Report on

IST 687 M001

Introduction to Data Science

Final Project

By
Bhanu Kirrann Garikipati
Sachin Samant
Snehal Rajesh Yadav
Vishal Reddy Vanga

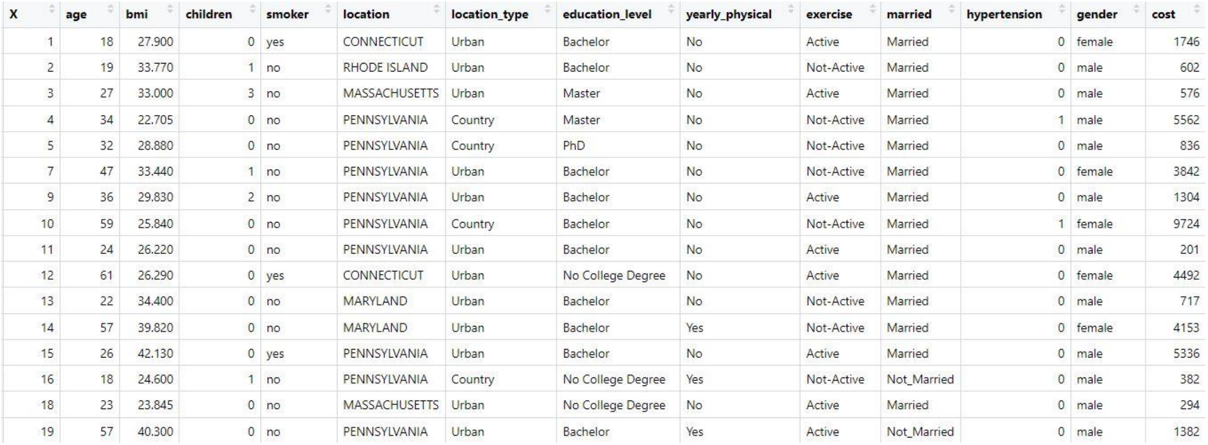
CONTENTS

Project Abstract

This project involves analyzing a healthcare cost dataset from a Health Management Organization (HMO) to gain insights into the key drivers of expensive healthcare costs and predict which customers are likely to be expensive. The project has two main goals: to predict people who will spend a lot of money on healthcare next year and to provide actionable insight to the HMO on how to lower their total healthcare costs by giving specific recommendations.

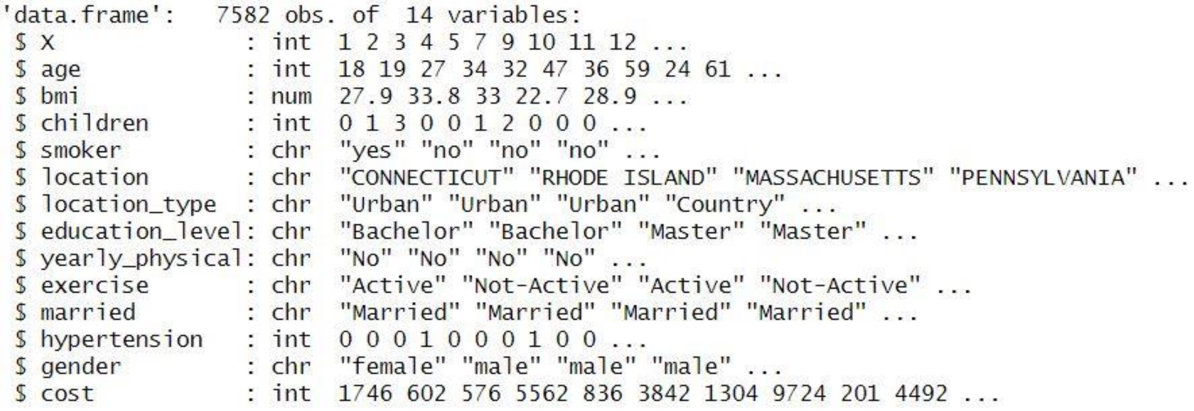
The dataset includes variables such as age, location, exercise habits, smoking status, BMI, yearly physical, hypertension, gender, education level, marital status, number of children, and total healthcare costs. The team will perform exploratory data analysis, including histograms, box plots, and mapping visualizations, to understand the dataset's characteristics. They will also use several machine learning techniques to predict which people will have high healthcare costs.

Exploratory Analysis



X	age	bmi	children	smoker	location	location_type	education_level	yearly_physical	exercise	married	hypertension	gender	cost
1	18	27.900	0	yes	CONNECTICUT	Urban	Bachelor	No	Active	Married	0	female	1746
2	19	33.770	1	no	RHODE ISLAND	Urban	Bachelor	No	Not-Active	Married	0	male	602
3	27	33.000	3	no	MASSACHUSETTS	Urban	Master	No	Active	Married	0	male	576
4	34	22.705	0	no	PENNSYLVANIA	Country	Master	No	Not-Active	Married	1	male	5562
5	32	28.880	0	no	PENNSYLVANIA	Country	PhD	No	Not-Active	Married	0	male	836
7	47	33.440	1	no	PENNSYLVANIA	Urban	Bachelor	No	Not-Active	Married	0	female	3842
9	36	29.830	2	no	PENNSYLVANIA	Urban	Bachelor	No	Active	Married	0	male	1304
10	59	25.840	0	no	PENNSYLVANIA	Country	Bachelor	No	Not-Active	Married	1	female	9724
11	24	26.220	0	no	PENNSYLVANIA	Urban	Bachelor	No	Active	Married	0	male	201
12	61	26.290	0	yes	CONNECTICUT	Urban	No College Degree	No	Active	Married	0	female	4492
13	22	34.400	0	no	MARYLAND	Urban	Bachelor	No	Not-Active	Married	0	male	717
14	57	39.820	0	no	MARYLAND	Urban	Bachelor	Yes	Not-Active	Married	0	female	4153
15	26	42.130	0	yes	PENNSYLVANIA	Urban	Bachelor	No	Active	Married	0	male	5336
16	18	24.600	1	no	PENNSYLVANIA	Country	No College Degree	Yes	Not-Active	Not_Married	0	male	382
18	23	23.845	0	no	MASSACHUSETTS	Urban	No College Degree	No	Active	Married	0	male	294
19	57	40.300	0	no	PENNSYLVANIA	Urban	Bachelor	Yes	Active	Not_Married	0	male	1382

To analyze the data, firstly we must retrieve the data from the link. so, we used the read.csv function in the tidyverse library. This function reads and retrieves the data from the link and stores it as data set form in a dataset named vector.



```
'data.frame':   7582 obs. of  14 variables:
 $ X              : int  1 2 3 4 5 7 9 10 11 12 ...
 $ age            : int  18 19 27 34 32 47 36 59 24 61 ...
 $ bmi            : num  27.9 33.8 33 22.7 28.9 ...
 $ children       : int  0 1 3 0 0 1 2 0 0 0 ...
 $ smoker         : chr  "yes" "no" "no" "no" ...
 $ location       : chr  "CONNECTICUT" "RHODE ISLAND" "MASSACHUSETTS" "PENNSYLVANIA" ...
 $ location_type  : chr  "Urban" "Urban" "Urban" "Country" ...
 $ education_level: chr  "Bachelor" "Bachelor" "Master" "Master" ...
 $ yearly_physical: chr  "No" "No" "No" "No" ...
 $ exercise       : chr  "Active" "Not-Active" "Active" "Not-Active" ...
 $ married        : chr  "Married" "Married" "Married" "Married" ...
 $ hypertension   : int  0 0 0 1 0 0 0 1 0 0 ...
 $ gender         : chr  "female" "male" "male" "male" ...
 $ cost           : int  1746 602 576 5562 836 3842 1304 9724 201 4492 ...
```

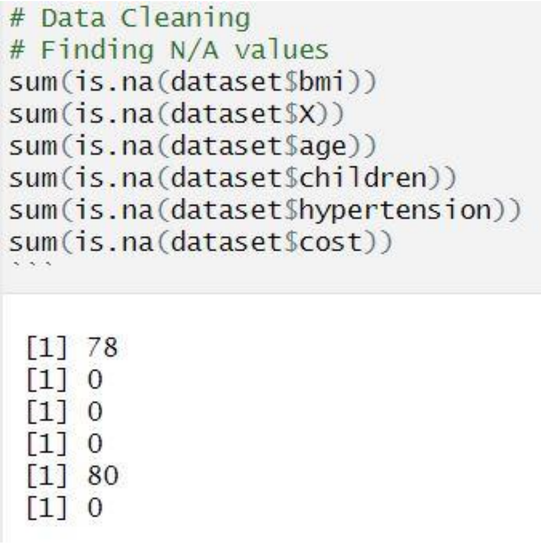
By using str functions, we get to know there are 14 columns and 7582 rows in the dataset. The whole dataset gives information of the expenses of a person with different habits.in the dataset there are integer, numerical and character data type variables. The variables are

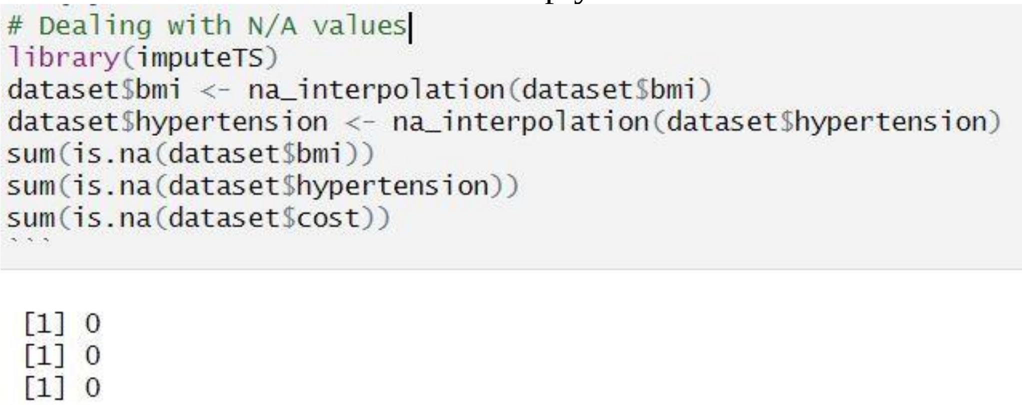
- **X**: Integer, Unique identified for each person
- **age**: Integer, The age of the person (at the end of the year).
- **location**: Categorical, the name of the state (in the United States) where the person lived (at the end of the year)
- **location_type**: Categorical, a description of the environment where the person lived (urban or country).

- **exercise**: Categorical, “Not Active” if the person did not exercise regularly during the year, “Active” if the person did exercise regularly during the year.
- **smoker**: Categorical, “yes” if the person smoked during the past year, “no” if the person didn’t smoke during the year.
- **bmi**: Integer, the body mass index of the person. The body mass index (BMI) is a measure that uses your height and weight to work out if your weight is healthy.
- **yearly_physical**: Categorical, “yes” if the person had a well visit (yearly physical) with their doctor during the year. “no” if the person did not have a well visit with their doctor.
- **Hypertension**: “0” if the person did not have hypertension.
- **gender**: Categorical, the gender of the person
- **education_level**: Categorical, the amount of college education ("No College Degree", "Bachelor", "Master", "PhD")
- **married**: Categorical, describing if the person is “Married” or “Not_Married”
- **num_children**: Integer, Number of children
- **cost**: Integer, the total cost of health care for that person, during the past year.

Data Cleaning

Once we are done with the data exploration then the next step is to check if there are any empty cells in the variables. If there are empty cells, then we must clean the data. The following function will give output of number of cells that are empty in the mentioned variable. The results show that there are 78 and 80 empty cells in the bmi and hypertension variables.

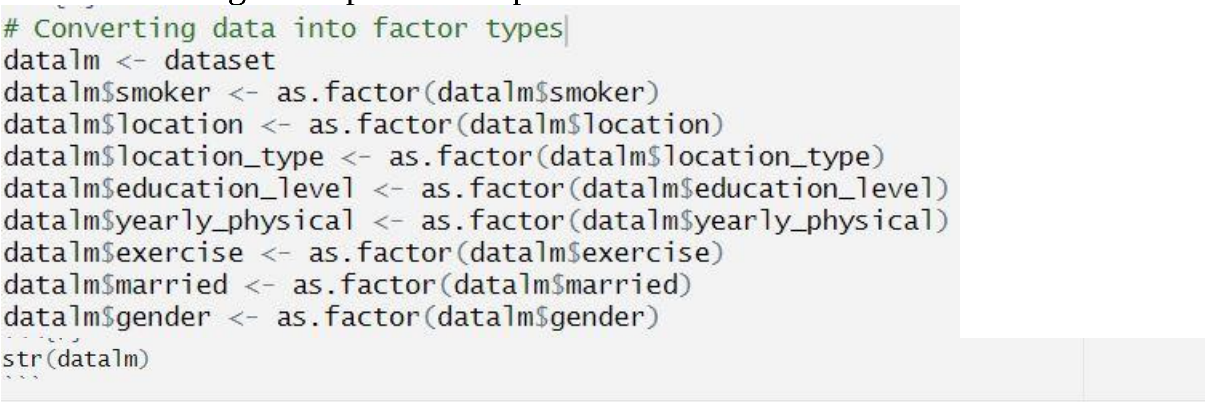


~~~~
# Data Cleaning
# Finding N/A values
sum(is.na(dataset$bmi))
sum(is.na(dataset$X))
sum(is.na(dataset$age))
sum(is.na(dataset$children))
sum(is.na(dataset$hypertension))
sum(is.na(dataset$cost))
```
~~~~

```
[1] 78
[1] 0
[1] 0
[1] 0
[1] 80
[1] 0
```

Now, we must clean those empty cells in the bmi and hypertension variables. To do the cleaning we chose to use na_interpolation function in the imputeTS library. This function will clean the data in those mentioned variables. Again, used is.na() function to verify whether the na_interpolation is worked is or not and the result shows there are no empty cells in the variables.



~~~~
# Dealing with N/A values
library(imputeTS)
dataset$bmi <- na_interpolation(dataset$bmi)
dataset$hypertension <- na_interpolation(dataset$hypertension)
sum(is.na(dataset$bmi))
sum(is.na(dataset$hypertension))
sum(is.na(dataset$cost))
```
~~~~

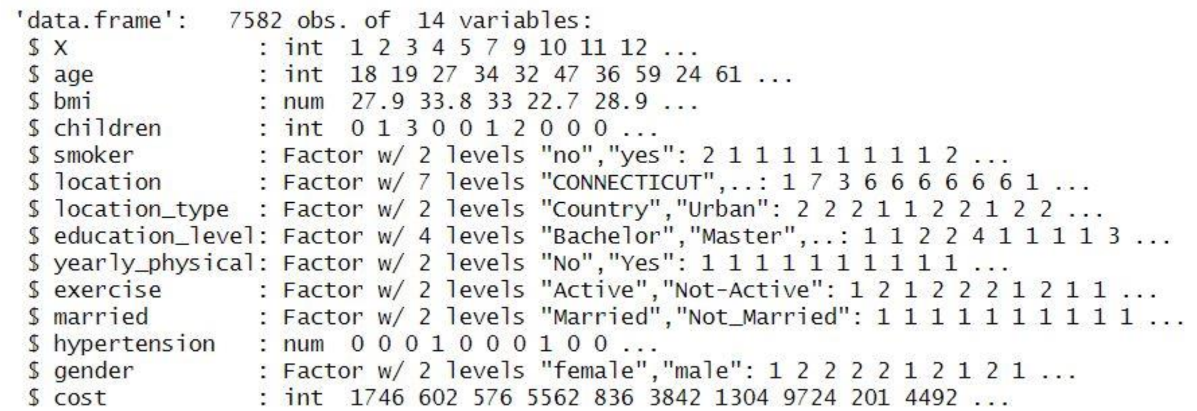
```
[1] 0
[1] 0
[1] 0
```

Predictive Analysis

We stored the dataset in the datalm and then converted all the character data type into the factor data type so that it will be best to find out which variables will be affecting the expenses if a person.

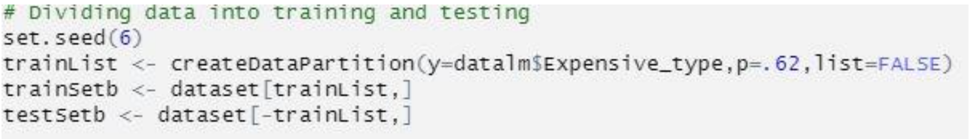


```
# Converting data into factor types
datalm <- dataset
datalm$smoker <- as.factor(datalm$smoker)
datalm$location <- as.factor(datalm$location)
datalm$location_type <- as.factor(datalm$location_type)
datalm$education_level <- as.factor(datalm$education_level)
datalm$yearly_physical <- as.factor(datalm$yearly_physical)
datalm$exercise <- as.factor(datalm$exercise)
datalm$married <- as.factor(datalm$married)
datalm$gender <- as.factor(datalm$gender)
str(datalm)
```



```
'data.frame':   7582 obs. of  14 variables:
 $ X              : int  1 2 3 4 5 7 9 10 11 12 ...
 $ age            : int  18 19 27 34 32 47 36 59 24 61 ...
 $ bmi            : num  27.9 33.8 33 22.7 28.9 ...
 $ children       : int  0 1 3 0 0 1 2 0 0 0 ...
 $ smoker         : Factor w/ 2 levels "no","yes": 2 1 1 1 1 1 1 1 1 2 ...
 $ location       : Factor w/ 7 levels "CONNECTICUT",..: 1 7 3 6 6 6 6 6 6 1 ...
 $ location_type  : Factor w/ 2 levels "Country","Urban": 2 2 2 1 1 2 2 1 2 2 ...
 $ education_level: Factor w/ 4 levels "Bachelor","Master",..: 1 1 2 2 4 1 1 1 1 3 ...
 $ yearly_physical: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ exercise       : Factor w/ 2 levels "Active","Not-Active": 1 2 1 2 2 2 1 2 1 1 ...
 $ married        : Factor w/ 2 levels "Married","Not_Married": 1 1 1 1 1 1 1 1 1 1 ...
 $ hypertension   : num  0 0 0 1 0 0 0 1 0 0 ...
 $ gender         : Factor w/ 2 levels "female","male": 1 2 2 2 2 1 2 1 2 1 ...
 $ cost           : int  1746 602 576 5562 836 3842 1304 9724 201 4492 ...
```

After creating HMO, we then divided it into two parts. We used the first part for training, and the last part for testing. In the testing dataset, we removed the cost exp variable, as we are testing for the same factor.



```
# Dividing data into training and testing
set.seed(6)
trainList <- createDataPartition(y=datalm$Expensive_type,p=.62,list=FALSE)
trainSetb <- dataset[trainList,]
testSetb <- dataset[-trainList,]
```

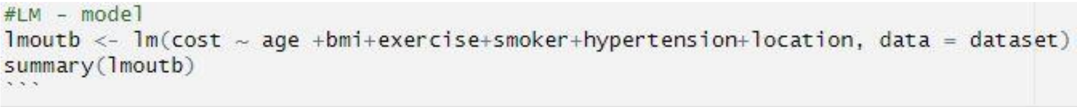
After the division of data frame, we now have the dataset to train the models as well as test them.

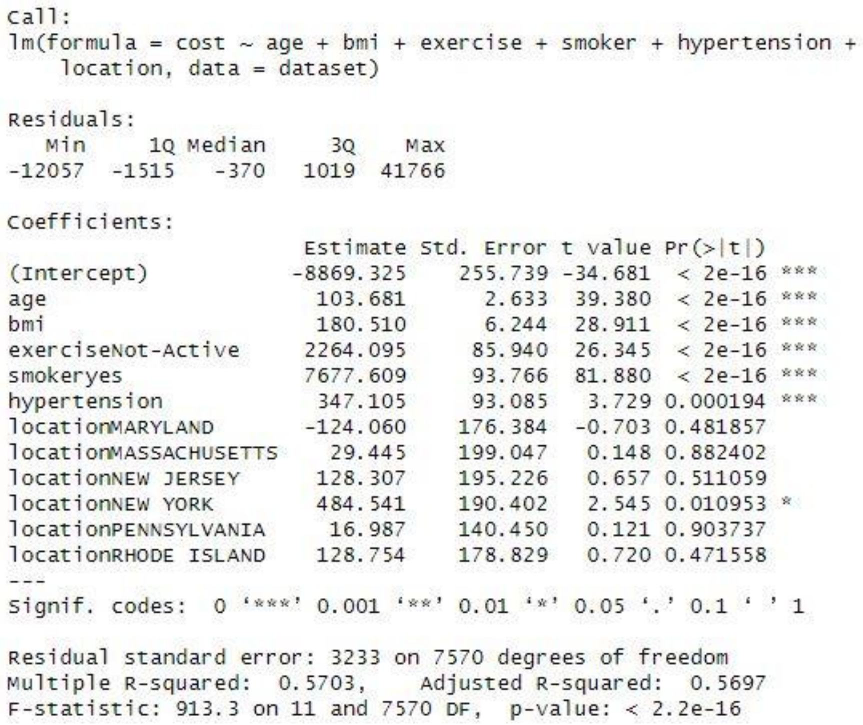
For our project, we considered three models, SVM model, Tree Model, and Linear Model.

Models

Linear model

To get more statistical information between the cost and other variables in the dataset we choose the linear regression model. In this model we used the datalm dataset where there are no chr data types. The resulting linear model is significant due to its p-value is less than 0.05 and its r-squared value is 56.97%. By looking at the p-value of the variables we can determine which values are significant and these variables will be considered to determine the expenses in the further models we used.



```
#LM - model
lmoutb <- lm(cost ~ age +bmi+exercise+smoker+hypertension+location, data = dataset)
summary(lmoutb)
```
```



```
Call:
lm(formula = cost ~ age + bmi + exercise + smoker + hypertension +
    location, data = dataset)

Residuals:
   Min   1Q Median   3Q   Max
-12057 -1515   -370   1019 41766

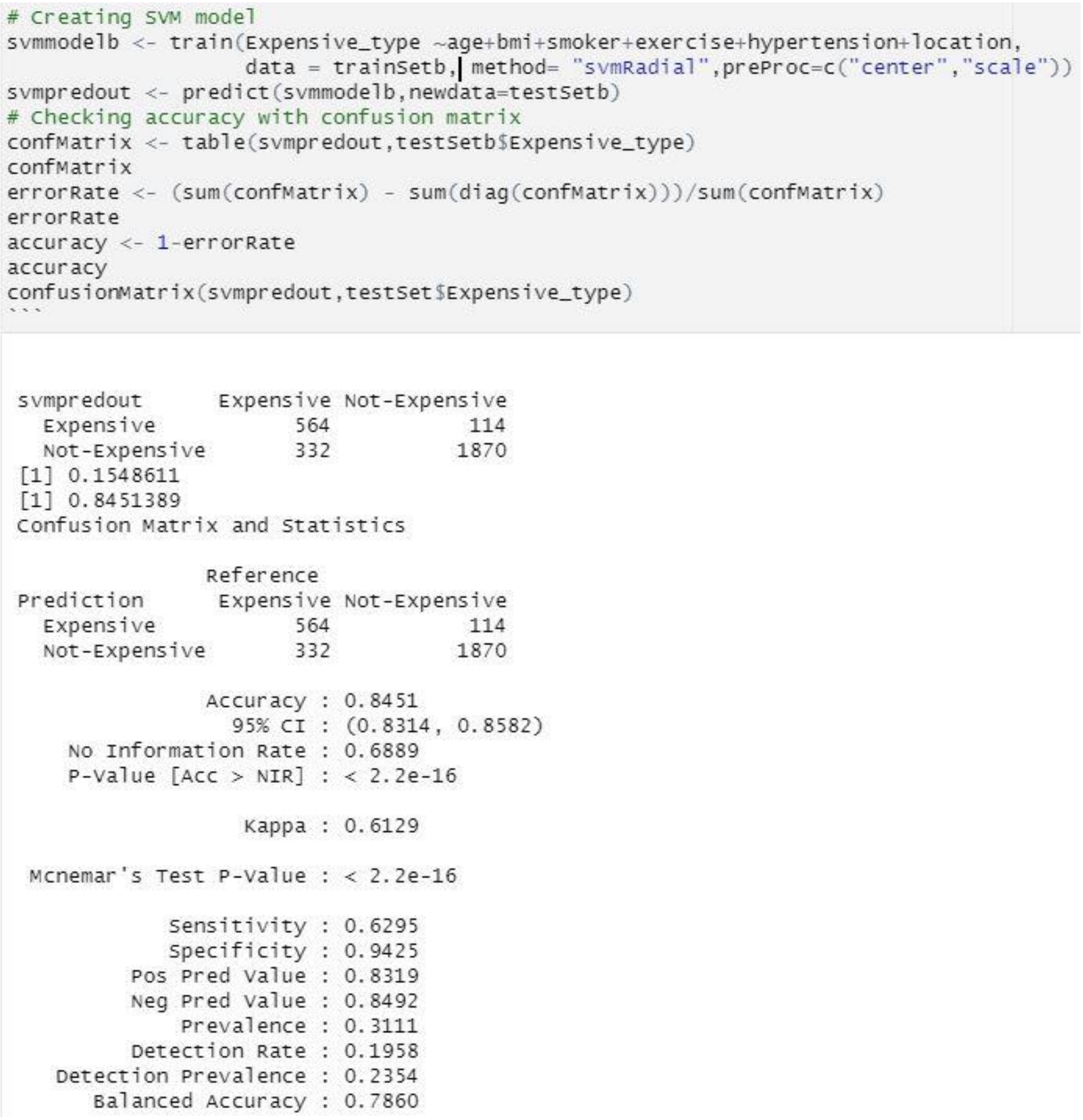
Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)   -8869.325   255.739 -34.681 < 2e-16 ***
age   103.681   2.633 39.380 < 2e-16 ***
bmi   180.510   6.244 28.911 < 2e-16 ***
exerciseNot-Active   2264.095   85.940 26.345 < 2e-16 ***
smokeryes   7677.609   93.766 81.880 < 2e-16 ***
hypertension   347.105   93.085   3.729 0.000194 ***
locationMARYLAND   -124.060   176.384 -0.703 0.481857
locationMASSACHUSETTS   29.445   199.047   0.148 0.882402
locationNEW JERSEY   128.307   195.226   0.657 0.511059
locationNEW YORK   484.541   190.402   2.545 0.010953 *
locationPENNSYLVANIA   16.987   140.450   0.121 0.903737
locationRHODE ISLAND   128.754   178.829   0.720 0.471558

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3233 on 7570 degrees of freedom
Multiple R-squared: 0.5703,	Adjusted R-squared: 0.5697
F-statistic: 913.3 on 11 and 7570 DF, p-value: < 2.2e-16
```
```

SVM Model

The following code will test the svm model using testSet. We created the confusion Matrix from the testing results so that we can see how much accuracy and sensitivity this model has. This confusionMatrix function will give use the accuracy without any calculation.

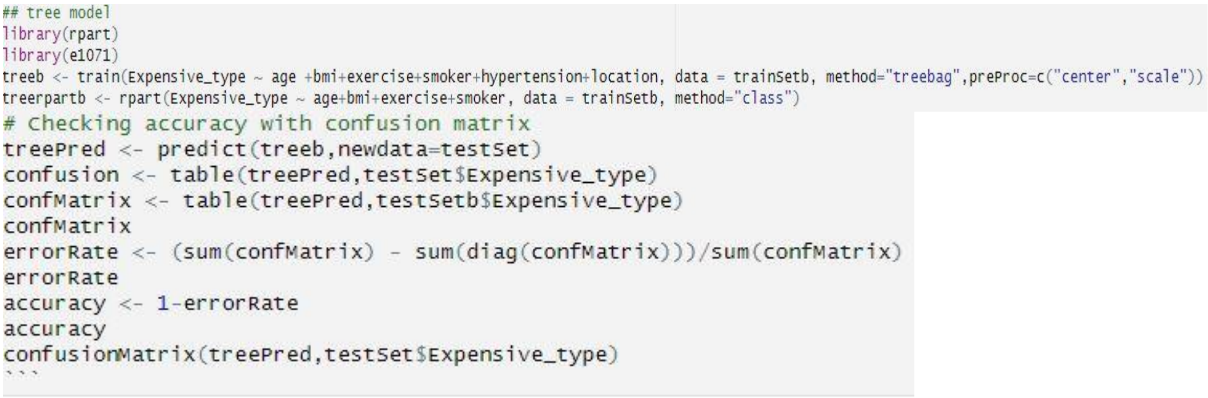


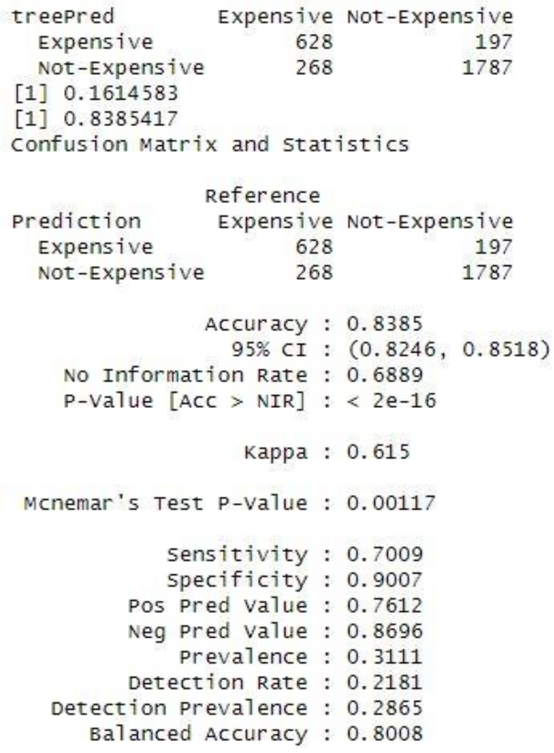
~~~
# Creating SVM model
svmmodelb <- train(Expensive_type ~age+bmi+smoker+exercise+hypertension+location,
                   data = trainSetb, method= "svmRadial",preProc=c("center","scale"))
svmpredout <- predict(svmmodelb,newdata=testSetb)
# Checking accuracy with confusion matrix
confMatrix <- table(svmpredout,testSetb$Expensive_type)
confMatrix
errorRate <- (sum(confMatrix) - sum(diag(confMatrix)))/sum(confMatrix)
errorRate
accuracy <- 1-errorRate
accuracy
confusionMatrix(svmpredout,testSet$Expensive_type)
```
~~~

```
svmpredout      Expensive Not-Expensive
  Expensive           564           114
  Not-Expensive       332          1870
[1] 0.1548611
[1] 0.8451389
Confusion Matrix and Statistics

               Reference
Prediction      Expensive Not-Expensive
  Expensive           564           114
  Not-Expensive       332          1870

               Accuracy : 0.8451
                 95% CI : (0.8314, 0.8582)
    No Information Rate : 0.6889
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6129

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.6295
            Specificity : 0.9425
         Pos Pred Value : 0.8319
         Neg Pred Value : 0.8492
             Prevalence : 0.3111
         Detection Rate : 0.1958
   Detection Prevalence : 0.2354
      Balanced Accuracy : 0.7860
```

Treebag Model

The following code will test the tree model using testSet. We created the confusion Matrix from the testing results so that we can see how much accuracy and sensitivity this model has. This confusionMatrix function will give use the accuracy without any calculation.



```
## tree model
library(rpart)
library(e1071)
treeb <- train(Expensive_type ~ age +bmi+exercise+smoker+hypertension+location, data = trainSetb, method="treebag",preProc=c("center","scale"))
treerpartb <- rpart(Expensive_type ~ age+bmi+exercise+smoker, data = trainSetb, method="class")
# Checking accuracy with confusion matrix
treePred <- predict(treeb,newdata=testSet)
confusion <- table(treePred,testSet$Expensive_type)
confMatrix <- table(treePred,testSetb$Expensive_type)
confMatrix
errorRate <- (sum(confMatrix) - sum(diag(confMatrix)))/sum(confMatrix)
errorRate
accuracy <- 1-errorRate
accuracy
confusionMatrix(treePred,testSet$Expensive_type)
```
```



```
treePred        Expensive Not-Expensive
  Expensive           628           197
  Not-Expensive       268          1787
[1] 0.1614583
[1] 0.8385417
Confusion Matrix and Statistics

               Reference
Prediction      Expensive Not-Expensive
  Expensive           628           197
  Not-Expensive       268          1787

               Accuracy : 0.8385
                 95% CI : (0.8246, 0.8518)
    No Information Rate : 0.6889
    P-Value [Acc > NIR] : < 2e-16

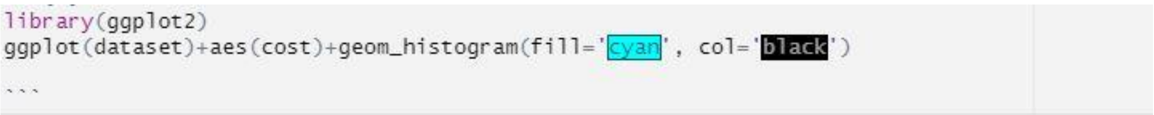
                  Kappa : 0.615

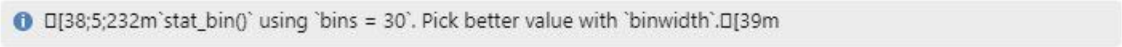
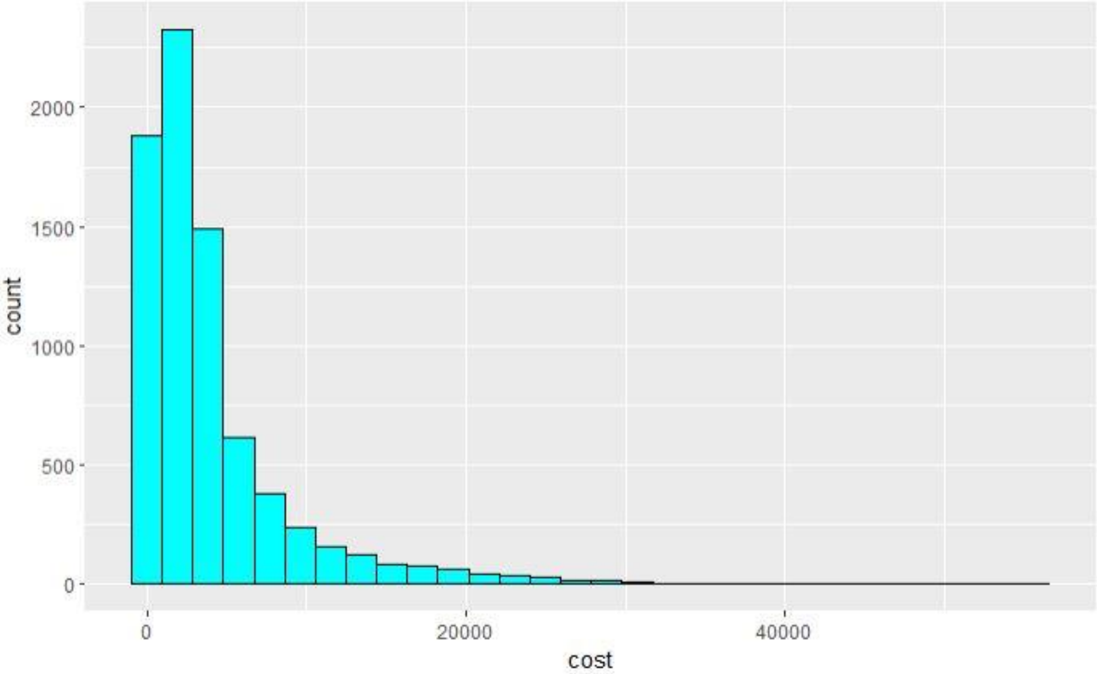
 Mcnemar's Test P-Value : 0.00117

            Sensitivity : 0.7009
            Specificity : 0.9007
         Pos Pred Value : 0.7612
         Neg Pred Value : 0.8696
             Prevalence : 0.3111
         Detection Rate : 0.2181
   Detection Prevalence : 0.2865
      Balanced Accuracy : 0.8008
```
```

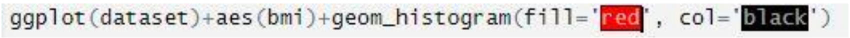
# Visualizations

## Cost Histogram

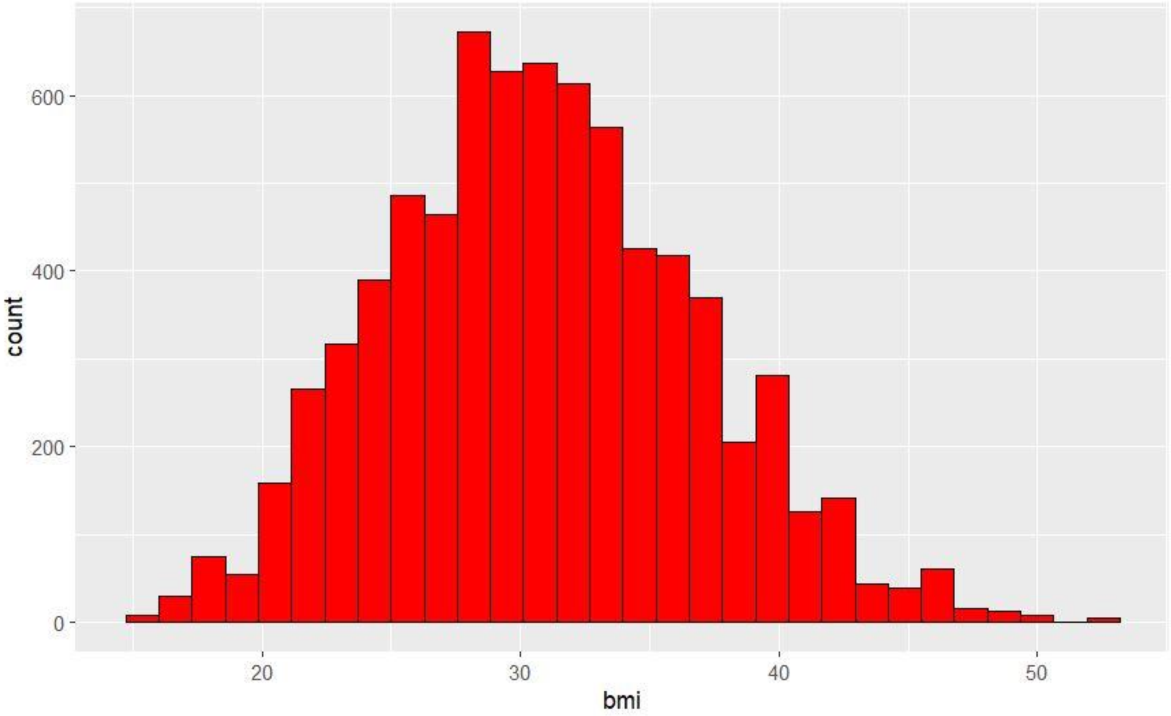


```
library(ggplot2)
ggplot(dataset)+aes(cost)+geom_histogram(fill='cyan', col='black')

```
```



```
□[38;5;232m`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.□[39m
```



To get more clarity on the cost variable, we created a histogram graph, and the resulting graph is a skewed shaped graph which means most of the bars are on the left side of the graph. these most frequent bars are in range of between 4000 to 5000 and the graphs there are out-liners with only frequency of 1 so we choose to mean of the cost variable as a margin cost to determine whether the expenses are expensive are not.
```

# BMI histogram

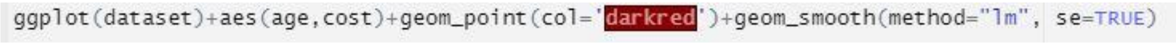


```
ggplot(dataset)+aes(bmi)+geom_histogram(fill='red', col='black')
```

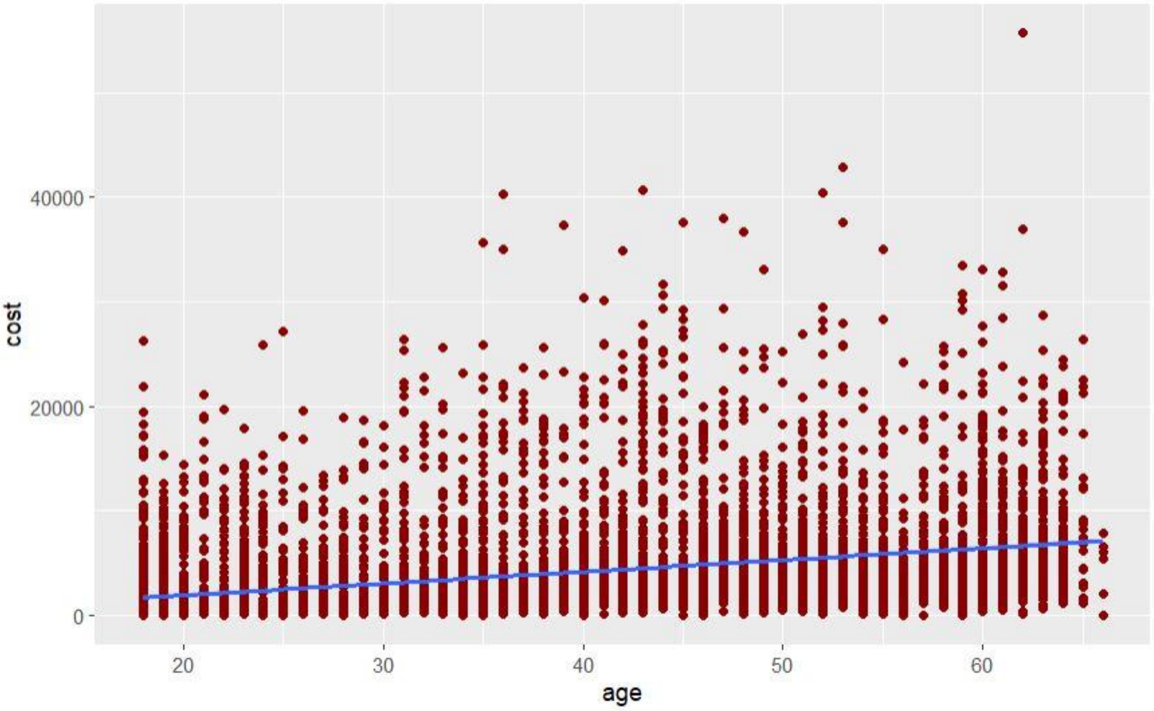


This is a histogram of a bmi variable, and the resulting graph is bell-curve shaped which means most of the frequent bmi values are situated around the median of the variable, but the median is 30.50. As per the standard chart, if bmi is greater than 30 then that person is suffering with obesity.

Scatter plot of Age vs Cost

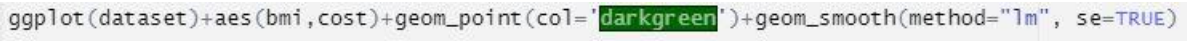


```
ggplot(dataset)+aes(age,cost)+geom_point(col='darkred')+geom_smooth(method="lm", se=TRUE)
```

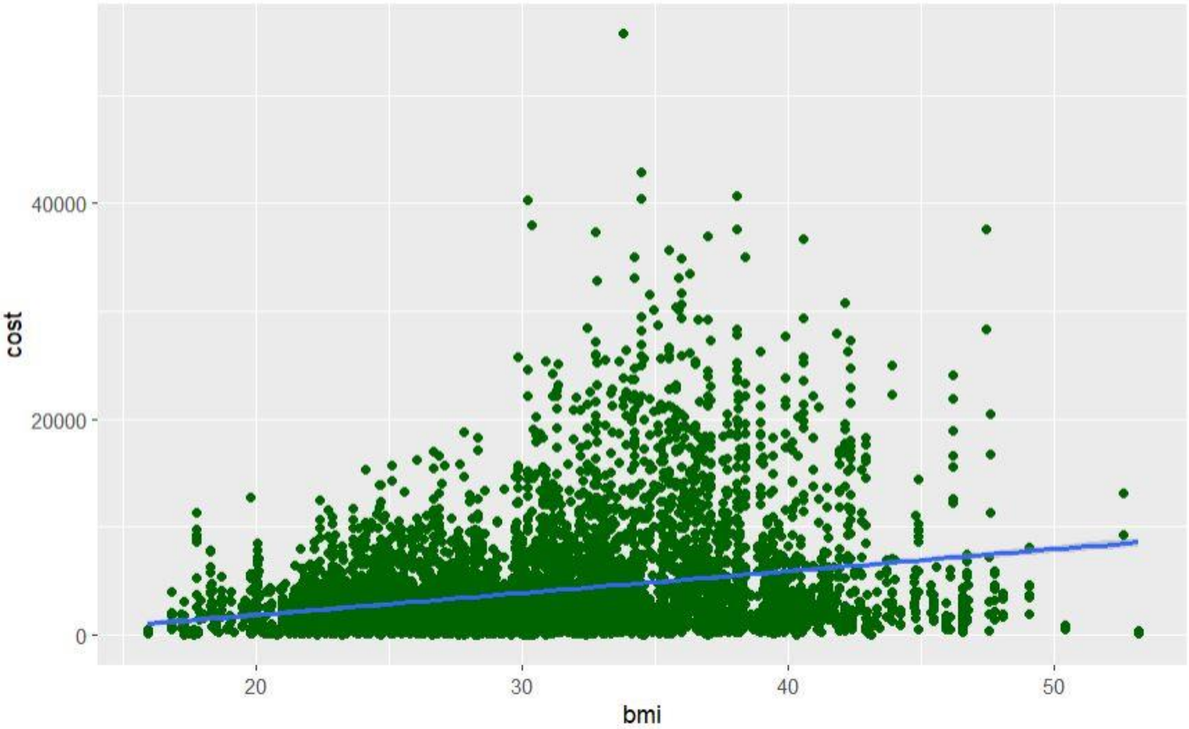


We can clearly see in the plot, as age is increasing the healthcare expenses increase. When they are younger, their body can cope with the sedentary lifestyle, but as they age their body gives out which results in medical issues.

Scatter plot of BMI vs Cost

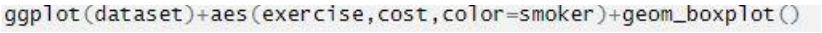


```
ggplot(dataset)+aes(bmi,cost)+geom_point(col='darkgreen')+geom_smooth(method="lm", se=TRUE)
```

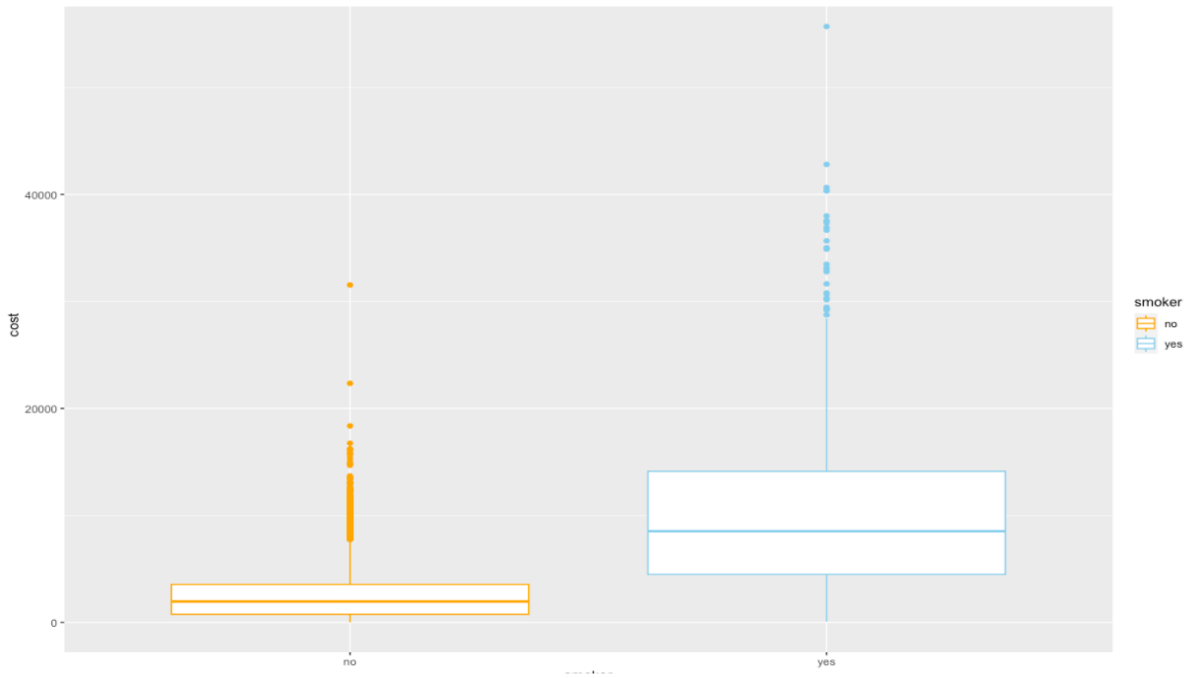


In the plot, we could clearly see that as the BMI goes on increasing, the cost also increased. The healthy range of BMI is 18.5-24.9, but the BMI in our dataset in on the higher end. As higher BMI is associated with poor health, it is reasonable to assume that by tackling BMI, we can reduce the health care cost. HMO should encourage their customers to reduce weight to reduce their BMI.

Boxplot of Smoker vs Cost

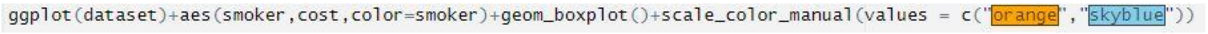


```
ggplot(dataset)+aes(exercise,cost,color=smoker)+geom_boxplot()
```

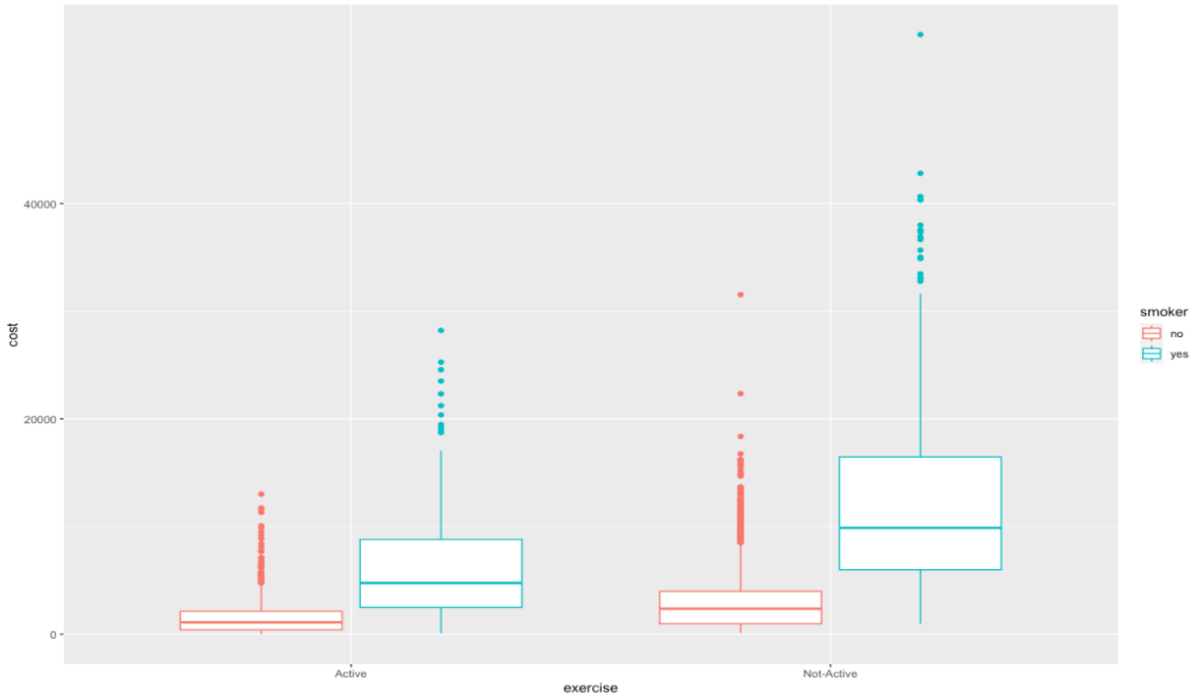


In the boxplot above we can see a clear difference between the expenses of an active smoker and the expenses of a non-smoker. The healthcare expenses of a smoker are significantly high when compared to that of a non-smoker. HMO should encourage people to quit smoking by educating them about the long-term impact of smoking and conducting de-addiction programs.

Boxplot of Exercise vs Cost



```
ggplot(dataset)+aes(smoker,cost,color=smoker)+geom_boxplot()+scale_color_manual(values = c("orange","skyblue"))
```

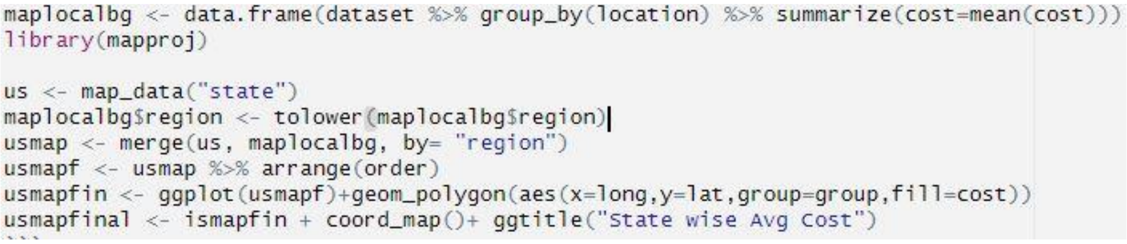


In the above plot, we can se thee impact of exercise and smoking have on health and medical expenses. The order of expenses of are as follows:

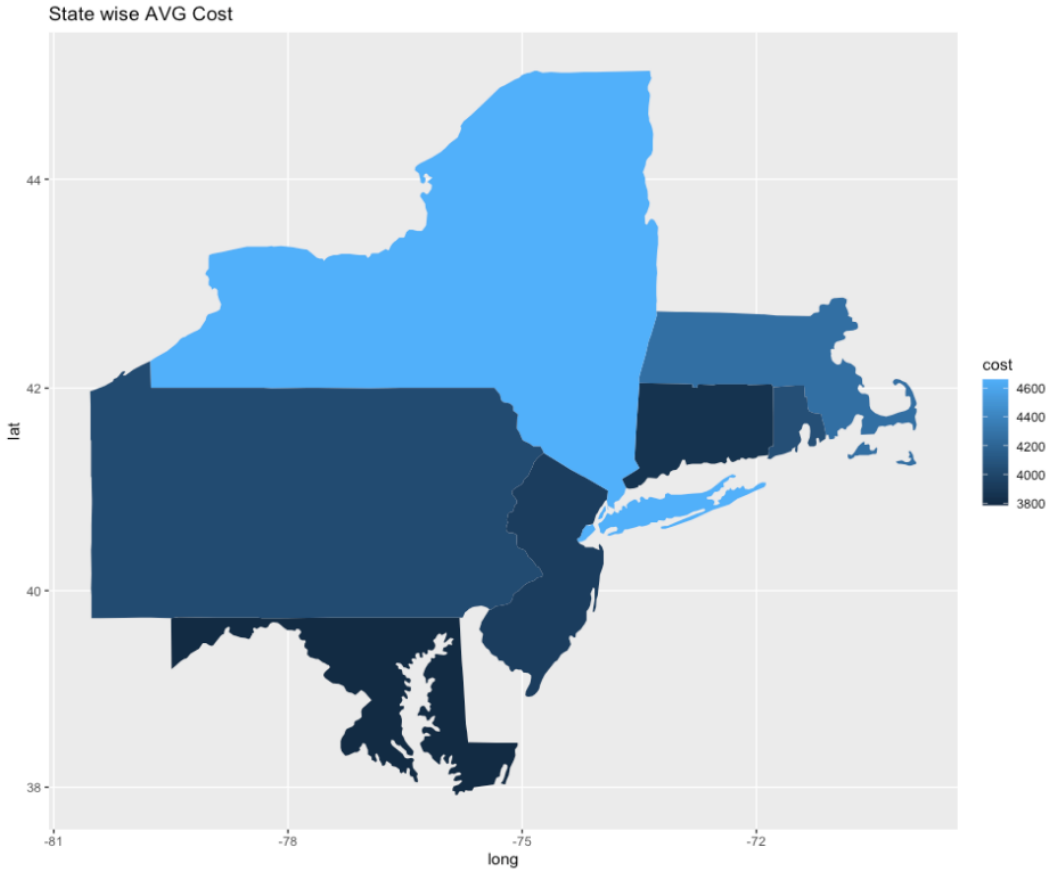
(Person who smokes and doesn’t exercise) > (person who is a non-smoker and doesn’t exercise) > (person who smokes and exercises actively) > (person who is a non-smoker and exercises actively).

Accordingly, it would be very helpful if HMO would conduct monthly workshops and educate people regarding the insights and show them how their day-to-day activities affect their healthcare expenses.

US Map of cost per state

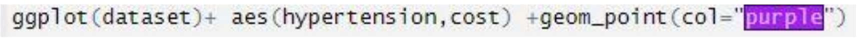


```
maplocalbg <- data.frame(dataset %>% group_by(location) %>% summarize(cost=mean(cost)))
library(mapproj)

us <- map_data("state")
maplocalbg$region <- tolower(maplocalbg$region)
usmap <- merge(us, maplocalbg, by= "region")
usmapf <- usmap %>% arrange(order)
usmapfin <- ggplot(usmapf)+geom_polygon(aes(x=long,y=lat,group=group,fill=cost))
usmapfinal <- ismapfin + coord_map()+ ggtitle("State wise Avg Cost")
```

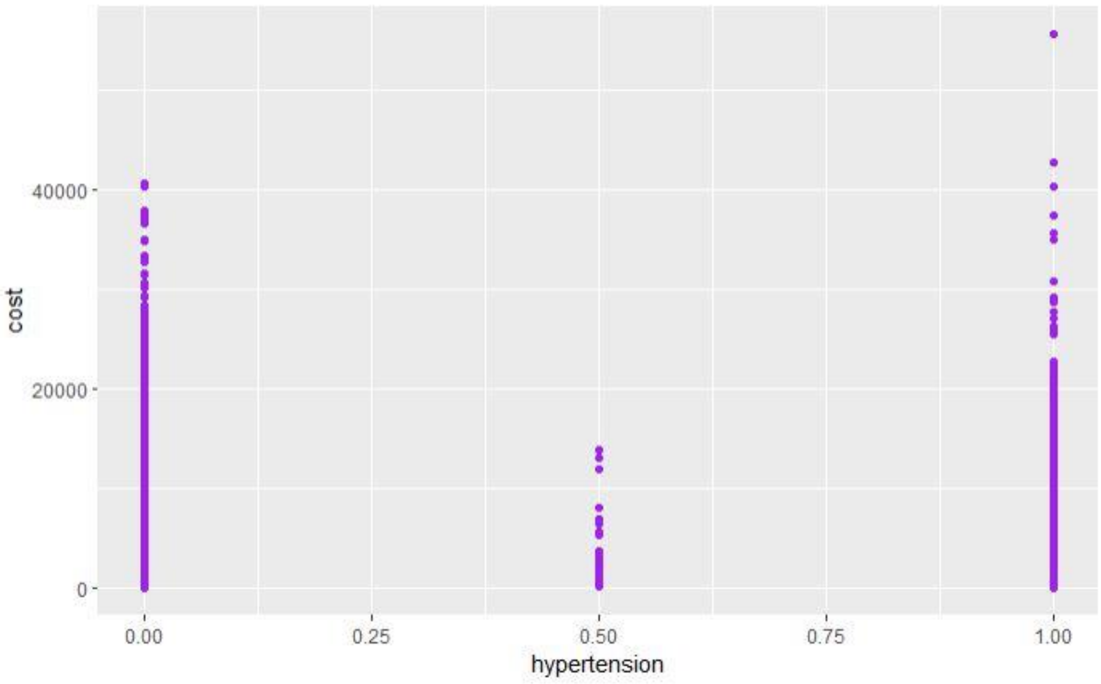


New Yorker's mean healthcare expenses are costlier compared to the other states.

Scatter plot of hypertension vs Cost



```
ggplot(dataset)+ aes(hypertension,cost) +geom_point(col="purple")
```



The above plot depicts the relation between cost and how different stages of hypertension affect it.

Recommendations

We recommend the following course of actions for HMO:

1) Fitness importance: HMOs should hold monthly workshops and educate people about keeping active and not smoking, as well as the direct relationship of those to their healthcare expenses.
2) BMI: Design a healthcare program which encourages obese people to lose weight. Providing an incentive for a healthier BMI would encourage people to take a necessary course of action to receive that incentive.
3) Hypertension: HMO could create an awareness among the people on how to avoid hypertension for the people with hypertension=0. For hypertension=0.5, HMO can monitor their healthcare and give them proper guidance towards bettering their health and correcting their hypertension.

Project summary

The objective of this project was to analyze factors that influence healthcare costs and develop a predictive model to identify individuals with high-cost associations. The initial focus was on Body Mass Index (BMI) as a major factor, but through exploratory data analysis, it was found that other factors such as age, number of children in a family, smoking habits, city, education level, physical activity, annual physical checkup, marital status, hypertension, and gender also played a significant role.

After cleaning and processing the data, a new data frame was created, and several models were developed to identify individuals with high-cost association. Support Vector Machine (SVM) model was found to have the highest accuracy and sensitivity in detecting high-cost association.

The relationships between the different factors were visualized to provide insights and recommendations for the Health Maintenance Organization (HMO). The results of this analysis could help the HMO to develop targeted interventions and policies to improve health outcomes and reduce costs.