BMEN 411

Project 4: Exploratory Data Analysis (EDA)

Remy Bell, Anthony Gisolfi, John Paul, Emma Grace Pittard

December 08, 2023

Introduction

The volatile nature of healthcare spending trends during the pandemic has brought analysis of national healthcare spending and personal medical costs to the national spotlight. Globally, the United States is regarded as being one of the largest healthcare markets, with over 17% of its GDP being spent on healthcare alone[1]. Over 2019-2021, healthcare spending in the United States rose nearly 13.2% to $4.3 trillion. However, between 2020 and 2021 healthcare spending rose only 2.7% which is substantially lower than the 10.3% increase from 2019-2020[2,3]. This deceleration in healthcare spending is attributed to a decline in government-related expenditures offsetting increases in utilization of healthcare services that occurred due to pent-up demand from the pandemic[3,4]. Analysis of longer intervals of time shows that healthcare spending has increased from $74.1 billion in 1970 to $1.4 trillion by 2000 and to the aforementioned $4.3 trillion by 2021[3]. These national spending trends can be used to create a generalized model of how personal medical costs have varied over time. However, there are identifiable trends that separate the spending for certain demographics and populations living in different geographical regions of the United States[5].

Though broad generalizations of medical costs for the country as a whole are useful, oftentimes the ability to predict personal medical costs for specific demographics and geographical locations is more desirable. Such a model would primarily be for creating forecasts of future medical costs for insurance companies and for analysis of historical trends, but the model could also be used to aid patients in accurately estimating their medical costs throughout a given year. Similar models are already used in the medical industry to predict health insurance premiums[6]. To create this model, an appropriate data set that tracks medical bills and relevant parameters such as age, body mass index (BMI), sex, geographic region, number of dependents on insurance plans, and other healthcare-related parameters must be used. Therefore, to create this model, a medical costs dataset that tracks patient age, sex, BMI, number of children, whether or not they are a smoker, geographic region, and medical charges was used[7]. To create predictions based on the dataset, various models, such as decision trees, K-nearest neighbor, and multiple linear regression models can be used. These models will then be able to identify relationships between the patient parameters in the data set and the medical charges associated with each patient such that predictions of future medical costs for new patients can be created.

Project 4

2023-11-30

```
knitr::opts_chunk$set(fig.height = 4,warning = FALSE)

library(ggplot2)
library(caret)

## Loading required package: lattice

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────────────
tidyverse 2.0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.4
## ✔ forcats   1.0.0     ✔ stringr   1.5.1
## ✔ lubridate 1.9.3     ✔ tibble    3.2.1
## ✔ purrr     1.0.2     ✔ tidyr     1.3.0

## ── Conflicts ──────────────────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ✖ purrr::lift()   masks caret::lift()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors

library(psych)

##
## Attaching package: 'psych'
##
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha

library(Amelia)

## Loading required package: Rcpp
## ##
## ## Amelia II: Multiple Imputation
## ## (Version 1.8.1, built: 2022-11-18)
## ## Copyright (C) 2005-2023 James Honaker, Gary King and Matthew Blackwell
## ## Refer to http://gking.harvard.edu/amelia/ for more information
## ##
```

Part 0: Importing Data

We will begin the exploratory data analysis by importing the dataset. The data set chosen is named “Medical Cost Personal Datasets” and can be found on Kaggle.com. The data is imported twice, once as a numeric data set labeled “data” and once again as a Factor data set labeled “data.cat”. The children column is factored as it may seem numeric but is actually categorical in nature.

```r
## Numeric Dataset
data<-read.csv("insurance.csv",stringsAsFactors = TRUE,header=TRUE)
data$children <- factor(data$children)
##Factors Dataset
data.cat <-read.csv("insurance.csv",stringsAsFactors = FALSE,header=TRUE)
data.cat$children <- factor(data.cat$children)
```

Part 1: Data Description

The variables of the data analysis were: age, sex, body mass index (BMI), children, smokers, region, and charges. The age variable represented the age of the primary beneficiary. The sex was either M/F and represented the gender of the insurance contractor. The BMI variable provides an overall understanding of the patient’s body. The objective body weight index $(\frac{kg}{m^2})$ used the ratio of height to weight and ideally was between 18.5 and 24.9. The number of children covered by health insurance and/or the number of dependents was represented in the children variable. The smoker variable recorded patients actively smoking. The region was broken up into northeast, southeast, southwest, and northwest and represented the residential area of the beneficiary. Lastly, and likely the most important variable, the charges of the individual’s medical costs billed by health insurance were represented by the charges variable.

1.2 Explore Data Set and Variables

To begin the analysis of the data, we need to have an understanding of what our data looks like. Using the str(), summary(), and describe() functions, we can achieve this. The str()

function provides us with an overview of the data frame and the types of data in each column. For example, we can see that the age column consists of integer values and the smoker column is a factor with two levels that correspond to smoker/nonsmoker. The summary() function provides a basic summary of statistical data associated with the data set. This includes mean, median, and mode, as well as the count of non-numeric data. Finally, the describe() function provides a similar statistical output but also includes variance and standard deviation. These outputs provide us with a decent overview of the data we have imported and we can now proceed with analysis.

```
str(data)
```

```
## 'data.frame':    1338 obs. of  7 variables:
##  $ age     : int  19 18 28 33 32 31 46 37 37 60 ...
##  $ sex     : Factor w/ 2 levels "female","male": 1 2 2 2 2 1 1 1 2 1 ...
##  $ bmi     : num  27.9 33.8 33 22.7 28.9 ...
##  $ children: Factor w/ 6 levels "0","1","2","3",..: 1 2 4 1 1 1 2 4 3 1
...
##  $ smoker  : Factor w/ 2 levels "no","yes": 2 1 1 1 1 1 1 1 1 1 ...
##  $ region  : Factor w/ 4 levels "northeast","northwest",..: 4 3 3 2 2 3 3
2 1 2 ...
##  $ charges : num  16885 1726 4449 21984 3867 ...
```

```
summary(data)
```

```
##       age             sex           bmi        children smoker
##  Min.   :18.00   female:662   Min.   :15.96   0:574    no :1064
##  1st Qu.:27.00   male  :676   1st Qu.:26.30   1:324    yes: 274
##  Median :39.00                Median :30.40   2:240
##  Mean   :39.21                Mean   :30.66   3:157
##  3rd Qu.:51.00                3rd Qu.:34.69   4: 25
##  Max.   :64.00                Max.   :53.13   5: 18
##        region       charges
##  northeast:324   Min.   : 1122
##  northwest:325   1st Qu.: 4740
##  southeast:364   Median : 9382
##  southwest:325   Mean   :13270
##                  3rd Qu.:16640
##                  Max.   :63770
```

```
describe(data)
```

```
##           vars    n     mean       sd  median  trimmed     mad     min
max
## age          1 1338    39.21    14.05   39.00    39.01    17.79   18.00
```

```
64.00
## sex*         2 1338     1.51     0.50    2.00     1.51    0.00    1.00
2.00
## bmi          3 1338    30.66     6.10   30.40    30.50    6.20   15.96
53.13
## children*    4 1338     2.09     1.21    2.00     1.94    1.48    1.00
6.00
## smoker*      5 1338     1.20     0.40    1.00     1.13    0.00    1.00
2.00
## region*      6 1338     2.52     1.10    3.00     2.52    1.48    1.00
4.00
## charges      7 1338 13270.42 12110.01 9382.03 11076.02 7440.81 1121.87
63770.43
##              range  skew kurtosis     se
## age          46.00  0.06    -1.25   0.38
## sex*          1.00 -0.02    -2.00   0.01
## bmi          37.17  0.28    -0.06   0.17
## children*     5.00  0.94     0.19   0.03
## smoker*       1.00  1.46     0.14   0.01
## region*       3.00 -0.04    -1.33   0.03
## charges   62648.55  1.51     1.59 331.07
```

1.3 Data Manipulation

It appears that after the importation of our data, no variable needs to be adjusted or changed from numeric to factor. It should be noted however that the "children" column was changed to a factor when the data was imported. Although there are no variables that need to be changed, we need to manipulate the data differently. If we wish to predict the price of medical charges, we will experience more success and predictability if we split these prices into three categories. The first of these categories, deemed *low* prices, will account for total charges below $5,000. The second category, *intermediate* prices, will encompass charges between $5,001 and $15,000. The last category, *high* prices, will account for values above $15,001. These price values were chosen based on the quartile data of the *charges* column which will be observed later (Fig. 15). By splitting the data like this, we have given ourselves the ability to make a more predictable model.

```
breaks <- c(-Inf, 5000, 15000, Inf)

# Create a new column 'charge_group' based on the breaks
#data$charge_group <- cut(data$charge, breaks = breaks, labels = c("<
$5,000", "$5,001 - $15,000", "> $15,000"), include.lowest = TRUE)
data$charge_group <- cut(data$charge, breaks = breaks, labels = c("Low",
"Intermediate", "High"), include.lowest = TRUE)
data<- data.frame(data, stringsAsFactors = TRUE)
# View the resulting data frame
head(data)
```

```
##   age    sex    bmi children smoker    region   charges charge_group
## 1  19 female 27.900        0    yes southwest 16884.924         High
## 2  18   male 33.770        1     no southeast  1725.552          Low
## 3  28   male 33.000        3     no southeast  4449.462          Low
## 4  33   male 22.705        0     no northwest 21984.471         High
## 5  32   male 28.880        0     no northwest  3866.855          Low
## 6  31 female 25.740        0     no southeast  3756.622          Low
```

1.4 Missing Data

When performing an exploratory data analysis, it is of utmost importance to check for missing data within the data set. Any missing values can be detrimental to the results of the study and can hinder progress significantly. Below is the code conducted to sum up the missing values in the data set as well as a missing data map to visualize the gaps in the data set (Fig. 1). Note that our data set has no missing values and is therefore ready to be analyzed further.

```
sum(is.na(data))
```

```
## [1] 0
```

```
missmap(data)
```

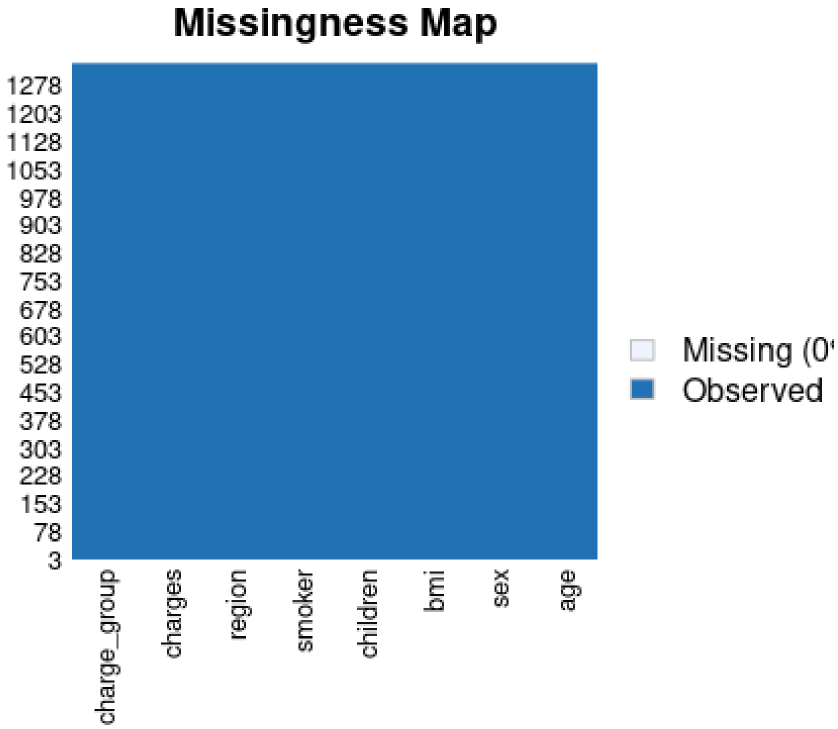

Figure 1. Missingness Map. Visualizes the missing data in the data set.

Part 2: Univariate Analysis

Now that we have preprocessed our data, we will now dive deeper into the individual variables in the data set. Although this may not be necessary, it will allow us to notice any uniqueness or possible trends among the variables.

2.1: Basic Structures

Here we utilize the summary() function once again for quick reference. We then begin to analyze the age variable through the determination of its mean, minimum value, and maximum values. This process is also completed for the other numeric variables: BMI and charges. Since these statistical values cannot be determined for categorical variables like sex, smoker, children, region, and the charge_group, we will provide a count of each variable’s levels using the count() function.

```
summary(data)

##       age            sex           bmi        children smoker
##  Min.   :18.00   female:662   Min.   :15.96   0:574    no :1064
##  1st Qu.:27.00   male  :676   1st Qu.:26.30   1:324    yes: 274
##  Median :39.00                Median :30.40   2:240
##  Mean   :39.21                Mean   :30.66   3:157
##  3rd Qu.:51.00                3rd Qu.:34.69   4: 25
##  Max.   :64.00                Max.   :53.13   5: 18
##        region       charges            charge_group
##  northeast:324   Min.   : 1122   Low         :359
##  northwest:325   1st Qu.: 4740   Intermediate:621
##  southeast:364   Median : 9382   High        :358
##  southwest:325   Mean   :13270
##                  3rd Qu.:16640
##                  Max.   :63770

## AGE
mean(data$age)

## [1] 39.20703

max(data$age)

## [1] 64

min(data$age)

## [1] 18

data.cat %>% count(data.cat$sex)

##   data.cat$sex   n
## 1       female 662
## 2         male 676

## BMI
max(data$bmi)

## [1] 53.13

min(data$bmi)

## [1] 15.96

mean(data$bmi)

## [1] 30.6634
## Children
data.cat %>% count(data.cat$children)
```

```
##   data.cat$children   n
## 1                 0 574
## 2                 1 324
## 3                 2 240
## 4                 3 157
## 5                 4  25
## 6                 5  18
```

```
## Smoker
data.cat %>% count(data.cat$smoker)
```

```
##   data.cat$smoker    n
## 1              no 1064
## 2             yes  274
```

```
## Region
data.cat %>% count(data.cat$region)
```

```
##   data.cat$region   n
## 1       northeast 324
## 2       northwest 325
## 3       southeast 364
## 4       southwest 325
```

```
## Charges
mean(data$charges)
```

```
## [1] 13270.42
```

```
max(data$charges)
```

```
## [1] 63770.43
```

```
min(data$charges)
```

```
## [1] 1121.874
```

2.2 Identify Outliers

When analyzing data, it is important to identify any outliers. To do so, we utilized box plots and histograms of the numeric data in the data set (Fig. 2-7). Outliers in the data are plotted in red in each boxplot. Note that the charges variable has a large amount of outliers which further justifies the splitting of the data into three price groups. By doing this outliers can be grouped and predicted separately.

```
summary(data)

##       age            sex           bmi        children smoker    
##  Min.   :18.00   female:662   Min.   :15.96   0:574    no :1064  
##  1st Qu.:27.00   male  :676   1st Qu.:26.30   1:324    yes: 274  
##  Median :39.00                Median :30.40   2:240              
##  Mean   :39.21                Mean   :30.66   3:157              
##  3rd Qu.:51.00                3rd Qu.:34.69   4: 25              
##  Max.   :64.00                Max.   :53.13   5: 18              
##        region       charges               charge_group
##  northeast:324   Min.   : 1122   Low         :359     
##  northwest:325   1st Qu.: 4740   Intermediate:621     
##  southeast:364   Median : 9382   High        :358     
##  southwest:325   Mean   :13270                        
##                  3rd Qu.:16640                        
##                  Max.   :63770                        

p<- ggplot(data,aes(age))
p+ geom_boxplot(outlier.colour = "red")
```

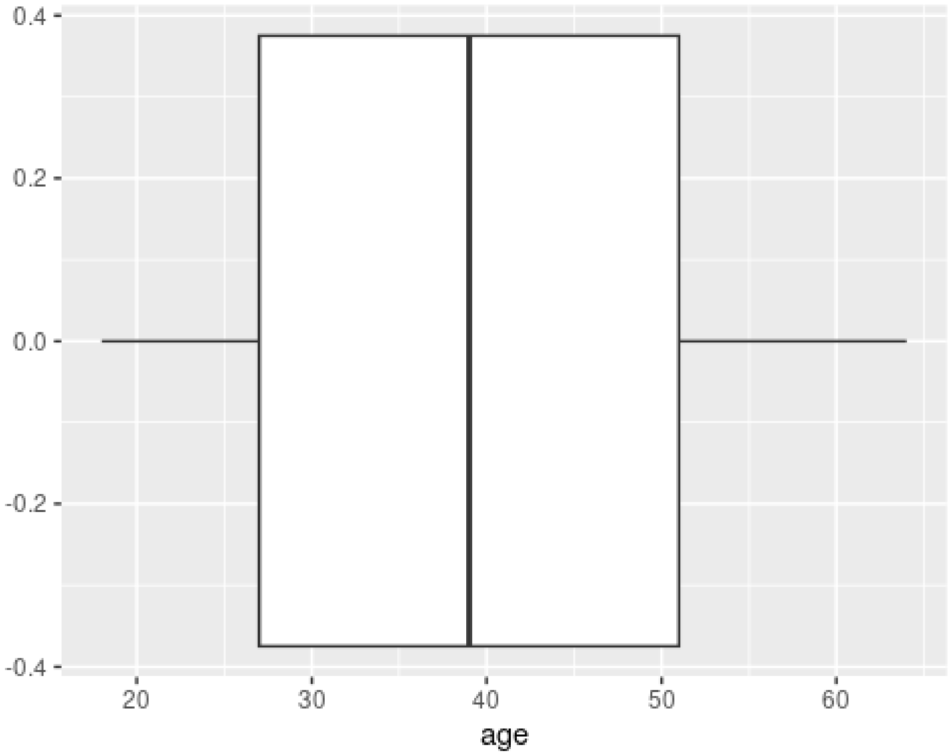

Figure 2. Boxplot of age. Visualizes the age data in a box plot.

```
p+ geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

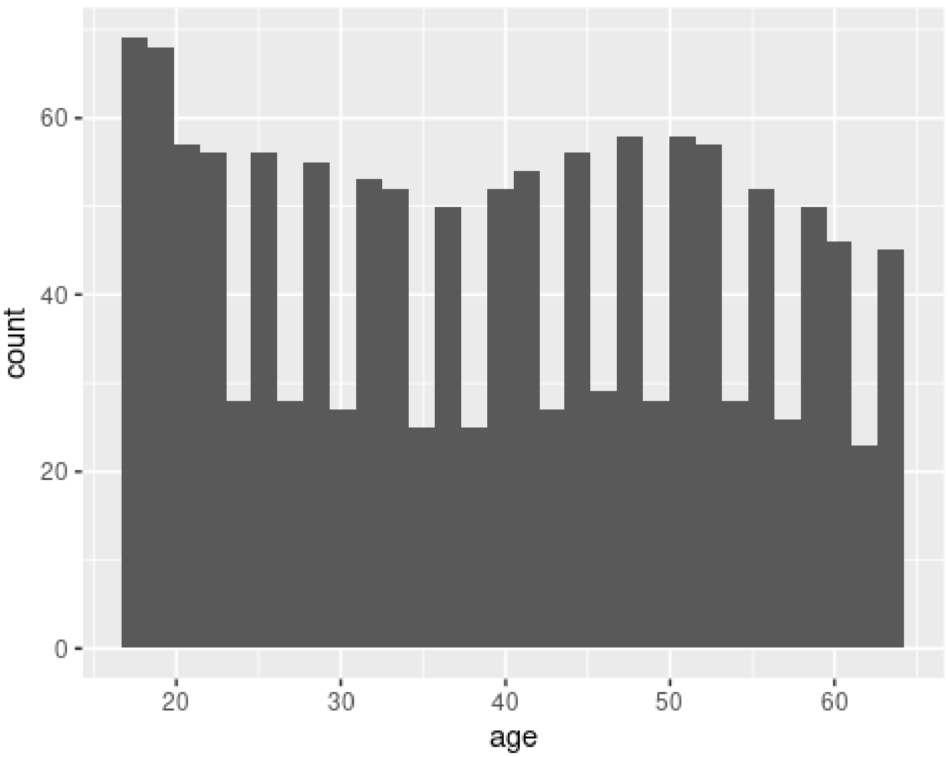

Figure 3. Age Histogram. Histogram of age data.

```
 p<- ggplot(data,aes(bmi))
p+ geom_boxplot(outlier.colour = "red")
```

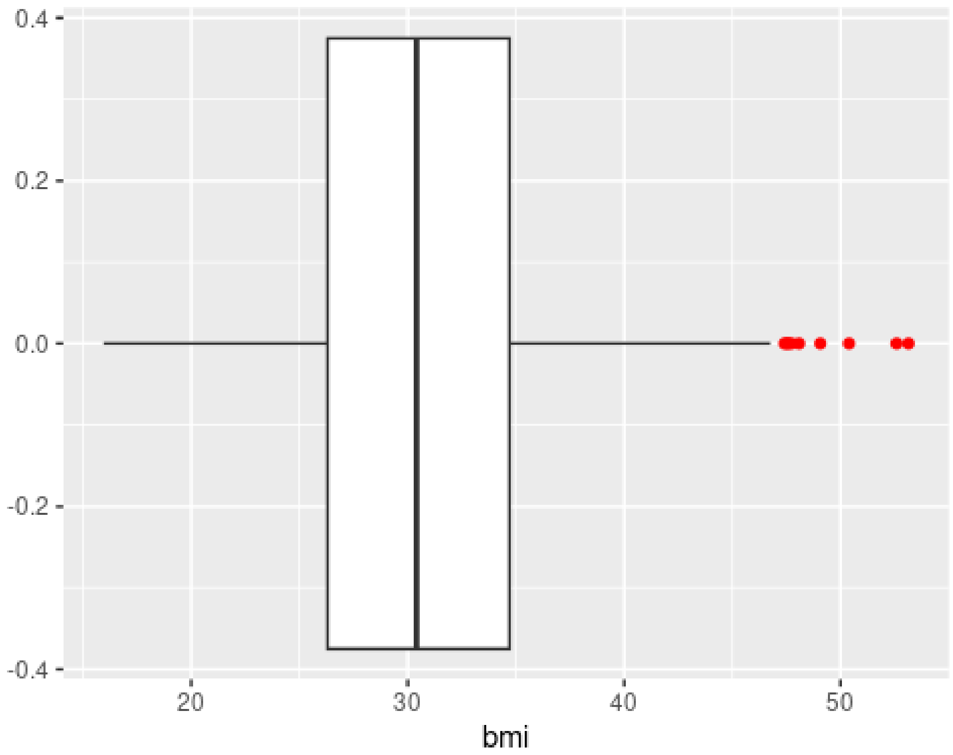

Figure 4. Boxplot of BMI. A boxplot of the BMI data.

```
p+ geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

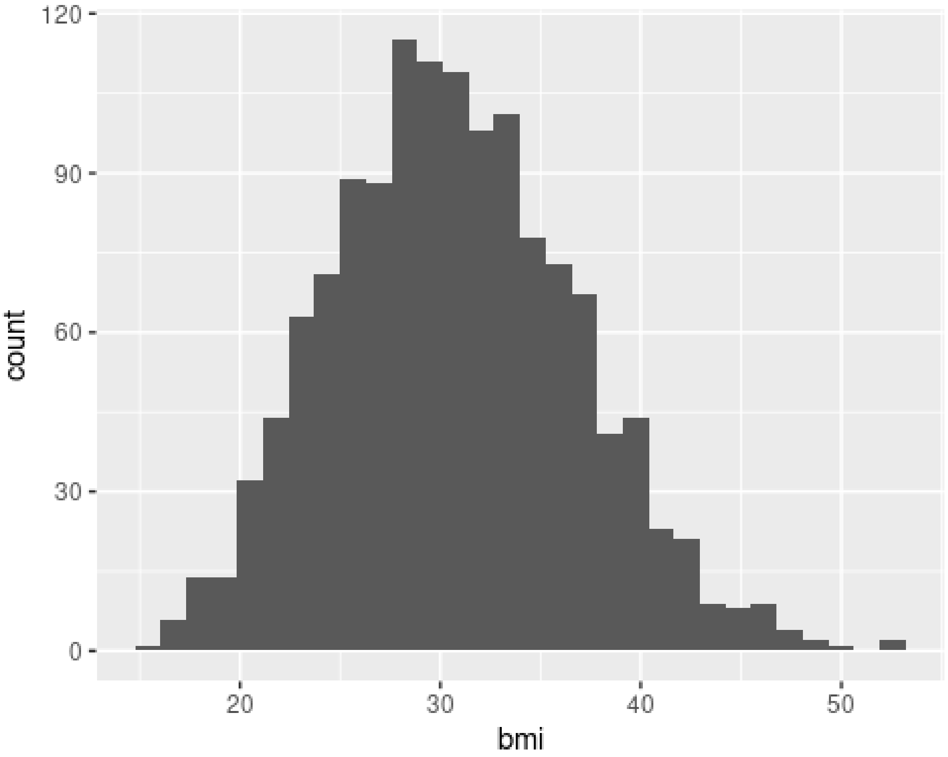

Figure 5.Histogram of BMI. Histogram of the BMI data.

```
 p<- ggplot(data,aes(charges))
p+ geom_boxplot(outlier.colour = "red")
```

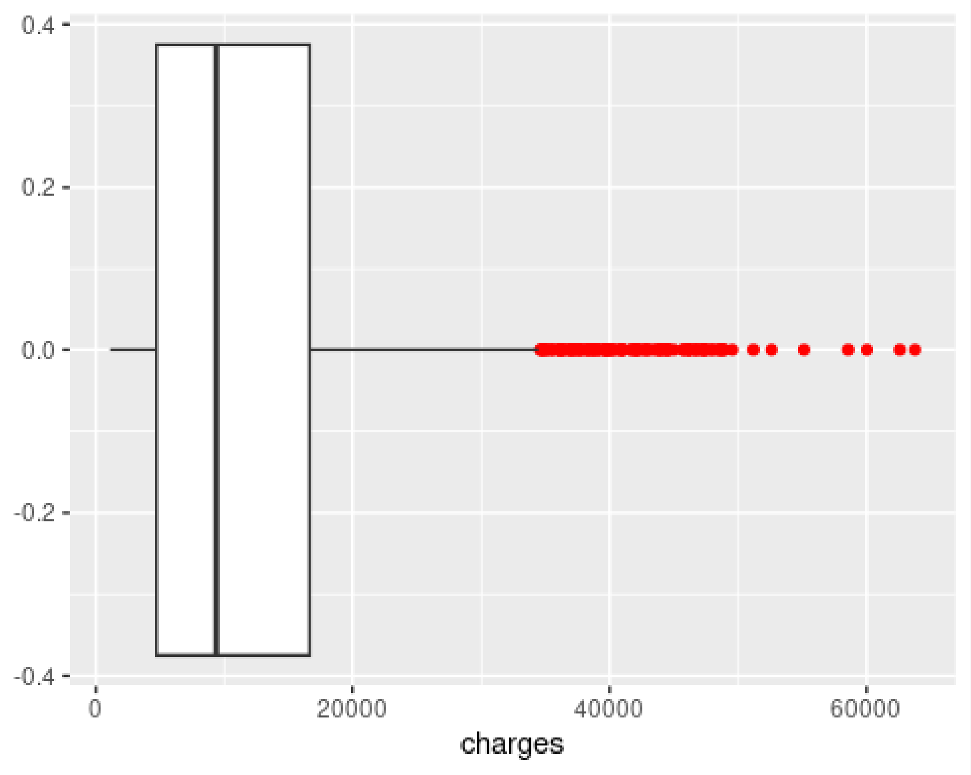

Figure 6. Charges Boxplot. A boxplot of the BMI data.

```
p+ geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

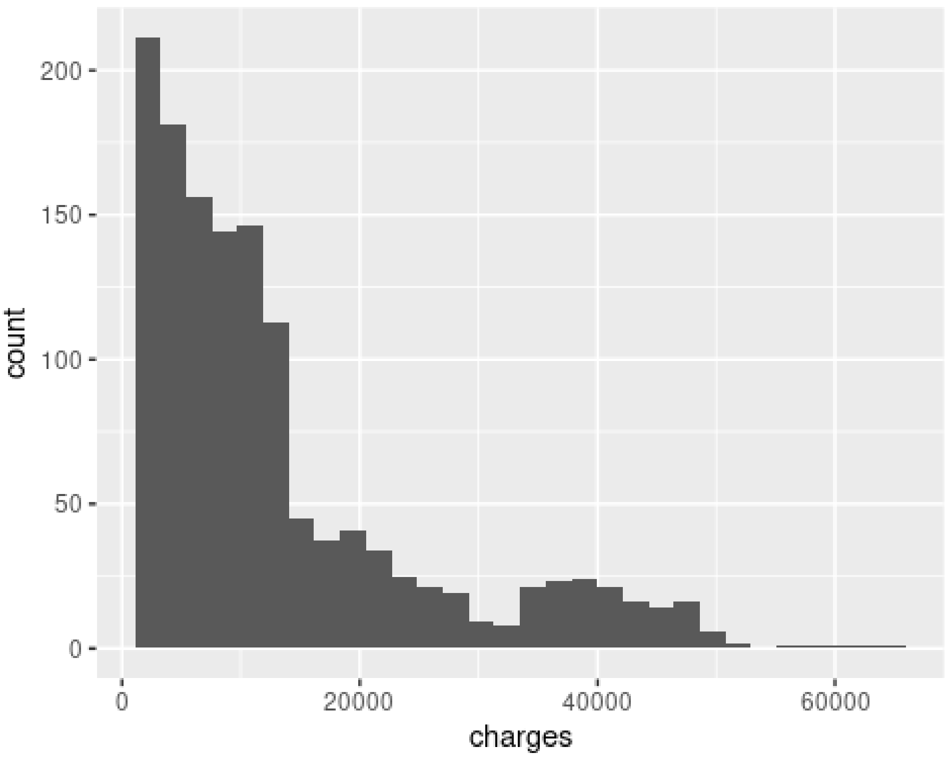

Figure 7. Histogram of Charges. Histogram of the charges data.

2.3: Frequency Distributions

To further investigate the data, let us assess the frequency distributions of the data. Using the pairs() function, we can visualize a matrix of scatterplots of each variable (Fig. 8). Furthermore, by utilizing the var() and sd() functions of the numerical data, we can obtain the variance and standard deviations of the data respectively. Then using the ggplot library, a histogram and density plot of each numeric variable can be created (Fig. 9-15). The density plot of the children variable is particularly interesting as it visualizes the data depending on the number of dependencies (children) (Fig. 13). The density graph of the charges variable is also interesting as it reveals that most of the data is below the $15,000 mark (Fig. 15). This chart helped determine how to split the three categories of price into meaningful groups.

```
pairs(data)
```

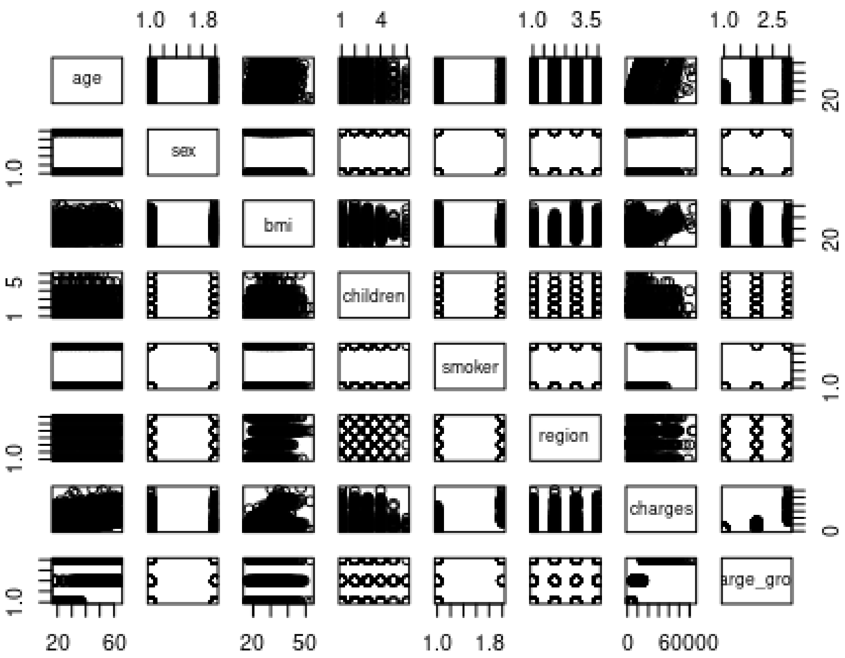

Figure 8. Matrix of Scatterplots. Scatterplots of each variable of the dataset.

```
#AGE

var(data$age)

## [1] 197.4014

sd(data$age)

## [1] 14.04996

p<- ggplot(data,aes(age))
p+ geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

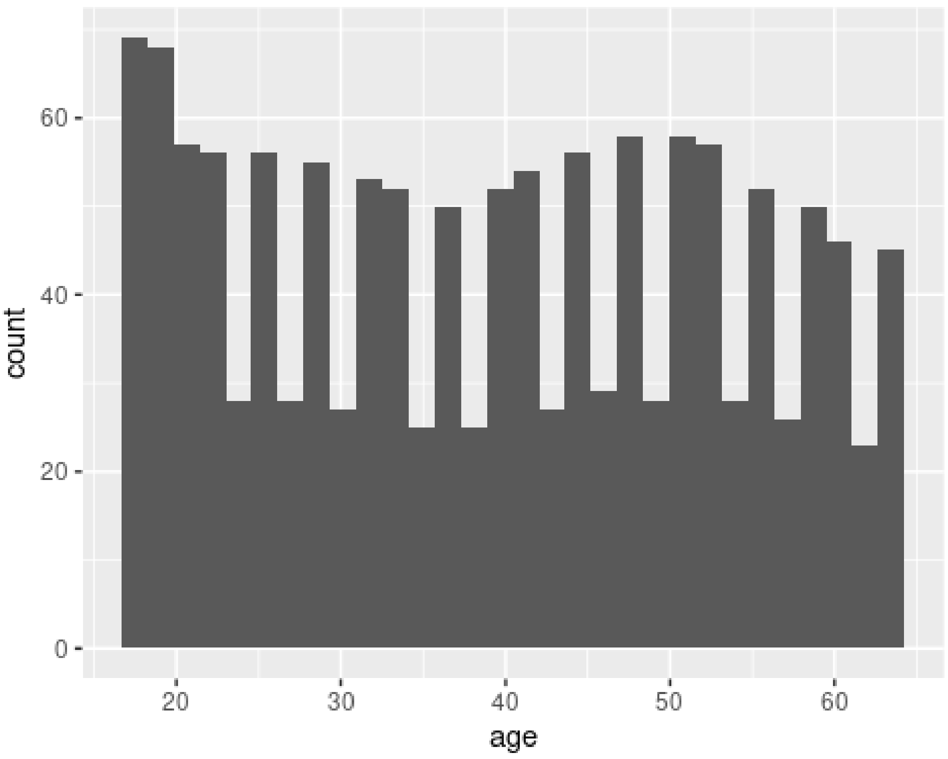

Figure 9. Histogram of Age data. Quantifies the different ages in the dataset.

```
p+ geom_density()
```

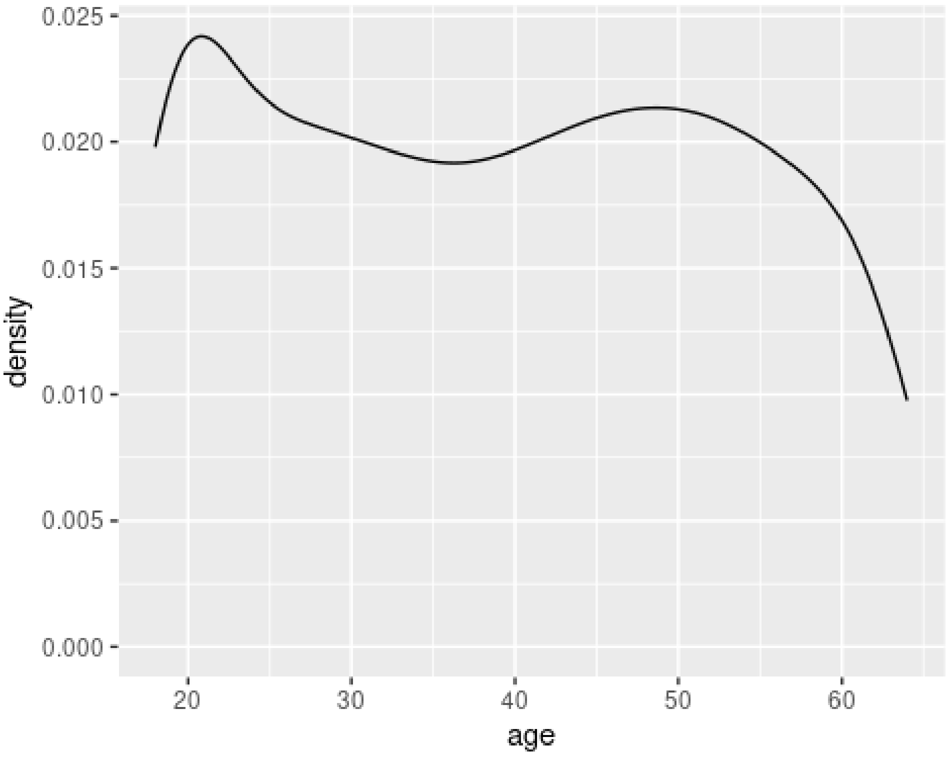

Figure 10. Density of Age. Shows the density of the age variable.

```
#BMI
var(data$bmi)

## [1] 37.18788

sd(data$bmi)

## [1] 6.098187

p<- ggplot(data,aes(bmi))
p+ geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

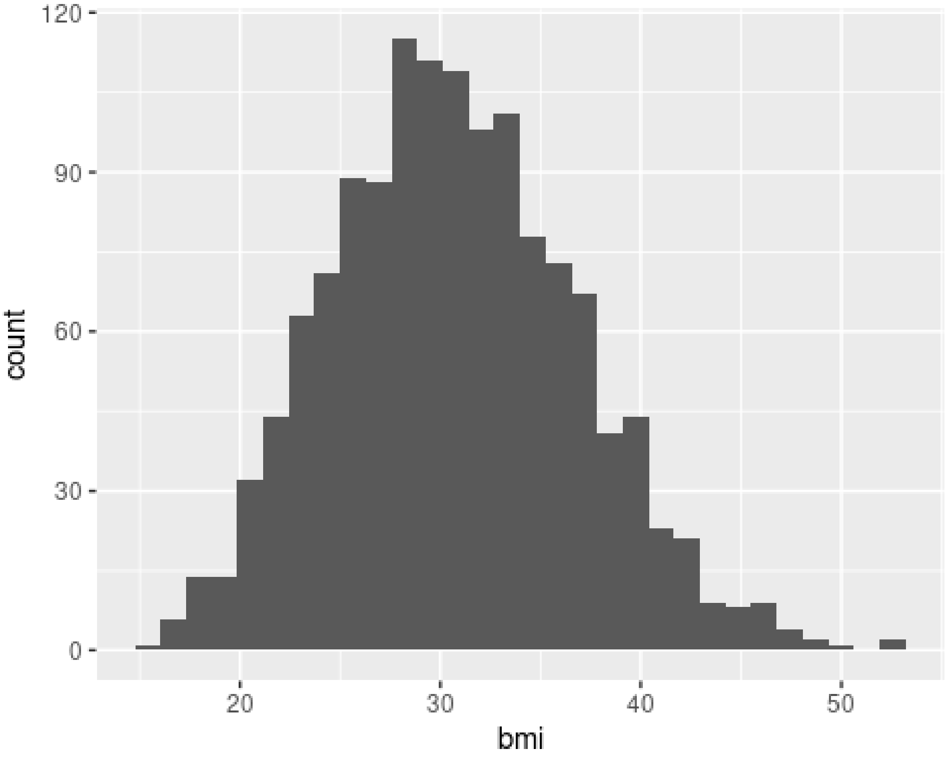

Figure 11. Histogram of BMI. Shows the histogram of the age variable.

```
p+ geom_density()
```

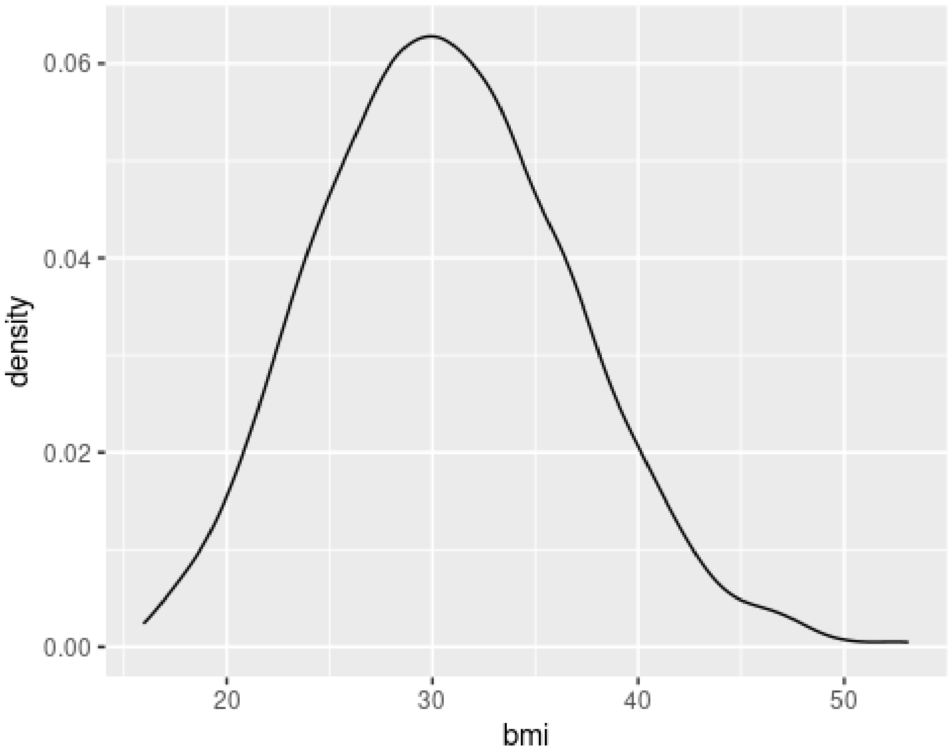

Figure 12. Density of BMI. The BMI is normally distributed around a BMI of 30.

```
#children
p<- ggplot(data,aes(children))
p+ geom_density()
```

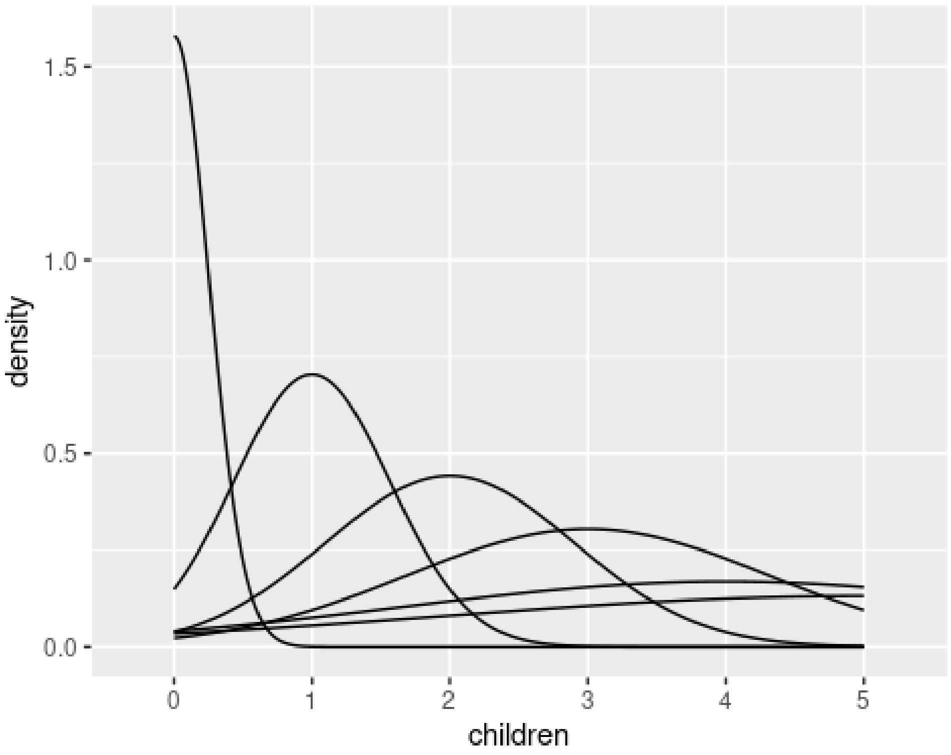

Figure 13. The density of the number of children. Most patients had zero dependencies.

```
#CHARGES
var(data$charges)

## [1] 146652372

sd(data$charges)

## [1] 12110.01

p<- ggplot(data,aes(charges))
p+ geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

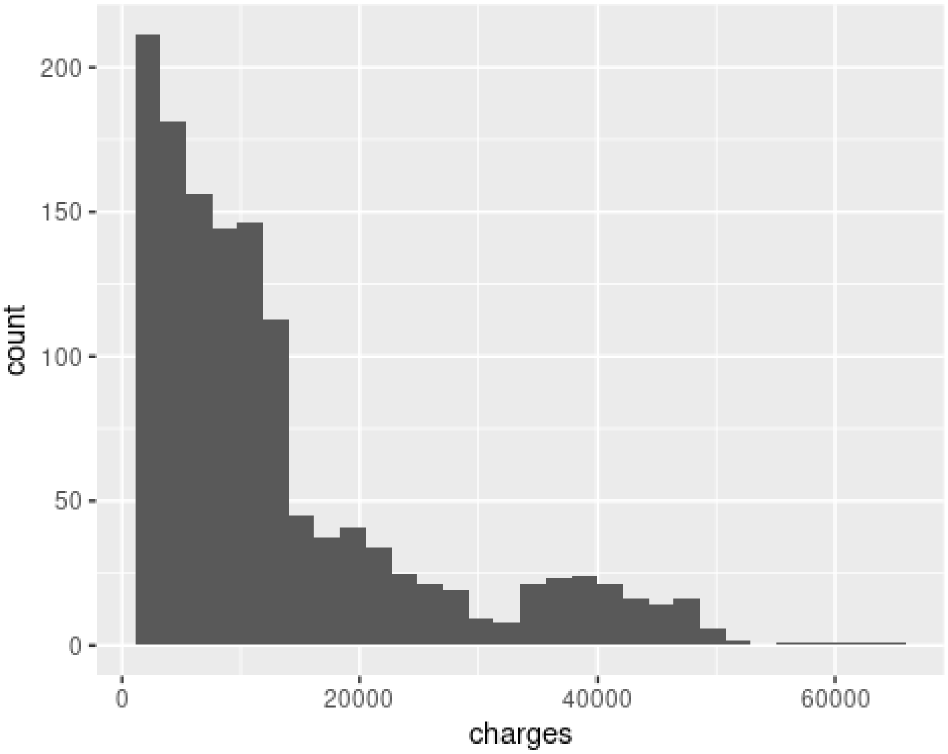

Figure 14. Quantification of insurance charges. Most charges fell below $15,000.

```
p+ geom_density()
```

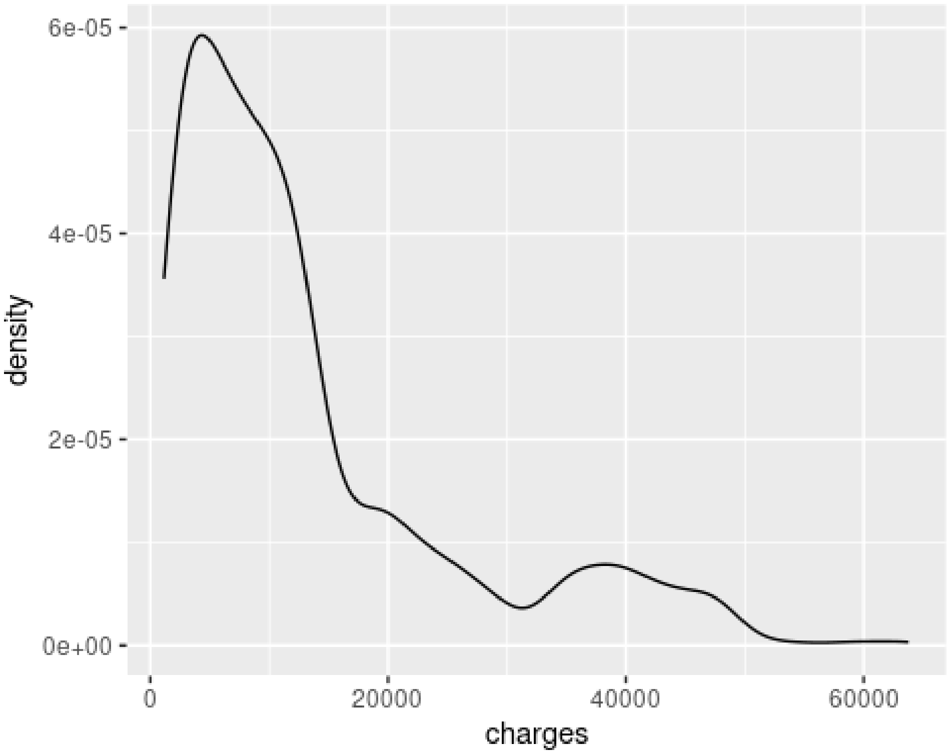

Figure 15. Density of Charges. Density chart of charges data.

```
summary(data$charges)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    1122    4740    9382   13270   16640   63770

data %>% count(data$charge_group)

##   data$charge_group   n
## 1               Low 359
## 2      Intermediate 621
## 3              High 358
```

Normalization of Data

Later in our analysis, we will require normalized data as we will be conducting a K-nearest neighbor model. There are several ways to normalize data, one of which is a min-max normalization, which was chosen for this project. This method takes all numeric data and places it within 0 and 1. This normalization method doesn't affect the integrity of the data and therefore our future models. Note that the children column needs to be refactored after the normalization of the data.

```
## using min-max normalization
library(dplyr)

min_max_normalize <- function(x) {
  (x - min(x, na.rm = TRUE)) / (max(x, na.rm = TRUE) - min(x, na.rm = TRUE))
}

normalized_data <- data %>%
  mutate(across(where(is.numeric), min_max_normalize))

## refactor children column
normalized_data$children<- factor(normalized_data$children)

data_encoded <- model.matrix(~ . - 1, data = normalized_data)
## Prove it works
str(normalized_data)

## 'data.frame':    1338 obs. of  8 variables:
##  $ age         : num  0.0217 0 0.2174 0.3261 0.3043 ...
##  $ sex         : Factor w/ 2 levels "female","male": 1 2 2 2 2 1 1 1 2 1
...
##  $ bmi         : num  0.321 0.479 0.458 0.181 0.348 ...
##  $ children    : Factor w/ 6 levels "0","1","2","3",..: 1 2 4 1 1 1 2 4 3
1 ...
##  $ smoker      : Factor w/ 2 levels "no","yes": 2 1 1 1 1 1 1 1 1 1 ...
##  $ region      : Factor w/ 4 levels "northeast","northwest",..: 4 3 3 2 2
3 3 2 1 2 ...
##  $ charges     : num  0.25161 0.00964 0.05312 0.33301 0.04382 ...
##  $ charge_group: Factor w/ 3 levels "Low","Intermediate",..: 3 1 1 3 1 1 2
2 2 3 ...
```

Part 3: Multivariate Analysis

Now that we have investigated the individual variables in the data set, let us now assess potential relationships between these variables. The following code analyzes the relationship between *age* versus *charges* and *BMI* versus *charges*. Analysis was also performed between *age* versus *charge_group* and *BMI* versus *charge_group* to observe how this would aid our results. Data points were plotted using ggplot and the *lm*() and *corr*() functions were used to create a linear regression model and assess the correlation between the respective variables. The linear model of the *age* versus *charges* showed an R^2 value of about 8%, whereas the *age* versus *charge_group* outputted a value of 45%. The linear model of the *BMI* versus *charges* showed an R^2 value of about 3%, whereas the *BMI* versus *charge_group* outputted a value of 6%. Although these values are quite low and would not prove effective models of medical bill price, it is important to note that the relationships with the *charge_group* performed better than those with the *charges* group. It is important to note that linear regression models and correlation cannot be determined on non-numeric variables.

```
p<- ggplot(data,aes(age,charges))
p+ geom_point()
```

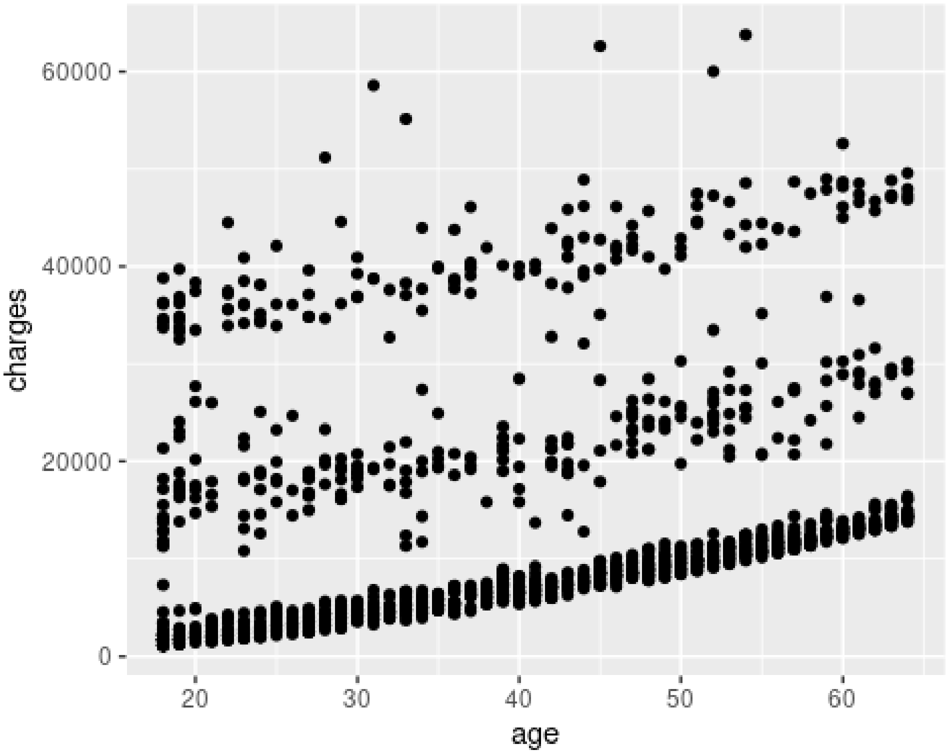

Figure 15. Linear regression of the charges versus age. As age increases, charge increases.

```
age.lm<- lm(age~charges, data=data)
summary(age.lm)
```

```
##
## Call:
## lm(formula = age ~ charges, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -30.0609 -11.4222   0.1691  11.1759  24.6013
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.460e+01  5.441e-01   63.60   <2e-16 ***
## charges     3.469e-04  3.029e-05   11.45   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.41 on 1336 degrees of freedom
## Multiple R-squared:  0.08941,    Adjusted R-squared:  0.08872
## F-statistic: 131.2 on 1 and 1336 DF,  p-value: < 2.2e-16
```

```
cor(x=data$age,y=data$charges)
```

```
## [1] 0.2990082
```

```
p<- ggplot(data,aes(age,charge_group))
p+ geom_point()
```

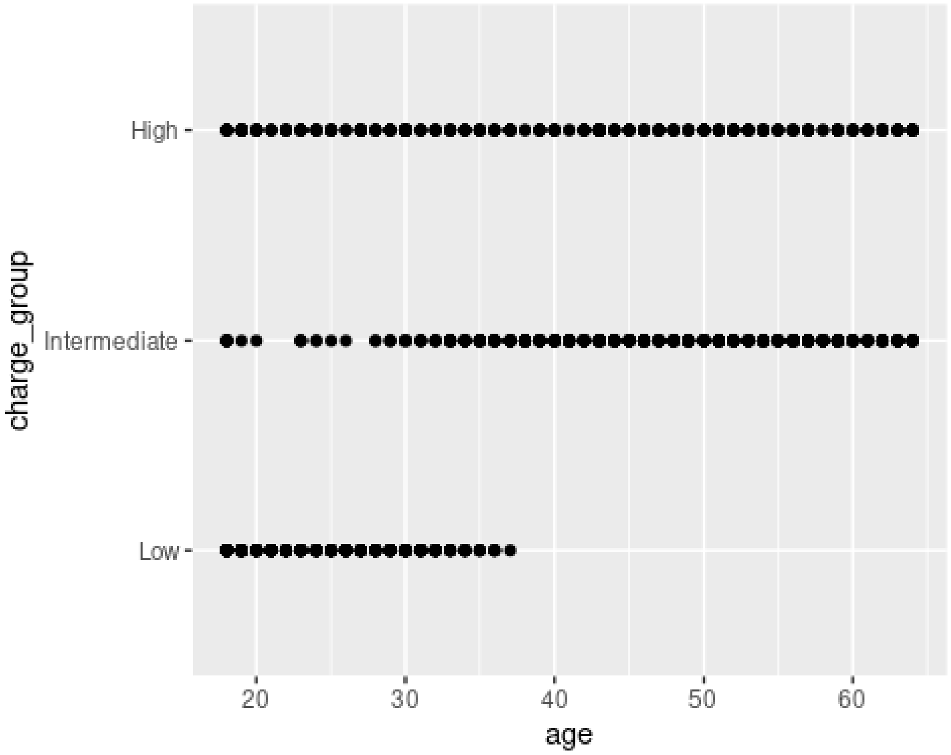

Figure 16. Charge groups versus age. The costs of charges are grouped into 3 separate groups to simplify data analysis.

```
age.lm<- lm(age~charge_group, data=data)
summary(age.lm)

##
## Call:
## lm(formula = age ~ charge_group, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.262  -6.287  -0.229   7.486  23.805
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               24.2869     0.5469   44.41   <2e-16 ***
## charge_groupIntermediate  22.9756     0.6871   33.44   <2e-16 ***
## charge_groupHigh          15.9086     0.7740   20.55   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.36 on 1335 degrees of freedom
## Multiple R-squared:  0.4568, Adjusted R-squared:  0.456
## F-statistic: 561.4 on 2 and 1335 DF,  p-value: < 2.2e-16

cor(x=data$age,y=data$charges)

## [1] 0.2990082

p<- ggplot(data,aes(bmi,charges))
p+ geom_point()
```

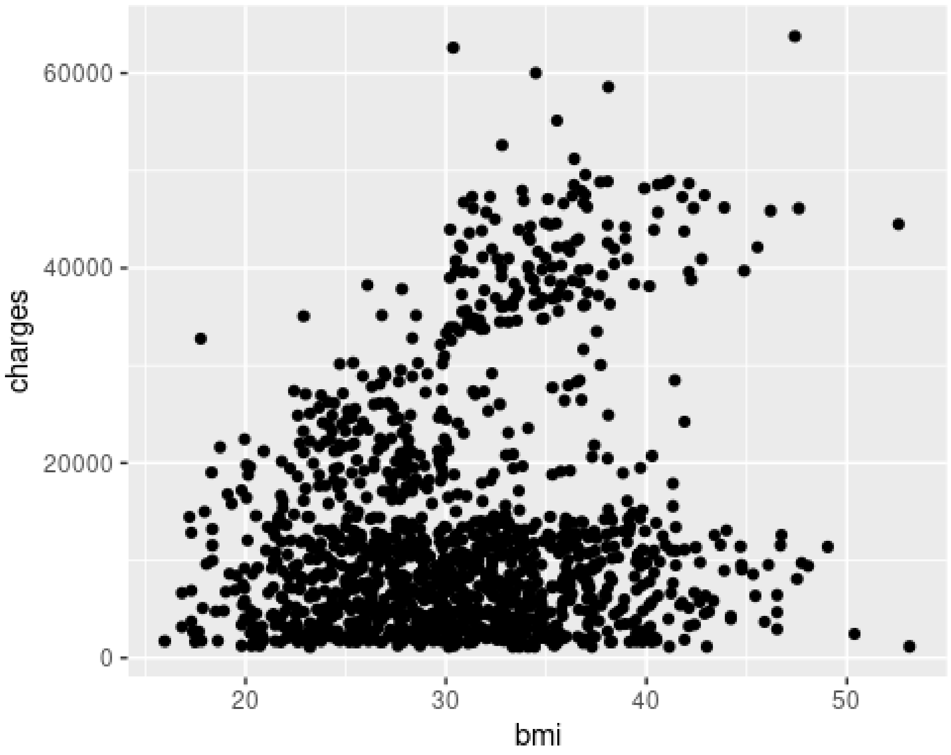

Figure 17. Charges versus BMI. Some positive correlation is suggested by the graph.

```
bmi.lm<- lm(bmi~charges, data=data)
summary(bmi.lm)

##
## Call:
## lm(formula = bmi ~ charges, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.8424  -4.1030  -0.2401   3.8467  23.6758
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.934e+01  2.426e-01 120.956  < 2e-16 ***
## charges     9.988e-05  1.350e-05   7.397 2.46e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.979 on 1336 degrees of freedom
## Multiple R-squared:  0.03934,    Adjusted R-squared:  0.03862
## F-statistic: 54.71 on 1 and 1336 DF,  p-value: 2.459e-13

cor(x=data$bmi,y=data$charges)
```

```
## [1] 0.198341

p<- ggplot(data,aes(bmi,charge_group))
p+ geom_point()
```

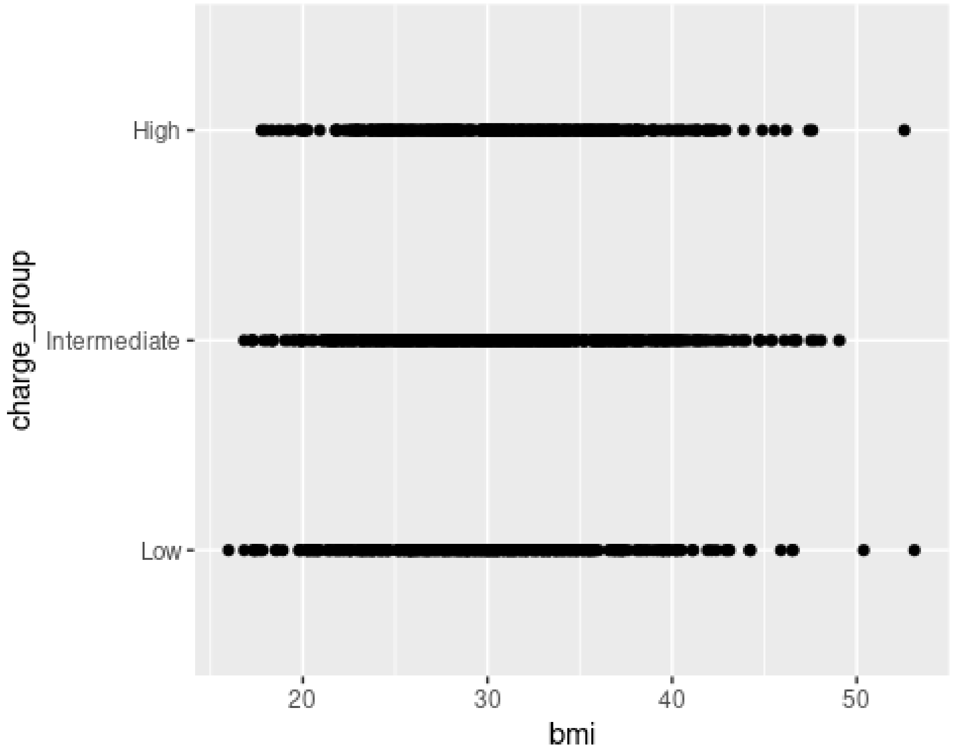

Figure 18. Charge groups versus BMI. This allows analyses to be simplified.

```
bmi.lm<- lm(bmi~charge_group, data=data)
summary(bmi.lm)

##
## Call:
## lm(formula = bmi ~ charge_group, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.0284  -4.3207  -0.2548   4.0276  23.2446
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               29.8854     0.3211  93.082  < 2e-16 ***
## charge_groupIntermediate   0.9579     0.4033   2.375  0.01769 *
## charge_groupHigh           1.2459     0.4544   2.742  0.00619 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 6.083 on 1335 degrees of freedom
## Multiple R-squared:  0.006351,   Adjusted R-squared:  0.004862
## F-statistic: 4.266 on 2 and 1335 DF,  p-value: 0.01423

cor(x=data$bmi,y=data$charges)

## [1] 0.198341

data %>% count(data$children)

##   data$children   n
## 1             0 574
## 2             1 324
## 3             2 240
## 4             3 157
## 5             4  25
## 6             5  18

p<- ggplot(data,aes(children,charges))
p+ geom_point()
```

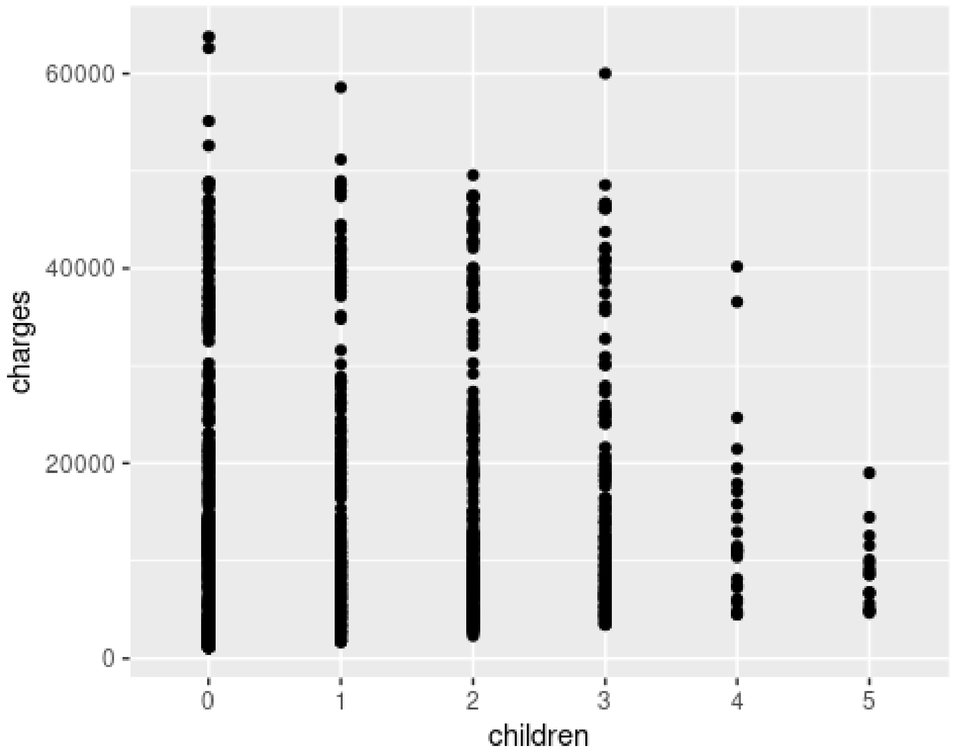

Figure 19. Charges versus number of children. Patients with greater numbers of children tended to have lesser charges.

```
p<- ggplot(data,aes(region,charges))
p+ geom_point()
```

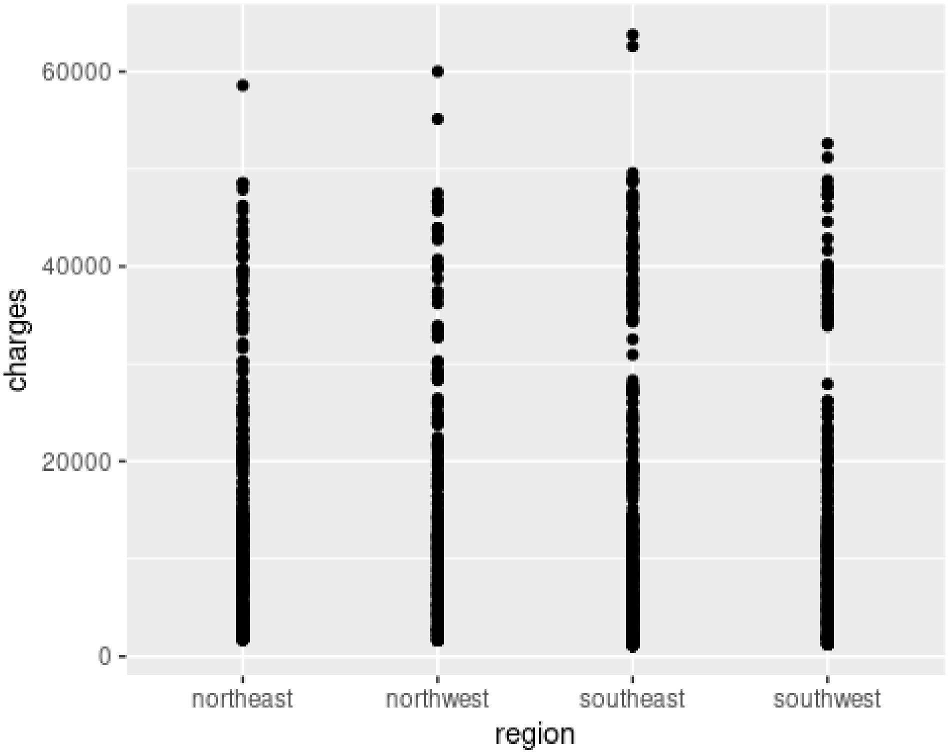

Figure 20. Charges versus region. Not much correlation is seen.

Part 4: Predictive Modeling

4.1: Split your data

To create more reliable predictive models, three advanced models will be created. These models include a multiple linear regression, a decision tree, and a K-nearest neighbor model. Prior to creating these models, we first need to split our normalized data into training and test data respectively. We will split the data into 70% training data and 30% testing data.

```
# Set Seed
set.seed(100)
#Seperate data
spt<-sample(1:nrow(data),size=nrow(data)*0.7,replace=FALSE)
#set Train data
```

```
train.data<-normalized_data[spt,]
#set test data
test.data<-normalized_data[-spt,]
```

4.2: Build your model

Model 1 - Multiple Linear Regression

A multiple linear regression of all parameters was performed to potentially describe the relationship between a patient's medical bill and the parameters of the data set. Note that linear regression-based models are incapable of predicting categorical data so to predict data into one of these three categories, we must explore other model options.

```
## Multiple linear regression of all parameters
mreg1<-
lm(data$charges~data$age+data$bmi+data$sex+data$children+data$region+data$smo
ker, data = data)
summary(mreg1)

##
## Call:
## lm(formula = data$charges ~ data$age + data$bmi + data$sex +
##     data$children + data$region + data$smoker, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11689.4  -2902.6   -943.7   1492.2  30042.7
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -11927.17     993.66 -12.003  < 2e-16 ***
## data$age                257.19      11.91  21.587  < 2e-16 ***
## data$bmi                336.91      28.61  11.775  < 2e-16 ***
## data$sexmale           -128.16     332.83  -0.385 0.700254
## data$children1          390.98     421.35   0.928 0.353619
## data$children2         1635.78     466.67   3.505 0.000471 ***
## data$children3          964.34     548.10   1.759 0.078735 .
## data$children4         2947.37    1239.16   2.379 0.017524 *
## data$children5         1116.04    1456.02   0.767 0.443514
## data$regionnorthwest   -380.04     476.56  -0.797 0.425318
## data$regionsoutheast  -1033.14     479.14  -2.156 0.031245 *
## data$regionsouthwest   -952.89     478.15  -1.993 0.046483 *
## data$smokeryes        23836.41     414.14  57.557  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6059 on 1325 degrees of freedom
## Multiple R-squared:  0.7519, Adjusted R-squared:  0.7497
## F-statistic: 334.7 on 12 and 1325 DF,  p-value: < 2.2e-16
```

Model 2 - K-Nearest Neighbor

Our second model will be a K-nearest neighbor. To begin we must choose a starting number of K to be our centers. We will obtain this starting value by taking the square root of the number of rows in the training data which we will round up to 31. Then we create training and test classes of the value we wish to predict which is the *charge_group*. The K-nearest neighbor function is then used to set up the model using the training, test, and class data with a k value of 31.

```
## how many centers?
sqrt(nrow(train.data))

## [1] 30.59412

library(class)
#set Train data
train.data_class<-train.data$charge_group
#set test data
test.data_class<-test.data$charge_group

train.data <- as.data.frame(lapply(train.data, as.numeric))
test.data <- as.data.frame(lapply(test.data, as.numeric))

suppressWarnings(knn.31<-knn(train=train.data,test=test.data,cl=train.data_cl
ass,k=31))
```

Model 3 - Decision Tree

Our final model is a decision tree. To have an accurate reading in the decision tree, we must remove the charges column of the data as it is deterministic of which charge group a data point will be assigned to. After removing the column, a decision tree with an initial cp of 0 was then created using the rpart library (Fig. 21). This tree is too convoluted so we will find the optimal cp value of 0.056 (Fig. 22). We will then recreate the tree using the optimal cp value (Fig. 23).

```
library(rpart)
library(rpart.plot)
# Remove charges column since it determinates the outcome
train.data_tree<- train.data
train.data_tree$charges<-NULL
# using all the predictors and setting cp = 0
tree <- rpart(charge_group ~ ., data = train.data_tree, method = "class", cp
= 0.00)
rpart.plot(tree)
```

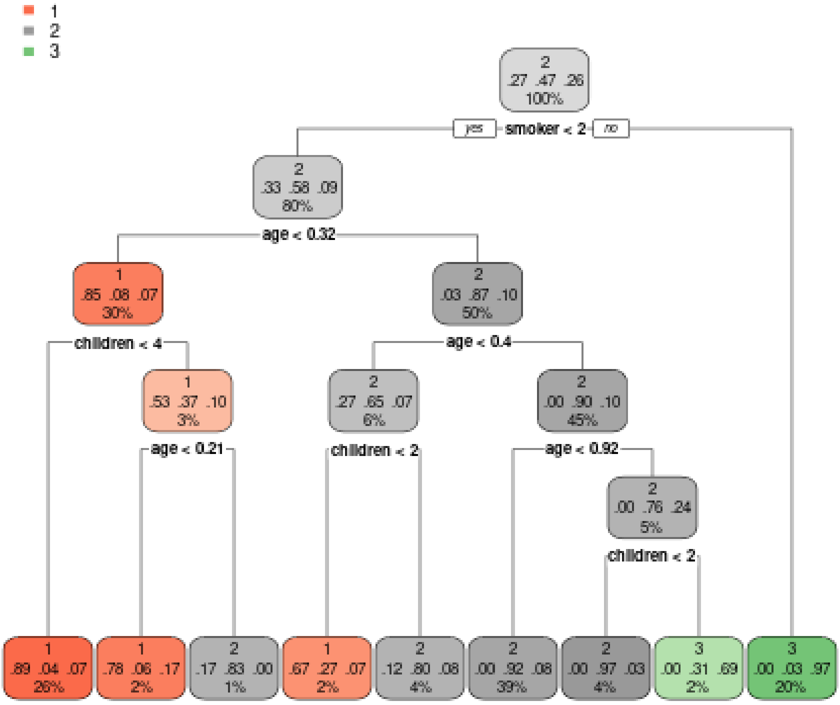

Figure 21. Decision Tree. Decision tree with a cp value of 0 .

```
plotcp(tree, lty = 3, col = 2, upper = "splits" )
```

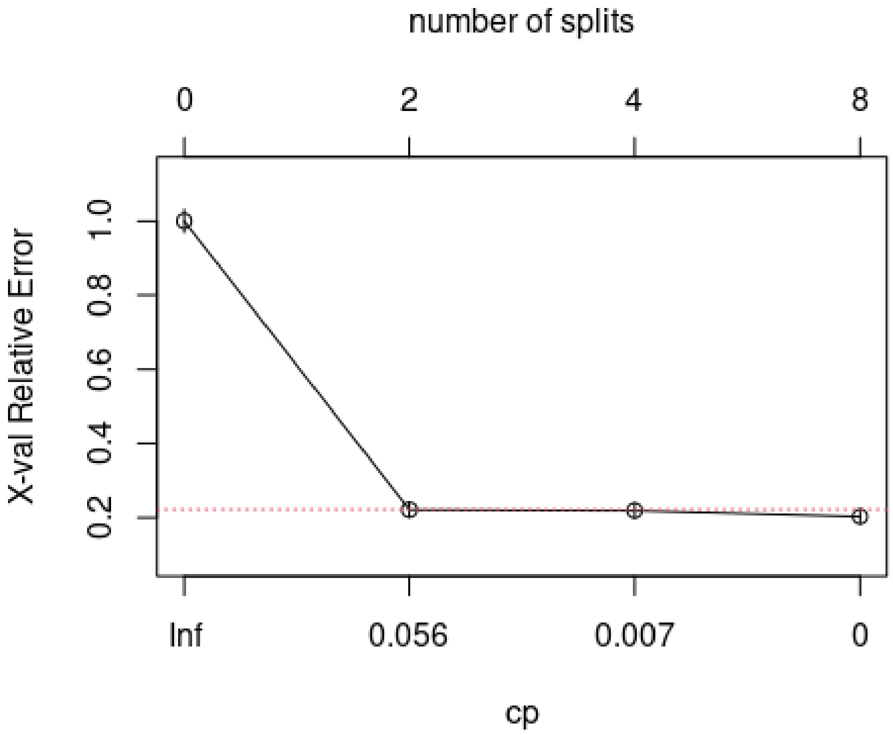

Figure 22. Optimal CP. Shows the relative error with respect to different cp values.

```r
tree <- rpart(charge_group ~ ., data = train.data_tree, method = "class", cp
= 0.056)
rpart.plot(tree)
```

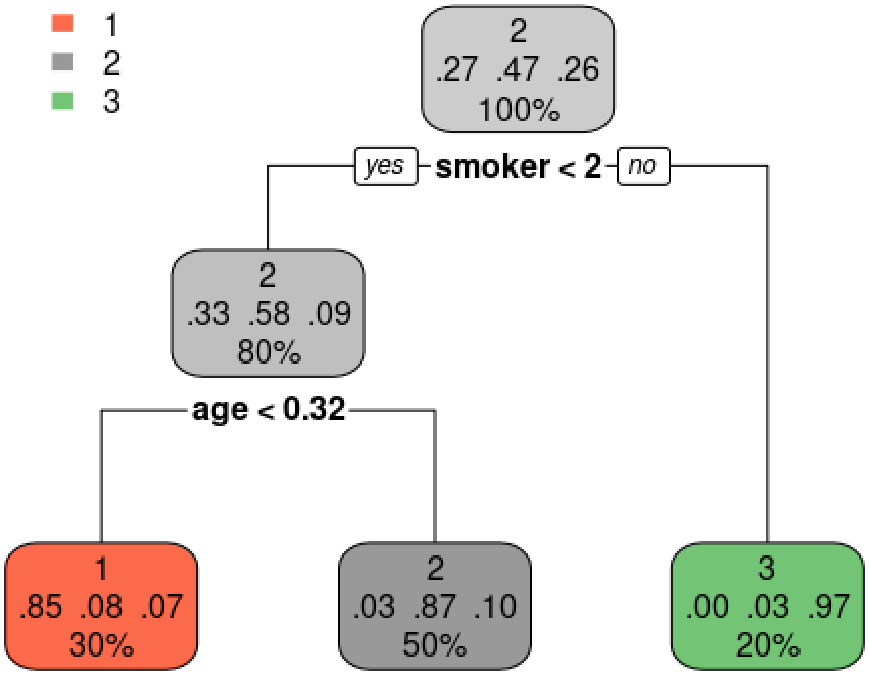

Figure 23. Final Decision Tree. A decision tree with a cp value of 0.056.

4.3 Model Evaluation

Now that our three models have been created, let's test their predictive success.

Multiple Linear Regression Evaluation

The *summary*() function and some visuals have been provided to assess the success of this model. Since the R^2 value was 75%, there seems to be a decent relationship between these parameters and the price of a medical bill. Although this value is decent, it would be difficult to use for predictive purposes as it relates the parameters to a numeric value of medical bills and not the three charge categories we created earlier for predictive purposes.

```
summary(mreg1)

##
## Call:
```

```
## lm(formula = data$charges ~ data$age + data$bmi + data$sex + 
##     data$children + data$region + data$smoker, data = data)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -11689.4  -2902.6   -943.7   1492.2  30042.7 
## 
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)    
## (Intercept)          -11927.17     993.66 -12.003  < 2e-16 ***
## data$age                257.19      11.91  21.587  < 2e-16 ***
## data$bmi                336.91      28.61  11.775  < 2e-16 ***
## data$sexmale           -128.16     332.83  -0.385 0.700254    
## data$children1          390.98     421.35   0.928 0.353619    
## data$children2         1635.78     466.67   3.505 0.000471 ***
## data$children3          964.34     548.10   1.759 0.078735 .  
## data$children4         2947.37    1239.16   2.379 0.017524 *  
## data$children5         1116.04    1456.02   0.767 0.443514    
## data$regionnorthwest   -380.04     476.56  -0.797 0.425318    
## data$regionsoutheast  -1033.14     479.14  -2.156 0.031245 *  
## data$regionsouthwest   -952.89     478.15  -1.993 0.046483 *  
## data$smokeryes        23836.41     414.14  57.557  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 6059 on 1325 degrees of freedom
## Multiple R-squared:  0.7519, Adjusted R-squared:  0.7497 
## F-statistic: 334.7 on 12 and 1325 DF,  p-value: < 2.2e-16
```

```
plot(mreg1$fitted.values, sqrt(abs(mreg1$residuals)),
     main = "Residuals vs. Fitted Values",
     xlab = "Fitted Values",
     ylab = "SQRT of Residuals")
```

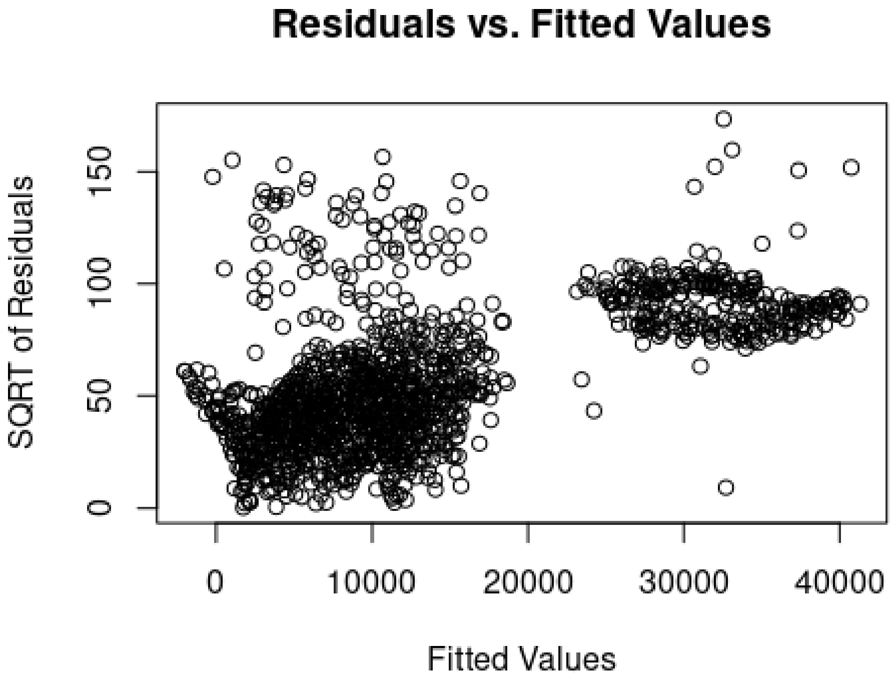

Figure 24. The square root of residuals versus fitted values. Two main populations can be seen.

```r
# Residuals vs. Each Predictor Variable
par(mfrow = c(2, 2))  # Create a 2x2 grid for multiple plots
plot(mreg1, which = 1:4)
```

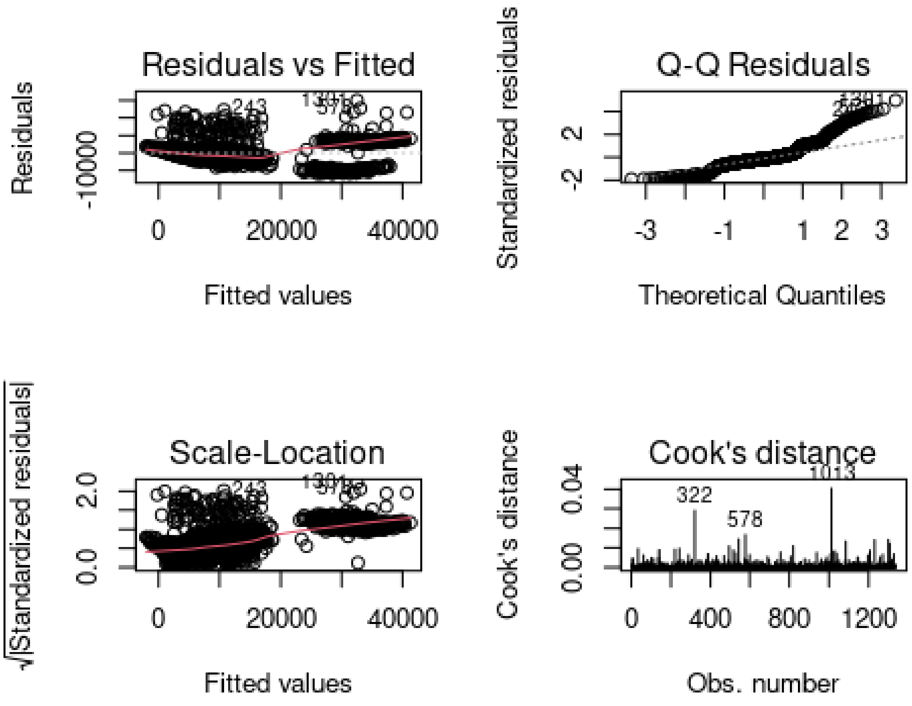

Figure 25. Multiple plots to assess types of multivariate analyses. Residuals versus fitted values; Q-Q residuals; Scale-Location; and Cook's distance.

Decision Tree Evaluation

Below is a confusion matrix of the decision tree created earlier. The classification error based on the confusion matrix was then calculated to be almost 4%. This means that the decision tree model was successful in predicting the correct charge group 96% of the time.

```
tree.predict <- predict(tree, test.data, type = "class")
## confusion matrix
conf.matrix <- table(test.data$charge_group, tree.predict)
rownames(conf.matrix) <- paste("Actual", rownames(conf.matrix), sep = ":")
colnames(conf.matrix) <- paste("Predicted", colnames(conf.matrix), sep = ":")
print(conf.matrix)

##             tree.predict
##              Predicted:1 Predicted:2 Predicted:3
##    Actual:1          100           9           0
##    Actual:2            7         175           1
##    Actual:3            8          16          86
```

```
# caclulating the classification error
classification_error<- (conf.matrix[1, 2] + conf.matrix[2, 1]) /
sum(conf.matrix)
print(classification_error)

## [1] 0.039801
```

KNN Evaluation

Finally, to assess the predictability of the KNN model, the following code was written. The model was found to successfully predict the groupings 97% of the time. Although this is very good, we can do better by optimizing K. A graph of accuracy vs. K values was created and the optimal K value of 3 was chosen which makes sense as there are three categories to choose from (Fig. 26). A new KNN model was created using a K of 3 and produced a 99.5% accuracy.

```
acc31<- 100*sum(test.data_class==knn.31)/nrow(test.data)
acc31

## [1] 97.26368

i<- 1
k.optm<-1
for(i in 1:50){
knn.mod <- knn(train=train.data,test=test.data,cl=train.data_class, k=i)
k.optm[i] <- 100 * sum(test.data_class==knn.mod)/nrow(test.data)
k<-i
cat(k,'=',k.optm[i],' ')
}

## 1 = 99.50249  2 = 99.25373  3 = 99.50249  4 = 99.25373  5 = 99.00498  6 =
99.00498  7 = 98.75622  8 = 98.75622  9 = 98.00995  10 = 98.25871  11 =
98.00995  12 = 98.00995  13 = 97.76119  14 = 97.76119  15 = 97.51244  16 =
97.76119  17 = 98.00995  18 = 98.00995  19 = 98.00995  20 = 97.76119  21 =
97.51244  22 = 97.51244  23 = 97.51244  24 = 97.76119  25 = 97.51244  26 =
97.51244  27 = 97.01493  28 = 97.01493  29 = 97.01493  30 = 97.01493  31 =
97.26368  32 = 97.26368  33 = 97.01493  34 = 96.51741  35 = 96.51741  36 =
96.51741  37 = 96.26866  38 = 97.01493  39 = 97.01493  40 = 96.76617  41 =
97.01493  42 = 96.51741  43 = 96.51741  44 = 96.51741  45 = 96.0199  46 =
96.26866  47 = 95.77114  48 = 95.77114  49 = 95.52239  50 = 95.02488

#View accuracy plot
ggplot(data=data.frame(k.optm))+
geom_line(mapping=aes(y=k.optm,x=1:length(k.optm))) + labs(x="K-Value",
y="Accuracy level %")
```

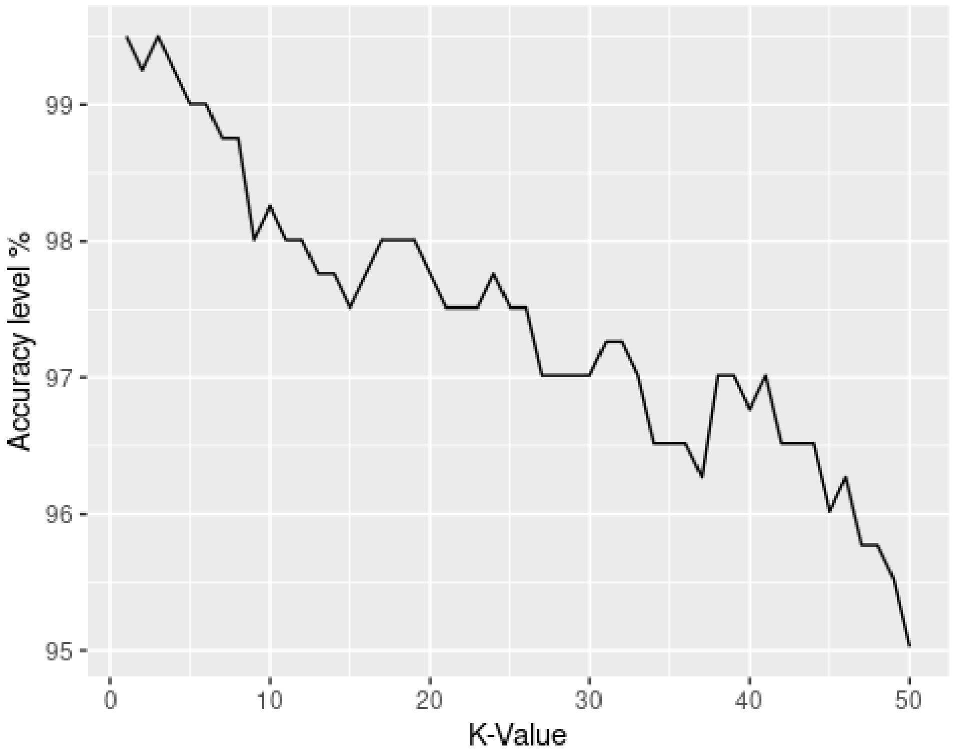

Figure 26. K-Values vs. Accuracy. Visualization of accuracy levels as K value increases.

```
suppressWarnings(knn.31<-knn(train=train.data,test=test.data,cl=train.data_cl
ass,k=3))
acc31<- 100*sum(test.data_class==knn.31)/nrow(test.data)
acc31
```

```
## [1] 99.50249
```

Part 5

5.1 Principal Results/Findings

The goal of this EDA was to attempt to create predictive models of medical bill charges using the following parameters: age, sex, BMI, smoker, number of children dependencies, and geographic region. Three models including a multiple linear regression, a decision tree, and a K-nearest neighbor were created. The K-nearest neighbor model showed the most success with the ability to assign data to the proper charge group 99.5% of the time. In a close second, the decision tree model had a predictive success of 96%. Finally, the multiple linear regression had the lowest success rate with a 75% correlation of data. It is also noteworthy that the multiple linear regression cannot predict the charge group and is therefore not an ideal model.

5.2 Limitations

One of the largest limitations of this dataset would be that the variables did not embody all of the factors that could contribute to the total medical cost for each patient. This could cause false conclusions to be made about the weight of importance for certain variables that may have in reality been attributed to a different cause. Additionally, the data set contained a limited number of data points, which could be improved upon with an increased sample population. The three price ranges also contain a large variability and lack of consistency in the groups. The first group ranges from $0-$5000, the second group ranges from $5,001-$15,000, and the third group is any value above $15,001. The inconsistency between these groups could skew the data and decrease the statistical significance of the findings. As with any machine learning model, the predictive success of the model is only as good as the data it is fed. To improve success we would need to feed the models more data and continue testing with data the model has yet to receive. Still, even with more data, these models can only be so predictive, and looking into different model options may also be beneficial.

5.3 Conclusion

The goal of this analysis was to explore the “Medical Cost Personal Datasets” dataset from Kaggle.com and develop predictive models from the given data. Specifically, a model of the price of the medical bill was desired. Since the results of attempting to model the exact charge of a medical bill would be dismal, predictive models were made the determine if a medical charge would be in a low group (<$5,000), intermediate group ($5,001-$15,000), or a high group (>$15,001). Most data points fell within the low group. It was found that a K-nearest neighbor model was the most successful in predicting these charge groups with a success rate of 99.5% compared to the success rates of the decision tree and multiple linear regression models, which had success rates of 96% and 75%, respectively. Due to the multivariate nature of this data set, inferences of the charge groups that single data types belong to could not be inferred. Further exploration of this data set might involve other model types as well as a larger dataset for more accurate models.

References

1. Dieleman, J.L., Squires, E., Bui, A.L., Campbell, M., Chapin, A., Hamavid, H., Horst, C., Li, Z., Matyasz, T., Reynolds, A., Sadat, N., Schneider, M.T., Murray, C.J., 2017. Factors associated with increases in US health care spending, 1996-2013. JAMA 318, 1668. doi:10.1001/jama.2017.15927
2. N.d. . National Health Expenditure Accounts: Methodology Paper, 2021 - CMS. URL https://www.cms.gov/files/document/definitions-sources-and-methods.pdf (accessed 12.8.23).
3. McCullough, J.M., Speer, M., Magnan, S., Fielding, J.E., Kindig, D., Teutsch, S.M., 2020. Reduction in US health care spending required to meet the Institute of Medicine's 2030 target. American Journal of Public Health 110, 1735–1740. doi:10.2105/ajph.2020.305793
4. 20, Updatedm., 2023. Trends in health care spending [WWW Document]. American Medical Association. URL https://www.ama-assn.org/about/research/trends-health-care-spending (accessed 12.8.23).
5. Cygańska, M., Kludacz-Alessandri, M., Pyke, C., 2023. Healthcare costs and health status: Insights from the share survey. International Journal of Environmental Research and Public Health 20, 1418. doi:10.3390/ijerph20021418
6. Kaushik, K., Bhardwaj, A., Dwivedi, A.D., Singh, R., 2022. Machine learning-based regression framework to predict health insurance premiums. International Journal of Environmental Research and Public Health 19, 7898. doi:10.3390/ijerph19137898
7. Choi, M., 2018. Medical Cost Personal Datasets [WWW Document]. Kaggle. URL https://www.kaggle.com/datasets/mirichoi0218/insurance (accessed 12.8.23).