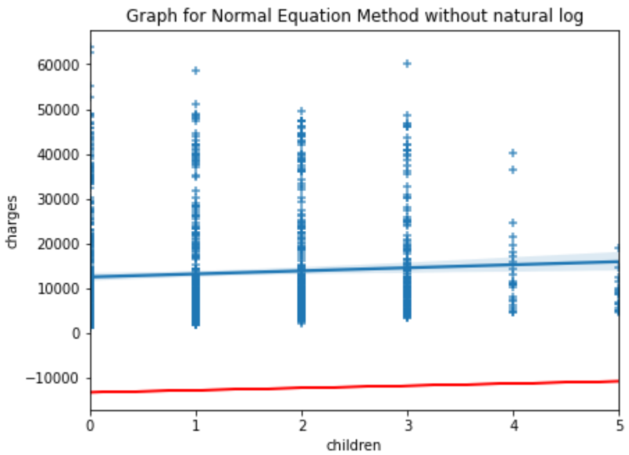

Graph for Normal Equation Method with natural log

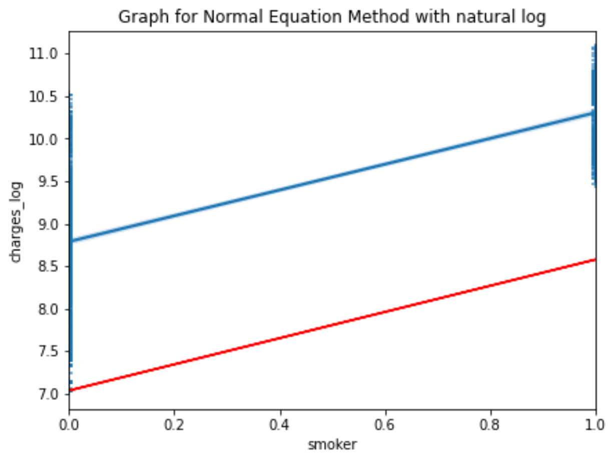

Graph for Normal Equation Method without natural log

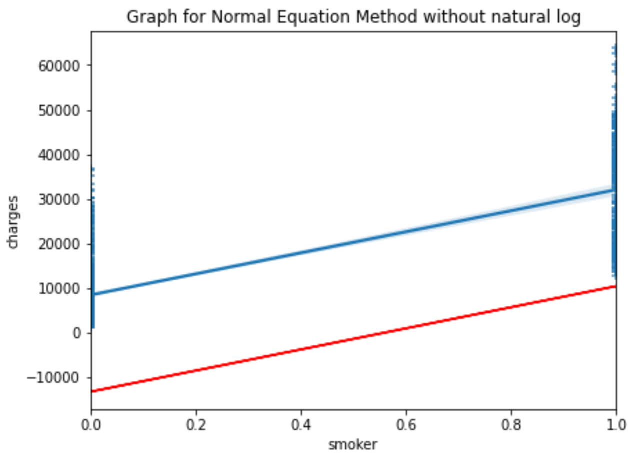

Graph for Normal Equation Method with natural log

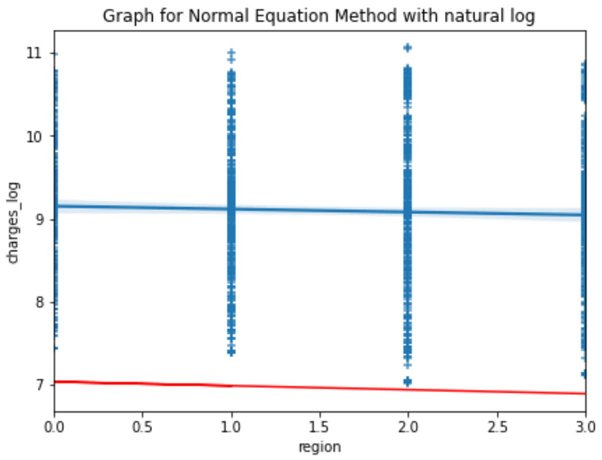

Graph for Normal Equation Method without natural log

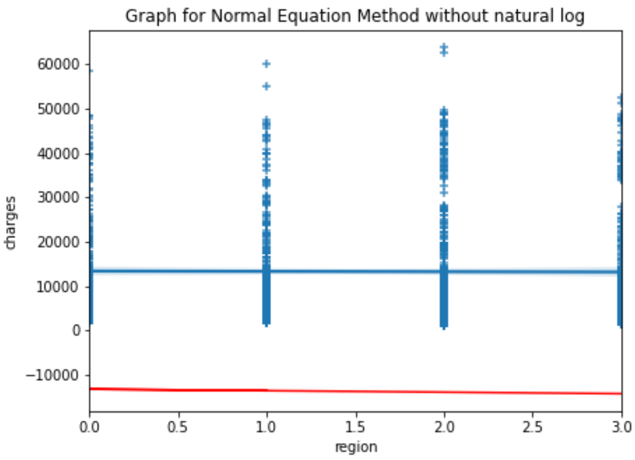

Graph for Normal Equation Method with natural log


Graph for Normal Equation Method without natural log

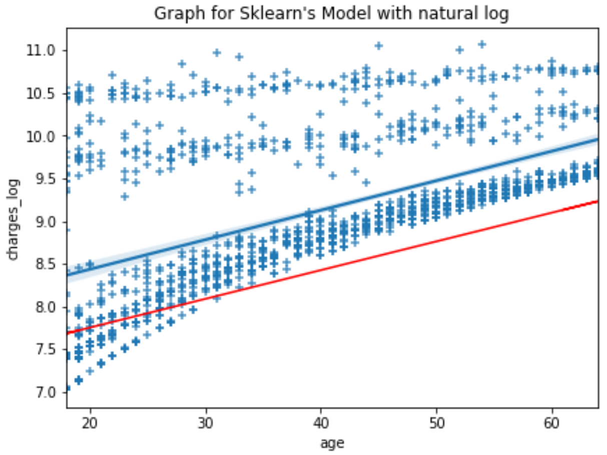

Graph for Normal Equation Method with natural log

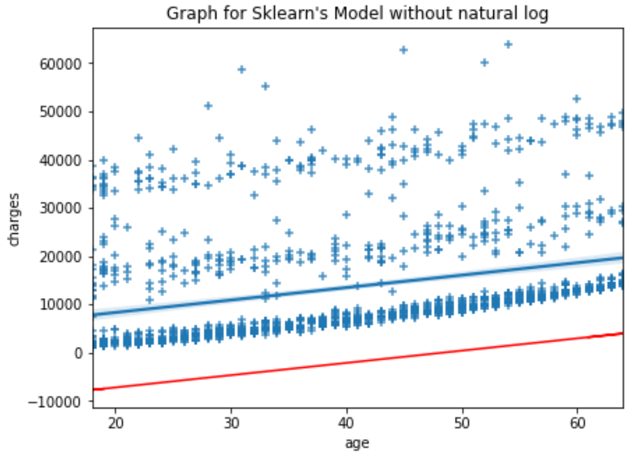

Graph for Normal Equation Method without natural log

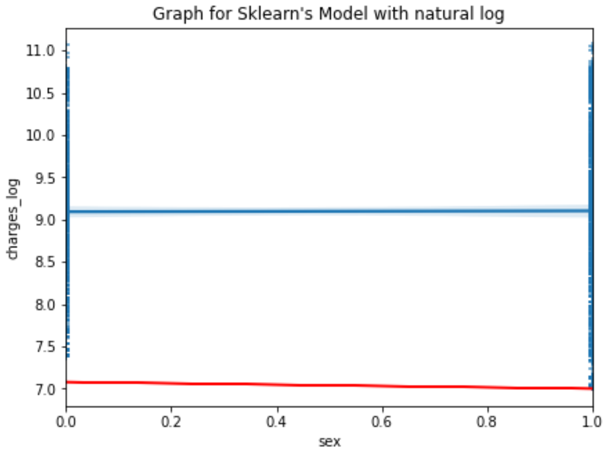


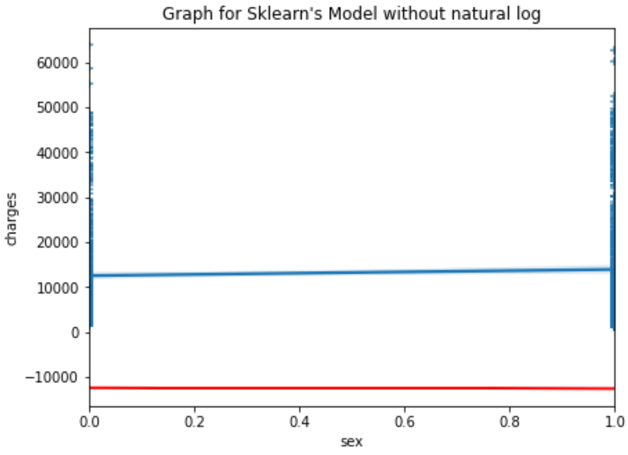


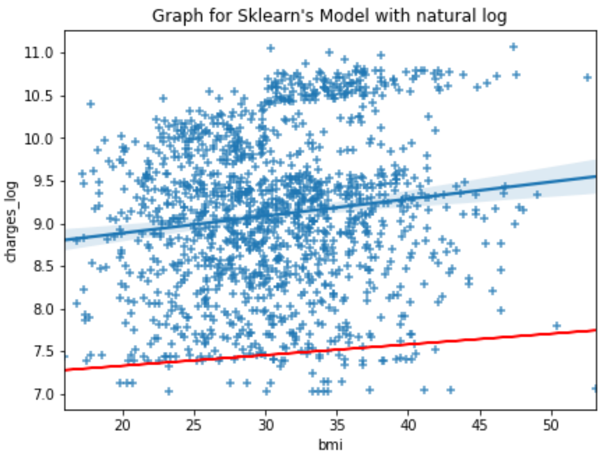
2. *Sklearn's Linear Regression Model , with and without Natural Log :-*

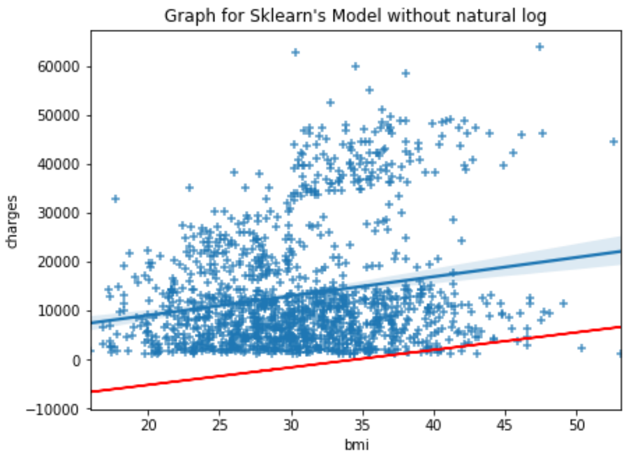


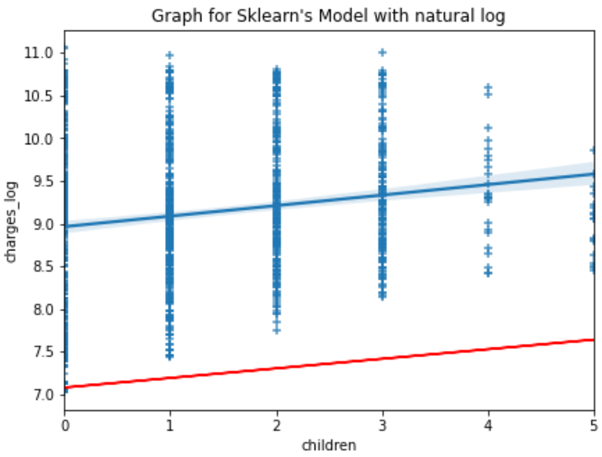


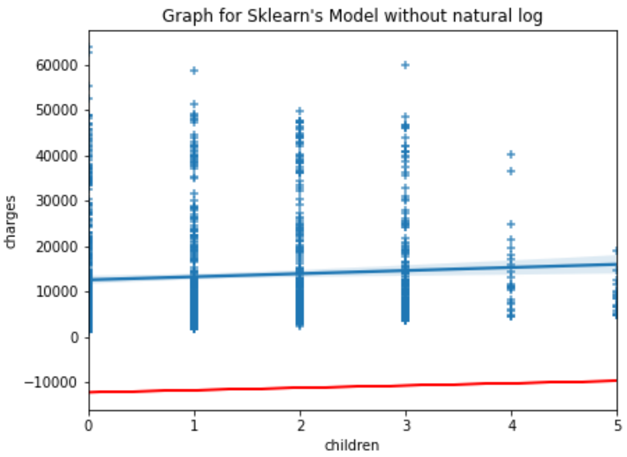


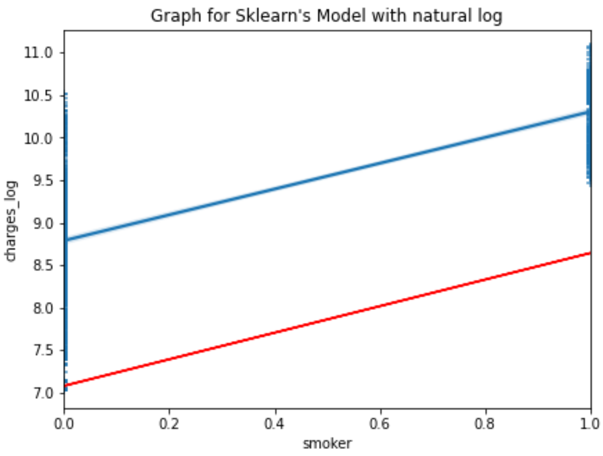


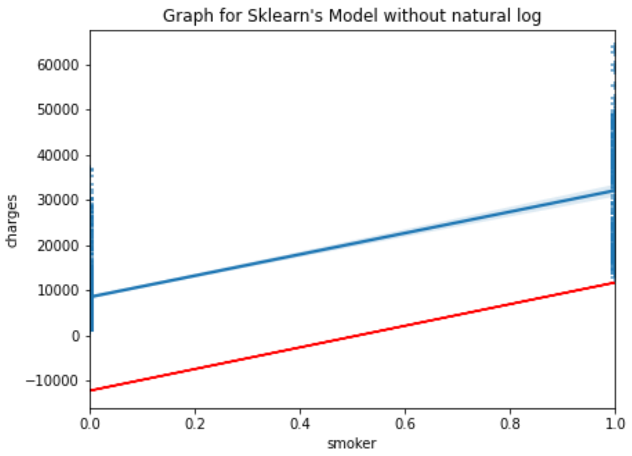

Graph for Sklearn's Model with natural log
charges_log
11.0
10.5
10.0
9.5
9.0
8.5
8.0
7.5
7.0
20
25
30
35
40
45
50
bmi

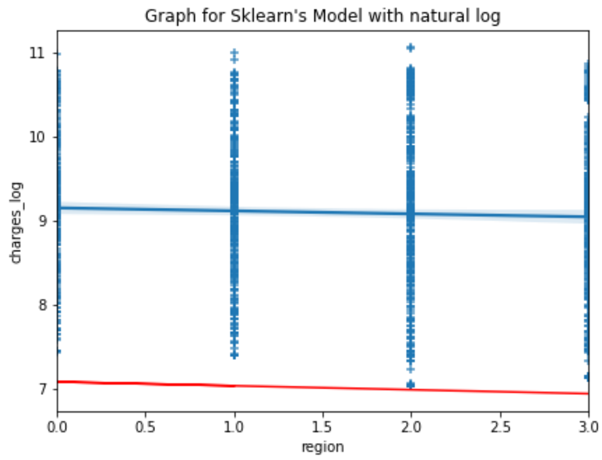

Graph for Sklearn's Model without natural log
charges
60000
50000
40000
30000
20000
10000
0
−10000
20
25
30
35
40
45
50
bmi

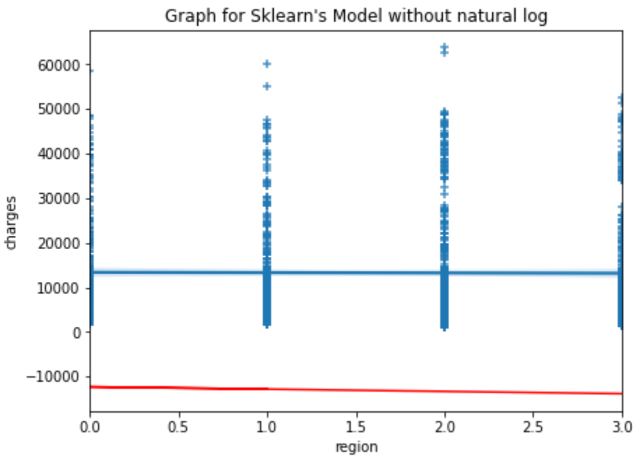

Graph for Sklearn's Model with natural log
charges_log
11.0
10.5
10.0
9.5
9.0
8.5
8.0
7.5
7.0
0
1
2
3
4
5
children


Graph for Sklearn's Model without natural log
charges
60000
50000
40000
30000
20000
10000
0
−10000
0
1
2
3
4
5
children


Graph for Sklearn's Model with natural log
charges_log
11.0
10.5
10.0
9.5
9.0
8.5
8.0
7.5
7.0
0.0
0.2
0.4
0.6
0.8
1.0
smoker


Graph for Sklearn's Model without natural log
charges
60000
50000
40000
30000
20000
10000
0
−10000
0.0
0.2
0.4
0.6
0.8
1.0
smoker





Conclusion

Here were the following observations, made from these implementations :-

1. When the Feature-Wise Regression Plot line was sketched alongside Hypothesis Function’s Plot Line, they appeared different. This happened because the Plot Line generated by Hypothesis Function, was only just the projection Curve for that particular Feature against Dependent Variable (Charges), whereas the Regression Plot Line focused only on that Particular Feature Data Points, not dependent on other Features’ Projections too.

2. The values of Mean Squared Error and Parameters , when natural log was taken, seemed to be way less than that without applying natural log (many-folds). It was because natural log changes the distribution of data and show the tendency to centralize the distribution, if the original distribution was left or right skewed. Thus, helping the Model to work with smaller values and reaching the minima effectively.

3. The effects of log scaling on the plots can be seen too, alongside the representation and distribution of data points, in a drastic fashion.