REGRESSION ANALYSIS PROJECT

Variable Model Selection

Submitted by – Group 8

Members	**Roll Number**
Arijit Ray	23N0078
Rajrana Barman	23N0077
Saikat Datta	23N0056
Subhajit Karmakar	23N0076

Supervisor: Prof. Siuli Mukhopadhyay

Department of Mathematics

INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

Contents

1 ABSTRACT

"Essentially, All models are wrong, but some are useful"

— George Box

The main objective of this project is to establish the statistical relationship between policyholder charges and several explanatory variables using Multiple Linear Regression. To accomplish this, we've selected a dataset from Kaggle, which we'll discuss in detail later on.

At first, we sort predictor variables into categories: categorical and non-categorical. Then, we split the dataset into two parts: 70% for training and 30% for testing. We focus our analysis on the training data. We study each numerical predictor variable against the response to see if there's any relationship between them. we have also used added variable plots to pick the variables for the multiple regression model.

We choose the best subset model based on measures like Adjusted R^2, AIC, and BIC. Then, we conduct outlier and residual analysis to verify the validity of the linear regression assumptions for all of three models. Additionally, we assess whether multicollinearity impacts our model by examining the Variance Inflation Factor(VIF). Afterward, we apply each model to the test dataset and choose the most accurate one based on their MSPR values and other considerations.

2 Description of the Dataset

2.1 Source

The data is openly available in multiple online sources. We have collected this specific dataset from Kaggle. The link of the dataset is provided below.

https://www.kaggle.com/datasets/simranjain17/insurance

2.2 About Variables

A brief description of the variables is given below.

Variables	Type	Description
age	Continuous	The Age of the policyholder
sex	Character	The Gender of the policyholder
bmi	Continuous	The Body Mass Index of the Policyholder
children	Integer	Number of Children of the Policyholder
smoker	Character	Indicates whether the Policyholder is Smoker or No Smoker
region	Character	The Region where the Policyholder belongs to
charges	Continuous	The Premium Charged to the Policyholder

Table 1: Description of the variables

3 Relationship among the variables

In this section, we will explore the association between the predictors and the response. We have taken **charges** as the response.

3.1 Univariate Analysis

1. **'region'**

	Region	Count	Proportion
0	southeast	364	27.23 %
1	southwest	325	24.31 %
2	northwest	324	24.23 %
3	northeast	324	24.23 %

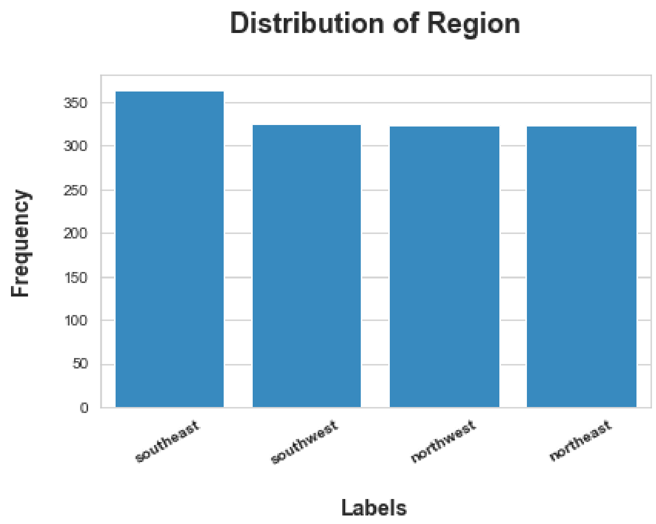


Figure 1: **Distribution of Region**

Comment: From the diagram, we observe that the proportion of policyholders is more or less the same for every region.

2. **'children'**

	Children	Count	Proportion
0	0	573	42.86 %
1	1	324	24.23 %
2	2	240	17.95 %
3	3	157	11.74 %
4	4	25	1.87 %
5	5	18	1.35 %

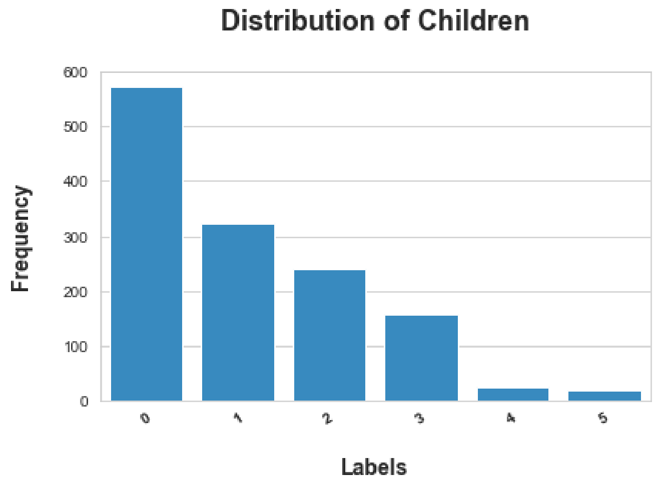


Figure 2: **Distribution of Children**

Comment: The distribution of children in the diagram is positively skewed, indicating that there are more families with fewer children, while fewer families have a higher number of children.

3. **'smoker'**

	Smoker	Count	Proportion
0	no	1063	79.51 %
1	yes	274	20.49 %

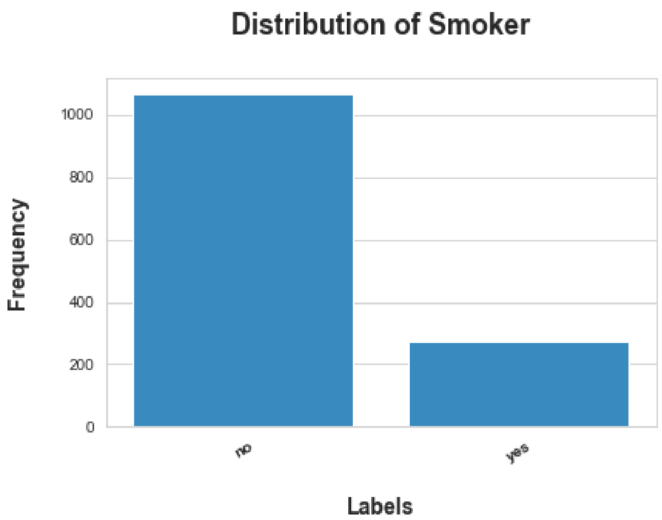


Figure 3: **Distribution of Smoker**

Comment: From the diagram, we observe that most of the policyholders are non-smokers.

4. **'sex'**

	Sex	Count	Proportion
0	male	675	50.49 %
1	female	662	49.51 %

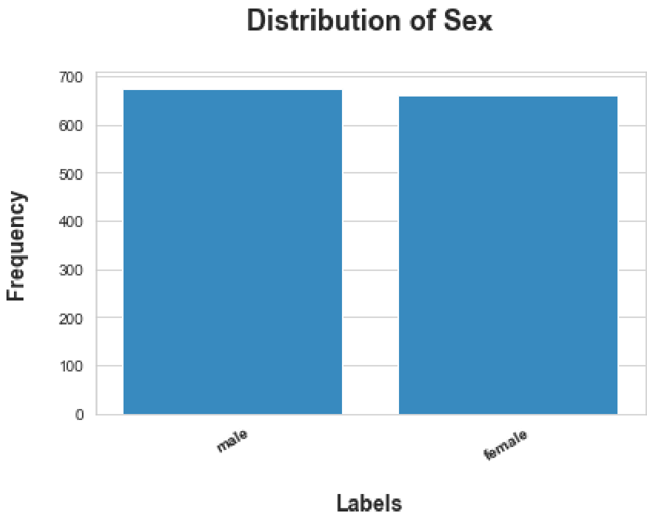
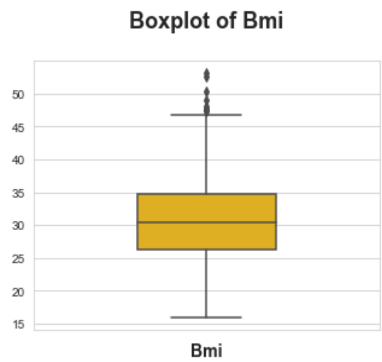


Figure 4: **Distribution of Sex**

Comment: From the diagram, we observe that the proportions of males and females are almost same in this case.

5. **'bmi'**

	Statistics	Values
0	mean	30.66
1	std	6.1
2	min	15.96
3	10%	22.99
4	25%	26.29
5	50%	30.4
6	75%	34.7
7	90%	38.63
8	max	53.13



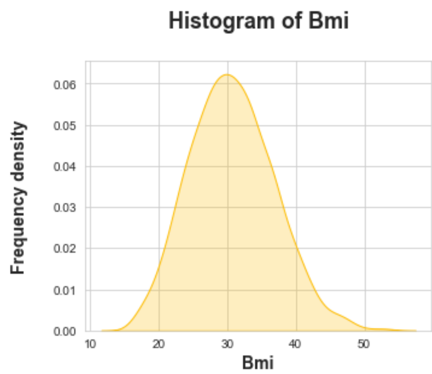


Figure 5: **Distribution of BMI**

Comment: From the boxplot, we see that the distribution is almost symmetrical and the histogram supports that. The presence of some outliers is also evident.

6. **'age'**

	Statistics	Values
0	mean	39.22
1	std	14.04
2	min	18.0
3	10%	19.0
4	25%	27.0
5	50%	39.0
6	75%	51.0
7	90%	59.0
8	max	64.0

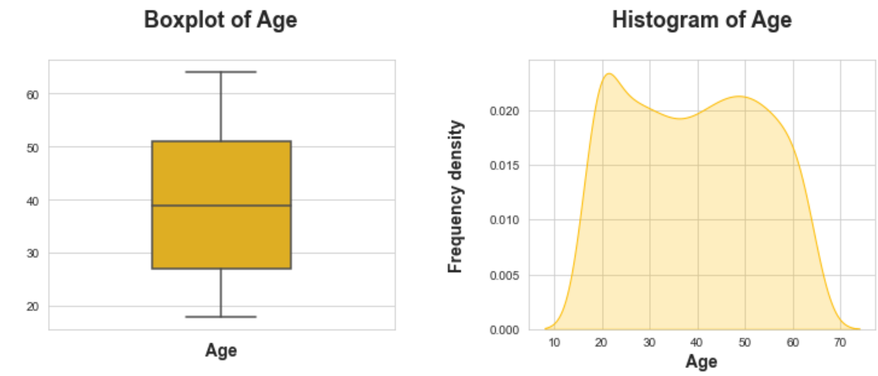


Figure 6: **Distribution of BMI**

Comment: From the above figure, we observe that the distribution is almost symmetric and there are no outliers.

7. **'charges'**

	Statistics	Values
0	mean	13279.12
1	std	12110.36
2	min	1121.87
3	10%	2358.52
4	25%	4746.34
5	50%	9386.16
6	75%	16657.72
7	90%	34832.74
8	max	63770.43

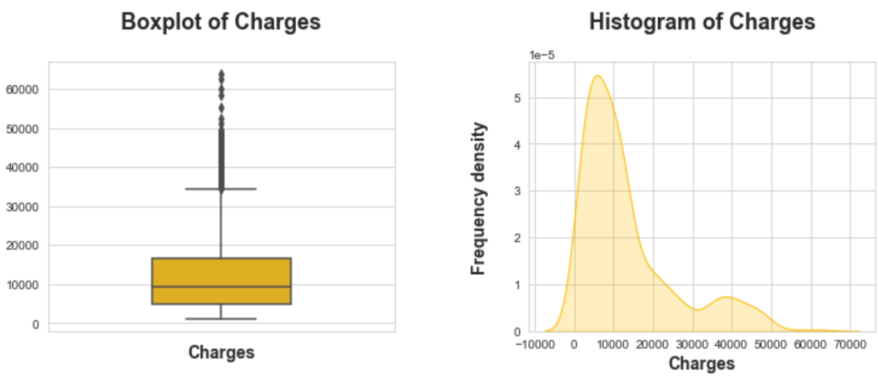


Figure 7: **Distribution of Charges**

Comment: From the above diagram, we observe that the distribution of charges is highly positively skewed with the evidence of the presence of large number of outliers.

3.2 Paired plots

Here we are interested in examining the association between the predictors and the response variable.

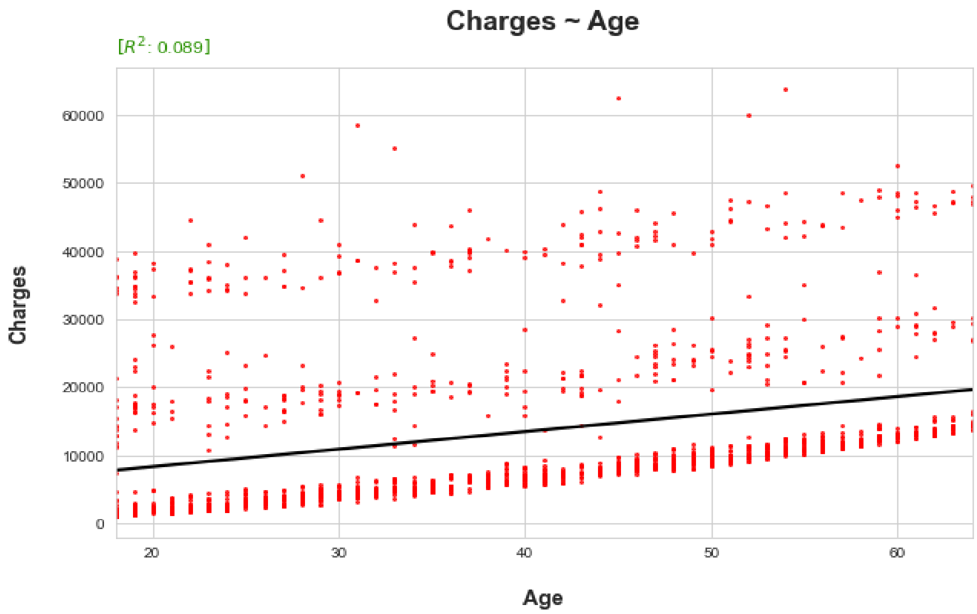


Figure 8: **Relationship between Age and Charges**

1. **Comment:** From the above plot, we can see that the relationship between Age and Charges is almost linear.

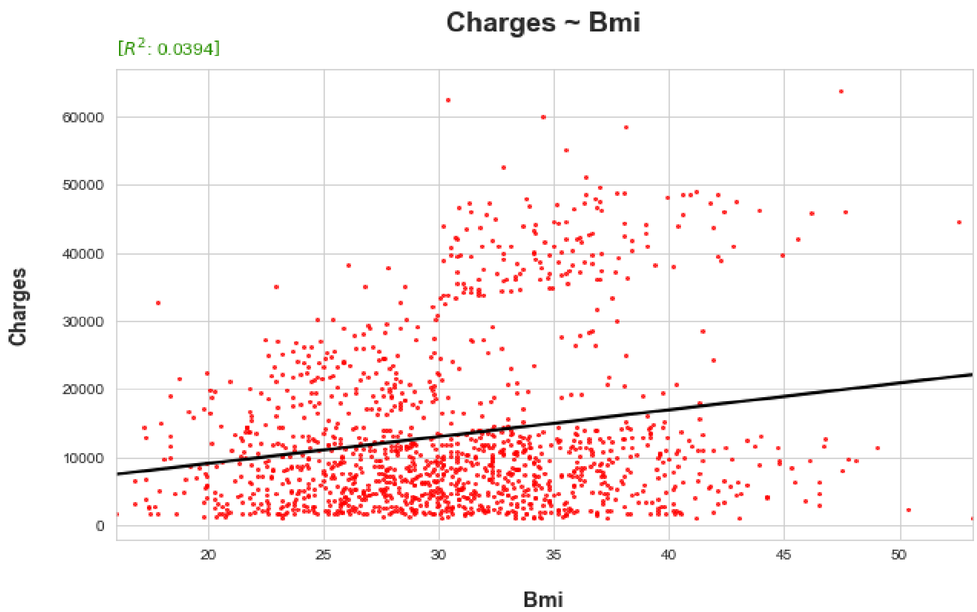


Figure 9: Relationship between BMI and Charges

2. **Comment:** From the above plot, we can see that there is a slight positive correlation between BMI and the response Charges.

4 Train-Test Split

Here we add dummy variables corresponding to each categorical column.

Now, we have split the whole dataset into two parts - one consisting of randomly chosen 70% of the rows (called the train data) and the other consisting of the remaining 30% of the rows (called the test data).

The main analysis will be based on the training dataset and models will be fitted using this data.

Later, the validity and accuracy of the models will be verified using the test data.

5 Box-Cox Transformation

Previously we have observed that the distribution of Charges is highly positively skewed. So, fitting a regression line with the original data would be inappropriate as the response will violate the normality assumption. Hence, the Box-Cox transformation is used to make it more symmetric.
Transformation Formula:

$$y^{(\lambda)} = \begin{cases} \dfrac{y^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0 \\ \log_e(y), & \text{if } \lambda = 0 \end{cases}$$

The distribution of Charges before and after the transformation is shown below.

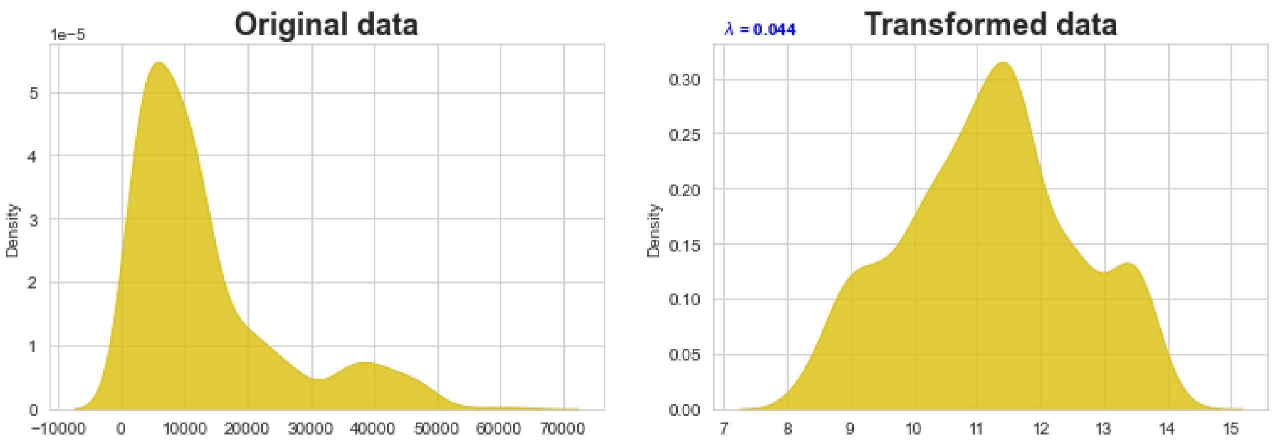


Figure 10: **The effect of transformation on the response**

Comment: Observe that the distribution of the response has become almost symmetrical.

6 Box-Tidwell Transformation

After implementing the Box-Cox transformation, from the scatter plot of continuous variables with the response, we observe that there is no significant linear relationship. To get linearity among them, we use the classical Box-Tidwell transformation.

Transformation Formula:

$$x^* = x^\alpha, \quad \alpha \in \mathbb{R}$$

To see how the transformation has helped to attain linearity between the response and the continuous explanatory variables, we look at the scatter plots before and after the transformation.

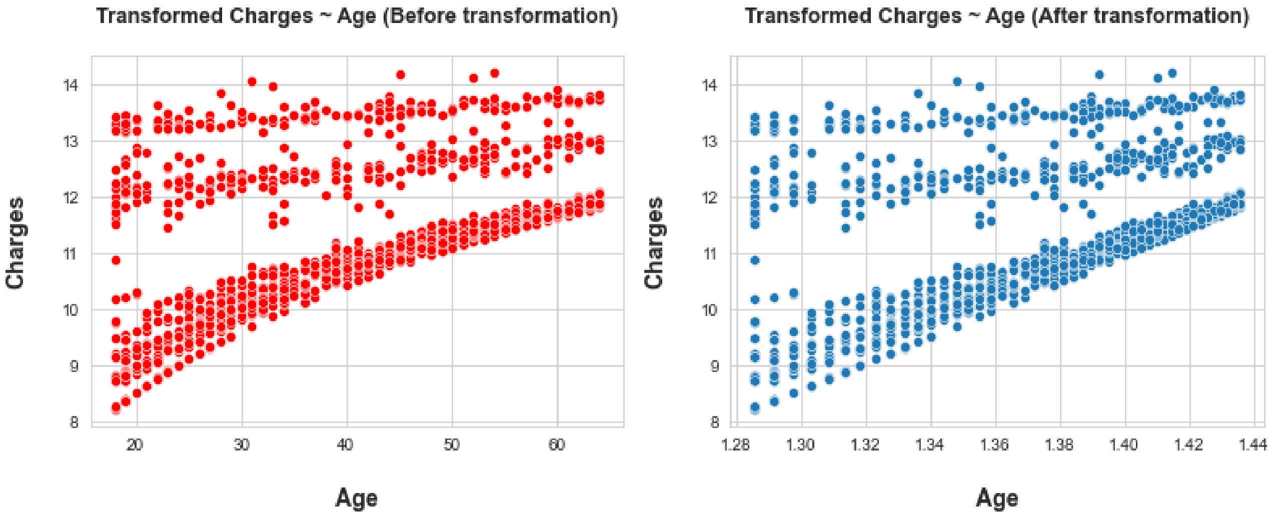


Figure 11: **Effect of Box-Tidwell Transformation (Age)**

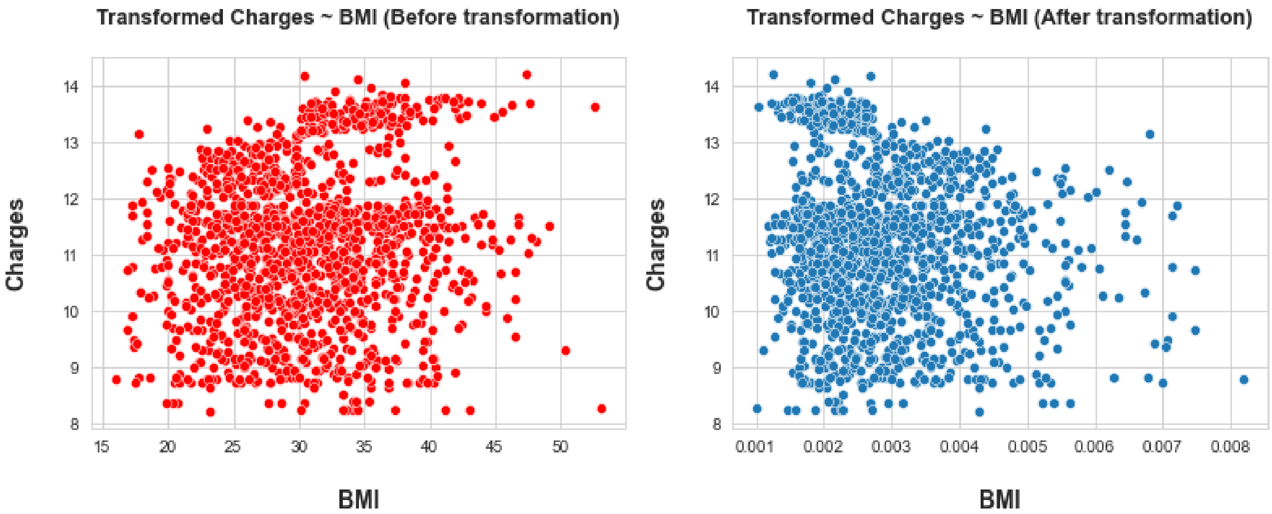


Figure 12: **Effect of Box-Tidwell Transformation (BMI)**

Comment: The above diagrams suggest that the degree of linear relationship has increased for both cases. Further, the nature of the linear relationship between transformed charges and BMI has changed after the transformation.

7 Added Variable Plot

7.1 Why?

Added variable plots (partial regression plots) are refined residual plots that provide graphic information about the marginal importance of a predictor variable X_k given the other predictor variables already in the model.

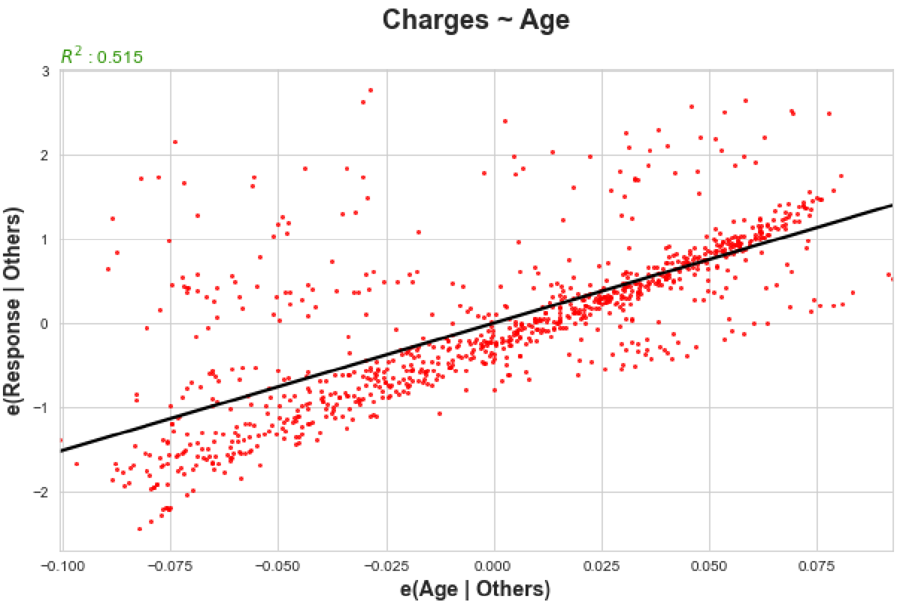


Figure 13: **Effect of Box-Tidwell Transformation (BMI)**

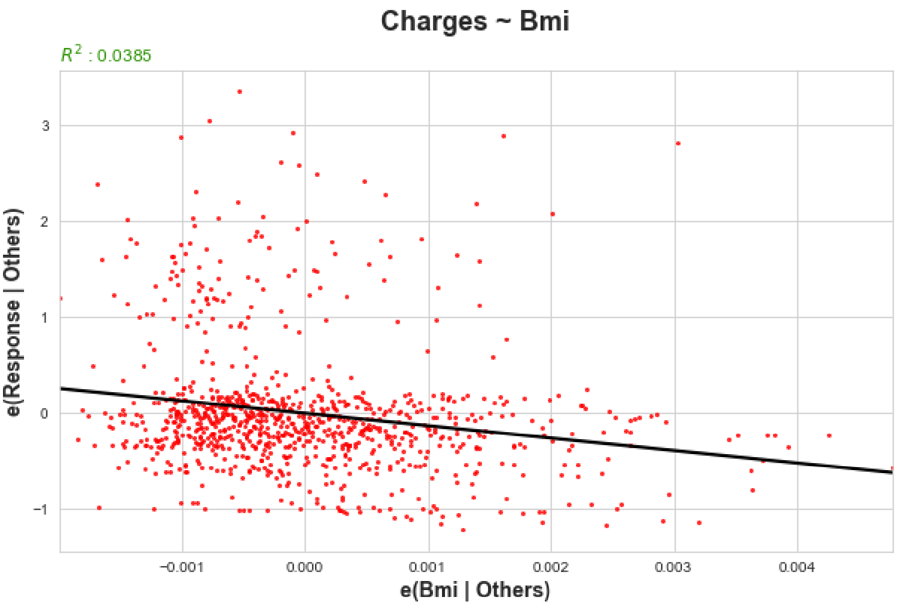


Figure 14: **Effect of Box-Tidwell Transformation (BMI)**

Comment: The Added variable plots of Transformed charges with respect to age and BMI show a linear relationship. Therefore, the marginal influences of these continuous predictors on the response are moderately strong.

8 Model Construction

In this section, we are aiming to build a robust model with maximum interpretability. To achieve our goal, we will use the Best Subset Selection method to find the most important predictors and consider only those for our final model.

8.1 Criteria for Selecting Best Model

8.1.1 Terminology

- n : Total number of rows in the Train data.
- p : Total number of parameters in the model.
- SSE_p : Sum of Squares due Errors of a p-subset model. (Number of parameters is $(p-1)$)
- SSTO : Total Sum of Squares.
- MSE_{full} : Mean Square Error of the Full Model.

8.1.2 Chosen Criterion

1. **Adjusted $\mathbf{R^2}$**

   The formula for adjusted R^2 is given by:

$$R^2_{a,p} = 1 - \left(\frac{n-1}{n-p}\right)\frac{\text{SSE}_p}{\text{SSTO}}$$

2. **Akaike Information Criterion (AIC)**

   The formula of **AIC** is given by:

$$\text{AIC}_p = n\log\left(\frac{\text{SSE}_p}{n}\right) + 2p$$

3. **Bayesian Information Criterion (BIC)**

   The formula of **BIC** is given by:

$$\text{BIC}_p = n\log(\text{SSE}_p) - n\log(n) + [\log(n)]p$$

8.2 Selected Models

- The best subset of predictors obtained by using the $\mathbf{R^2_{a,p}}$ criterion are listed below:
    1. Age
    2. Bmi
    3. Children
    4. Sex_male
    5. Smoker_yes
    6. Region_northwest
    7. Region_southeast
    8. Region_southwest
- The best subset of predictors obtained by using the **AIC** criterion are listed below:
    1. Age
    2. Bmi
    3. Children
    4. Sex_male
    5. Smoker_yes
    6. Region_southeast
    7. Region_southwest
- The best subset of predictors obtained by using the **BIC** criterion are listed below:
    1. Age
    2. Bmi
    3. Children
    4. Smoker_yes

Comment: Notice that, $R^2_{a,p}$ selection criteria is returning the full set of predictors. Whereas AIC and BIC selection criteria return a proper subset of the predictors. Further, the number of predictors returned by BIC is the lowest.

9 Residual Analysis

Here, we will be checking whether the following assumption holds:

$$\epsilon_i \overset{iid}{\sim} \mathcal{N}(0, \sigma^2),\ \forall\, i\ (\sigma^2 \text{ is a constant})$$

To verify the assumption, the following plots and tests are done.

9.1 Necessary Plots

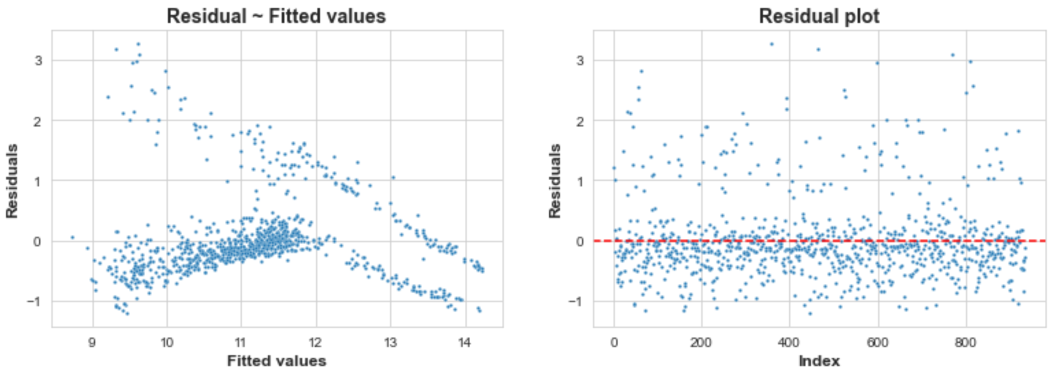


Figure 15: **Residual vs. Fitted plot and Residual Plot** $\left(\mathbf{For\ R^2_{a,p}}\right)$

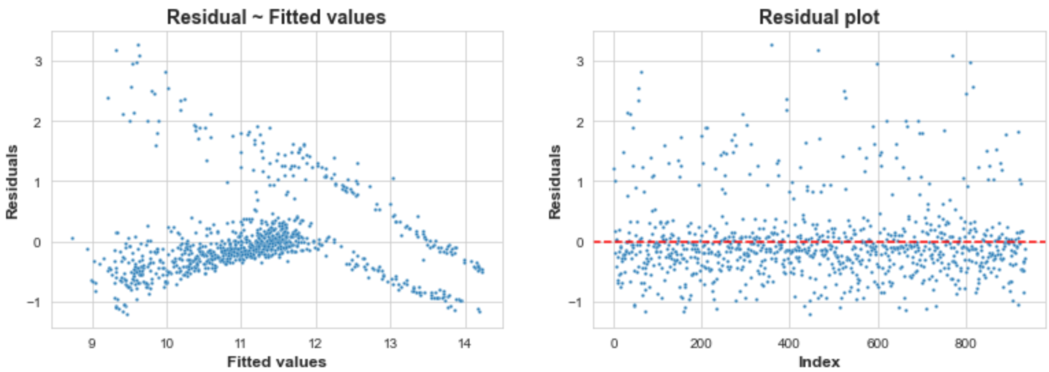


Figure 16: **Residual vs. Fitted plot and Residual Plot** $\left(\mathbf{For\ AIC}\right)$

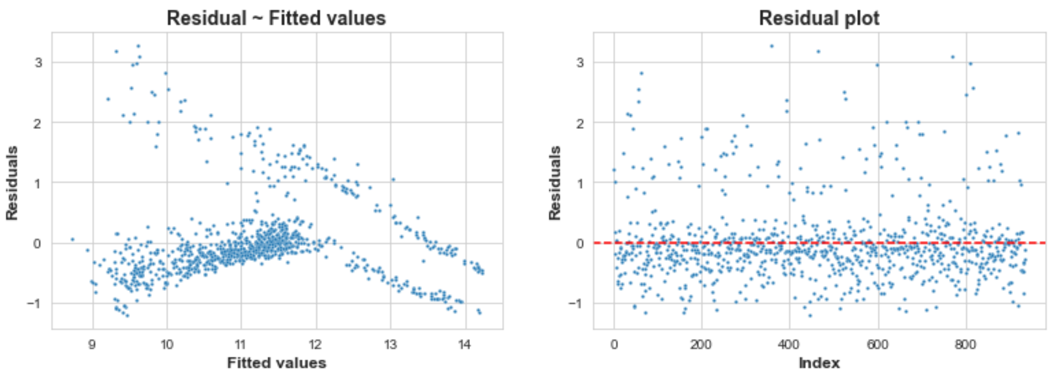


Figure 17: **Residual vs. Fitted plot and Residual Plot** $\left(\mathbf{For\ BIC}\right)$

Comment: As we can observe the plots are quite similar for each of the selection criteria. In each of these cases, the residual versus fitted plot shows a funnel-like shape with some evident clusters. Also, the residual plot indicates the absence of auto-correlation.

9.2 Brown-Forsythe Test

This test is implemented to check the homogeneity of error variance, i.e. to test,

$$H_0 : \sigma_i^2 = \sigma^2(\text{Constant})\ \forall\ i = 1(1)n \quad \text{against} \quad H_1 : \text{At least one inequality}$$

Test Statistic

$$F = \frac{(N-k)}{(k-1)} \frac{\sum_{j=1}^{k} n_j(\bar{z}_j - \bar{z})^2}{\sum_{j=1}^{k}\sum_{i=1}^{n_j}(z_{ij} - \bar{z}_j)^2}$$

Where,

- $z_{ij} = |y_{ij} - \tilde{y}_j| \quad \big(\tilde{y}_j$ denotes median of the group $j,\ j = 1, 2, \ldots, k\big)$
- k : Number of groups
- n_j : Number of observations under group j
- N : Total number of instances
- $\bar{z}_j$: Mean of group j
- $\bar{z}$: Overall mean

Here,

- $F \overset{H_0}{\sim} F_{k-1,N-k}$
- $N = 935$
- $k = 3$
- Groups are of the same size.
- Here we consider 5% level of significance.

Observations:

1. **For $\mathbf{R^2_{a,p}}$**
   - Value of F-statistic $= 15.24$
   - p-value $= 3.12 \times 10^{-7}$
2. **For AIC**
   - Value of F-statistic $= 15.53$
   - p-value $= 2.33 \times 10^{-7}$

3. **For BIC**

   - Value of F-statistic $= 13.68$
   - p-value $= 1.41 \times 10^{-6}$

Comment: As we can see, in all of the above cases, the assumption of homoscedasticity is getting violated.

9.3 Inspecting Normality of Residuals

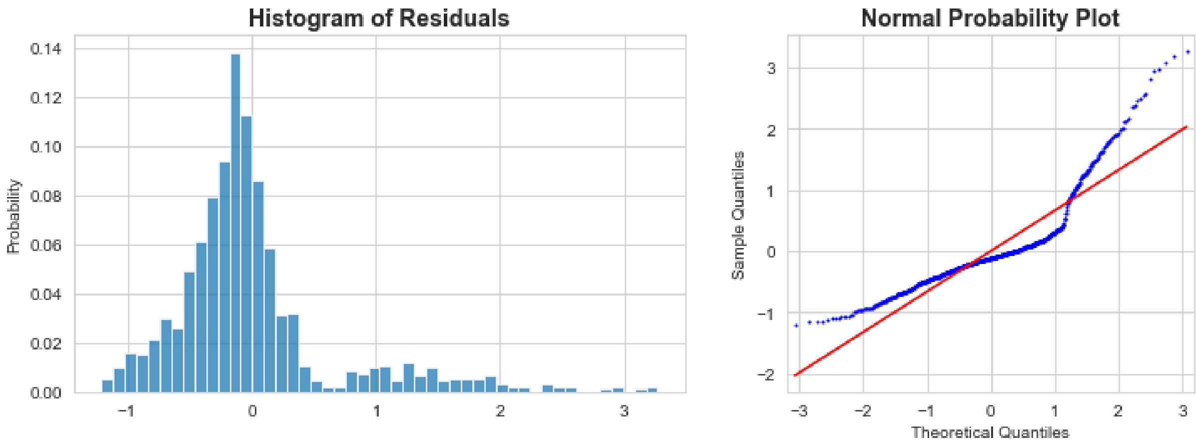


Figure 18: **Distribution and the QQ Plot of the Residuals** $\left(\text{For } \mathbf{R^2}_{a,p}\right)$

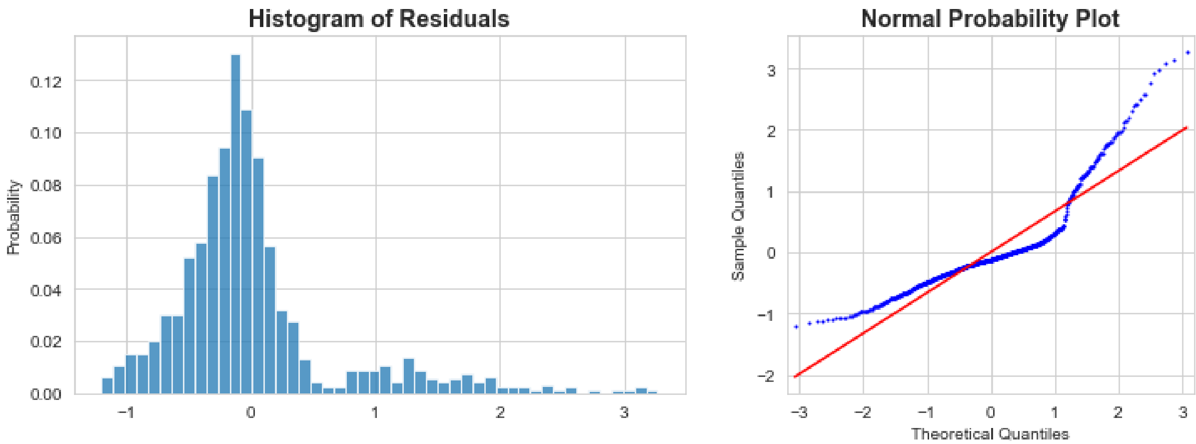


Figure 19: **Distribution and the QQ Plot of the Residuals** $\left(\textbf{For AIC}\right)$

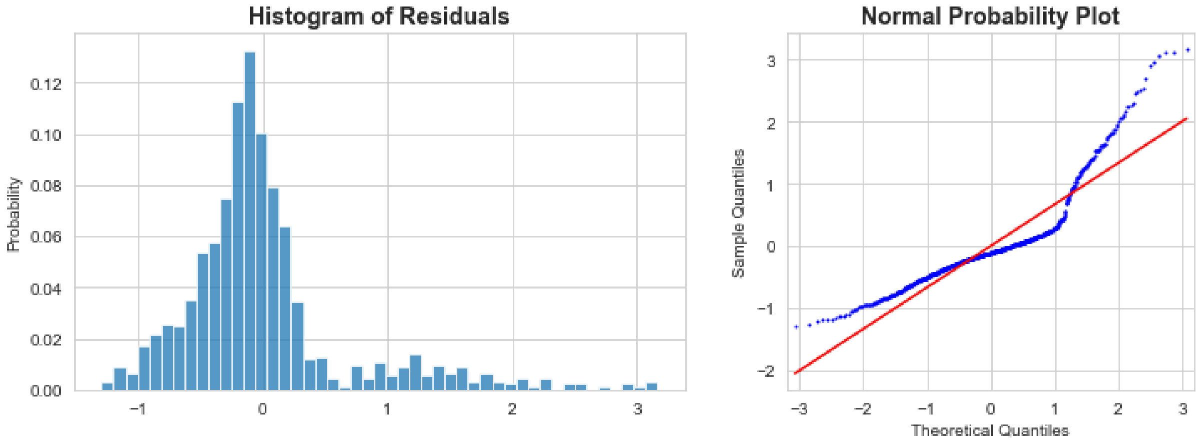


Figure 20: **Distribution and the QQ Plot of the Residuals (For BIC)**

Comment: The above diagrams suggest that the distribution of residuals is positively skewed in each of the cases. So the assumption of normality is violated, which is also evident from the QQ plot.

10 Outlier Analysis

10.1 Outlier w.r.t. the Response

Here, our aim is to find out the outliers with respect to the response. By outliers, we mean those observations for which $|r_i| > 2$, where $|r_i|$ is the studentized residual value for i^{th} observation.

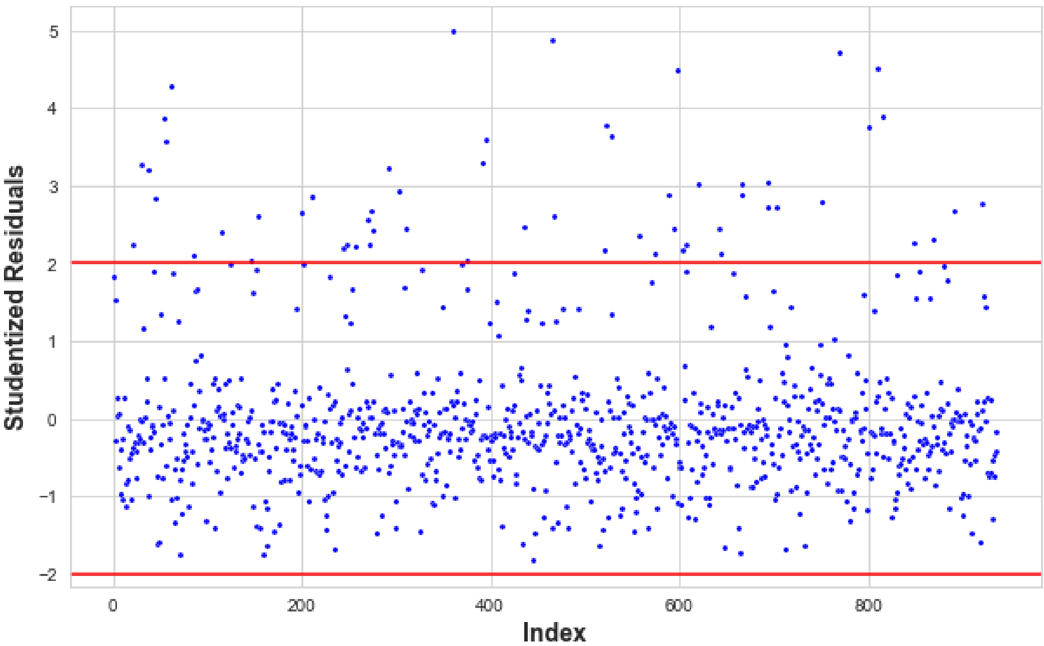


Figure 21: **Studentized Residual Plot (For $\mathbf{R^2}_{a,p}$)**

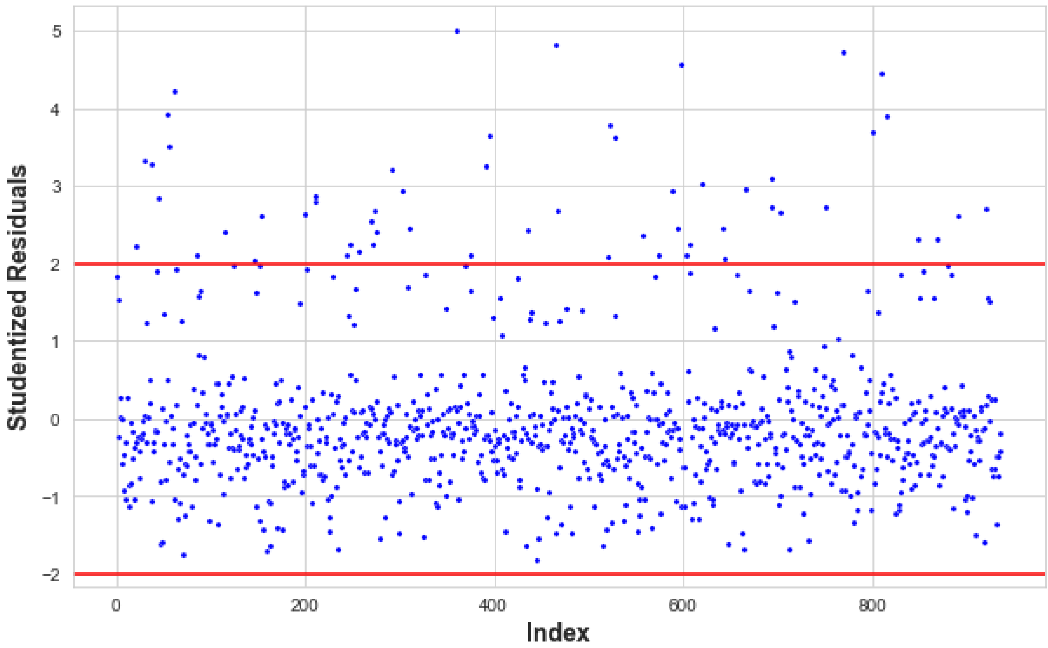


Figure 22: **Studentized Residual Plot (For AIC)**

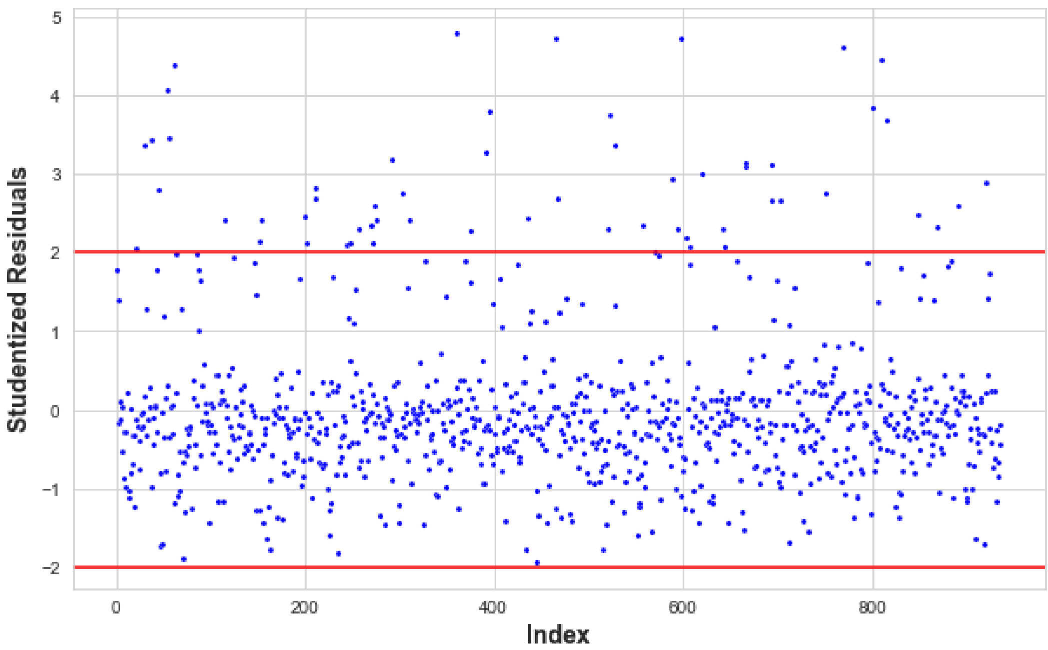


Figure 23: **Studentized Residual Plot (For BIC)**

Comment: The presence of outliers is evident in all of the three models.

10.2 Outlier w.r.t. the Predictors

Here, we compute leverage values for each of the observations for all three models and mark those observations as outliers for which the leverage value is more than $\frac{2p}{n}$. $\big($Where p is the number of parameters in the model and n is the number of observations in the data$\big)$

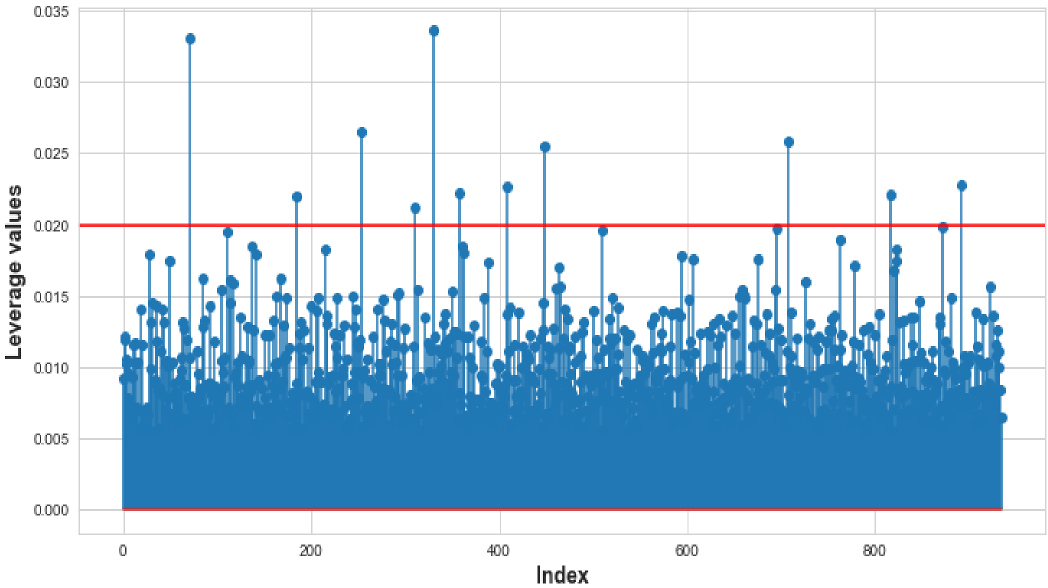


Figure 24: **Plot of the Leverages** $\big($**For** $\mathbf{R^2}_{a,p}\big)$

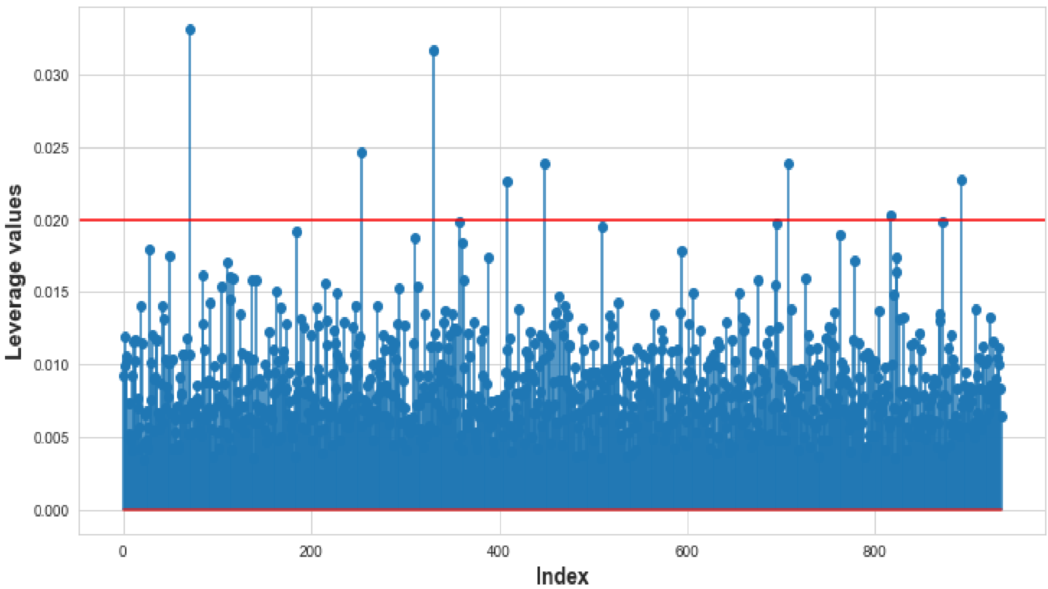


Figure 25: **Plot of the Leverages** $\big($**For AIC**$\big)$

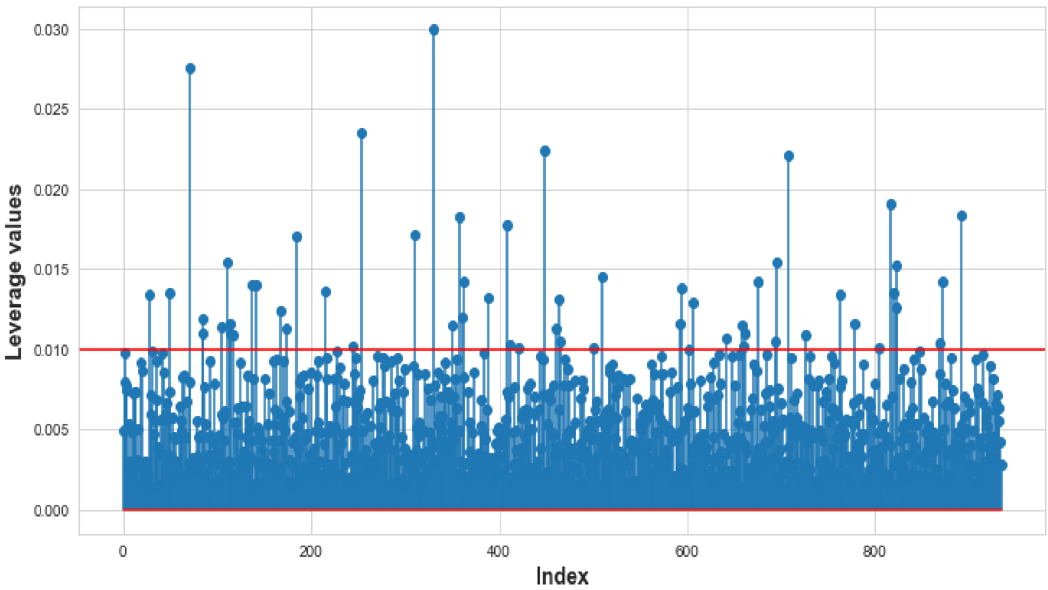


Figure 26: **Plot of the Leverages** (**For BIC**)

Comment: For the best model w.r.t adjusted R^2 and AIC, the threshold, $\frac{2p}{n}$ turns out to be 0.02 and that for BIC is 0.01 since, number of parameters is least for BIC. This leads to the occurrence of a greater number of outliers in the case of BIC compared to other models.

10.3 Cook's Distance

Cook's distance is a summary of how much the regression model gets changed when i^{th} observation is removed. It takes into account both the leverage and residual of each of the observations.

In this section, we will try to find out if the outliers are influential points using Cook's Distance.

Formula:

$$D_i = \frac{\sum_{j=1}^{n} \left(\hat{Y}_j - \hat{Y}_{j(i)}\right)^2}{p \cdot \text{MSE}} \sim F_{p,n-p}$$

Where the notations carry their usual meaning.

If the percentile value of F-distribution is less than 10% or 20% for i^{th} case, then we conclude that i^{th} case is not much influential. However, if the percentile is 50 or more then it will be influential.

Model	**10%**	**20%**
AIC	0.44	0.57
BIC	0.32	0.47
$R^2_{a,p}$	0.46	0.61

Table 2: Table showing the 10^{th} and 20^{th} percentile of respective F-distribution

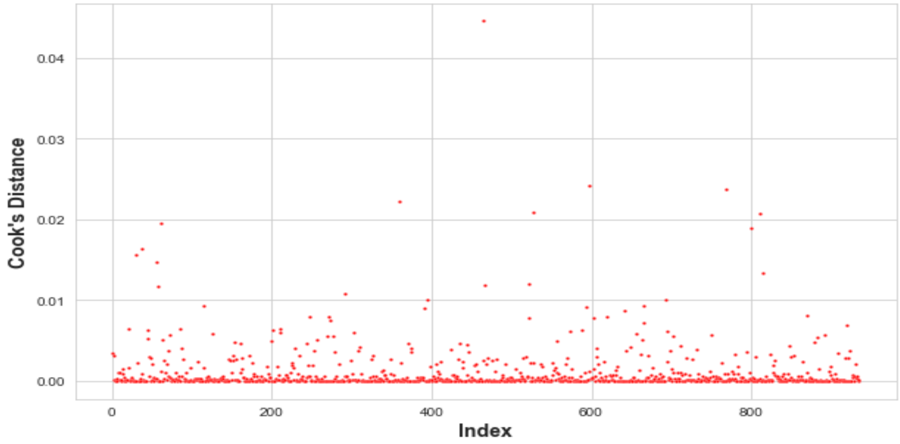


Figure 27: **Plot of Cooks Distance $\left(\text{For } \mathbf{R^2_{a,p}}\right)$**

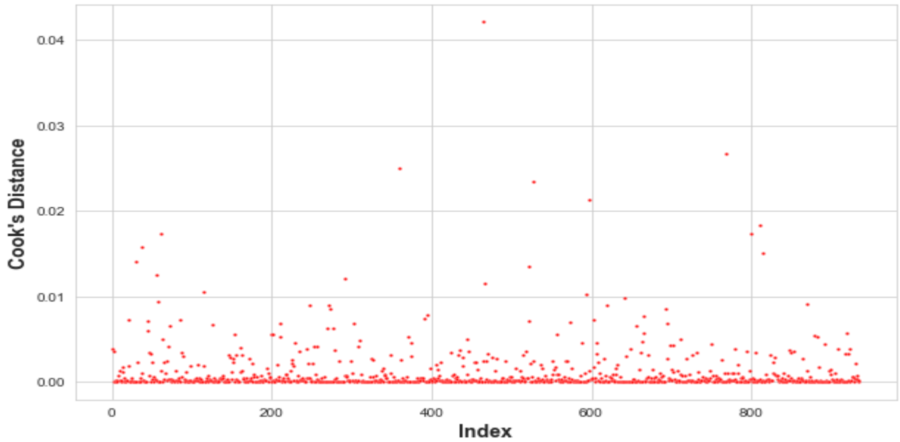


Figure 28: **Plot of Cooks Distance (For AIC)**

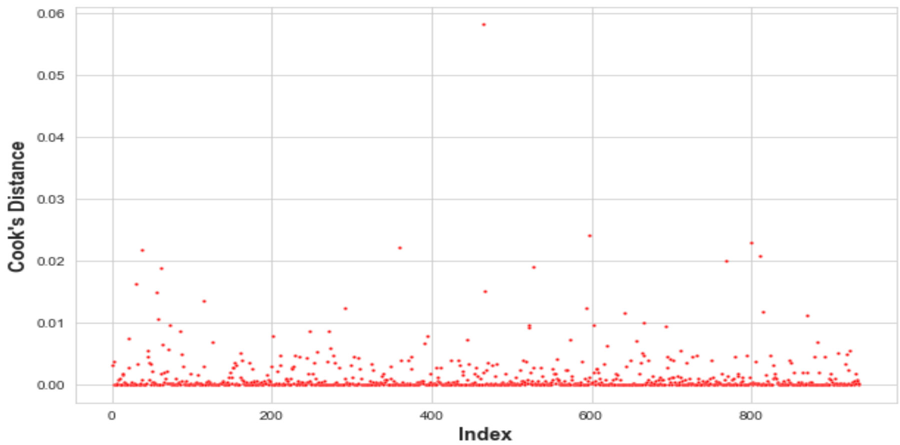


Figure 29: **Plot of Cooks Distance (For BIC)**

Comment: We can observe from the above plots that no influential point exists in the best models obtained by $R^2_{a,p}$ and AIC criteria. However, there is one influential point present in the case of the best model selected by the BIC criterion.

For further analysis of the model selected by the BIC criterion, we will be removing the influential instance and re-build the model.
The plot of Cook's Distance after removing the influential point is given below.

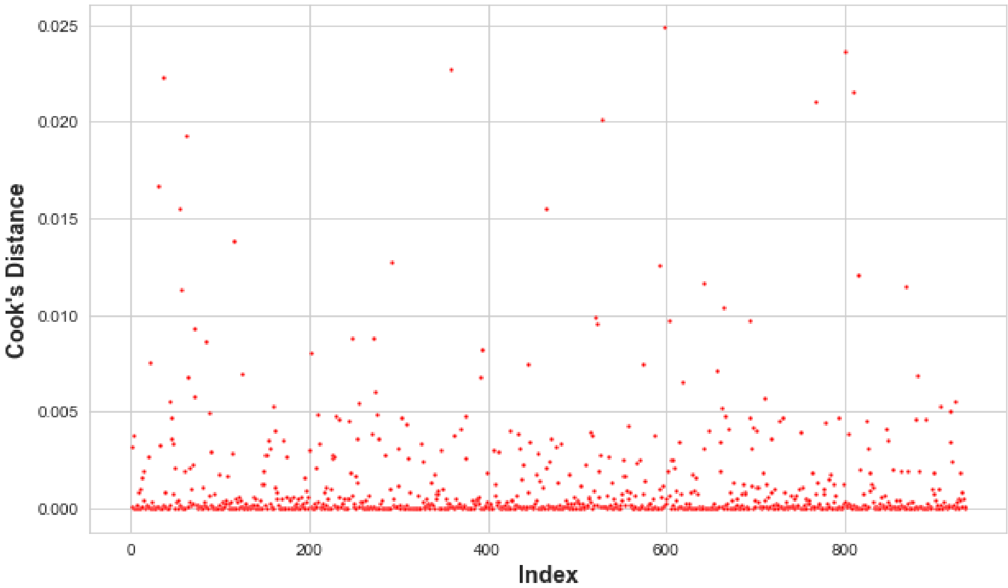


Figure 30: **Plot of Cooks Distance (For BIC after removing influential point)**

Comment: Notice that there is no influential point present here.

11 Multicollinearity

By multicollinearity, we mean moderate or high correlation among two or more of the predictors in a regression model.

Here, to check for multicollinearity, we use the VIF technique. **Formula:**

$$(\text{VIF})_k = \frac{1}{1 - R_k^2}$$

Where R_k^2 is the multiple coefficient of determination when X_k is regressed on other $p-2$ predictors, $(k = 1, 2, \ldots, p-1)$

If $\max(\text{VIF})_k$ is greater than 15, then we can assume that there is multicollinearity present in the dataset.

Observation

Features	VIF
Age	9.15
Bmi	8.23
Children	1.83
Smoker_yes	1.26

Table 3: **VIF values for different predictors (BIC)**

Features	VIF
Age	13.12
Bmi	8.65
Children	1.83
Sex_male	2.11
Smoker_yes	1.28
Region_southeast	1.65
Region_southwest	1.52

Table 4: **VIF values for different predictors (AIC)**

Features	VIF
Age	15.54
Bmi	8.65
Children	1.83
Sex_male	2.11
Smoker_yes	1.28
Region_northwest	2.07
Region_southeast	2.25
Region_southwest	2.07

Table 5: **VIF values for different predictors $\mathbf{R^2_{a,p}}$**

Comment: From the above tables, it is clear that there is no multicollinearity present among the predictors in the best models selected by AIC and BIC. However, the same is not the case for the model obtained by $R^2_{a,p}$ selection criterion. In this case, VIF value for the predictor 'Age' is more than 15. So, before proceeding further with model validation, we remove 'Age' from the set of predictors.

12 Model Validation

In this section, we will validate the models by checking their performance on the Test Data. We have taken MSPR as the metric for judging the performance of our models on unseen data.

Formula:

$$\text{MSPR} = \frac{1}{n^*} \sum_{i=1}^{n^*} \left(Y_i - \hat{Y}_i\right)^2$$

Where, n^* is the number of instances in the Test Data.

Model	MSPR
AIC	0.3966
BIC	0.4110
$R^2_{a,p}$	0.9404

Table 6: **MSPR values for different models**

Comment: We can observe that the value of MSPR is significantly higher for $R^2_{a,p}$ model compared to the other models. This is because of the omission of a significant predictor (viz. 'Age').

13 Final Model

In the previous section, we have seen that the MSPR values for the models selected by AIC and BIC are almost similar. Moreover, the best subset of predictors is lesser in the case of BIC than that of AIC which suggests greater interpretability (i.e. lesser complexity).

Considering all the evidence and outcomes discussed above, we choose the best model obtained by BIC as our final model.

Model: $\boxed{Y = -9.87 + 15.21X_1 - 122.16X_2 + 0.11X_3 + 2.32X_4}$

Here,

- $Y = \dfrac{(\text{charges})^{0.044} - 1}{0.044}$
- $X_1 = (\text{age})^{0.087}$
- $X_2 = (\text{bmi})^{-1.735}$
- $X_3 = \text{Children}$
- $X_4 = \text{smoker\_yes}$

14 Python Codes

```python
# Importing necessary Libraries:
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats as sts
from scipy.stats import f
from pandas.plotting import table
import statsmodels.api as sm
from statsmodels.formula.api import ols
from itertools import permutations as pmr
from itertools import chain, combinations
from sklearn.model_selection import train_test_split
from statsmodels.stats.outliers_influence import variance_inflation_factor

pd.set_option('display.max_rows', None)
sns.set_style('whitegrid')
#==================================================================================================
# Calling the data:
df = pd.read_csv('D:\Datasets\Insurance\insurance.csv')
#==================================================================================================
# Visualizing the first and last 5 rows of the data:
#==================================================================================================
print(df.head(5))
print('_'*65)
print(df.tail(5))
#==================================================================================================
# Checking the data:
#==================================================================================================
df.shape
df.info()
df.duplicated()
df = df.drop_duplicates()
df.shape
df.isnull().sum()
df.columns = df.columns.str.title()
#==================================================================================================
# Function to plot the univariate distributions:
#==================================================================================================

    ## CASE I (Categorical)
def uni_func(x):
    d = df[x].value_counts().reset_index().rename(columns = {'index':x,
                                                              x:'Count'})
    d['Proportion'] = (100*d['Count']/sum(d['Count'])).round(2).astype(str) + ' %'

    fig = plt.figure(figsize = (12, 4))
    gs = fig.add_gridspec(1, 2, width_ratios = [1, 2])

    # creating the table:
    ax1 = fig.add_subplot(gs[0])
    plt.axis('off')
    t = table(ax1, d, loc = 'center')
    t.auto_set_font_size(False)
    t.scale(1.5, 1.8)
    t.set_fontsize(14)

    # creating the barplot:
```

```
    ax2 = fig.add_subplot(gs[1])
    s = sns.barplot(x = x, y = 'Count',
           data = d, color = '#218FD6', ax = ax2)
    s.set_title('Distribution of ' + x + '\n',
           fontdict = {'weight':'bold', 'size':18})
    s.set_xlabel('\nLabels', fontdict = {'weight':'bold', 'size':14})
    s.set_ylabel('Frequency\n', fontdict = {'weight':'bold', 'size':14})
    s.set_xticklabels(s.get_xticklabels(), rotation = 30,
                     fontdict = {'weight':'bold'})

    plt.subplots_adjust(wspace = 0.5)
    plt.show()

    # Viewing the distributions
uni_func('Region')
uni_func('Children')
uni_func('Sex')
uni_func('Smoker')


    ## CASE II (Continuous)
def cont_func(x):
    d = df[x].describe(percentiles = [0.1,0.25,0.5,0.75,0.9]).drop('count').reset_index(
    ).rename(columns = {'index':'Statistics', x:'Values'}).round(2)

    fig = plt.figure(figsize = (16, 4))
    gs = fig.add_gridspec(1, 3, width_ratios = [1, 3, 3])

    # creating the table:
    ax1 = fig.add_subplot(gs[0])
    plt.axis('off')
    t = table(ax1, d, loc = 'center')
    t.auto_set_font_size(False)
    t.scale(1.5, 1.8)
    t.set_fontsize(14)

    # creating the boxplot:
    ax2 = fig.add_subplot(gs[1])
    s = sns.boxplot(y = df[x], ax = ax2, color = '#FDBD04',
                width = 0.4)
    s.set_title('Boxplot of '+ x + '\n',
               fontdict = {'weight':'bold', 'size':18})
    s.set_xlabel(x, fontdict = {'weight':'bold', 'size':14})
    s.set_ylabel(' ')

    # creating the histogram:
    ax3 = fig.add_subplot(gs[2])
    s = sns.kdeplot(x = df[x], fill = True, ax = ax3, color = '#FDBD04')
    s.set_title('Histogram of '+ x + '\n',
               fontdict = {'weight':'bold', 'size':18})
    s.set_xlabel(x, fontdict = {'weight':'bold', 'size':14})
    s.set_ylabel('Frequency density\n', fontdict = {'weight':'bold', 'size':14})

    plt.subplots_adjust(wspace = 0.5)
    plt.show()

    # Viewing the distributions
cont_func('Age')
cont_func('Bmi')
cont_func('Charges')
```

```
#=============================================================================================
# Visualizing the relationship between the Predictors and the Response
#=============================================================================================

    ## CASE I (Continuous Predictors vs. Continuous Response)
def cont_cont(x):
    r_sq = (df['Charges'].corr(df[x])**2).round(4)

    plt.figure(figsize = (10,6))
    s = sns.regplot(x = x, y = 'Charges',
               data = df, color = '#DD1400', ci = False,
                   scatter_kws = {'s':4, "color": "red"}, line_kws = {"color": "black"})
    s.set_title('Charges ~ ' + x + '\n',
                fontdict = {'weight':'bold', 'size':18})
    s.set_xlabel('\n' + x, fontdict = {'weight':'bold',
                                        'size':14})
    s.set_ylabel('Charges\n', fontdict = {'weight':'bold',
                                        'size':14})
    plt.text(np.min(df[x]), 69000, r'$[R^2$: ' + str(r_sq) + '$]$',
             fontsize = 12, color = '#2C9203')
    plt.show()

    # fiting model:
    X = sm.add_constant(df[x])
    Y = df['Charges']

    # Fit the linear regression model
    model = sm.OLS(Y, X).fit()

    # View the summary table
    print(model.summary())

    ## Visualization:
cont_cont('Bmi')
cont_cont('Age')

    ## CASE II (Categorical Predictors vs. Continuous Response)
def cont_cat(x):
    plt.figure(figsize = (10,5))
    s = sns.boxplot(x = x, y = 'Charges', data = df, width = 0.5,
            flierprops = dict(marker = 'o', markerfacecolor = '#5207C4', markersize = 4),
           color = '#18ADE5')
    s.set_title('Charges ~ ' + x + '\n',
                fontdict = {'weight':'bold', 'size':18})
    s.set_xlabel('\n' + x, fontdict = {'weight':'bold',
                                        'size':14})
    s.set_ylabel('Charges\n', fontdict = {'weight':'bold',
                                        'size':14})
    plt.show()

    print('-'*100,'\nANOVA Result:')
    # ANOVA
    model = ols('Charges ~ ' + x, data = df).fit()
    print(sm.stats.anova_lm(model, type = 2))

    # Visualization:
cont_cat('Region')
cont_cat('Smoker')
cont_cat('Sex')
cont_cat('Children')
```

```
#==========================================================================================
# Box-Cox Transformation
#==========================================================================================
Y = df['Charges']
Y_new, lambda_hat = sts.boxcox(Y)

fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (13,4))
s1 = sns.kdeplot(x = Y, fill = True, alpha = 0.8, ax = ax[0], color = '#D9C008')
s1.set_title('Original data', fontdict = {'weight':'bold', 'size':18})
s1.set_xlabel(' ')
s2 = sns.kdeplot(x = Y_new, fill = True, alpha = 0.8, ax = ax[1], color = '#D9C008')
s2.set_title('Transformed data', fontdict = {'weight':'bold', 'size':18})
plt.text(7, 0.34, r'$\lambda$ = ' + str(round(lambda_hat, 3)),
        fontdict = {'weight':'bold', 'color':'blue'})

plt.show()

    # saving the transformed data:
df['Y_new'] = Y_new
df_copy = df.drop('Charges', axis = 1)
#==========================================================================================
# Box-Tidwell Transformation
#==========================================================================================
for i in df_copy[['Age','Bmi']].columns:
    d1 = pd.DataFrame({"1":np.ones_like(df_copy[i]),"2":df_copy[i]})
    d2 = pd.DataFrame({"1":np.ones_like(df_copy[i]),"2":df_copy[i],"3":df_copy[i]*np.log(df_copy[
    i])})
    beta1 = np.linalg.lstsq(d1.values, df_copy['Y_new'].values, rcond=None)[0]
    beta2 = np.linalg.lstsq(d2.values, df_copy['Y_new'].values, rcond=None)[0]
    alpha = (beta2[-1]/beta1[-1])+1
    print(round(alpha,3))
    df_copy[i] = df_copy[i]**alpha

    # Scatterplot of transformed predictors and transformed Response
fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (13,4))

s1 = sns.scatterplot(x = 'Age', y = 'Y_new', data = df, ax = ax[0],
                    color = 'red')
s1.set_title('Transformed Charges ~ Age (Before transformation)' + '\n',
                fontdict = {'weight':'bold', 'size':13})
s1.set_xlabel('\nAge', fontdict = {'weight':'bold',
                                        'size':14})
s1.set_ylabel('Charges\n', fontdict = {'weight':'bold',
                                        'size':14})
s2 = sns.scatterplot(x = 'Age', y = 'Y_new', data = df_copy, ax = ax[1])
s2.set_title('Transformed Charges ~ Age (After transformation)' + '\n',
                fontdict = {'weight':'bold', 'size':13})
s2.set_xlabel('\nAge', fontdict = {'weight':'bold',
                                        'size':14})
s2.set_ylabel('Charges\n', fontdict = {'weight':'bold',
                                        'size':14})
plt.show()



fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (13,4))

s1 = sns.scatterplot(x = 'Bmi', y = 'Y_new', data = df, ax = ax[0],
                    color = 'red')
s1.set_title('Transformed Charges ~ BMI (Before transformation)' + '\n',
```

```
                fontdict = {'weight':'bold', 'size':13})
s1.set_xlabel('\nBMI', fontdict = {'weight':'bold',
                                        'size':14})
s1.set_ylabel('Charges\n', fontdict = {'weight':'bold',
                                        'size':14})
s2 = sns.scatterplot(x = 'Bmi', y = 'Y_new', data = df_copy, ax = ax[1])
s2.set_title('Transformed Charges ~ BMI (After transformation)' + '\n',
                fontdict = {'weight':'bold', 'size':13})
s2.set_xlabel('\nBMI', fontdict = {'weight':'bold',
                                        'size':14})
s2.set_ylabel('Charges\n', fontdict = {'weight':'bold',
                                        'size':14})
plt.show()
#===============================================================================================
# Train-Test Split
#===============================================================================================
Y = df_copy['Y_new']
X = df_copy.drop('Y_new', axis = 1)

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.30, random_state = 42)

X_train.head(5)
X_train.shape
#===============================================================================================
# Added Variable Plot
#===============================================================================================
def add_var(x):
    def resid(x,y):
        x = sm.add_constant(pd.get_dummies(x, drop_first = True))
        M = sm.OLS(y,x).fit()
        return M.resid

    # Storing the set of other variables:
    X_new = X_train.drop(x, axis = 1)

    # y ~ x_others
    r1 = resid(X_new, Y_train)

    # x ~ x_others
    r2 = resid(X_new, X_train[x])

    r_sq = (r1.corr(r2)**2).round(4)

    plt.figure(figsize = (10,6))
    s = sns.regplot(x = r2, y = r1, color = '#DD1400', ci = False,
                   scatter_kws = {'s':4, "color": "red"}, line_kws = {"color": "black"})
    s.set_title('Charges ~ ' + x + '\n',
                fontdict = {'weight':'bold', 'size':18})
    s.set_xlabel('e(' + x + ' | Others)', fontdict = {'weight':'bold',
                                        'size':14})
    s.set_ylabel('e(Response | Others)', fontdict = {'weight':'bold',
                                        'size':14})
    plt.text(np.min(r2), np.max(r1) + 0.3, r'$R^2$ : ' + str(r_sq),
             fontsize = 12, color = '#2C9203')
    plt.show()


    # Visualization:
add_var('Age')
add_var('Bmi')
```

```
#===============================================================================================
# Best Subset Selection
#===============================================================================================
def powerset(x):
    S = chain.from_iterable(combinations(x, r) for r in range(len(x)+1))
    return [list(k) for k in list(S)][1:]

X_train_dumm = pd.get_dummies(X_train, drop_first = True)
C = powerset(X_train_dumm.columns)

AIC = []
BIC = []
Adj_Rsq = []
Rsq = []

for var in C:
    X = sm.add_constant(X_train_dumm[var])
    Y = Y_train

    M = sm.OLS(Y,X).fit()

    Adj_Rsq.append(round(M.rsquared_adj,3))
    Rsq.append(round(M.rsquared,3))
    AIC.append(round(M.aic,3))
    BIC.append(round(M.bic,3))

best_df = pd.DataFrame({
    'Variable set': C,
    'Adjusted Rsq': Adj_Rsq,
    'R_Square': Rsq,
    'AIC': AIC, 'BIC': BIC
})

    # Best variables concerning different criteria:
best_subset_vars_adj = best_df.at[best_df['Adjusted Rsq'].idxmax(),'Variable set']
best_subset_vars_adj

best_subset_vars_BIC = best_df.at[best_df['BIC'].idxmin(),'Variable set']
best_subset_vars_BIC

best_subset_vars_AIC = best_df.at[best_df['AIC'].idxmin(),'Variable set']
best_subset_vars_AIC

    # Creating dummy variables:
X_train_dumm_adj = X_train_dumm[best_subset_vars_adj]
X_train_dumm_BIC = X_train_dumm[best_subset_vars_BIC]
X_train_dumm_AIC = X_train_dumm[best_subset_vars_AIC]
#===============================================================================================
# Model fitting with the best subset of variables
#===============================================================================================

    # For Adjusted R^2:
X = sm.add_constant(X_train_dumm_adj)
Y = Y_train
model_adj = sm.OLS(Y,X).fit()

resid_adj = model_adj.resid
yhat_adj = model_adj.fittedvalues

index = np.arange(1,len(resid_adj)+1)
```

```
fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (13,4))
s1 = sns.scatterplot(x = yhat_adj, y = resid_adj, s = 6, ax = ax[0])
s1.set_title('Residual ~ Fitted values', fontdict = {'weight':'bold',
                                                     'size':14})
s1.set_xlabel('Fitted values', fontdict = {'weight':'bold',
                                                     'size':12})
s1.set_ylabel('Residuals', fontdict = {'weight':'bold', 'size':12})
s2 = sns.scatterplot(x = index, y = resid_adj, s = 6, ax = ax[1])
s2.set_title('Residual plot', fontdict = {'weight':'bold',
                                                     'size':14})
s2.set_xlabel('Index', fontdict = {'weight':'bold', 'size':12})
s2.set_ylabel('Residuals', fontdict = {'weight':'bold', 'size':12})
plt.axhline(y = 0, color = 'r', linestyle = '--')
plt.show()

    # For BIC:
X = sm.add_constant(X_train_dumm_BIC)
Y = Y_train
model_BIC = sm.OLS(Y,X).fit()

resid_BIC = model_BIC.resid
yhat_BIC = model_BIC.fittedvalues

index = np.arange(1,len(resid_adj)+1)
fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (13,4))
s1 = sns.scatterplot(x = yhat_adj, y = resid_adj, s = 6, ax = ax[0])
s1.set_title('Residual ~ Fitted values', fontdict = {'weight':'bold',
                                                     'size':14})
s1.set_xlabel('Fitted values', fontdict = {'weight':'bold',
                                                     'size':12})
s1.set_ylabel('Residuals', fontdict = {'weight':'bold', 'size':12})
s2 = sns.scatterplot(x = index, y = resid_adj, s = 6, ax = ax[1])
s2.set_title('Residual plot', fontdict = {'weight':'bold',
                                                     'size':14})
s2.set_xlabel('Index', fontdict = {'weight':'bold', 'size':12})
s2.set_ylabel('Residuals', fontdict = {'weight':'bold', 'size':12})
plt.axhline(y = 0, color = 'r', linestyle = '--')
plt.show()

    # For AIC:
X = sm.add_constant(X_train_dumm_AIC)
Y = Y_train
model_AIC = sm.OLS(Y,X).fit()

resid_AIC = model_AIC.resid
yhat_AIC = model_AIC.fittedvalues

index = np.arange(1,len(resid_adj)+1)
fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (13,4))
s1 = sns.scatterplot(x = yhat_adj, y = resid_adj, s = 6, ax = ax[0])
s1.set_title('Residual ~ Fitted values', fontdict = {'weight':'bold',
                                                     'size':14})
s1.set_xlabel('Fitted values', fontdict = {'weight':'bold',
                                                     'size':12})
s1.set_ylabel('Residuals', fontdict = {'weight':'bold', 'size':12})
s2 = sns.scatterplot(x = index, y = resid_adj, s = 6, ax = ax[1])
s2.set_title('Residual plot', fontdict = {'weight':'bold',
                                                     'size':14})
s2.set_xlabel('Index', fontdict = {'weight':'bold', 'size':12})
s2.set_ylabel('Residuals', fontdict = {'weight':'bold', 'size':12})
```

```
plt.axhline(y = 0, color = 'r', linestyle = '--')
plt.show()
#================================================================================================
# Brown-Forsythe test
#================================================================================================
k = len(Y_train) // 3

    # For BIC:
d = pd.DataFrame({'Y_fit':yhat_BIC, 'res':resid_BIC})
y = d.sort_values(by = 'Y_fit')['res']
print(sts.levene(y[:k], y[k:2*k], y[2*k:3*k], center = 'median'))

    # For AIC:
d = pd.DataFrame({'Y_fit':yhat_AIC, 'res':resid_AIC})
y = d.sort_values(by = 'Y_fit')['res']
print(sts.levene(y[:k], y[k:2*k], y[2*k:3*k], center = 'median'))

    # For Adj R^2:
d = pd.DataFrame({'Y_fit':yhat_adj, 'res':resid_adj})
y = d.sort_values(by = 'Y_fit')['res']
print(sts.levene(y[:k], y[k:2*k], y[2*k:3*k], center = 'median'))
#================================================================================================
# Check for Normality
#================================================================================================
    # For BIC:
fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (12,4),
                      gridspec_kw = {'width_ratios': [1.5, 1]})
sns.histplot(resid_BIC, stat = 'probability', ax = ax[0])
sm.qqplot(resid_BIC, line = 's', ax = ax[1])
plt.setp(fig.axes[1].get_lines(), markersize = 1)
ax[0].set_title('Histogram of Residuals', fontdict = {'weight':'bold',
                                                     'size':14})
ax[1].set_title('Normal Probability Plot', fontdict = {'weight':'bold',
                                                     'size':14})
plt.show()

    # For AIC
fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (12,4),
                       gridspec_kw = {'width_ratios': [1.5, 1]})
sns.histplot(resid_AIC, stat = 'probability', ax = ax[0])
sm.qqplot(resid_AIC, line = 's', ax = ax[1])
plt.setp(fig.axes[1].get_lines(), markersize = 1)
ax[0].set_title('Histogram of Residuals', fontdict = {'weight':'bold',
                                                     'size':14})
ax[1].set_title('Normal Probability Plot', fontdict = {'weight':'bold',
                                                     'size':14})
plt.show()

    # For Adjusted R^2
fig, ax = plt.subplots(nrows = 1, ncols = 2, figsize = (12,4),
                       gridspec_kw = {'width_ratios': [1.5, 1]})
sns.histplot(resid_adj, stat = 'probability', ax = ax[0])
sm.qqplot(resid_adj, line = 's', ax = ax[1])
plt.setp(fig.axes[1].get_lines(), markersize = 1)
ax[0].set_title('Histogram of Residuals', fontdict = {'weight':'bold',
                                                     'size':14})
ax[1].set_title('Normal Probability Plot', fontdict = {'weight':'bold',
                                                     'size':14})
plt.show()
```

```
#==============================================================================================
# Outlier Analysis
#==============================================================================================

    # For AIC:
resid_std_AIC = model_AIC.outlier_test()['student_resid']

index = np.arange(1, len(resid_std_AIC)+1)

plt.figure(figsize = (10,6))
s = sns.scatterplot(x = index, y = resid_std_AIC, s = 10, color = 'blue')
plt.axhline(y = -2, color = 'red')
plt.axhline(y = 2, color = 'red')
s.set_xlabel('Index', fontdict = {'weight':'bold', 'size':14})
s.set_ylabel('Studentized Residuals', fontdict = {'weight':'bold', 'size':14})
plt.show()

    # For BIC:
resid_std_BIC = model_BIC.outlier_test()['student_resid']

index = np.arange(1, len(resid_std_BIC)+1)

plt.figure(figsize = (10,6))
s = sns.scatterplot(x = index, y = resid_std_BIC, s = 10, color = 'blue')
plt.axhline(y = -2, color = 'red')
plt.axhline(y = 2, color = 'red')
s.set_xlabel('Index', fontdict = {'weight':'bold', 'size':14})
s.set_ylabel('Studentized Residuals', fontdict = {'weight':'bold', 'size':14})
plt.show()

    # For Adjusted R^2
resid_std_adj = model_adj.outlier_test()['student_resid']

index = np.arange(1, len(resid_std_adj)+1)

plt.figure(figsize = (10,6))
s = sns.scatterplot(x = index, y = resid_std_adj, s = 10, color = 'blue')
plt.axhline(y = -2, color = 'red')
plt.axhline(y = 2, color = 'red')
s.set_xlabel('Index', fontdict = {'weight':'bold', 'size':14})
s.set_ylabel('Studentized Residuals', fontdict = {'weight':'bold', 'size':14})
plt.show()

### Checking for Leverage values:

    # For AIC:
lev = model_AIC.get_influence().hat_matrix_diag
thrs = round(2*(len(best_subset_vars_AIC)+1)/935, 2)

plt.figure(figsize = (12,6))
plt.stem(lev)
plt.axhline(y = thrs, color = 'r')
plt.xlabel('Index', fontdict = {'weight':'bold', 'size':14})
plt.ylabel('Leverage values', fontdict = {'weight':'bold', 'size':14})
plt.show()

    # For Adjusted R^2
lev = model_adj.get_influence().hat_matrix_diag
thrs = round(2*(len(best_subset_vars_adj)+1)/935, 2)

```

```
plt.figure(figsize = (12,6))
plt.stem(lev)
plt.axhline(y = thrs, color = 'r')
plt.xlabel('Index', fontdict = {'weight':'bold', 'size':14})
plt.ylabel('Leverage values', fontdict = {'weight':'bold', 'size':14})
plt.show()

    # For BIC:
lev = model_BIC.get_influence().hat_matrix_diag
thrs = round(2*(len(best_subset_vars_BIC)+1)/935, 2)

plt.figure(figsize = (12,6))
plt.stem(lev)
plt.axhline(y = thrs, color = 'r')
plt.xlabel('Index', fontdict = {'weight':'bold', 'size':14})
plt.ylabel('Leverage values', fontdict = {'weight':'bold', 'size':14})
plt.show()


### Cook's Distance

    ## For AIC:
cooks_AIC = model_AIC.get_influence().cooks_distance[0]
p = len(X_train_dumm_AIC.columns)

index = np.arange(1, len(resid_AIC)+1)

plt.figure(figsize = (10,6))
s = sns.scatterplot(x = index, y = cooks_AIC, s = 6, color = 'red')
s.set_xlabel('Index', fontdict = {'weight':'bold', 'size':14})
s.set_ylabel("Cook's Distance", fontdict = {'weight':'bold', 'size':14})
plt.show()

    # For BIC:
cooks_BIC = model_BIC.get_influence().cooks_distance[0]
p = len(X_train_dumm_BIC.columns)

index = np.arange(1, len(resid_BIC)+1)

plt.figure(figsize = (10,6))
s = sns.scatterplot(x = index, y = cooks_BIC, s = 6, color = 'red')
s.set_xlabel('Index', fontdict = {'weight':'bold', 'size':14})
s.set_ylabel("Cook's Distance", fontdict = {'weight':'bold', 'size':14})
plt.show()

c = pd.DataFrame(cooks_BIC).rename(columns = {0:'cooks'})
d = pd.concat([X_train_dumm_BIC.reset_index().drop('index', axis = 1),
           pd.DataFrame(Y_train).reset_index().drop('index', axis = 1), c],
           axis = 1)
d = d[d['cooks'] < 0.05]

# After update:
X_train_dumm_BIC = d.drop(['Y_new','cooks'], axis = 1)
Y_train_BIC = d['Y_new']

X = sm.add_constant(X_train_dumm_BIC)
Y = Y_train_BIC
model_BIC_new = sm.OLS(Y,X).fit()

cooks_BIC = model_BIC_new.get_influence().cooks_distance[0]
```

```
index = np.arange(1, len(resid_BIC))

plt.figure(figsize = (10,6))
s = sns.scatterplot(x = index, y = cooks_BIC, s = 6, color = 'red')
s.set_xlabel('Index', fontdict = {'weight':'bold', 'size':14})
s.set_ylabel("Cook's Distance", fontdict = {'weight':'bold', 'size':14})
plt.show()

    # For Adjusted R^2:
cooks_adj = model_adj.get_influence().cooks_distance[0]
p = len(X_train_dumm_adj.columns)

index = np.arange(1, len(resid_adj)+1)

plt.figure(figsize = (10,6))
s = sns.scatterplot(x = index, y = cooks_adj, s = 6, color = 'red')
s.set_xlabel('Index', fontdict = {'weight':'bold', 'size':14})
s.set_ylabel("Cook's Distance", fontdict = {'weight':'bold', 'size':14})
plt.show()
#================================================================================================
# Variance Inflation Factor (VIF)
#================================================================================================
    ## For BIC:
vif = pd.DataFrame()
vif["Variable"] = X_train_dumm_BIC.columns
vif["VIF"] = [variance_inflation_factor(X_train_dumm_BIC.values, i)
              for i in range(X_train_dumm_BIC.shape[1])]

vif

    # For AIC:
# AIC
vif = pd.DataFrame()
vif["Variable"] = X_train_dumm_AIC.columns
vif["VIF"] = [variance_inflation_factor(X_train_dumm_AIC.values, i)
              for i in range(X_train_dumm_AIC.shape[1])]

vif

    ## For Adjusted R^2:
vif = pd.DataFrame()
vif["Variable"] = X_train_dumm_adj.columns
vif["VIF"] = [variance_inflation_factor(X_train_dumm_adj.values, i)
              for i in range(X_train_dumm_adj.shape[1])]

vif

# Update:
X_train_dumm_adj = X_train_dumm_adj.drop('Age', axis = 1)

vif = pd.DataFrame()
vif["Variable"] = X_train_dumm_adj.columns
vif["VIF"] = [variance_inflation_factor(X_train_dumm_adj.values, i)
              for i in range(X_train_dumm_adj.shape[1])]

vif

X = sm.add_constant(X_train_dumm_adj)
Y = Y_train
model_adj = sm.OLS(Y,X).fit()
```

```
#==========================================================================================
# Model Validation
#==========================================================================================
X_test_dumm = pd.get_dummies(X_test, drop_first = True)
X_test_dumm.head(5)

    ## MSPR
n = len(Y_test)

def mspr_calc(var):
    M = globals()['model_' + var[17:]]
    df_new = sm.add_constant(X_test_dumm[M.model.exog_names[1:]])
    yhat = M.predict(df_new)
    return round(np.sum((Y_test - yhat)**2)/n, 4)

print('For AIC: ' + str(mspr_calc('best_subset_vars_AIC')))
print('For BIC: ' + str(mspr_calc('best_subset_vars_BIC')))
print('For adj: ' + str(mspr_calc('best_subset_vars_adj')))

    # Printing the model parameters:
print(model_BIC.params)
print('_'*30)
print(model_BIC_new.params)
```

15 Summary Table for Final Models

Adjusted R^2

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Y_new   R-squared:                       0.759
Model:                            OLS   Adj. R-squared:                  0.757
Method:                 Least Squares   F-statistic:                     363.8
Date:                Thu, 02 May 2024   Prob (F-statistic):          1.10e-279
Time:                        14:48:37   Log-Likelihood:                -942.34
No. Observations:                 935   AIC:                             1903.
Df Residuals:                     926   BIC:                             1946.
Df Model:                           8
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const               -9.4702      0.675    -14.038      0.000     -10.794      -8.146
Age                 15.0557      0.482     31.212      0.000      14.109      16.002
Bmi               -129.5855     21.297     -6.085      0.000    -171.381     -87.790
Children             0.1101      0.018      6.012      0.000       0.074       0.146
Sex_male            -0.1013      0.044     -2.313      0.021      -0.187      -0.015
Smoker_yes           2.3225      0.054     42.910      0.000       2.216       2.429
Region_northwest    -0.0826      0.063     -1.314      0.189      -0.206       0.041
Region_southeast    -0.1855      0.064     -2.918      0.004      -0.310      -0.061
Region_southwest    -0.1642      0.063     -2.593      0.010      -0.289      -0.040
==============================================================================
Omnibus:                      353.982   Durbin-Watson:                   1.927
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1275.302
Skew:                           1.826   Prob(JB):                    1.18e-277
Kurtosis:                       7.405   Cond. No.                     2.20e+03
==============================================================================
```

Figure 31: **For Training Data (Before removing 'Age')**

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Y_new   R-squared:                       0.505
Model:                            OLS   Adj. R-squared:                  0.501
Method:                 Least Squares   F-statistic:                     135.0
Date:                Thu, 02 May 2024   Prob (F-statistic):          9.32e-137
Time:                        16:06:59   Log-Likelihood:                -1278.4
No. Observations:                 935   AIC:                             2573.
Df Residuals:                     927   BIC:                             2611.
Df Model:                           7
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const               11.4071      0.126     90.775      0.000      11.160      11.654
Bmi               -222.5775     30.191     -7.372      0.000    -281.828    -163.327
Children             0.1722      0.026      6.603      0.000       0.121       0.223
Sex_male            -0.1100      0.063     -1.754      0.080      -0.233       0.013
Smoker_yes           2.2500      0.077     29.063      0.000       2.098       2.402
Region_northwest    -0.1411      0.090     -1.568      0.117      -0.318       0.035
Region_southeast    -0.3334      0.091     -3.674      0.000      -0.512      -0.155
Region_southwest    -0.2131      0.091     -2.351      0.019      -0.391      -0.035
==============================================================================
Omnibus:                        0.873   Durbin-Watson:                   1.956
Prob(Omnibus):                  0.646   Jarque-Bera (JB):                0.928
Skew:                           0.013   Prob(JB):                        0.629
Kurtosis:                       2.848   Cond. No.                     1.82e+03
==============================================================================
```

Figure 32: **For Training Data (After removing ‘Age’)**

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Y_new   R-squared:                       0.525
Model:                            OLS   Adj. R-squared:                  0.516
Method:                 Least Squares   F-statistic:                     62.12
Date:                Thu, 02 May 2024   Prob (F-statistic):           8.02e-60
Time:                        16:07:00   Log-Likelihood:                -554.67
No. Observations:                 402   AIC:                             1125.
Df Residuals:                     394   BIC:                             1157.
Df Model:                           7
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const               11.2760      0.180     62.552      0.000      10.922      11.630
Bmi               -173.1094     43.607     -3.970      0.000    -258.841     -87.378
Children             0.1651      0.040      4.118      0.000       0.086       0.244
Sex_male            -0.1964      0.099     -1.976      0.049      -0.392      -0.001
Smoker_yes           2.4431      0.123     19.884      0.000       2.202       2.685
Region_northwest    -0.0549      0.141     -0.390      0.697      -0.332       0.222
Region_southeast    -0.2234      0.136     -1.648      0.100      -0.490       0.043
Region_southwest    -0.2191      0.139     -1.577      0.116      -0.492       0.054
==============================================================================
Omnibus:                        5.259   Durbin-Watson:                   1.738
Prob(Omnibus):                  0.072   Jarque-Bera (JB):                3.796
Skew:                          -0.095   Prob(JB):                        0.150
Kurtosis:                       2.564   Cond. No.                     1.71e+03
==============================================================================
```

Figure 33: **For Testing Data**

AIC

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Y_new   R-squared:                       0.758
Model:                            OLS   Adj. R-squared:                  0.756
Method:                 Least Squares   F-statistic:                     415.2
Date:                Thu, 02 May 2024   Prob (F-statistic):          1.23e-280
Time:                        14:44:59   Log-Likelihood:                -943.21
No. Observations:                 935   AIC:                             1902.
Df Residuals:                     927   BIC:                             1941.
Df Model:                           7
Covariance Type:            nonrobust
=====================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------
const                -9.5426      0.673    -14.187      0.000     -10.863      -8.222
Age                  15.0746      0.482     31.252      0.000      14.128      16.021
Bmi                -128.2363     21.280     -6.026      0.000    -169.999     -86.473
Children              0.1100      0.018      6.002      0.000       0.074       0.146
Sex_male             -0.1030      0.044     -2.352      0.019      -0.189      -0.017
Smoker_yes            2.3253      0.054     42.979      0.000       2.219       2.431
Region_southeast     -0.1420      0.054     -2.615      0.009      -0.249      -0.035
Region_southwest     -0.1212      0.054     -2.235      0.026      -0.228      -0.015
==============================================================================
Omnibus:                      351.154   Durbin-Watson:                   1.928
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1250.779
Skew:                           1.815   Prob(JB):                    2.49e-272
Kurtosis:                       7.351   Cond. No.                     2.18e+03
==============================================================================
```

Figure 34: **For Training Data**

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Y_new   R-squared:                       0.800
Model:                            OLS   Adj. R-squared:                  0.796
Method:                 Least Squares   F-statistic:                     225.0
Date:                Thu, 02 May 2024   Prob (F-statistic):          2.07e-133
Time:                        15:10:04   Log-Likelihood:                -380.72
No. Observations:                 402   AIC:                             777.4
Df Residuals:                     394   BIC:                             809.4
Df Model:                           7
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
const              -10.7250      0.950    -11.290      0.000     -12.593      -8.857
Age                 15.9221      0.684     23.292      0.000      14.578      17.266
Bmi               -111.3008     28.408     -3.918      0.000    -167.151     -55.450
Children             0.1426      0.026      5.507      0.000       0.092       0.193
Sex_male            -0.0902      0.064     -1.405      0.161      -0.216       0.036
Smoker_yes           2.3833      0.080     29.893      0.000       2.227       2.540
Region_southeast    -0.2612      0.078     -3.349      0.001      -0.415      -0.108
Region_southwest    -0.2148      0.080     -2.686      0.008      -0.372      -0.058
==============================================================================
Omnibus:                      184.168   Durbin-Watson:                   1.834
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              792.381
Skew:                           2.032   Prob(JB):                    8.64e-173
Kurtosis:                       8.549   Cond. No.                     2.04e+03
==============================================================================
```

Figure 35: **For Testing Data**

BIC

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Y_new   R-squared:                       0.755
Model:                            OLS   Adj. R-squared:                  0.754
Method:                 Least Squares   F-statistic:                     715.0
Date:                Thu, 02 May 2024   Prob (F-statistic):          6.51e-282
Time:                        15:50:38   Log-Likelihood:                -950.04
No. Observations:                 935   AIC:                             1910.
Df Residuals:                     930   BIC:                             1934.
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -9.8213      0.671    -14.638      0.000     -11.138      -8.505
Age           15.1569      0.484     31.324      0.000      14.207      16.107
Bmi         -112.5968     20.699     -5.440      0.000    -153.220     -71.974
Children       0.1113      0.018      6.039      0.000       0.075       0.147
Smoker_yes     2.3122      0.054     42.676      0.000       2.206       2.418
==============================================================================
Omnibus:                      343.342   Durbin-Watson:                   1.927
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1213.530
Skew:                           1.772   Prob(JB):                    3.06e-264
Kurtosis:                       7.311   Cond. No.                     2.03e+03
==============================================================================
```

Figure 36: **For Training Data (Before removing Influential point)**

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Y_new   R-squared:                       0.760
Model:                            OLS   Adj. R-squared:                  0.759
Method:                 Least Squares   F-statistic:                     736.1
Date:                Thu, 02 May 2024   Prob (F-statistic):          3.41e-286
Time:                        15:51:36   Log-Likelihood:                -938.42
No. Observations:                 934   AIC:                             1887.
Df Residuals:                     929   BIC:                             1911.
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -9.8741      0.663    -14.883      0.000     -11.176      -8.572
Age           15.2108      0.479     31.785      0.000      14.272      16.150
Bmi         -122.1572     20.565     -5.940      0.000    -162.517     -81.797
Children       0.1136      0.018      6.231      0.000       0.078       0.149
Smoker_yes     2.3175      0.054     43.254      0.000       2.212       2.423
==============================================================================
Omnibus:                      335.806   Durbin-Watson:                   1.914
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1157.134
Skew:                           1.742   Prob(JB):                    5.39e-252
Kurtosis:                       7.194   Cond. No.                     2.04e+03
==============================================================================
```

Figure 37: **For Training Data (After removing Influential point)**

After removing the influential point, we get the final model which is provided above for the training dataset. Now, the summary table of the corresponding model for test data is given below:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Y_new   R-squared:                       0.791
Model:                            OLS   Adj. R-squared:                  0.789
Method:                 Least Squares   F-statistic:                     376.1
Date:                Thu, 02 May 2024   Prob (F-statistic):          1.46e-133
Time:                        15:11:24   Log-Likelihood:                -389.30
No. Observations:                 402   AIC:                             788.6
Df Residuals:                     397   BIC:                             808.6
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const        -10.9641      0.962    -11.398      0.000     -12.855      -9.073
Age           15.9276      0.694     22.957      0.000      14.564      17.292
Bmi          -84.1681     27.975     -3.009      0.003    -139.165     -29.171
Children       0.1368      0.026      5.215      0.000       0.085       0.188
Smoker_yes     2.3547      0.080     29.266      0.000       2.196       2.513
==============================================================================
Omnibus:                      174.503   Durbin-Watson:                   1.852
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              701.659
Skew:                           1.937   Prob(JB):                    4.33e-153
Kurtosis:                       8.184   Cond. No.                     1.91e+03
==============================================================================
```

Figure 38: **For Testing Data**