Healthcare Management Organisation (HMO)

GROUP 5
GROUP MEMBERS

Bhavya Shah
Sharvari Kairnar
Tejas Amrutkar
Khushi Shetty
Samriddhi Tiwari

CONTENTS

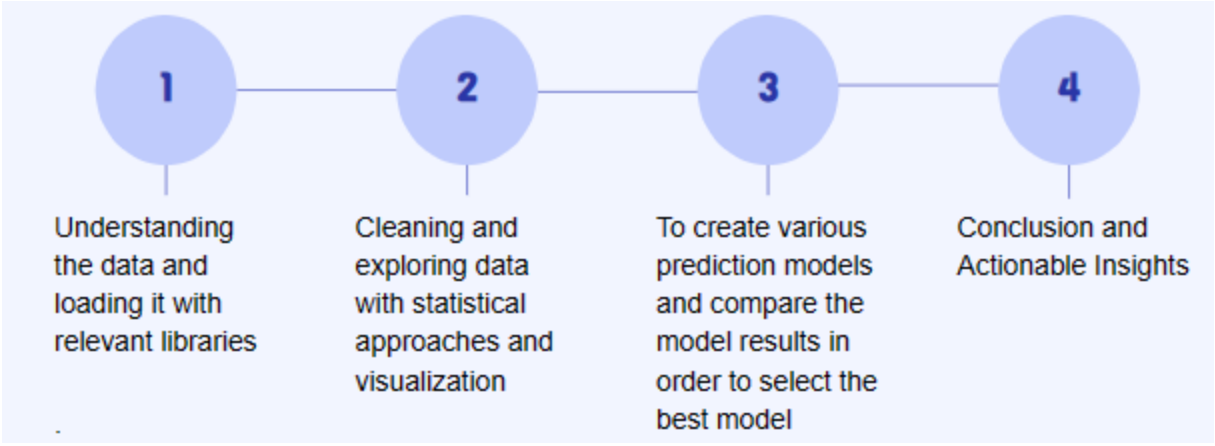
Overview

This project's primary objective is to identify the distinctive factors that make some people have higher healthcare costs than others, predict which people will have higher healthcare costs in the upcoming year, and provide the HMO with useful advice on how to cut costs.

Project Goals

The overall goal of the case is to provide actionable insight, based on the data available, as well as accurately predict which people (customers) will be expensive. The dataset utilized in our analysis contains healthcare cost information from an HMO (Health Management Organization) with several attributes such as degree level, activity level, smoker, etc.

Our team's goal is to understand the key drivers for why some people are more expensive as well as predict which people will be expensive in terms of health care costs. Our team completed the project by completing the following phases:



Once the data set was imported and cleaned up, copies and modifications were made to the data set to better fit the predictive model type. The team utilized a mixture of linear regression, support vector machine, and tree model to determine the best method to predict high costs patients. The model outputs were evaluated to predict high medical costs based on prediction accuracy, sensitivity, model significance, and statistically significant variables (i.e. p-value < 0.05). Finally, the team generated conclusions and recommendations for the insurance company to cover the higher cost patients based on our data science analysis.

Data Sets and Rstudio Package Libraries

Dataset

The data appears like this when viewed.

X	age	bmi	children	smoker	location	location_type	education_level	yearly_ph	exercise	married	hypertension	gender	cost
1	18	27.9	0	yes	CONNECTICUT	Urban	Bachelor	No	Active	Married	0	female	1746
2	19	33.77	1	no	RHODE ISLAND	Urban	Bachelor	No	Not-Active	Married	0	male	602
3	27	33	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	28.88	0	no	PENNSYLVANIA	Country	PhD	No	Not-Active	Married	0	male	836
7	47	33.44	1	no	PENNSYLVANIA	Urban	Bachelor	No	Not-Active	Married	0	female	3842
9	36	29.83	2	no	PENNSYLVANIA	Urban	Bachelor	No	Active	Married	0	male	1304
10	59	25.84	0	no	PENNSYLVANIA	Country	Bachelor	No	Not-Active	Married	1	female	9724
11	24	26.22	0	no	PENNSYLVANIA	Urban	Bachelor	No	Active	Married	0	male	201
12	61	26.29	0	yes	CONNECTICUT	Urban	No College Degree	No	Active	Married	0	female	4492
13	22	34.4	0	no	MARYLAND	Urban	Bachelor	No	Not-Active	Married	0	male	717
14	57	39.82	0	no	MARYLAND	Urban	Bachelor	Yes	Not-Active	Married	0	female	4153
15	26	42.13	0	yes	PENNSYLVANIA	Urban	Bachelor	No	Active	Married	0	male	5336
16	18	24.6	1	no	PENNSYLVANIA	Country	No College Degree	Yes	Not-Active	Not_Marri	0	male	382
18	23	23.845	0	no	MASSACHUSETTS	Urban	No College Degree	No	Active	Married	0	male	294
19	57	40.3	0	no	PENNSYLVANIA	Urban	Bachelor	Yes	Active	Not_Marri	0	male	1382
20	31	35.3	0	yes	PENNSYLVANIA	Urban	PhD	No	Not-Active	Married	0	male	15058
21	60	36.005	0	no	PENNSYLVANIA	Urban	PhD	No	Active	Married	0	female	3384
22	30	32.4	1	no	PENNSYLVANIA	Urban	Master	No	Active	Married	0	female	761

The original data set imported into Rstudio included 7,582 observations with 14 variables. The variables are described as the below:

1. **X**: Integer
    - Unique identifier for each person
2. **age**: Integer
    - The age of the person (at the end of the year).
3. **location**:
    - Categorical, the name of the state (in the United States) where the person lived (at the end of the year)
4. **location_type**: Categorical
    - a description of the environment where the person lived (urban or country).
5. **exercise**: Categorical
    - “Not-Active” if the person did not exercise regularly during the year, “Active” if the person did exercise regularly during the year.

6. **smoker**: Categorical
    - “yes” if the person smoked during the past year, “no” if the person didn’t smoke during the year.
7. **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.
8. **yearly_physical**: Categorical
    - “yes” if the person had a good visit (yearly physical) with their doctor during the year. “no” if the person did not have a good visit with their doctor.
9. **hypertension**:
    - “0” if the person did not have hypertension.
10. **gender**: Categorical
    - the gender of the person
11. **education_level**: Categorical
    - the amount of college education ("No College Degree", "Bachelor", "Master", "Ph.D.")
12. **married**: Categorical
    - describing if the person is “Married” or “Not_Married”
13. **num_children**: Integer
    - Number of children
14. **cost**: Integer
    - the total cost of healthcare for that person, during the past year.

Package Libraries

Several package libraries were needed to access the functions and capabilities in order to construct the models for this project. These are the package libraries:

PACKAGE NAME	PURPOSE
Dplyr	dplyr is a grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges: mutate() adds new variables that are functions of existing variables. select() picks variables based on their names. filter() picks cases based on their values.
Ggplot2	ggplot2 is an R package for producing statistical, or data, graphics. Our team generated box plots, bar charts, scatter plots, and maps
Tidyverse	Among other capabilities, tidyverse provides the ability to 'pipe' results from one command / function to another command / function, making code more readable. The "tidyverse" is a collection of R packages that helps reorganize and visualize data. Developed by RStudio's chief scientist Hadley Wickham, the tidyverse uses a consistent approach to build an ecosystem of packages. The tidyverse consists of many packages to help with data manipulation (e.g., dplyr), working with data types (e.g., stringr for strings) and data visualization (e.g., ggplot2).
rsample	The rsample package provides functions to create different types of resamples and corresponding classes for their analysis. The goal is to have a modular set of methods that can be used for: resampling for estimating the sampling distribution of a statistic.
Caret	the 'caret' package, which provides a robust framework for easily trying different machine learning algorithms. The package raises the abstraction level and, as such, makes it easier to write R code. Our team utilized createpartition() function to create training and test data sets for the rpart and svm model

Kernlab	kernlab is an extensible package for kernel-based machine learning methods in R. It takes advantage of R's new S4 object model and provides a framework for creating and using kernel- based algorithmsSupervised model. For our purpose, we used the kernlab library to access the ksvm() function to create our supervised model
e1071	e1071 is a package for R programming that provides functions for statistic and probabilistic algorithms like a fuzzy classifier, naive Bayes classifier, bagged clustering, short-time Fourier transform, support vector machine, etc..
Arules	Rules Association - The arules package for R provides the infrastructure for representing, manipulating and analyzing transaction data and patterns (frequent itemsets and association rules)
ArulesViz	Visualizing Association Rules and Frequent Itemsets with R. This R package extends package arules with various visualization techniques for association rules and itemsets. The package also includes several interactive visualizations for rule exploration.
imputeTS	The package offers multiple missing data replacement (imputation) functions which our team utilized to address gaps in our data set.
rio	Rio allows the data import and export in R painless and quick. This objective is mainly reached when rio makes assumptions about the file format.
rpart	Rpart is a powerful machine learning library in R that is used for building classification and regression trees. This library implements recursive partitioning and is very easy to use. This package allowed the team to create a rpart model and rpart decision tree to visualize the model results.
rpart.plot	This function combines and extends the plot. rpart and text. rpart in the rpart package. It automatically scales and adjusts the displayed tree for best fit.

Data Import and Clean Up

First, we used read_csv to load the data into Rstudio. Our "MyData" file had 7,582 observations for 14 different variables. To comprehend the data more fully, we made use of the head, structure, and summary functions. This method is frequently employed to look for outliers, determine the types of data included in the dataset, examine the distributions of each variable, etc.

We conducted a more thorough analysis of the dataset using the structure function.

```
str(MyData)
```

```
spc_tbl_ [7,582 × 14] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ X               : num [1:7582] 1 2 3 4 5 7 9 10 11 12 ...
 $ age             : num [1:7582] 18 19 27 34 32 47 36 59 24 61 ...
 $ bmi             : num [1:7582] 27.9 33.8 33 22.7 28.9 ...
 $ children        : num [1:7582] 0 1 3 0 0 1 2 0 0 0 ...
 $ smoker          : chr [1:7582] "yes" "no" "no" "no" ...
 $ location        : chr [1:7582] "CONNECTICUT" "RHODE ISLAND" "MASSACHUSETTS" "PENNSYLVANIA" ...
 $ location_type   : chr [1:7582] "Urban" "Urban" "Urban" "Country" ...
 $ education_level: chr [1:7582] "Bachelor" "Bachelor" "Master" "Master" ...
 $ yearly_physical: chr [1:7582] "No" "No" "No" "No" ...
 $ exercise        : chr [1:7582] "Active" "Not-Active" "Active" "Not-Active" ...
 $ married         : chr [1:7582] "Married" "Married" "Married" "Married" ...
 $ hypertension    : num [1:7582] 0 0 0 1 0 0 0 1 0 0 ...
 $ gender          : chr [1:7582] "female" "male" "male" "male" ...
 $ cost            : num [1:7582] 1746 602 576 5562 836 ...
```

There are eight numerical columns, one of which is a category variable (hypertension), as can be seen. There are 7 character columns left.

The summary output, which contained descriptive statistics for each of the numerical variables, was also examined. This enabled us to understand the distributions, ranges, and skewness of the variables.

```
summary(MyData)
```

```
       X                  age              bmi            children     
 Min.   :        1   Min.   :18.00   Min.   :15.96   Min.   :0.000  
 1st Qu.:     5635   1st Qu.:26.00   1st Qu.:26.60   1st Qu.:0.000  
 Median :    24916   Median :39.00   Median :30.50   Median :1.000  
 Mean   :   712602   Mean   :38.89   Mean   :30.80   Mean   :1.109  
 3rd Qu.:   118486   3rd Qu.:51.00   3rd Qu.:34.77   3rd Qu.:2.000  
 Max.   :131101111   Max.   :66.00   Max.   :53.13   Max.   :5.000  
                                     NA's   :78                     
    smoker            location         location_type     
 Length:7582        Length:7582        Length:7582       
 Class :character   Class :character   Class :character  
 Mode  :character   Mode  :character   Mode  :character  


 education_level    yearly_physical      exercise        
 Length:7582        Length:7582        Length:7582       
 Class :character   Class :character   Class :character  
 Mode  :character   Mode  :character   Mode  :character  


   married           hypertension       gender              cost          
 Length:7582        Min.   :0.0000   Length:7582        Min.   :    2  
 Class :character   1st Qu.:0.0000   Class :character   1st Qu.:  970  
 Mode  :character   Median :0.0000   Mode  :character   Median : 2500  
                    Mean   :0.2005                      Mean   : 4043  
                    3rd Qu.:0.0000                      3rd Qu.: 4775  
                    Max.   :1.0000                      Max.   :55715  
                    NA's   :80                                         
```

As can be seen, the sample's age distribution had a mean of around 39 and ranged from 18 to 66. With a mean of 31, the BMI ranged from 16 to 53. The insurance cost has a wide range with a mean of $4,043, ranging from only $2 to $55,715. This shows that the cost variable is skewed to the right by a number of extraordinarily high values.

The smoker column is a binary column that indicates whether or not that particular person smokes. The sort of location might be either rural or urban. Numerous alternatives exist for education level. These include having a PHD, a bachelor's, a master's, or no college degree. Exercise and yearly physicals are likewise binary columns with yes, no, active, and not active alternatives. A person with hypertension is represented by a binary column with the value 1 and a person without it by the value 0. Married and gender are also self-explanatory binary variables. Cost is the sum of money spent on healthcare over the previous 12 months. This is the important variable.

It didn't seem like there were any outliers that needed to be eliminated. But, we did discover some NAs in the summary result.

We calculated the total of each individual variable and checked the dataset for NAs using the is.na function. As a result, we discovered that hypertension and bmi had NA values that need attention.

```
# Checking for NA values in all columns
colSums(is.na(MyData))
```

```
             X            age            bmi       children         smoker 
             0              0             78              0              0 
      location  location_type education_level yearly_physical       exercise 
             0              0              0              0              0 
       married   hypertension         gender           cost 
             0             80              0              0 
```

In order to remove NAs from the data, we applied the na_interpolation function.

```
# Removing NA values
MyData$bmi<- na_interpolation(MyData$bmi)
MyData$hypertension <- na_interpolation(MyData$hypertension)
```

Finally, we tested that we have addressed each NA using the same is.na method that was previously discussed. We checked that all of the code was working afterward and came to a conclusion that the NA values were no longer a problem.

```
# Checking for NA values in all columns after using na_interpolation to remove the NA valuesfi
colSums(is.na(MyData))
```

```
             X            age            bmi       children 
             0              0              0              0 
        smoker       location  location_type education_level 
             0              0              0              0 
yearly_physical       exercise        married   hypertension 
             0              0              0              0 
        gender           cost      expensive 
             0              0              0 
```

After removing the NA values within the dataset, we created a new variable named “expensive”. The 75th percentile of the cost variable was calculated using the quantile() function, with the result being the value 4775. This value is then used to define a new variable called "expensive" in the dataset, with each row being assigned either TRUE or FALSE depending on whether the cost for that row is greater than 4775.

```
MyData$expensive <- MyData$cost>4775
str(MyData)
```

```
tibble [7,582 × 15] (S3: tbl_df/tbl/data.frame)
 $ X              : num [1:7582] 1 2 3 4 5 7 9 10 11 12 ...
 $ age            : num [1:7582] 18 19 27 34 32 47 36 59 24 61 ...
 $ bmi            : num [1:7582] 27.9 33.8 33 22.7 28.9 ...
 $ children       : num [1:7582] 0 1 3 0 0 1 2 0 0 0 ...
 $ smoker         : chr [1:7582] "yes" "no" "no" "no" ...
 $ location       : chr [1:7582] "CONNECTICUT" "RHODE ISLAND" "MASSACHUSETTS" "PENNSYLVANIA" ...
 $ location_type  : chr [1:7582] "Urban" "Urban" "Urban" "Country" ...
 $ education_level: chr [1:7582] "Bachelor" "Bachelor" "Master" "Master" ...
 $ yearly_physical: chr [1:7582] "No" "No" "No" "No" ...
 $ exercise       : chr [1:7582] "Active" "Not-Active" "Active" "Not-Active" ...
 $ married        : chr [1:7582] "Married" "Married" "Married" "Married" ...
 $ hypertension   : num [1:7582] 0 0 0 1 0 0 0 1 0 0 ...
 $ gender         : chr [1:7582] "female" "male" "male" "male" ...
 $ cost           : num [1:7582] 1746 602 576 5562 836 ...
 $ expensive      : logi [1:7582] FALSE FALSE FALSE TRUE FALSE FALSE ...
```

Then we converted the TRUE/FALSE values in the "expensive" column to 1 or 0 using the str_replace_all() function. This is done to ensure that the "expensive" variable is represented numerically, which is necessary for statistical and machine learning analyses.

```
MyData <- MyData %>%mutate(expensive=str_replace_all(string=expensive,pattern="TRUE","1"))
MyData <- MyData %>%mutate(expensive=str_replace_all(string=expensive,pattern="FALSE","0"))

str(MyData)
```

```
tibble [7,582 × 15] (S3: tbl_df/tbl/data.frame)
 $ X              : num [1:7582] 1 2 3 4 5 7 9 10 11 12 ...
 $ age            : num [1:7582] 18 19 27 34 32 47 36 59 24 61 ...
 $ bmi            : num [1:7582] 27.9 33.8 33 22.7 28.9 ...
 $ children       : num [1:7582] 0 1 3 0 0 1 2 0 0 0 ...
 $ smoker         : chr [1:7582] "yes" "no" "no" "no" ...
 $ location       : chr [1:7582] "CONNECTICUT" "RHODE ISLAND" "MASSACHUSETTS" "PENNSYLVANIA" ...
 $ location_type  : chr [1:7582] "Urban" "Urban" "Urban" "Country" ...
 $ education_level: chr [1:7582] "Bachelor" "Bachelor" "Master" "Master" ...
 $ yearly_physical: chr [1:7582] "No" "No" "No" "No" ...
 $ exercise       : chr [1:7582] "Active" "Not-Active" "Active" "Not-Active" ...
 $ married        : chr [1:7582] "Married" "Married" "Married" "Married" ...
 $ hypertension   : num [1:7582] 0 0 0 1 0 0 0 1 0 0 ...
 $ gender         : chr [1:7582] "female" "male" "male" "male" ...
 $ cost           : num [1:7582] 1746 602 576 5562 836 ...
 $ expensive      : chr [1:7582] "0" "0" "0" "1" ...
```

Then we created two new subsets of the data, based on the "expensive" variable. The first subset is called "expensivePeople" and includes all rows where the "expensive" variable is equal to "1" (i.e., rows where the cost is greater than 4775). The second subset is called "inexpensivePeople" and includes all rows where the "expensive" variable is equal to "0" (i.e., rows where the cost is less than or equal to 4775).

Data Exploration and Visualizations

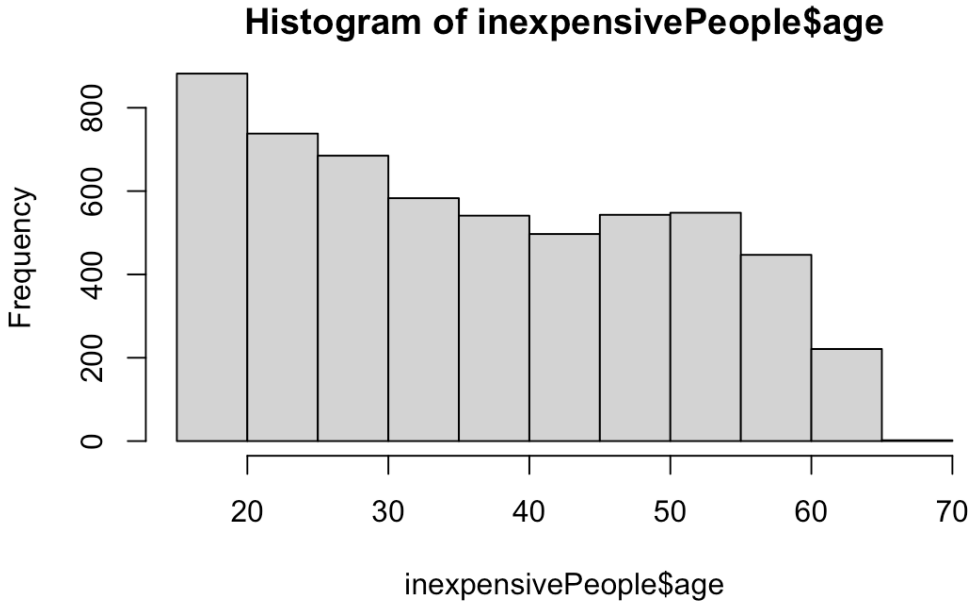
To better comprehend the data and effects on medical expenditures, visualizations and summary tables were created. The following sections provide an overview of all the analyses done after several relationships were examined.

I. Histogram of Expensive People and Age

A histogram was created for the age distribution of individuals under the "expensive" category. The analysis showed that individuals between the ages of 40 to 65 are more likely to pay higher healthcare costs.

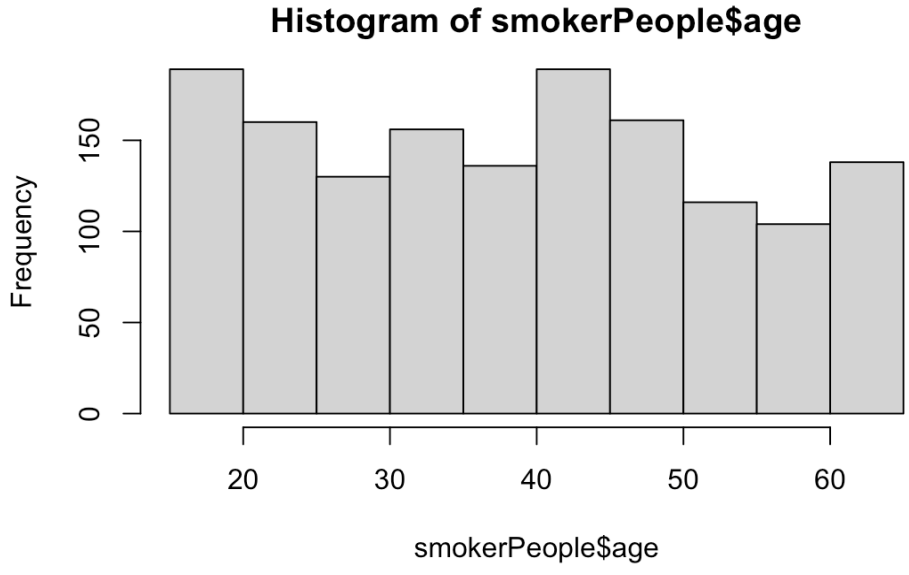
II. Histogram of Inexpensive People and Age

We created a histogram to analyze the age distribution of the group with inexpensive healthcare costs. It was observed that the majority of this group falls within the age range of 18 to 40 years.



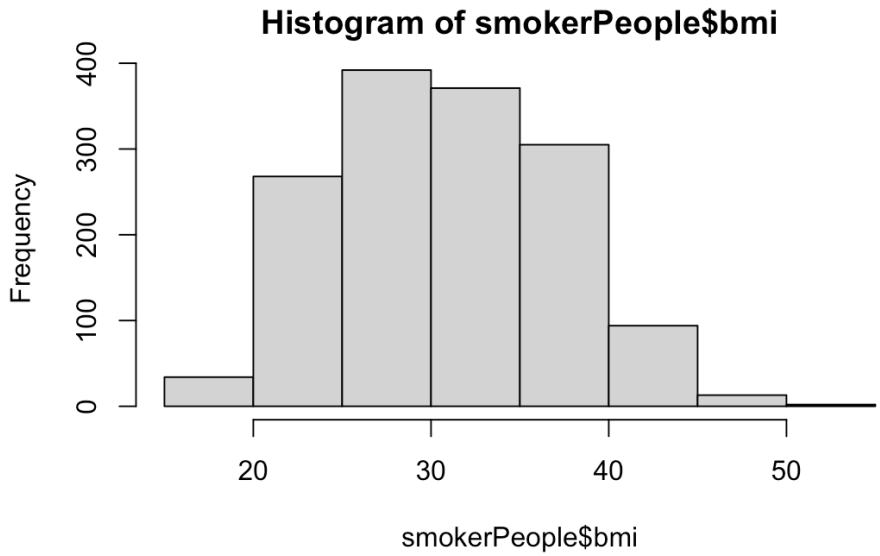
III. Histogram of Smoker People and Age

A histogram was created to analyze the age distribution of individuals who smoke. The results indicate that individuals between the ages of 18 to 25 and 40 to 45 tend to smoke more frequently than other age groups.



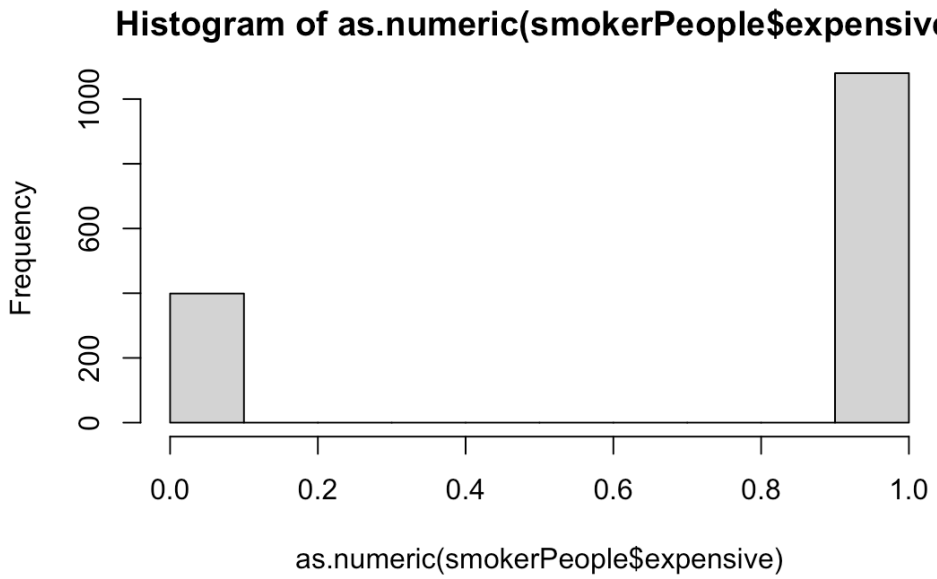
IV. Histogram of Smoker People and bmi

A histogram was created to analyze the BMI of individuals who smoke, revealing that the majority of smokers have a BMI outside of the healthy range of 18 to 25.



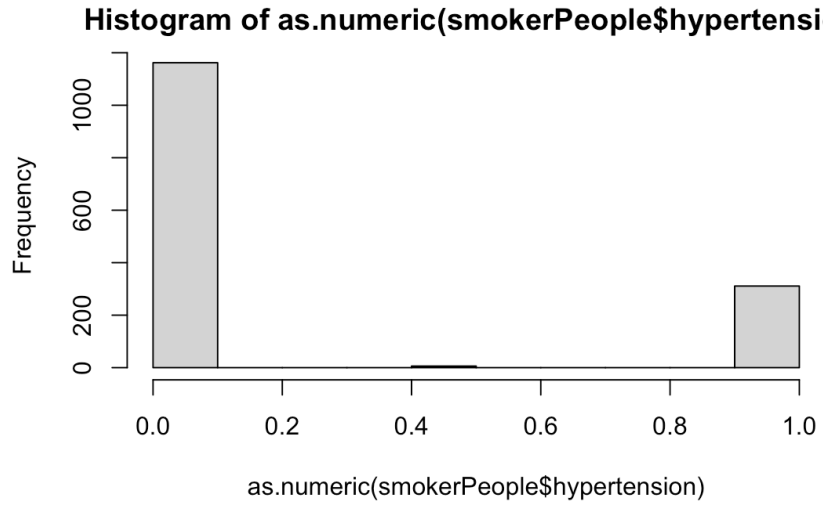
V. Histogram of Smoker People and expensive

We created a histogram to analyze the healthcare cost of smokers. It is observed that the majority of smokers tend to have higher healthcare costs compared to non-smokers.



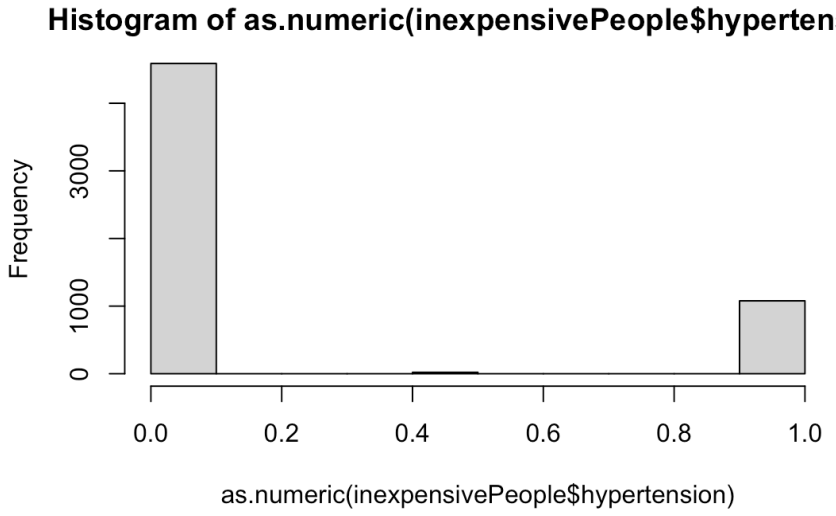
VI. Histogram of Smoker People and hypertension

Upon analyzing the relationship between smoking and hypertension through a histogram, we found that a majority of smokers do not have hypertension. While this suggests that hypertension may not be a direct factor in healthcare costs for people who smoke, other factors may still contribute to increased healthcare costs for this group.



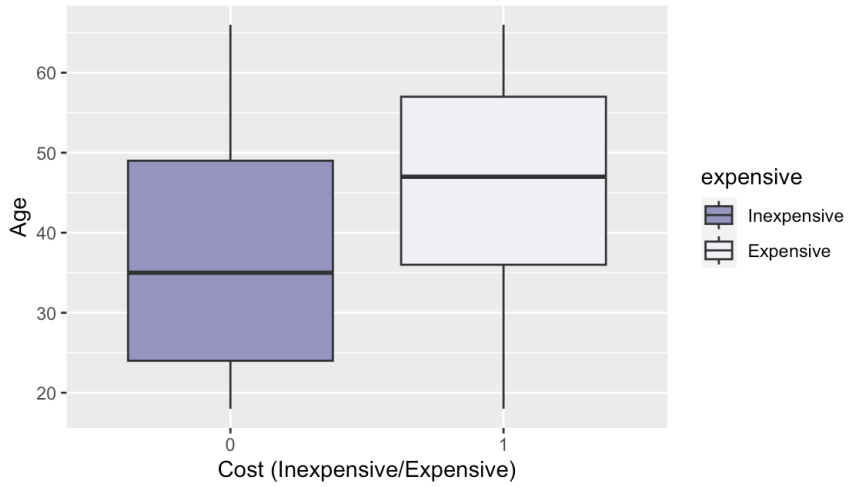
VII. Histogram of Inexpensive People and hypertension

A histogram was created to analyze the relationship between healthcare cost and hypertension. The plot showed that people who pay less for healthcare are less likely to have hypertension. Therefore, it can be inferred that people without hypertension tend to pay less for their healthcare.



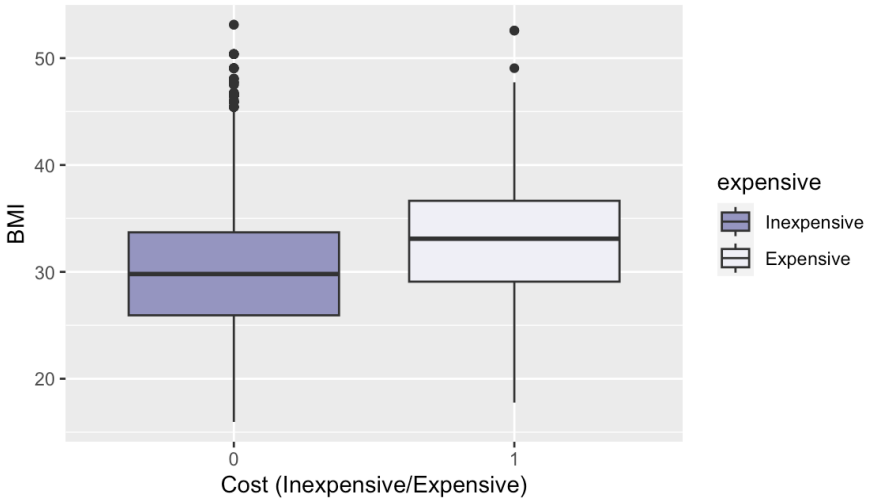
VIII. Boxplot for Healthcare Cost by Age

We used a box plot to compare the age distribution of people categorized as "expensive" and "inexpensive" in terms of their healthcare cost. The results showed that the cost by age is generally higher for those in the "expensive" category compared to those in the "inexpensive" category. This is indicated by the median age of approximately 47 for the "expensive" group, which is higher than the median age of 35 for the "inexpensive" group.



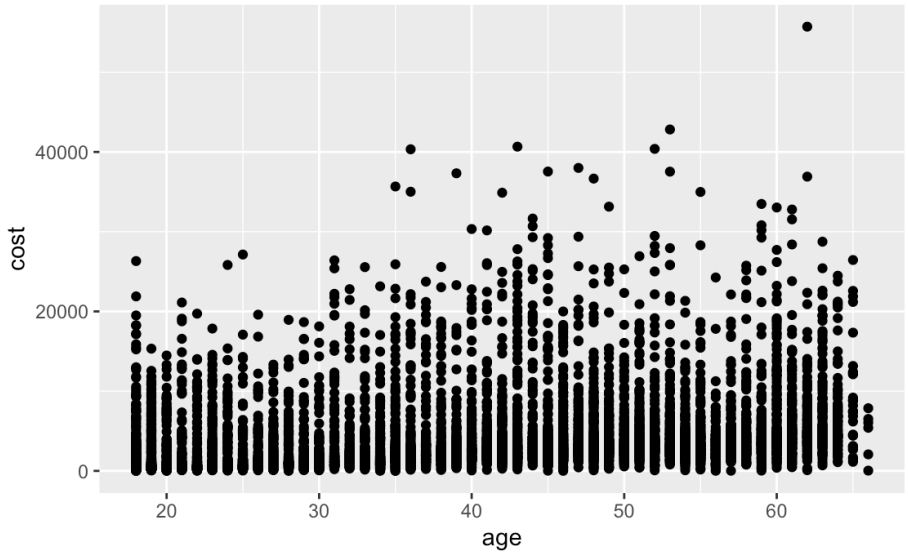
IX. Boxplot for Healthcare Cost by BMI

We also generated a box plot to compare the BMI distribution between "expensive" and "inexpensive" groups. The results revealed that the median BMI for "expensive" group is greater than that of "inexpensive" group, indicating that the healthcare cost tends to be higher for people with higher BMI.



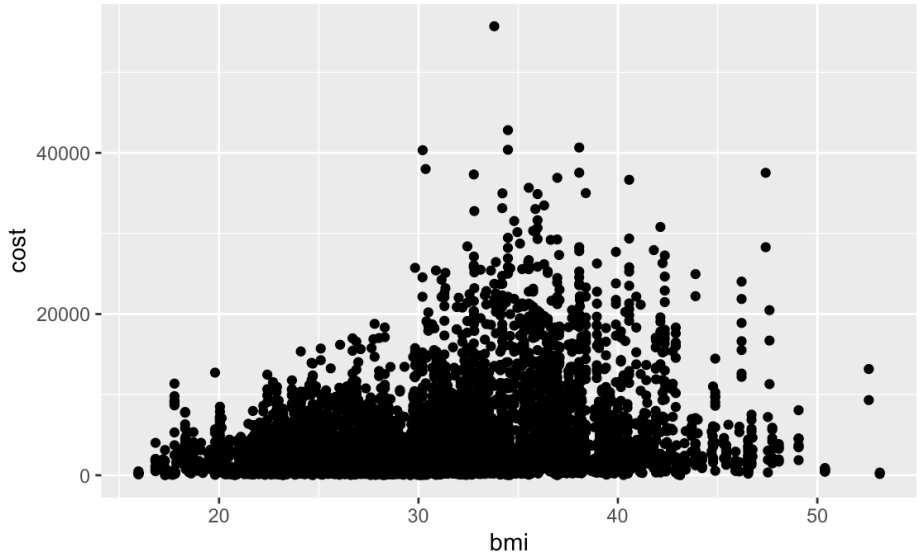
X. Scatterplot for Age Vs Cost

A scatterplot was generated to examine the relationship between age and healthcare costs. The plot revealed a positive correlation, indicating that as age increases, healthcare costs also tend to increase.



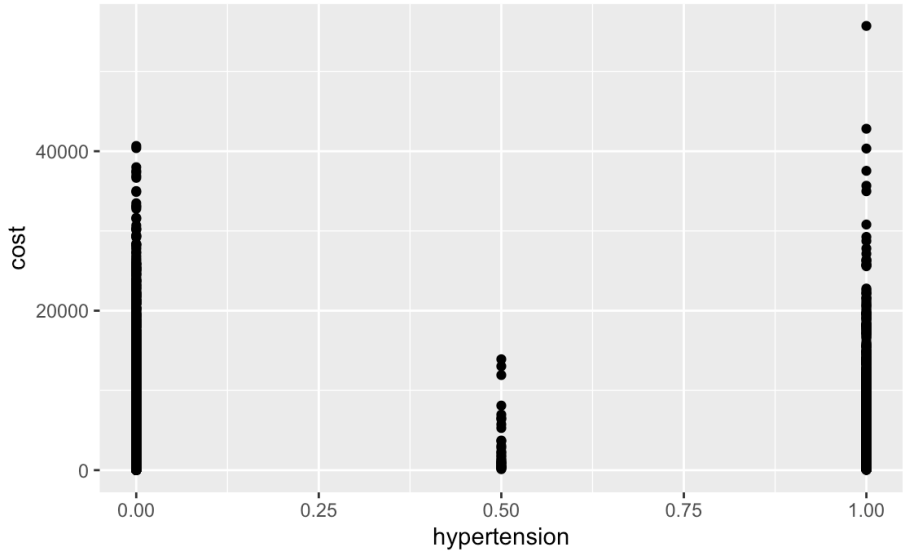
XI. Scatterplot for bmi Vs Cost

A scatterplot was created to compare BMI and healthcare cost. The plot showed a positive correlation between BMI and cost, with individuals having a BMI between 30-40 tending to have higher healthcare costs.



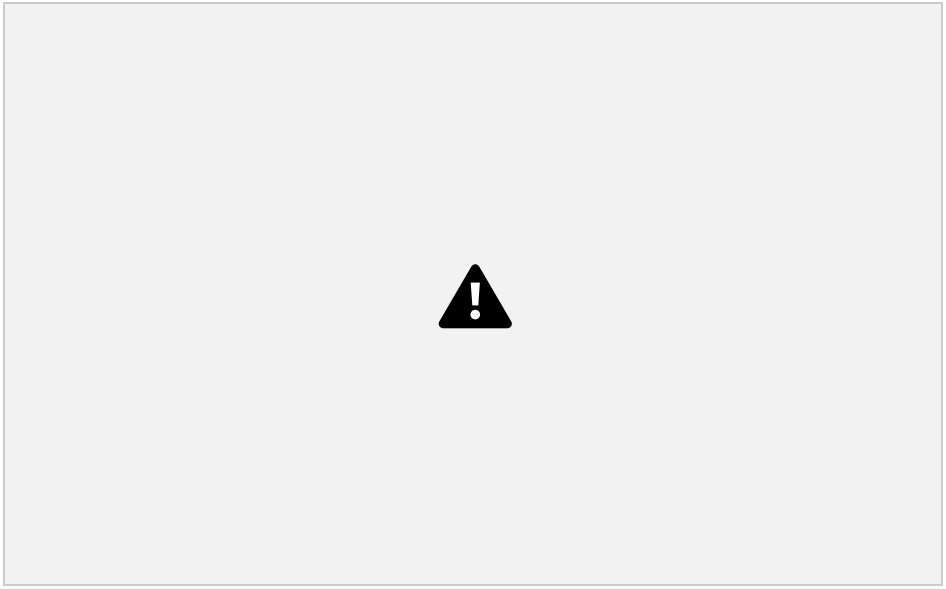
XII. Scatterplot for Hypertension Vs Cost

A scatterplot was created to analyze the relationship between hypertension and healthcare cost. The plot suggests that there is no significant correlation between hypertension and healthcare cost, indicating that hypertension may not be a major factor in determining healthcare expenses.



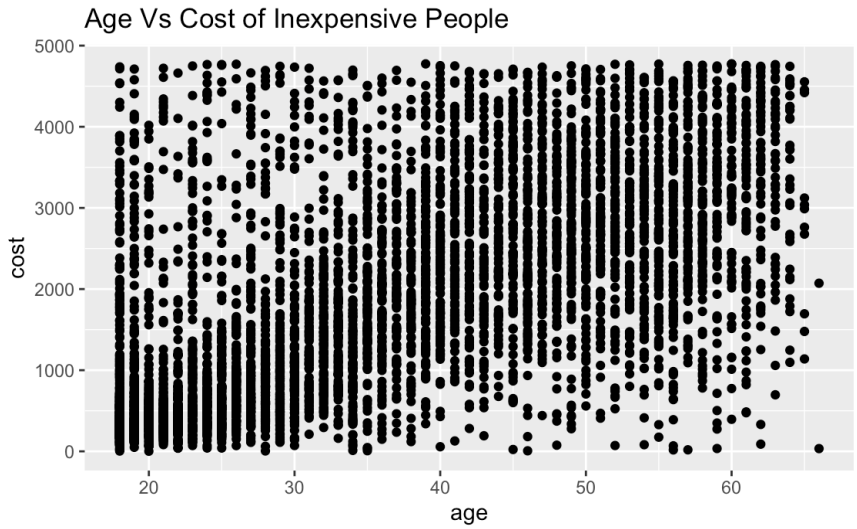
XIII. Barplot for expensive count

A bar plot was created to show the count of people falling under the "expensive" and "inexpensive" healthcare cost categories. The majority of people in the sample data were found to fall under the "inexpensive" category.



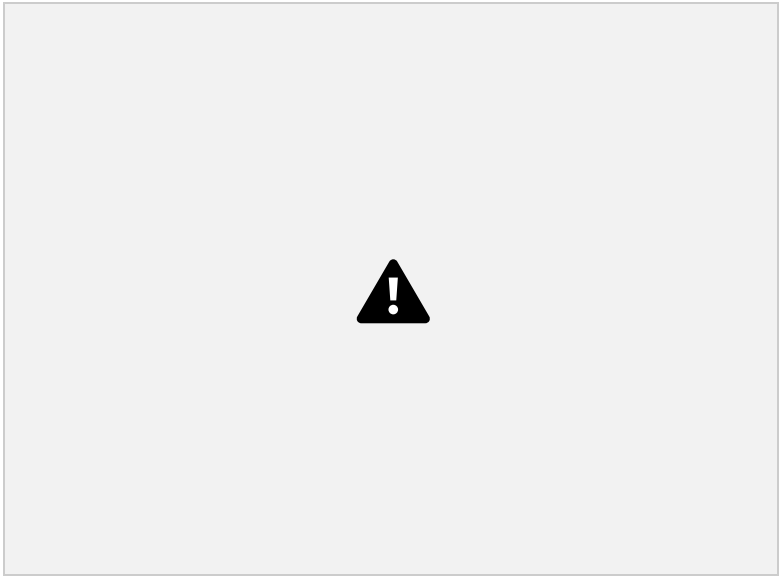
XIV. Scatterplot for Age Vs Cost of Inexpensive People

A scatterplot was created to analyze the relationship between age and healthcare cost for people categorized as 'inexpensive'. The plot revealed a positive correlation between age and cost, as evidenced by the dense concentration of points in the upper right-hand corner of the graph.



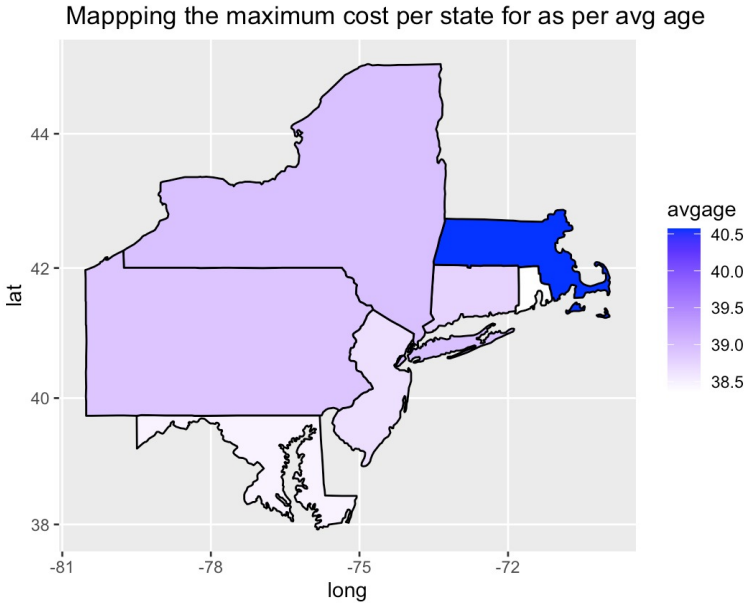
XV. Mapping the expensive and inexpensive people on the basis of maximum cost spent on healthcare per state

This map displays the maximum healthcare cost per state for both expensive and inexpensive groups, highlighting where healthcare spending is the highest. According to the map, Connecticut has the highest healthcare expenditure with a total cost of $55,000.



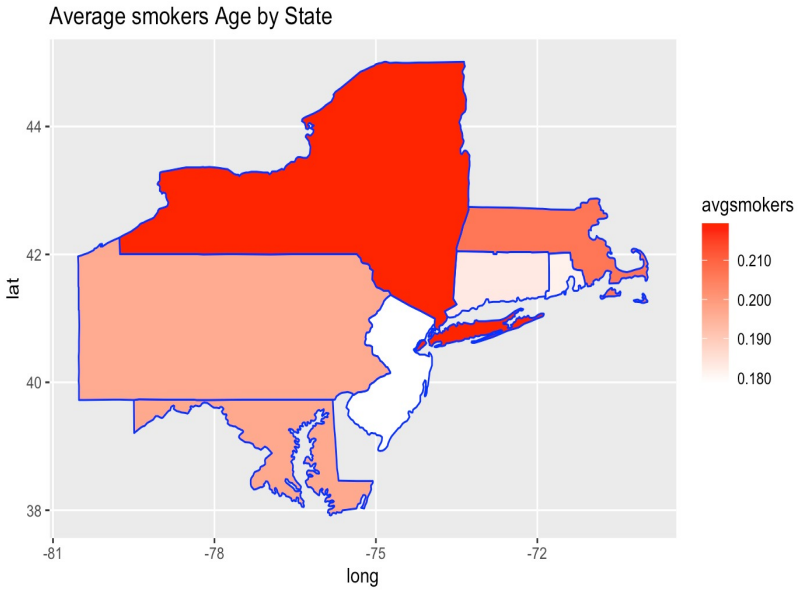
XVI. Mapping the age per state for the expensive and inexpensive people on the basis of average age of people

We created a map to visualize the average age of people using healthcare in each state, for both, expensive and inexpensive people. The map reveals that Massachusetts has the highest average age of people using healthcare, with a highest average age of 40.5, compared to other states.



XVII. Mapping the average smokers per state

We created a map to determine the average number of smokers per state, and found that New York has the highest number of average paying smokers.



Models

Data partition: Training data - 80%
Testing data - 20%

```
trainListS <- createDataPartition(y=HMOData$expensive,p=0.80,list=FALSE)
trainSetS <- HMOData[trainListS,]
testSetS <- HMOData[-trainListS,]
dim(trainSetS)
```

We have used 3 models:

1. Linear Regression
2. Support Vector Machine (SVM)
3. Tree Model

Model 1: Linear Regression

We performed linear regression analysis on a training dataset (trainSetS) to predict whether a person is "expensive" or not based on their age, body mass index (BMI), number of children, smoking status, hypertension, exercise habits, and frequency of yearly physicals.

At first, we converted the "expensive" variable in both the training and test datasets from a character or factor type to a numeric type, which is necessary for modeling purposes.

We created the linear regression model, with the variable "expensive" as the dependent variable to be predicted, and the remaining variables (age, bmi, children, smoker, hypertension, exercise, yearly_physical) as independent variables used to predict the dependent variable.

```
# Linear model
trainSetS$expensive<-as.numeric(trainSetS$expensive)
testSetS$expensive<-as.numeric(testSetS$expensive)
lmOut <- lm(expensive~age+bmi+children+smoker+hypertension+exercise+yearly_physical,data=tr
ainSetS)
summary(lmOut)
```

Below is the output of the summary() function applied to the linear regression model lmOut. The output provides information about the model's performance and the significance of each independent variable.

```
Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        0.3142587  0.0258982  12.134  < 2e-16
age                0.0075213  0.0003002  25.057  < 2e-16
bmi                0.0120483  0.0007064  17.056  < 2e-16
children           0.0138848  0.0034583   4.015 6.02e-05
smokeryes          0.5967618  0.0106872  55.839  < 2e-16
hypertension       0.0363817  0.0106351   3.421 0.000628
exerciseNot-Active 0.1683463  0.0097860  17.203  < 2e-16
yearly_physicalYes 0.0219387  0.0097779   2.244 0.024888

(Intercept)        ***
age                ***
bmi                ***
children           ***
smokeryes          ***
hypertension       ***
exerciseNot-Active ***
yearly_physicalYes *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3288 on 6058 degrees of freedom
Multiple R-squared:  0.4241,    Adjusted R-squared:  0.4235
F-statistic: 637.4 on 7 and 6058 DF,  p-value: < 2.2e-16
```

The "Coefficients" section shows the estimated coefficients, standard errors, t-values, and p-values for each independent variable. The intercept (or baseline) coefficient is 0.314, which means that when all the independent variables are equal to zero, the expected value of the dependent variable is 0.314 (which corresponds to the probability of being "expensive").

The "Signif. codes" section shows the level of significance (p-value) of each coefficient. All the coefficients in this model are highly significant ($p<0.001$) except for yearly_physicalYes, which has a p-value of 0.024888.

The "F-statistic" section shows the overall significance of the model. The lower the p-value, the more significant the model. In this case, the p-value is very small (<2.2e-16), indicating that the model is highly significant.

Overall, the output suggests that the model is statistically significant (p-value < 2.2e-16) and provides some predictive power for the expensive variable based on the selected predictors. However, the model's performance may need to be further evaluated on the test dataset to assess its generalizability.

Model 2: Support Vector Machine (SVM)

We built an SVM (Support Vector Machine) model using the e1071 package in R. The set.seed(123) sets the random seed to ensure reproducibility of the results. The svm() function is used to build the SVM model with the following arguments:

- data: The training dataset.
- expensive~.: The formula specifying the target variable (expensive) and the predictor variables (all other variables in the dataset).
- C: The cost parameter that controls the trade-off between achieving a low training error and a low testing error. In this case, the value of 5 is chosen.
- CV: The number of folds for cross-validation. In this case, 3-fold cross-validation is used.
- prob.model: A logical value indicating whether to train a probability model. In this case, it is set to TRUE.

After building the model, the predict() function is used to make predictions on the test dataset (testSetS).

```
# Building SVM model
set.seed(123)
library(e1071)
ksvm_model <- svm(data= trainSetS, expensive~.,C=5, CV=3, prob.model= TRUE)
svmPred<- predict(ksvm_model,newdata= testSetS, type= "response")
head(svmPred)
```

```
 4  5  7 14 18 19
 0  0  0  0  0  0
Levels: 0 1
```

The confusion matrix shows the performance of the SVM model in predicting the expensive or inexpensive health insurance plans correctly.

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1110  176
         1   27  203

               Accuracy : 0.8661
                 95% CI : (0.8479, 0.8828)
    No Information Rate : 0.75
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5891

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

            Sensitivity : 0.9763
            Specificity : 0.5356
         Pos Pred Value : 0.8631
         Neg Pred Value : 0.8826
             Prevalence : 0.7500
         Detection Rate : 0.7322
   Detection Prevalence : 0.8483
      Balanced Accuracy : 0.7559

       'Positive' Class : 0
```

The model has an accuracy of 86.61%. The sensitivity of the model is 97.63%, which means that the model has correctly identified 97.63% of the expensive plans. The specificity of the model is 53.56%, which means that the model has correctly identified only 53.56% of the inexpensive plans. Overall, the SVM model has a good accuracy, but it needs improvement in correctly identifying the inexpensive plans.

Model 3: Tree Model

This code builds a decision tree model using the rpart package in R. The rpart function is used to build the model with the formula expensive ~ age +bmi +children +smoker +hypertension +exercise +yearly_physical, which specifies that the target variable is expensive and the predictor variables are age, bmi, children, smoker, hypertension, exercise, and yearly_physical. The method = "class" argument specifies that the model is a classification tree.

```
# Building a tree model
rpart_model <- rpart(expensive ~
age+bmi+children+smoker+hypertension+exercise+yearly_physical, data = trainSetS, method =
"class")
rpartPred <- predict(rpart_model, newdata= testSetS, type= "class")
confusionMatrix(rpartPred, as.factor(testSetS$expensive))
```

The confusion matrix below shows the performance of the decision tree model in predicting whether a person's healthcare costs are expensive or not.

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1093  132
         1   44  247

               Accuracy : 0.8839
                 95% CI : (0.8667, 0.8996)
    No Information Rate : 0.75
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6644

 Mcnemar's Test P-Value : 5.458e-11

            Sensitivity : 0.9613
            Specificity : 0.6517
         Pos Pred Value : 0.8922
         Neg Pred Value : 0.8488
             Prevalence : 0.7500
         Detection Rate : 0.7210
   Detection Prevalence : 0.8080
      Balanced Accuracy : 0.8065

       'Positive' Class : 0
```

The model predicted 1093 true negatives (TN) and 247 true positives (TP), and misclassified 132 false negatives (FN) and 44 false positives (FP). The accuracy of the model is 88.39%, meaning that it correctly classified 88.39% of the instances. The sensitivity of the model is 96.13%, indicating that it correctly classified 96.13% of the instances that were actually positive. The specificity of the model is 65.17%, indicating that it correctly classified 65.17% of the instances that were actually negative. The positive predictive value (PPV) of the model is 89.22%, indicating that of all instances that the model classified as positive, 89.22% were actually positive. The negative predictive value (NPV) of the model is 84.88%, indicating that of all instances that the model classified as negative, 84.88% were actually negative. Overall, the decision tree model performs slightly better than the SVM model with an accuracy of 86.61%.

Conclusions

From the study of the information provided, we were able to figure out the reasons why some people pay more for healthcare services as well as the causes of expensive health care. This research makes it easier to understand how to reduce healthcare expenditures and help wealthy individuals see the value of forming healthy habits.

We were able to come to the conclusion that the three main variables that influence healthcare expenditures are BMI, age, and smoking status. Therefore, we can conclude that by keeping an eye on these parameters, we can lower overall healthcare expenses.

Actionable Insights

- Smokers ought to pay more in premiums. Relative to non-smokers, smokers spend much more on healthcare, therefore charging a higher premium to smokers would help offset the high cost of healthcare.
- If you smoke in New York, the premium needs to be substantially greater. Only New York had higher healthcare prices for people who smoke that were much higher than those in other states.
- If we already charge older people with higher premiums due to their age, we don't need to charge more for someone with a high BMI. This is the case because aging has a greater impact on BMI than smoking does.