

**Intermediary Statistical Modeling for Analytics**

December 15, 2022

**Making prediction method based on dependent and independent variables in the scope of the health insurance system by using “Medical Insurance Forecasting”**

Project Report for: STAT 4600- Intermediate Statistics

Students: Bargavi Kongara – Shailja Sharma – Srivalli Adigarla – Prakhar Kumar Ahuja

Instructor: Prof. Jin Fang

Data: <https://www.kaggle.com/code/shrutidandagi/medical-insurance-forecast-by-linear-regression/data>

**Index**

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Titles** | **Page No.** |
| 1 | Introduction | 3 |
| 2 | Organizing and graphing data | 10 |
| 3 | Results and Analysis | 17 |
| 4 | Methods | 20 |
| 5 | Conclusion | 28 |
| 6 | Appendices | 30 |

**Abstract**

It is an undeniable fact that forecasting or prediction plays a vital role in the scope of the health insurance system. People’s healthcare cost forecasting is now a valuable tool for improving healthcare accountability. The healthcare sector produces a very large amount of data related to patients, diseases, and diagnosis, but since it has not been analyzed properly, it does not provide the significance which it holds along with the patient healthcare cost.

In addition, health insurance forecasting requires reliable data, information, and appropriate analytical tools for the prediction of specific health conditions or situations. In this project, “Medical Insurance Forecasting” will be analyzed and predicted by some specific tools and methods containing “Python” and “Regression”. At the end of this project, results and the method’s accuracy will be provided.

**Chapter 1**

**Introduction**

**1. The research problem**

For a health insurance company to make money, it needs to collect more in yearly premiums than it spends on medical care for its beneficiaries. It is crucial to forecast medical expenses for the insured population. Medical expenses are difficult to estimate because the costliest conditions are rare and seemingly random. Still, some conditions are more prevalent for certain segments of the population. We decided to research the effect of people's age, sex, BMI, Children, smoking, and region on medical expenses such as diagnosis, treatment and drug costs.

**1.1 Project Questions**

* Which factors like age, smoker, region, and number of children significantly affect the medical expenses?
* Can prediction methods like the logistic regression method be efficient in forecasting medical expenses?

# **2. Problem Importance**

To begin with, the purpose of this research problem is to predict medical expenses through variables shown in the data: age, sex, BMI, Children, smoking, and region – to increase accurate results for medical insurance companies. This research is related to finding the average medical care expenses and it is beneficial for insurance companies. For example, we’ve chosen the sum of people’s age in all regions and the sum of medical charges to generate a comparative line chart. We clearly see a trend that people who smoke are having more medical expenses. Also, the visualization result tells us age is also a driving factor to determine the premiums.

Additionally, by analyzing the correlation between a smoker and medical charges, we did further research on medical expenses. Meanwhile, the dataset also allows us to inspect the relationship between BMI and charges. This study helps medical insurance companies to predict medical expenses and decide the insurance premiums and therefore reduce losses.

The article (Forecasting Health expenses to U.S Medicare system, 2002) treats the uncertainty of long-term forecasts for health spending more systematically than in the past through a stochastic approach. Also, it uses the relation of health care spending to time until death to incorporate the changing health status of the population in the forecasts. Similarly, our project utilizes a dataset that establishes a dependency on different parameters like people's age, sex, BMI, Children, smoking, and the region on medical charges.

Furthermore, changes in healthcare expenditure appear quickly after changes in smoking behavior. A 10% relative drop in smoking in every state is predicted to be followed by an expected $63 billion reduction (in 2012 US dollars) in healthcare expenditure the next year. (Smoking Behavior and Healthcare Expenditure in the United States,2016). In our project, the research also focuses on smoker’s expenditure on medical charges.

Lastly, predicting medical insurance costs is still a problem in the healthcare industry that needs to be investigated and improved. In this paper (A Computational Intelligence Approach for Predicting Medical Insurance Cost, 2021), by using a set of ML algorithms, a computational intelligence approach is applied to predict healthcare insurance costs. We will also be doing regression analysis on our dataset for various variables to determine the medical expenses. This will help the insurance companies to decide their premium charges.

# **3.** **Description of the dataset**

The type of dataset that will be used in this project is CSV format.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Set Characteristics:** | Multivariate | **Number of Instances:** | 1338 | **Area:** | Healthcare |
| **Attribute Characteristics:** | Categorical, Integer | **Number of Attributes:** | 7 | **Duplicates?** | No |
| **Associated Tasks:** | Classification | **Missing Values?** | No | **Number of Views** | 1007684 |

The data set is publicly available on the Kaggle website,

The dataset contains charges, age, BMI, sex, children, and smoking habits of people in 4 regions. There are no missing values and duplicates in the dataset. Hence, we decided to consider the entire data set which has 1338 rows and 7 columns.

The classification goal is to predict medical expenses. The data set provides the members who are enrolled in health insurance plans. Each attribute is a potential factor. There are demographic, personal, and financial factors.

## 3.1. **Dataset main information**

The data set which will be analyzed is collected from health insurance companies. In the present data set, some clinical features are collected to have a better approximation. Data set clinical features are listed below:

<https://www.kaggle.com/code/shrutidandagi/medical-insurance-forecast-by-linear-regression/data>

|  |  |  |  |
| --- | --- | --- | --- |
| **Variables** | **Description** | **Values/Range** | **Type** |
| age | age of beneficiary | 18 - 64 | int64 |
| Sex | gender of beneficiary | Male, Female | object |
| BMI | Body mass index, providing an understanding of the body, weights that are relatively high or low relative to height, objective index of body weight (kg / m ^ 2) using the ratio of height to weight | 15.96 - 53.13 | float64 |
| children | Number of children covered by health insurance / Number of dependents | 0 - 5 | int64 |
| smoker | the smoking habit of the beneficiary | yes, no | object |
| region | the beneficiary's residential area in the US, northeast, southeast, southwest, northwest | northwest, northeast,   southeast, southwest | object |
| charges | Medical Insurance Cost | 1121.8739 - 63770.42801 | float64 |

# **4**. **Analysis and modeling methods**

In this project, we are going to implement a recommender system using logistic regression and Random Forecast Regression to estimate the medical charges. Using the dataset, which can be found at <https://www.kaggle.com/code/shrutidandagi/medical-insurance-forecast-by-linear-regression/data>. All the data are contained in multiple comma-separated values (CSV) files. The aim is to find out the factors which are affecting the medical charges. In this project, correlation, logistic regression, and random forest regression analysis will be done. Different evaluation metrics are computed to assess the results with minimal error. The methodology of the proposed system is carried out in stages which include dataset processing and performance evaluation of classifiers.

# **5. Software**

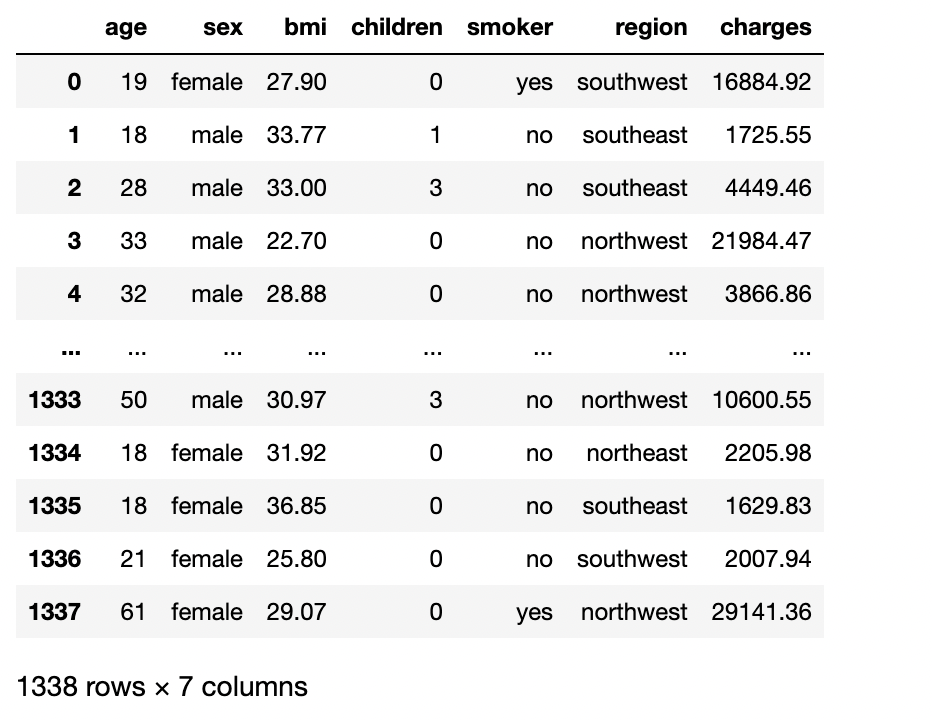
In this project, we will use Python 3 and Jupyter Notebook to explore the dataset and build the model.

# **6. Summary and Descriptive Statistics**

In this section, we explore the dataset and implement descriptive analysis to discover the properties of the data and have a better understanding of it.

## **6.1. Number of Observations (Figure 1)**

This data set includes data for 1338 patients, including male and female. For each observation, 7 different clinical features, including age, sex, number of children, region etc., is considered.

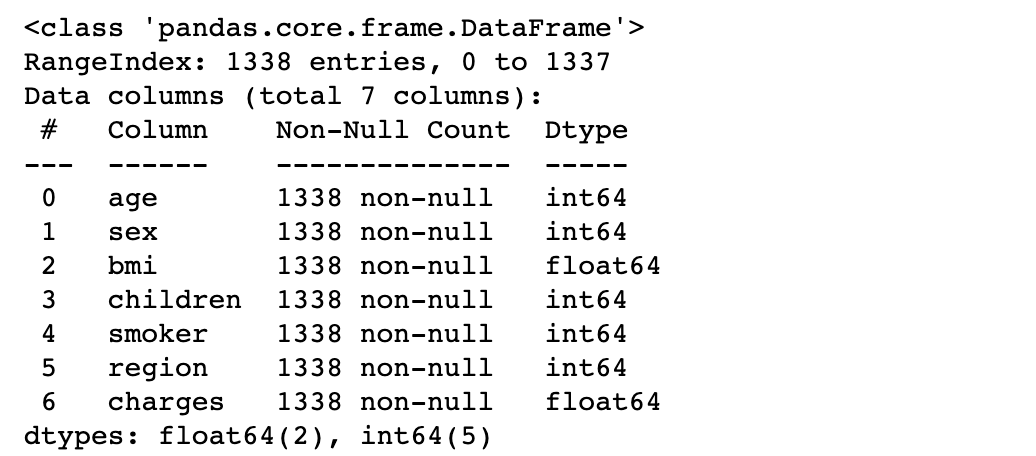


*Figure 1. Number of observations*

The dataset shape is (1338,7) and shows that there is 1338 patients' information having 7 criteria or columns. The type of features is mentioned below.

## **6.2 Type of features**

Each feature considered in our data set has a specific type, including “float”, “integer”, and “object” For more clarification, the image of the code in Jupyter notebook is shown below:

**

*Figure 2. Variables Type*

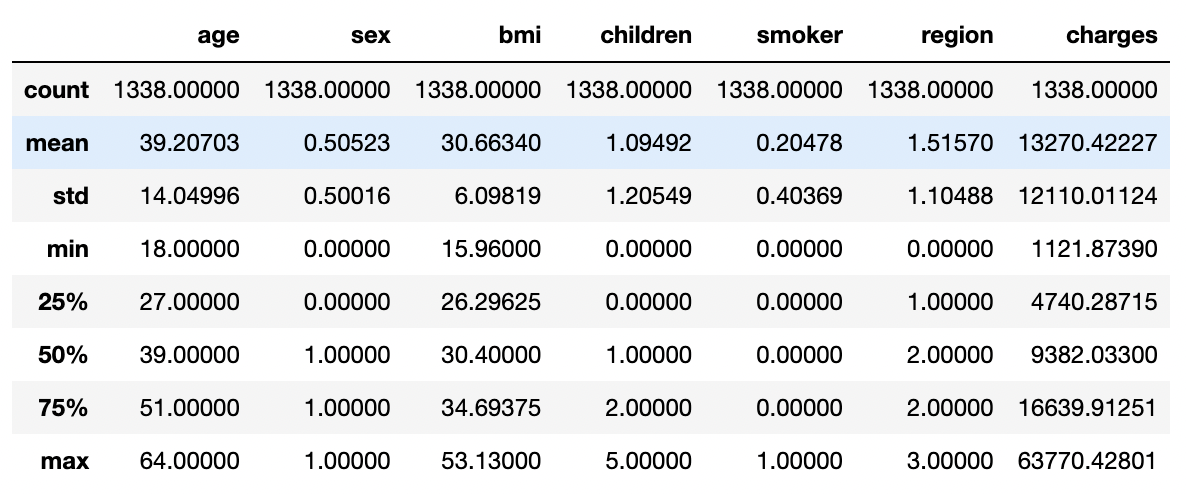
## 6.3 **Features statistical view**

Every feature has specific statistic attributes that are necessary for further analysis and making clear prediction model.

### 6.3.1 **Sample statistical Measures (Figure 3)**

Some of these statistical attributes are calculated for numerical variables and shown as follow:

Moreover, it may be interesting to calculate the mean of numerical variables in each category.



*Figure 3. Statistical Measures*

**Chapter 2**

**Organizing and graphing data**

**2.1 Frequency distribution table:**

Frequency distribution table for the qualitative variables in the dataset:

Table 1 shows the frequency distribution of sex in the dataset. The number of women is 662 and the rest of them are men.

|  |  |
| --- | --- |
| Sex | Frequency |
| Male | 676 |
| Female | 662 |

*Table 1. Frequency Distribution of Sex*

Table 2 shows the frequency distribution of two classes of smoker

|  |  |
| --- | --- |
| Smoker | Frequency |
| Yes | 274 |
| No | 1064 |

*Table 2. Frequency Distribution of Smoker*

Table 3 shows the frequency distribution of four classes of region

|  |  |
| --- | --- |
| Region | Frequency |
| northeast | 324 |
| northwest | 325 |
| southeast | 364 |
| southwest | 325 |

*Table 3. Frequency Distribution of Region*

**2.1.1 Relative frequency and percentage of the qualitative variables.**

|  |  |  |
| --- | --- | --- |
| Sex | Relative Frequency | Percentage |
| Female | 0.49 | 49 |
| Male | 0.51 | 51 |
| sum | 1 | 100 |

|  |  |  |
| --- | --- | --- |
| Smoker | Relative Frequency | Percentage |
| Yes | 0.20 | 20 |
| No | 0.80 | 80 |
| sum | 1 | 100 |

|  |  |  |
| --- | --- | --- |
| Region | Relative Frequency | Percentage |
| northeast | **0.24** | 24 |
| northwest | **0.25** | 25 |
| southeast | **0.27** | 27 |
| southwest | **0.24** | 24 |
| sum | 1 | 100 |

*Table 4. Relative Frequency and percentage of Qualitative Variables.*

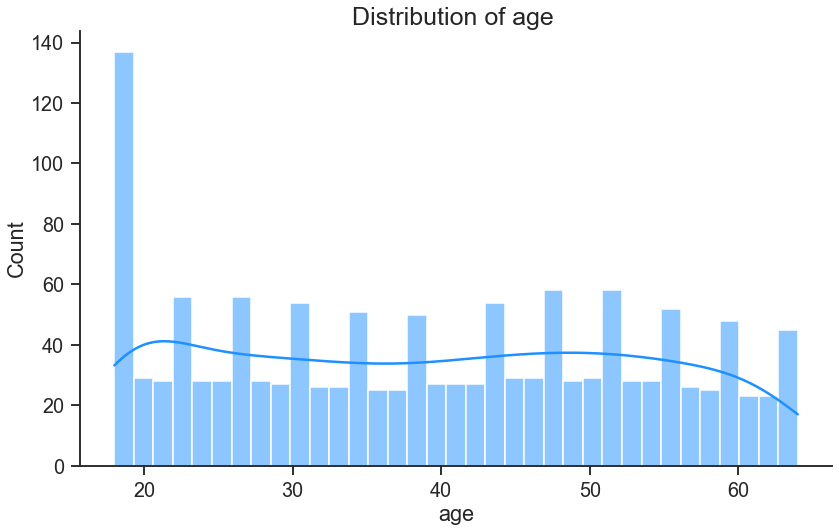
**2.2 Features graphical view**

Using matplotlib and pandas libraries, we try to draw different plots for the dataset variables to see their distributions and their relationship.

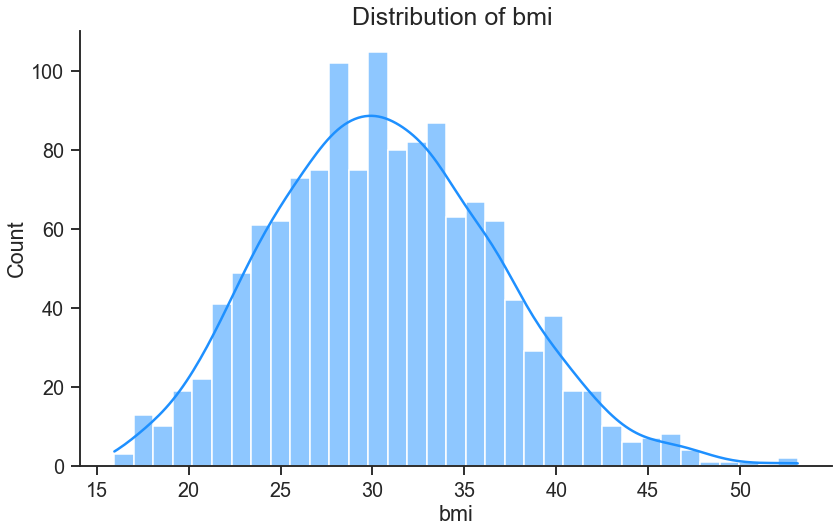
**2.2.1 Frequency distribution graph for the quantitative variables in the dataset.**

Graph 1,2,3,4 represents the frequency distribution of each quantitative variable in the dataset.

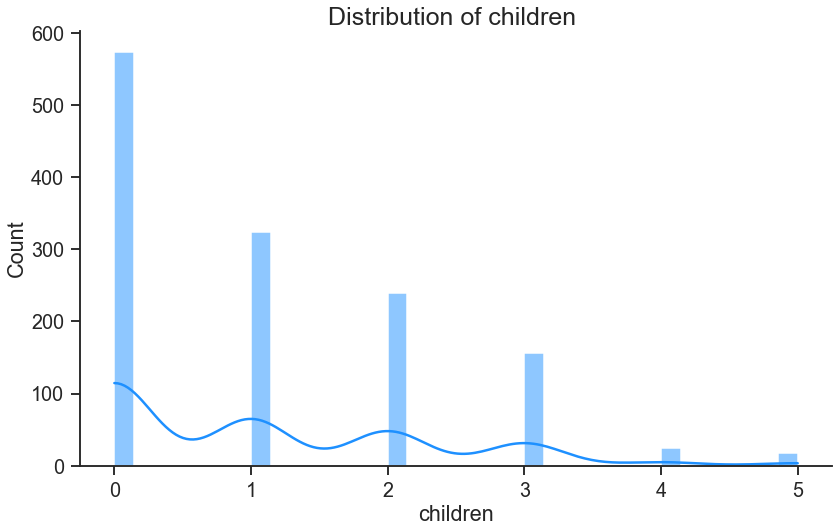
Age: min = 18, max = 64



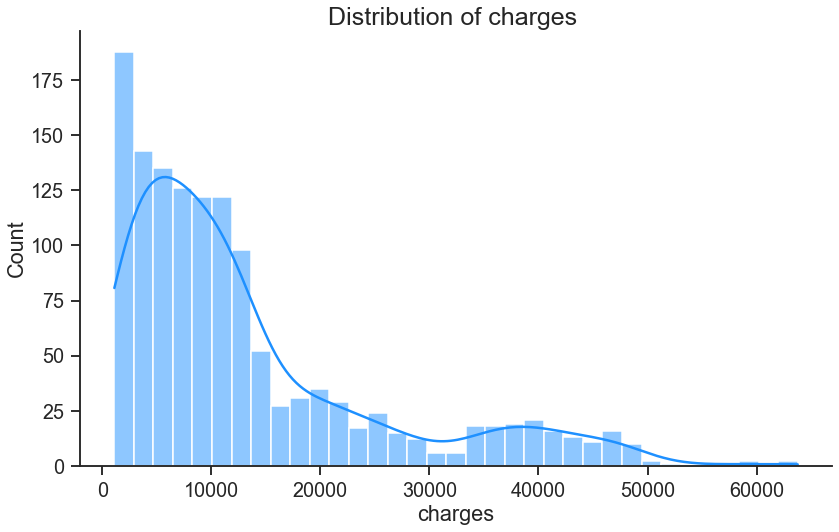
*Graph 1. Frequency Distribution of Age*



*Graph 2. Frequency Distribution of BMI*



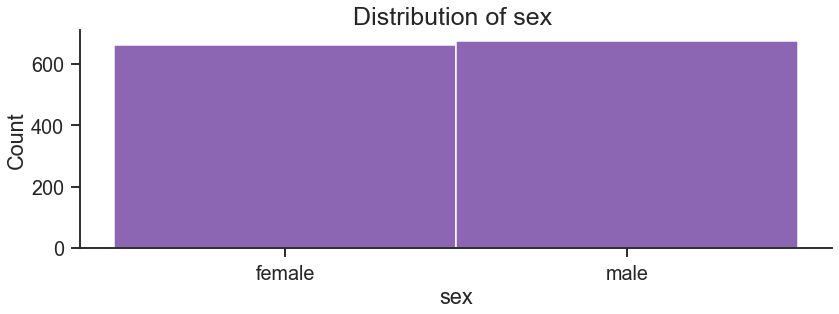
*Graph 3. Frequency Distribution of Children*



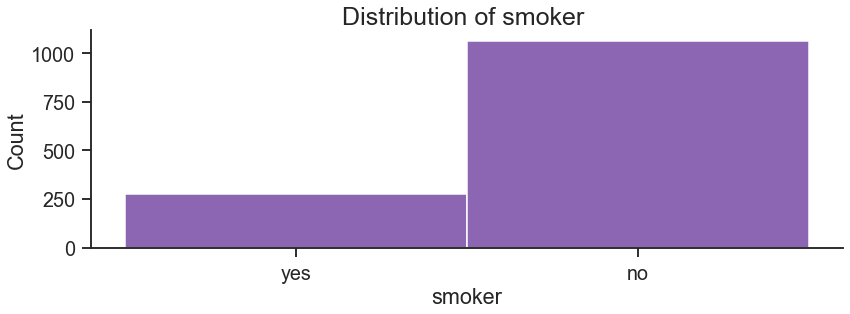
*Graph 4. Frequency Distribution of Charges*

**2.2.2 Frequency distribution graph for the qualitative variables in the dataset.**

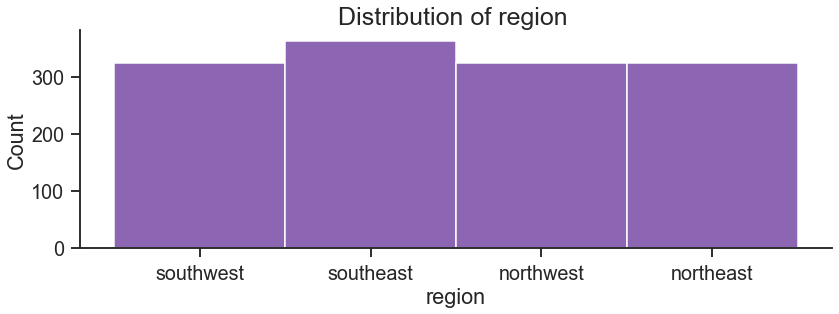
Graphs 5,6,7 represent the frequency distribution of each qualitative variable in the dataset.



*Graph 5. Frequency Distribution of Sex*

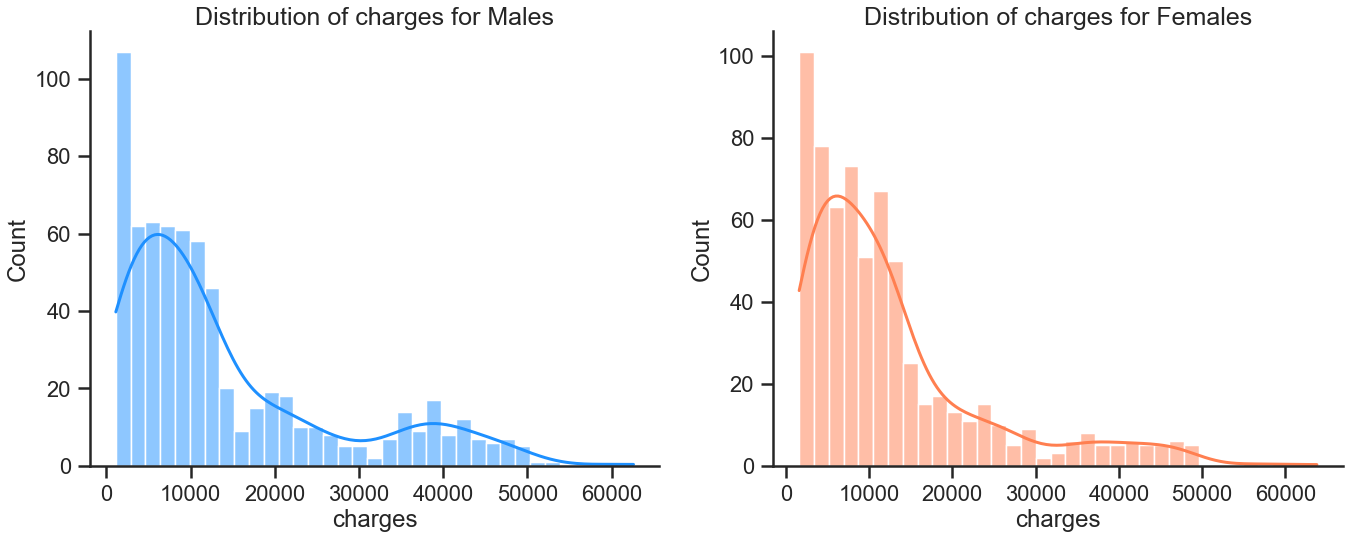


*Graph 6. Frequency Distribution of Smoker*

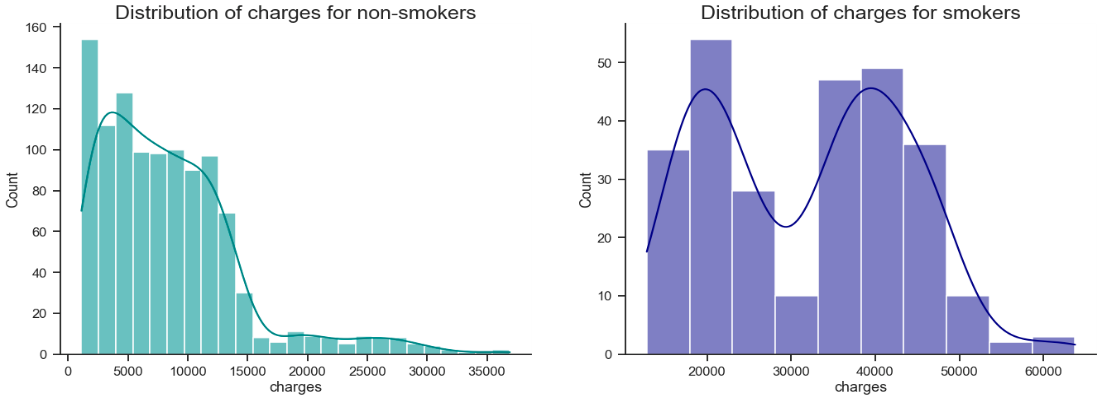


*Graph 7. Frequency Distribution of Region*

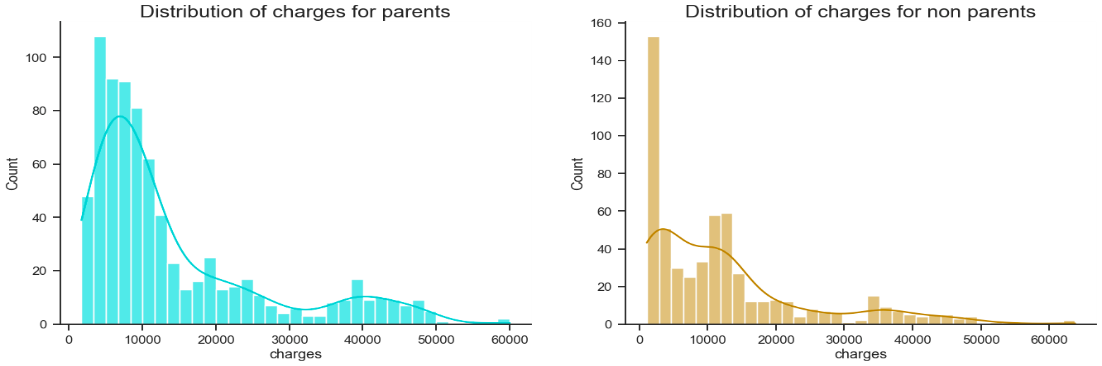
Using matplotlib and pandas libraries, we try to draw different plots for the dataset variables to see their distributions and their relationship. Graphs 8,9,10,11 contain the bar charts of variables in the dataset.



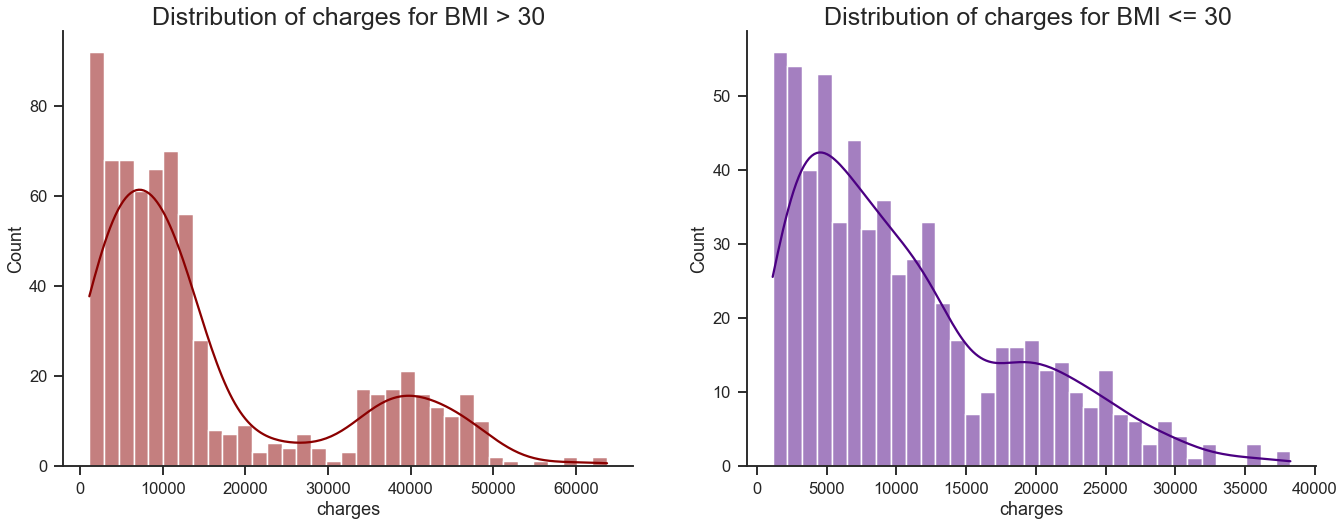
*Graph 8. Frequency Distribution of Charges v/s Sex*



*Graph 9. Frequency Distribution of charges v/s smokers and non-smokers*



*Graph 10. Frequency Distribution of charges v/s parents, non-parent*



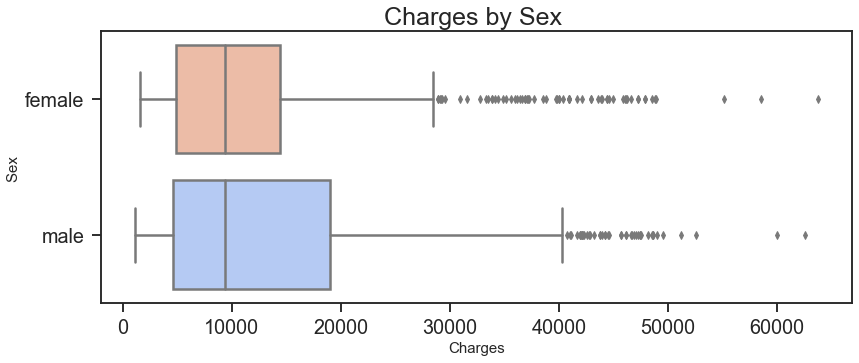
*Graph 11. Frequency Distribution of Charges v/s BMI*

**Chapter 3**

**Results and Analysis**

##### **3.1.1 Sex vs. Charges**

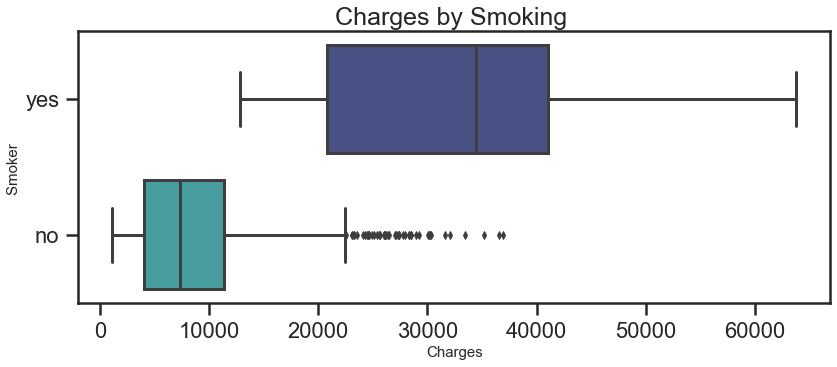
The medical insurance charges for the female gender are always greater than for the male as shown in Figure [4](https://www.hindawi.com/journals/mpe/2021/1162553/fig8/). It gives the sex types on the *y-axis* and the charges on the *x-axis.* The figure illustrates that the insurances charges for the female are 14000, and for the male, the charges are around 13000.



*Figure 4. Sex v/s Charges*

**3.1.2 Smoker vs charges**

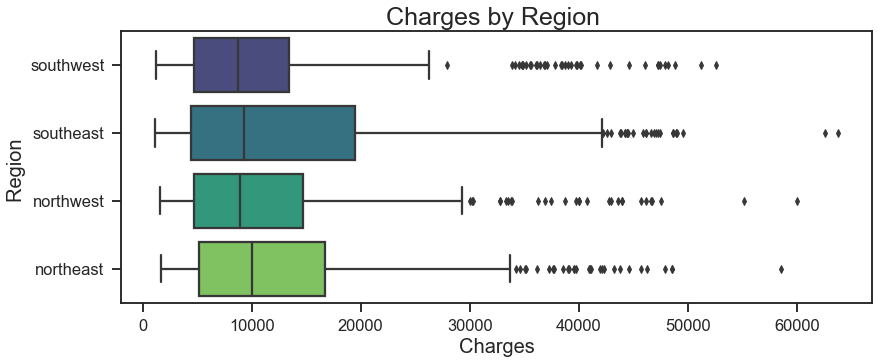
Figure [5](https://www.hindawi.com/journals/mpe/2021/1162553/fig7/) illustrates that as a normal smoker, the medical insurance cost varies slightly. However, men are more addicted and passionate to smoking as compared to women so the health insurance cost for females is greater as compared to the males. We can see in Figure [5](https://www.hindawi.com/journals/mpe/2021/1162553/fig7/) that with the increase of smoking habits, the insurance charges are going to be decreased for men and increased for women. Smokers’ values are shown on the *y-axis*, and charges are shown on the *x-axis*.



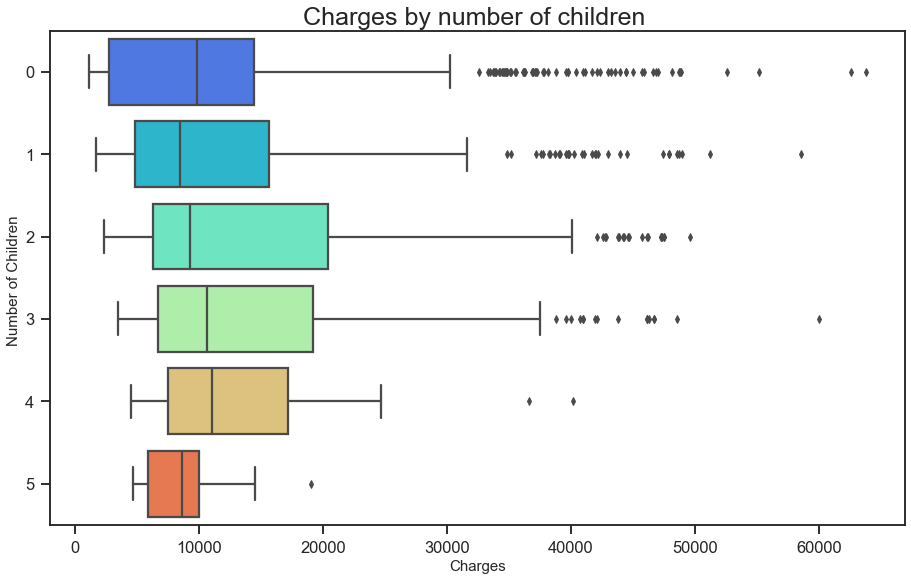
*Figure 5. Smokers v/s Charges*

##### **3.1.3 Region vs. Charges**

Insurance charges vary concerning certain regions as shown in Figure [6](https://www.hindawi.com/journals/mpe/2021/1162553/fig5/). The health insurance charges in the southeast are greater than in other regions. The region is displayed on the y*-axis*, and charges are shown on the x*-axis*. Figure 7 shows dependency o number of children over charges.



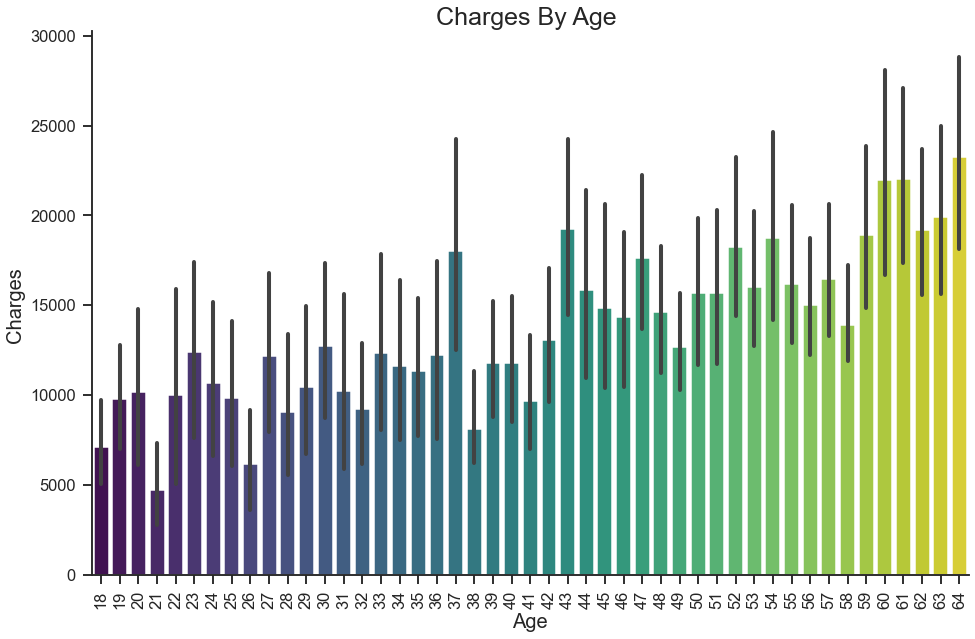
*Figure 6. Charges v/s Region*



*Figure 7. Charges v/s Children*

##### **3.1.4 Age vs. Charges**

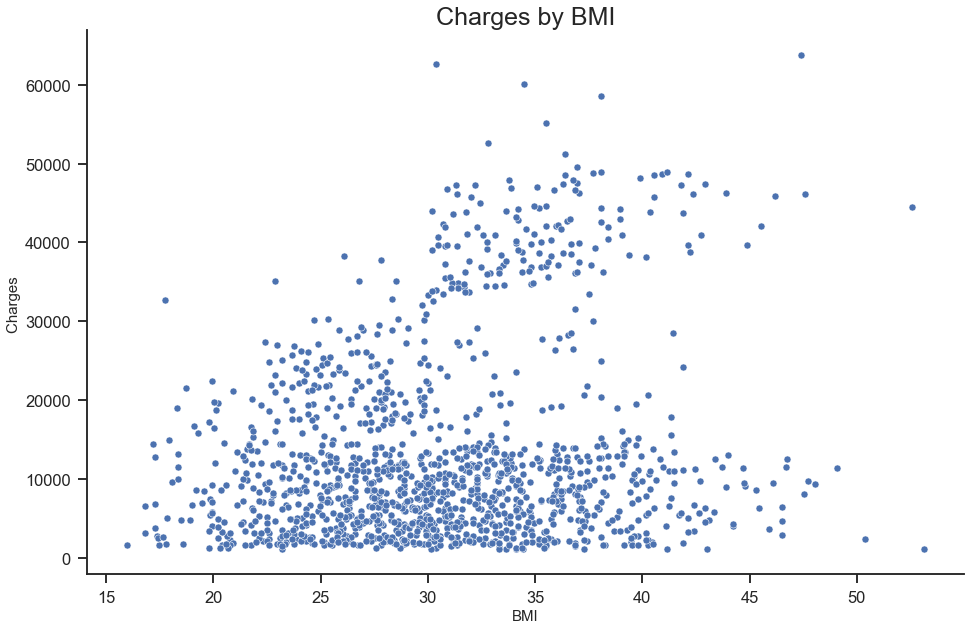
We can see in Figure 8 that with the growing age, the insurance charges are going to be increased. For example, when the age touches 64, the insurance charge is 23000, as shown in Figure 8. Age is shown on the *x-axis*, and charges are given on the *y-axis*.



*Figure 8. Charges v/s Age*

##### **3.1.5 BMI vs. Charges**

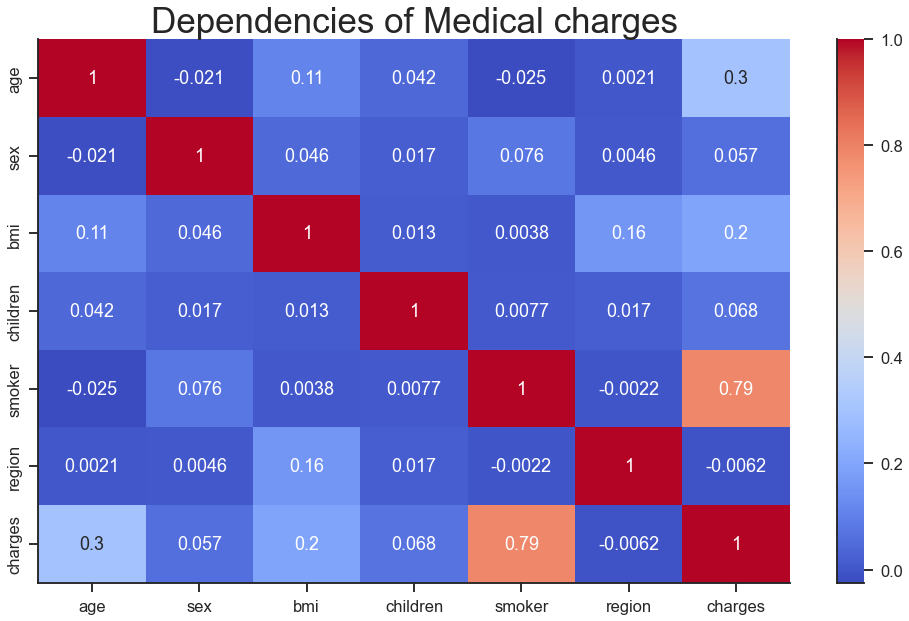
In Figure [9](https://www.hindawi.com/journals/mpe/2021/1162553/fig6/), the zero value is used to represent the females and one value is used for the males. The BMI values of sex or gender types (male and female) are given in the *x-axis*, and the charges are presented in the *y-axis*. It can be clearly seen that when the values of BMI are varied, the insurance charges will vary accordingly as shown in Figure [9](https://www.hindawi.com/journals/mpe/2021/1162553/fig6/).



*Figure 9. Charges v/s BMI*

##### **3.2 Feature Engineering and Correlation Matrix:**

When it comes to machine learning, feature engineering is the process of extracting features from raw data while applying domain expertise to improve the performance of ML algorithms. In the medical insurance cost dataset, attributes such as smoker, BMI, and age are the most important factors that determine charges. The heat map makes it easy to identify which features are most related to the other features or the target variable. Outcomes are shown in Figure [10.](https://www.hindawi.com/journals/mpe/2021/1162553/fig3/)

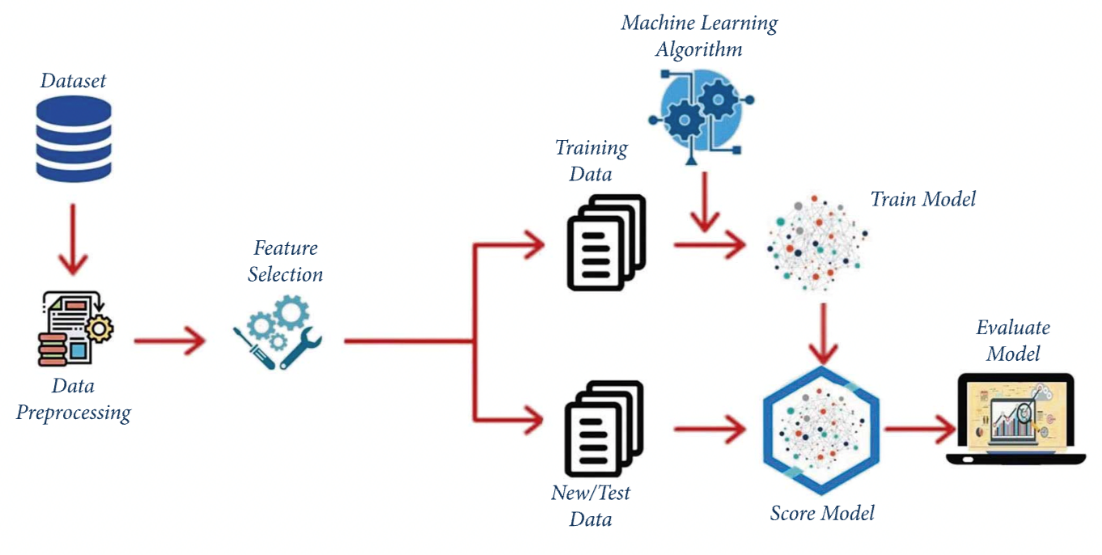


*Figure 10. Charges v/s BMI*

**Chapter 4**

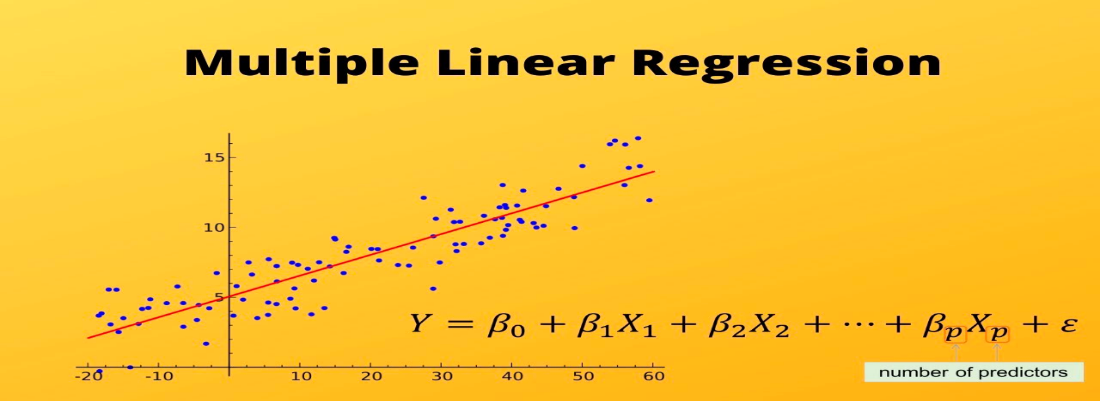
**Methods**

Machine learning techniques on medical insurance data have been performed. The medical insurance cost dataset is gained from KAGGLE’s repository and performed the data preprocessing. After preprocessing, selected the features by performing feature engineering. Then, the dataset is split into two parts: train, and test datasets; about 70% of the total data are used for training, while the rest is for testing. The training dataset is used to create a model that predicts medical insurance costs for the year, while the test dataset is used to evaluate the regression models. Figure 11 gives the basic knowledge of how the model will work.



*Figure 11. Model Flowchart*

4.1 **Multiple Linear Regression:** Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression is to model the linear relationship between the explanatory (independent) variables and response (dependent) variables. In essence, multiple regression is the extension of ordinary least-squares (OLS) regression because it involves more than one explanatory variable (Figure 12)

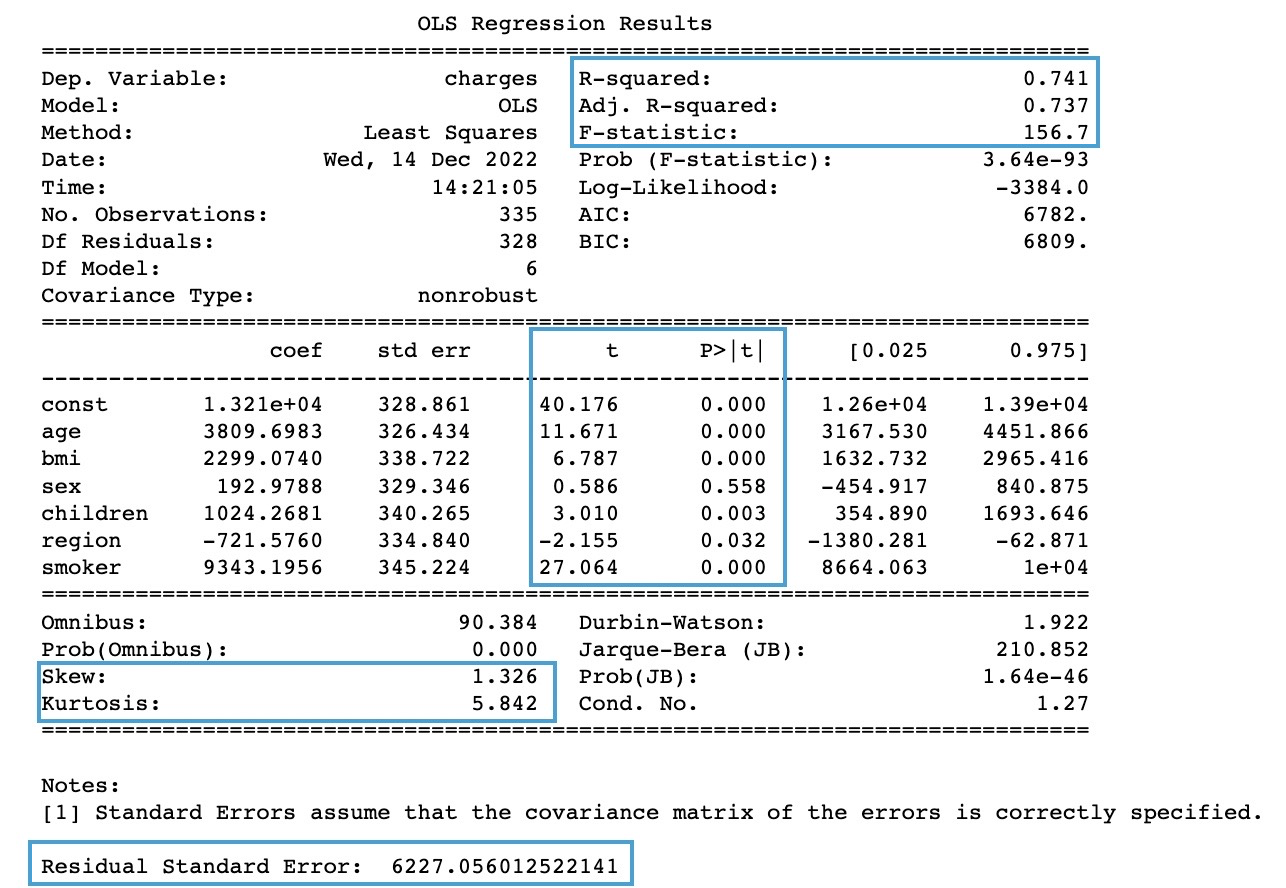


*Figure 12. Multiple Linear Regression*

**4.1.1 Dependent and Independent Variables:**

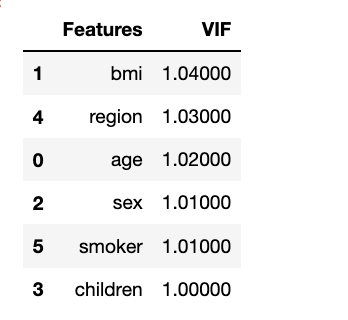
Independent Variables = sex, BMI, children, region, smoker, and age.

Another term should be defined, in statistics, least squares describe the variance in a prediction of the dependent variable as a function of the independent variable and the deviations from the fitted curve. Least Square method is executed in Figure 13.



*Figure 13. Regression Results*

**4.1.2 Variance Inflation Factor results (Figure 14):**



*Figure 14. VIF Results*

* A rule of thumb commonly used in practice is if a **VIF is > 10,** you have high multicollinearity. In our case, values are 1 and hence we are in good shape and can proceed with our regression.

**4.1.3 Interpretation:**

* **P-Values:**
  + All the **p values** are within the acceptable range. A low p-value (< 0.05) indicates that you can reject the null hypothesis. And all the p values we got are less than 0.05.
* **Multiple R-squared:**

0% indicates that the model explains none of the variability of the response data around its mean.100% indicates that the model explains all variability of the response data around its mean. Our model explains 74% of the variability of the response data around its mean. It does fit data well.

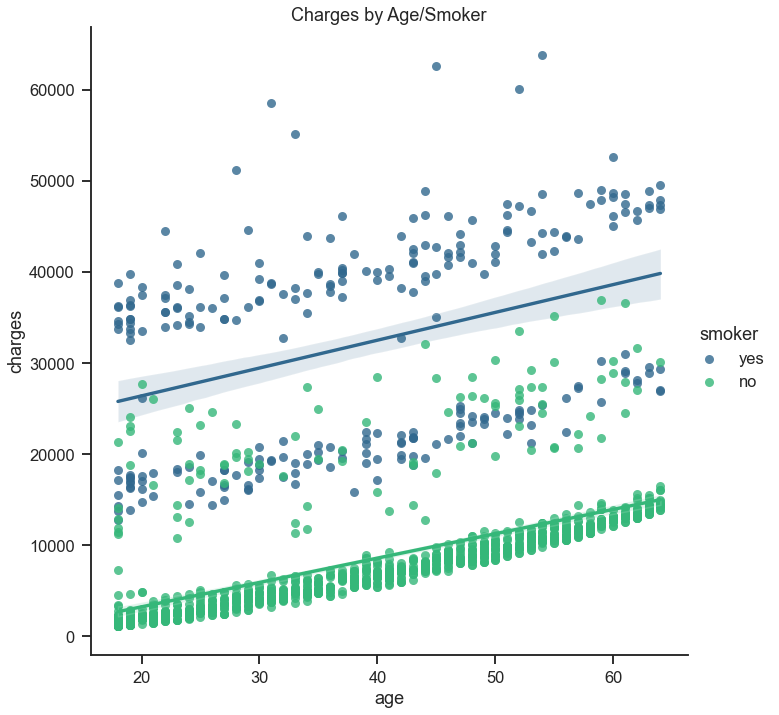
* **Adjusted R-squared:**

Also indicates how well terms fit a curve or line but adjusts for the number of terms in a model. If you add more useless variables to a model, then adjusted R-squared will decrease. If you add more useful variables, the adjusted R-squared will increase.

* **F-Statistics:**

Our model shows that F-statistic is 156.7 our p-value is so small, which is close to 0. This would lead us to reject H0 and conclude that there is strong evidence that a relationship does exist between charges and age, smoker.

* The **larger the t-value** is, the more confident we can be that the coefficient is not zero.
* The **Residual standard error** is the average amount that the observed values of y deviate from the true regression line. In general, we want the smallest residual standard error possible because that means our model's prediction line is close to the actual data values for our current model, we can see on average, the actual values are 6090.45 away from the predicted values (regression line)
* **Kurtosis** is 5.842 which means the distribution is longer or fatter, while Skewness is **1.326,** which means that data is highly skewed.



*Figure 15. Multiple Linear regression results*

**4.2 Random Forest Regression:** Random Forest Regression is a supervised learning algorithm that uses ensemble learning method for regression. Ensemble learning method is a technique that combines predictions from multiple machine learning algorithms to make a more accurate prediction than a single model

**How Random Forest Regression works:**

**Step 1**: The algorithm selects random samples from the dataset provided.

**Step 2:** The algorithm will create a decision tree for each sample selected. Then it will get a prediction result from each decision tree created. (Figure 16)

**Step 3:** Voting will then be performed for every predicted result. For a classification problem, it will use mode, and for a regression problem, it will use mean.

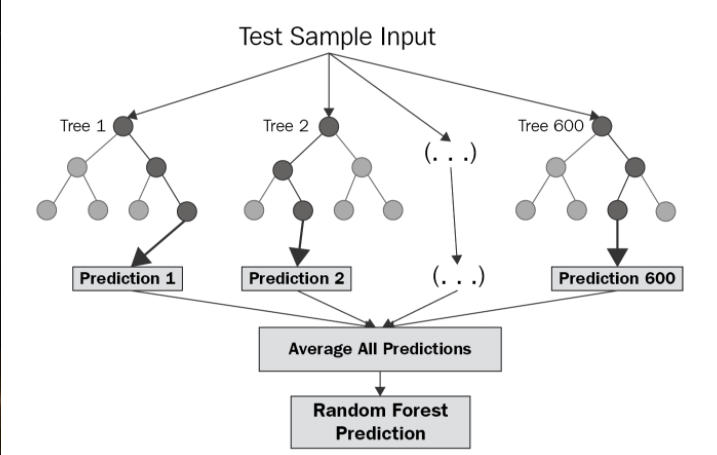
**Step 4**: And finally, the algorithm will select the most voted prediction result as the final prediction.

We will start by importing important packages that we will use to load the dataset and create a random forest classifier. We will use the scikit-learn library to load and use the random forest algorithm.

We create our random forest classifier and then train it on the train set. We will also pass the number of trees (20) in the forest, we want to use through theparameter called n\_estimators.

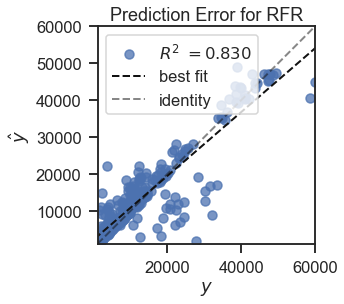
We check the accuracy using actual and predicted values from the test data.

Cross validation is applied to compare and select the best model. Learning curves are plots used to show a model's performance as the training set size increases.



*Figure 16. Random Forest Tree*

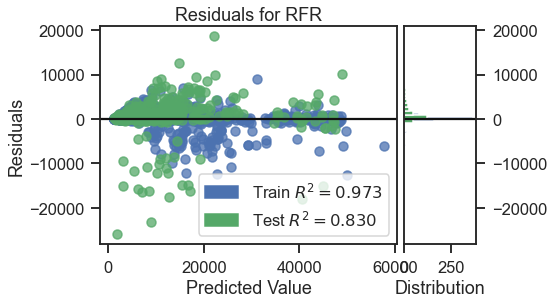
A prediction error plot in Figure 17 shows the actual targets from the dataset against the predicted values generated by our model. As observed from the above graph, R-Square value is 83% and this is the best fit model to predict the accurate medical charges.



*Figure 17. Prediction error for Random Forest Regressor*

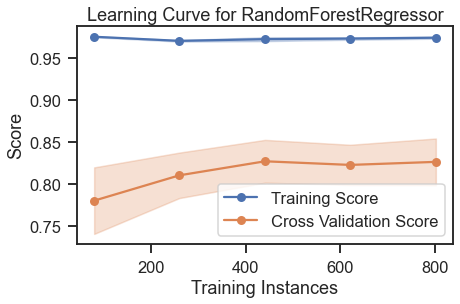
Residuals, in the context of regression models, are the difference between the observed value of the target variable (y) and the predicted value (ŷ), i.e., the error of the prediction. The residuals plot shows the difference between residuals on the vertical axis and the dependent variable on the horizontal axis, allowing you to detect regions within the target that may be susceptible to more or less error.

As observed from Figure 18, Train data R square value is 97% and Test data R squared value is 83% and it clearly states that there is not much difference between the observed value of target variable and the predicted value.



*Figure 18 Residuals for Random Forest Regressor*

A learning curve in Figure 19 shows the relationship of the training score versus the cross validated test score for an estimator with a varying number of training samples. The data is enough to accurately predict the charges as the training score curve is above 95%.



*Figure 19 Learning Curve for Random Forest Regressor*

**Chapter 5**

**Conclusion**

* From the regression analysis, we ﬁnd that region and gender do not bring signiﬁcant difference on charges.
* Age, BMI, number of children and smoking are the ones that drive the charges
* Smoking seems to have the most inﬂuence on the medical charges

In the final step, the multiple linear regression algorithm, random forest regression is implemented. We chose these models for analysis because the target variable is quantitative. The multiple linear regression model performed well with 74% accuracy and Random Forest Regression model gave the accuracy of 83%. All in all, we conclude that statistical algorithms such as Random Forest Regression performs well in prediction of medical expenses.

**Appendices**

***Appendix***

**Package Import**

import warnings

warnings.filterwarnings('ignore')

import pandas as pd

import numpy as np

import seaborn as sns

import lazypredict

import statsmodels.api as sm

import matplotlib.pyplot as plt

from sklearn.preprocessing import StandardScaler

from sklearn.model\_selection import train\_test\_split

from statsmodels.stats.outliers\_influence import variance\_inflation\_factor

from sklearn.metrics import mean\_squared\_error as mse # library for Mean squared error regression loss.

from sklearn import metrics #accuracy score,MAE,MSE,RMSQE

from yellowbrick.regressor import prediction\_error # Visualize the prediction error

from yellowbrick.regressor import residuals\_plot # Visualize the residuals between predicted and actual data

from yellowbrick.model\_selection import LearningCurve # Visualize Learning Curve

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score

plt.rcParams.update({'font.size': 30})

plt.style.use('fivethirtyeight')

sns.set(rc={'figure.figsize':(12,10)})

sns.set\_context("talk", font\_scale=1.2, rc={"lines.linewidth": 2.5})

sns.set\_style("ticks")

sns.despine()

%matplotlib inline

**Reading and display the data**

path = '/Users/bargavikongara/Documents/Clark/STAT4600-01-F22 - INTERMED STATS MODEL ANALYTICS/R Project/Medical Insurance\_LR copy.csv'

df = pd.read\_csv(path)

df

df.head()

#Checking missing values

df.isnull().sum()

**Getting all the unique values in each feature**

features = df.columns

for feature in features:

print(f"{feature} ---> {df[feature].nunique()}")

**EDA - Exploratory Data Analysis**

features = df.columns

for feature in features:

if df[feature].nunique() > 4:

plt.figure(figsize=(12,8))

sns.histplot(data = features, x = df[feature],

color = 'dodgerblue', bins = 35, kde = True,)

plt.title(f"Distribution of {feature} ",fontsize = 25)

sns.despine()

plt.show()

else:

plt.figure(figsize=(12,4))

sns.histplot(data = features, x = df[feature],

color = 'rebeccapurple',kde = False)

plt.title(f"Distribution of {feature} ",fontsize = 25)

sns.despine()

plt.show()

**Log distribution of charges:**

plt.figure(figsize=(12,8))

sns.histplot(data = df, x = 'charges', color = 'dodgerblue', kde = True, log\_scale=True)

plt.title("Log Distribution of Charges ",fontsize = 25)

sns.despine()

plt.show()

**Categorical Features Analysis**

#Sex Charges

gender = df.groupby('sex').agg(['mean','min','max'])['charges']

m = gender.at['female','mean']

f = gender.at['male','mean']

gender

(print(f'The mean charges for males is {m:.2f} and the mean charges for females is {f:.2f} dollars'))

**Plot to show charges v/s sex**

plt.figure(figsize=(12,5))

plt.title("Charges by Sex ", fontsize = 25)

sns.boxplot(data = df, x = 'charges', y = 'sex', palette = 'coolwarm\_r',orient = 'h')

plt.ylabel("Sex ", fontsize = 15)

plt.xlabel("Charges", fontsize = 15)

sns.set\_context("poster")

plt.show()

**Plot figure to show charges v/s sex**

f= plt.figure(figsize=(20,8))

ax=f.add\_subplot(121)

sns.histplot(df.query('sex == "male"'), x = 'charges', kde = True, color = 'dodgerblue', bins = 35)

ax.set\_title('Distribution of charges for Males', fontsize = 25, )

ax=f.add\_subplot(122)

sns.histplot(df.query('sex == "female"'), x = 'charges', kde = True, color = 'coral', bins = 35)

ax.set\_title('Distribution of charges for Females', fontsize = 25,)

sns.despine()

plt.show()

**Plot to show BMI v/s sex**

plt.figure(figsize=(12,4))

plt.title("BMI by Sex ",fontsize = 25)

sns.boxplot(data = df, x = 'bmi', y = 'sex', palette = 'coolwarm\_r',orient = 'h')

plt.ylabel("Sex ", fontsize = 15)

plt.xlabel("BMI", fontsize = 15)

plt.show()

**Plot to show gender count**

plt.title("Gender Counts",fontsize = 25)

sns.countplot(data = df, x = 'sex', palette = 'coolwarm\_r')

plt.ylabel("Count ", fontsize = 15)

plt.xlabel("Sex", fontsize = 15)

sns.despine()

plt.show()

**Smoker mean , min and max value:**

smokers = df.groupby('smoker').agg(['mean','min','max'])['charges']

smokers

**Plot to show smokers v/s charges**

plt.figure(figsize=(12,5))

plt.title("Charges by Smoking ",fontsize = 25)

sns.boxplot(data = df, x = 'charges', y = 'smoker', palette = 'mako', orient = 'h')

plt.ylabel("Smoker ", fontsize = 15)

plt.xlabel("Charges", fontsize = 15)

sns.set\_context("talk")

plt.show()

**Violin Plot to show charges v/s sex and charges v/s smoker**

f = plt.figure(figsize=(14,6))

ax = f.add\_subplot(121)

sns.violinplot(x='sex', y='charges',data=df,palette='Accent',ax=ax)

ax.set\_title('Violin plot of Charges vs Sex')

ax = f.add\_subplot(122)

sns.violinplot(x='smoker', y='charges',data=df,palette='inferno',ax=ax)

sns.despine()

ax.set\_title('Violin plot of Charges vs Smoker')

**Distribution of charges for smokers and non-smokers**

f= plt.figure(figsize=(20,8))

ax=f.add\_subplot(121)

sns.histplot(df.query('smoker == "no"'), x = 'charges', kde = True, color = 'teal')

ax.set\_title('Distribution of charges for non-smokers', fontsize = 25, )

ax=f.add\_subplot(122)

sns.histplot(df.query('smoker == "yes"'), x = 'charges', kde = True, color = 'navy',)

ax.set\_title('Distribution of charges for smokers', fontsize = 25, )

sns.despine()

plt.show()

**Count of Smokers:**

plt.figure(figsize=(8,6))

plt.title("Smoker Counts",fontsize = 25)

sns.countplot(data = df, x = 'smoker', palette = 'mako')

plt.ylabel("Count ", fontsize = 20)

plt.xlabel("Smoker", fontsize = 20)

sns.despine()

plt.show()

**Region mean, min and max values:**

region = df.groupby('region').agg(['mean','min','max'])['charges']

region

**Plot of Charges v/s region**

plt.figure(figsize=(12,5))

plt.title("Charges by Region ", fontsize = 25)

sns.boxplot(data = df, x = 'charges', y = 'region', palette = 'viridis',orient = 'h')

plt.ylabel("Region ", fontsize = 20)

plt.xlabel("Charges", fontsize = 20)

plt.show()

**Children mean, min and max values:**

kiddos = df.groupby('children').agg(['mean','min','max'])['charges']

kiddos

**Violin Plot of charges v/s no. of children**

plt.figure(figsize=(14,6))

plt.title('Violin Plot of Charges by Number of Children', fontsize = 25, );

sns.violinplot(x='children', y='charges',hue='sex',data=df,palette='rainbow',split=True)

plt.show()

**Bar Plot to show number of children v/s charges**

fig2 = plt.figure(figsize=(30,20))

fig2.add\_subplot(223)

plt.title("Charges by number of children ",fontsize = 25)

sns.boxplot(data = df, x = 'charges', y = 'children', palette = 'rainbow',orient = 'h')

plt.ylabel("Number of Children ", fontsize = 15)

plt.xlabel("Charges", fontsize = 15)

plt.show()

fig2.add\_subplot(224)

plt.title("Counts of Children", fontsize = 25)

sns.countplot(data = df, x = 'children', palette = 'rainbow',)

plt.ylabel("Count ", fontsize = 15)

plt.xlabel("Number of Children", fontsize = 15)

plt.show()

**Distribution of charges for parents and non-parents:**

f= plt.figure(figsize=(20,8))

ax=f.add\_subplot(121)

sns.histplot(df.query('children >=1'), x = 'charges', kde = True, color = 'darkturquoise',bins = 35)

ax.set\_title('Distribution of charges for parents', fontsize = 25)

ax=f.add\_subplot(122)

sns.histplot(df.query('children <= 0'), x = 'charges', kde = True, color = 'darkgoldenrod',bins = 35)

ax.set\_title('Distribution of charges for non parents', fontsize = 25)

sns.despine()

plt.show()

**Log distribution of charges for parents and non-parents**

f= plt.figure(figsize=(20,8))

ax=f.add\_subplot(121)

sns.histplot(df.query('children >=1'), x = 'charges', kde = True, color = 'darkturquoise',bins = 35,log\_scale = True)

ax.set\_title('Log Distribution of charges for parents', fontsize = 25,)

ax=f.add\_subplot(122)

sns.histplot(df.query('children <= 0'), x = 'charges', kde = True, color = 'darkgoldenrod',bins = 35,log\_scale = True)

ax.set\_title('Log Distribution of charges for non parents', fontsize = 25,)

sns.despine()

plt.show()

**Scatterplot of charges v/s BMI/Smoker**

plt.figure(figsize=(14,10))

plt.title("Charges by BMI/Smoking ",fontsize = 25)

sns.scatterplot(data = df, y = 'charges', x = 'bmi', hue = 'smoker',

palette = 'magma', s = 50)

plt.ylabel("Charges ", fontsize = 15)

plt.xlabel("BMI", fontsize = 15)

sns.despine()

plt.show()

sns.lmplot(x="bmi", y="charges", hue="smoker", data=df, palette = 'inferno', size = 10)

plt.show()

**Scatterplot of BMI v/s charges**

plt.figure(figsize=(14,10))

plt.title("Charges by BMI",fontsize = 25)

sns.scatterplot(data = df, y = 'charges', x = 'bmi',

palette = 'magma', s = 50)

plt.ylabel("Charges ", fontsize = 15)

plt.xlabel("BMI", fontsize = 15)

sns.despine()

plt.show()

sns.lmplot(x="bmi", y="charges", data=df, palette = 'inferno', size = 10)

plt.show()

**Histogram plot for BMI v/s charges**

plt.figure(figsize=(12,8))

plt.title("Distribution of BMI ", fontsize=25)

ax = sns.histplot(df["bmi"], color = 'indigo', kde = True,)

sns.despine()

**Histogram plot for BMI v/s charges . BMI < 30 and BMI>30**

f= plt.figure(figsize=(20,8))

ax=f.add\_subplot(121)

sns.histplot(df.query('bmi > 30'), x = 'charges', kde = True, color = 'darkred', bins = 35)

ax.set\_title('Distribution of charges for BMI > 30', fontsize = 25)

ax=f.add\_subplot(122)

plt.title("Charges by Age/Smoker ",fontsize = 25)

sns.histplot(df.query('bmi <= 30'), x = 'charges', kde = True, color = 'indigo', bins = 35)

ax.set\_title('Distribution of charges for BMI <= 30 ', fontsize = 25)

sns.despine()

plt.show()

**Scatterplot of charges v/s Age/Smoker**

plt.figure(figsize=(14,10))

plt.title("Charges by Age/Smoker ",fontsize = 25)

sns.scatterplot(data = df, y = 'charges', x = 'age', hue = 'smoker',

palette = 'viridis', s = 60)

plt.ylabel("Charges ", fontsize = 20)

plt.xlabel("Age", fontsize = 20)

sns.despine()

plt.show()

sns.lmplot(x="age", y="charges", hue="smoker", data=df, palette = 'viridis', size = 10).set(title='Charges by Age/Smoker')

plt.show()

**Barplot of Age v/s Charges**

plt.figure(figsize=(14,10))

plt.title("Charges By Age ", fontsize = 25)

sns.barplot(data = df, y = 'charges', x = 'age',

palette = 'viridis',)

plt.ylabel("Charges ", fontsize = 20)

plt.xlabel("Age", fontsize = 20)

sns.despine()

plt.xticks(rotation=90)

plt.show()

**Barplot of BMI v/s Age**

plt.figure(figsize=(14,10))

plt.title("BMI By Age", fontsize = 25)

sns.barplot(data = df, y = 'bmi', x = 'age',

palette = 'winter',)

plt.ylabel("BMI ", fontsize = 11)

plt.xlabel("Age", fontsize = 11)

sns.despine()

plt.xticks(rotation=90)

plt.show()

**Model Preparation:**

from sklearn.preprocessing import LabelEncoder

#sex

le = LabelEncoder()

le.fit(df.sex.drop\_duplicates())

df.sex = le.transform(df.sex)

# smoker or not

le.fit(df.smoker.drop\_duplicates())

df.smoker = le.transform(df.smoker)

#region

le.fit(df.region.drop\_duplicates())

df.region = le.transform(df.region)

**Correlation Matrix and Pairplot**

plt.figure(figsize=(16,10))

heatmap = sns.heatmap(df.corr()[['charges']].sort\_values(by='charges', ascending=False), annot=True, cmap='YlGnBu')

heatmap.set\_title('Features Correlating With Medical Charges', fontdict={'fontsize':35}, pad=16);

sns.despine()

plt.show()

plt.figure(figsize = (15, 10))

plt.title("Dependencies of Medical charges",fontsize = 35)

sns.heatmap(df.corr(), annot = True, cmap="coolwarm",cbar=True,

linewidths=0,linecolor='white',)

sns.despine()

plt.show()

df.corr()['charges'].sort\_values()

sns.pairplot(df)

sns.despine()

plt.show()

**Build Model:**

def build\_model(X,y):

"""Used to build linear regression model with stats models and print summary"""

X = sm.add\_constant(X) #Adding the constant

lm = sm.OLS(y,X).fit() # fitting the model

print(lm.summary()) # model summary

return lm

**Splitting data for test/train:**

X = df[['age', 'bmi', 'sex','children', 'region','smoker',]]

y = df['charges']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, train\_size=0.75, random\_state=1)

X\_train.to\_csv('x\_train.csv',index=False)

y\_train.to\_csv('y\_train.csv',index=False)

scaler = StandardScaler()

scaler.fit(X\_train)

X\_train = pd.DataFrame(scaler.transform(X\_train), index = X\_train.index, columns = X\_train.columns )

X\_test = pd.DataFrame(scaler.transform(X\_test), index = X\_test.index, columns = X\_test.columns )

len(X\_train)

len(X\_test)

len(y\_train)

len(y\_test)

X\_train.head()

**Create dataframe for VIF**

def checkVIF(X):

"""Creates dataframe of variance inflation factor results"""

vif = pd.DataFrame()

vif['Features'] = X.columns

vif['VIF'] = [variance\_inflation\_factor(X.values, i) for i in range(X.shape[1])]

vif['VIF'] = round(vif['VIF'], 2)

vif = vif.sort\_values(by = "VIF", ascending = False)

return(vif)

**Multiple Linear regression results:**

pd.options.display.float\_format = '{:.5f}'.format #To remove exponential values in the summary statistics

model1 = build\_model(X\_train,y\_train)

model = sm.OLS(y\_train, X\_train).fit()

print("\nResidual Standard Error: ", model.resid.std(ddof=X.shape[1]))

**To get the Residual Standard Error.**

model = sm.OLS(y\_train, X\_train).fit()

model.resid.std(ddof=X.shape[1])

model1

**After removing exponential values in summary statistics:**

model1 = build\_model(X\_test,y\_test)

model = sm.OLS(y\_test, X\_test).fit()

pd.options.display.float\_format = '{:.5f}'.format #To remove exponential values in the summary statistics

print("\nResidual Standard Error: ", model.resid.std(ddof=X.shape[1]))

**Improving model**

checkVIF(X\_train)

**RandomForestRegressor**

from sklearn.ensemble import RandomForestRegressor

regressor = RandomForestRegressor(n\_estimators=20, random\_state=0)

regressor.fit(X\_train, y\_train)

predictions = regressor.predict(X\_test)

print('Mean Absolute Error:', metrics.mean\_absolute\_error(y\_test, predictions))

print('Mean Squared Error:', metrics.mean\_squared\_error(y\_test, predictions))

print('Root Mean Squared Error:', np.sqrt(metrics.mean\_squared\_error(y\_test, predictions)))

print('Training Score:',regressor.score(X\_train, y\_train))

print('Testing Score:',regressor.score(X\_test, y\_test))

**Prediction error and Residuals for Random forest regressor:**

visualizer = prediction\_error(regressor, X\_train, y\_train, X\_test, y\_test,title = "Prediction Error for RFR")

viz = residuals\_plot(regressor, X\_train, y\_train, X\_test, y\_test,title = "Residuals for RFR")

**Learning Curve for Random Forest Regressor:**

visualizer = LearningCurve(regressor, scoring='r2')

visualizer.fit(X\_train, y\_train)

visualizer.show()