



## **IST 687 INTRODUCTION TO DATA SCIENCE**

**M003 | Group 3**

### **DATA ANALYSIS AND COST PREDICTION FOR HEALTH MANAGEMENT ORGANIZATION**

**Submitted By**

**Vedant Devidas Patil  
Chuan Tse Tsai  
Ritik Dhame  
Sai Sisira Pathakamuri  
Adrian Wagner**

## **Table of contents**

<b>1. Description</b>	<b>3</b>
<b>2. Project Scope and Objective</b>	<b>3</b>
<b>3. Project Deliverables</b>	<b>3</b>
<b>4. Data Acquisition</b>	<b>4</b>
<b>5. Data preprocessing</b>	<b>4</b>
<b>6. Data Analysis using Visualizations</b>	<b>7</b>
<b>7. Modelling Techniques</b>	<b>14</b>
<b>7.1 Association Rules</b>	<b>14</b>
<b>7.2 Linear Regression</b>	<b>15</b>
<b>7.3 Support Vector Machine</b>	<b>17</b>
<b>7.4 Decision Tree</b>	<b>18</b>
<b>8. Shiny App</b>	<b>20</b>
<b>9. Location Visualizations</b>	<b>22</b>
<b>10. Interpretation and Recommendation</b>	<b>25</b>

## **1. Description**

Based on the given data, the case's main objective is to offer invaluable insights and precisely estimate which individuals (clients) would be pricey. We intended to analyze and draw necessary factors that lead to actionable insights and provide recommendations in order to help an individual cut their costs.

## **2. Project Scope and objective**

The data employed in this project is real life data from a Healthcare management organization. This data consists of various factors that influence the patient's healthcare cost every year such as age, smoking habits, location where the person is staying at, type of environment in which the individual lives in, hypertension, body mass index, exercise habits, etc. We'll focus on identifying the elements that might lead to a patient having a big hospital bill for next year and using statistical methods to gain useful information.

The workings of the project consist of using various data analysis techniques and models to support the interpreted recommendations, driven by correlational trends between the various parameters present in the healthcare dataset.

### **Focus points**

- 1) Identifying people who might spend more on healthcare next year.
- 2) Provide actionable insight to the HMO, in terms of how to lower their total health care costs, by providing a specific recommendation on how to lower health care costs.

## **3. Project Deliverables**

- The R code for analysis.
- A presentation of the actionable insight achieved along with specific recommendations given in the presentation. Keeping sensitivity in mind over accuracy.
- A shiny app deployed on shinyapps.io that app can read in a user selected datafile

- Using a previously created and stored model to predict which people will be expensive. The app should also read in a second user selected file, which shows which people were actually expensive.
- Output of the Sensitivity (from the confusion matrix) for the new test dataset
- Output of the full confusion matrix of predictions.

#### **4. Data Acquisition**

Our course instructors supplied the data that we used. Information on healthcare clients from seven distinct US states—Pennsylvania, New Jersey, Connecticut, Massachusetts, Maryland, New York, and Rhode Island—is revealed in this set of data. It should also be noted that these states are located in the northeastern part of the United States and they all border one another. This data collection includes a number of health-related variables, such as the patient's BMI and whether or not they smoke often, in addition to other information, such as the state in which they reside and the amount of money they spend on healthcare. One patient is represented by each row in this data set. The variables that might be used from this data were thoroughly investigated. Following this first analysis, the data set was sent to the preprocessing stage, where any mistakes were fixed to make the data suitable for further analysis.

#### **5. Data preprocessing**

Before processing our data, we had 7582 rows and 14 columns. The data was from the healthcare management organization for the past year that had all the different attributes that influenced the healthcare cost of each person along with their last year's cost/expenses on their health care.

To obtain a more broad idea about the dataset in order to proceed with data wrangling, we performed some basic analysis to understand the structure of the dataset, possible outliers, etc.

The screenshot pictured below shows the implementation of `read.csv()` function used to read the data and display the data using `head` function.

```
##{r}
#Loading the dataset using read.csv function.
hmoDF <- read.csv("https://intro-datascience.s3.us-east-2.amazonaws.com/HMO_data.csv")

#Displaying the first five rows of the dataset.
head(hmoDF)
```

Description: df [6 x 14]

	X <int>	age <int>	bmi <dbl>	children <int>	smoker <chr>	location <chr>	location_type <chr>	education_level <chr>	yearly_physical <chr>
1	1	18	27.900	0	yes	CONNECTICUT	Urban	Bachelor	No
2	2	19	33.770	1	no	RHODE ISLAND	Urban	Bachelor	No
3	3	27	33.000	3	no	MASSACHUSETTS	Urban	Master	No
4	4	34	22.705	0	no	PENNSYLVANIA	Country	Master	No
5	5	32	28.880	0	no	PENNSYLVANIA	Country	PhD	No
6	7	47	33.440	1	no	PENNSYLVANIA	Urban	Bachelor	No

Next, we used the `summary()` function to understand the data types of each column along with finding out the minimums, maximums and the value range each column falls in a standard deviation.

```
##{r}
#The summary() function in R can be used to quickly summarize the values in a data frame.
summary(hmoDF, include='all')
```

X	age	bmi	children	smoker	location	location_type
Min. : 1	Min. :18.00	Min. :15.96	Min. :0.000	Length:7582	Length:7582	Length:7582
1st Qu.: 5635	1st Qu.:26.00	1st Qu.:26.60	1st Qu.:0.000	Class :character	Class :character	Class :character
Median : 24916	Median :39.00	Median :30.50	Median :1.000	Mode :character	Mode :character	Mode :character
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				
education_level	yearly_physical	exercise	married	hypertension	gender	cost
Length:7582	Length:7582	Length:7582	Length:7582	Min. :0.0000	Length:7582	Min. : 2
Class :character	Class :character	Class :character	Class :character	1st Qu.:0.0000	Class :character	1st Qu.: 970
Mode :character	Mode :character	Mode :character	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		

## Missing Values

After having a brief idea about the data, it becomes very important to clean and transform the data so that we could use our dataset in a proper form to begin our analysis with. Hence, we checked for the missing values in our dataset using `is.na()`. When we first got our data, there were missing values. We used `na_Interpolation` to remove the null values because it was the best way to replace the null values with some meaningful values that `na_interpolation` works on.

```
##{r}
#The is.na() method checks the null/NA values for all the columns in the dataset.
#The colSums counts the number of rows satisfying the condition.
colSums(is.na(hmoDF))
##
```

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

Above, we can see that `bmi` has 78 null values whereas `hypertension` has 80 null values.

The above image shows the distribution of null/NA/missing values in our dataset. In this manner, it becomes very important to manage null values using `na_interpolation()` method, as shown in the following graphic.

```
##{r}
library(imputeTS)

#Missing values get replaced by values of approx, spline or stinterp interpolation.
head(na_interpolation(hmoDF$bmi))
head(na_interpolation(hmoDF$hypertension))
##
```

```
Registered S3 method overwritten by 'quantmod':
  method      from
as.zoo.data.frame zoo
[1] 27.900 33.770 33.000 22.705 28.880 33.440
[1] 0 0 0 1 0 0
```

```
##{r}
#performing na_interpolation
hmoDF$bmi <- na_interpolation(hmoDF$bmi)
hmoDF$hypertension <- na_interpolation(hmoDF$hypertension)

#checking the null values after na_interpolation()
sum(is.na(hmoDF$bmi))
sum(is.na(hmoDF$hypertension))
##
```

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

After performing the `na_interpolation`, we could again cross verify the null values in our dataset which turns out to be zero.

Now, to begin the analysis with, we created an *isexpensive* column in the dataset on the basis of cost column that represents the last year data, so that we could use that particular *isexpensive* column to find an accuracy and decide on our model that we could use for predicting the outcomes on the actual data.

We used the quantile function to identify the 75th percentile of the cost from the cost column. All the values that were above 4775 were classified as being expensive, hence set as *isexpensive* “True” and the values which were below 4775 in the cost column were set as “False”.

```
##{r}
#Created a new Column isexpensive and stored 0 or 1 based on cost column
#If cost is greater than 4775(ie 75 percentile)
#then store 1 else store 0.

hmoDF$isexpensive <- with(hmoDF, ifelse(cost >= 4775, TRUE, FALSE))
View(hmoDF)
head(hmoDF)
##
```

Description: df [6 x 15]

location <chr>	location_type <chr>	education_level <chr>	yearly_physical <chr>	exercise <chr>	married <chr>	hypertension <dbl>	gender <chr>	cost <int>	isexpensive <lgl>
CONNECTICUT	Urban	Bachelor	No	Active	Married	0	female	1746	FALSE
RHODE ISLAND	Urban	Bachelor	No	Not-Active	Married	0	male	602	FALSE
MASSACHUSETTS	Urban	Master	No	Active	Married	0	male	576	FALSE
PENNSYLVANIA	Country	Master	No	Not-Active	Married	1	male	5562	TRUE
PENNSYLVANIA	Country	PhD	No	Not-Active	Married	0	male	836	FALSE
PENNSYLVANIA	Urban	Bachelor	No	Not-Active	Married	0	female	3842	FALSE

## 6. Data Analysis using Visualizations

For the purpose of making the data easier to understand, we have constructed numerous visualizations of the healthcare data. These visualizations include maps and scatter plots, and were generated through R-Studio. They are useful because they show how various factors, health-related and locational, impact the cost of healthcare.

## Maps

One of the most effective ways to visualize this kind of data is to use a map. Maps are especially useful for showing the differences in data from different locations. For this project maps allow us to compare the data in the different states.

## Scatter Plots

We have also created several scatter plots in order to showcase the relations between different attributes of the data.

### Analyzing BMI and smokers data

Here, we have used scatter plot using smoker and BMI attributes to find a relationship with the cost factor.

#### Code:

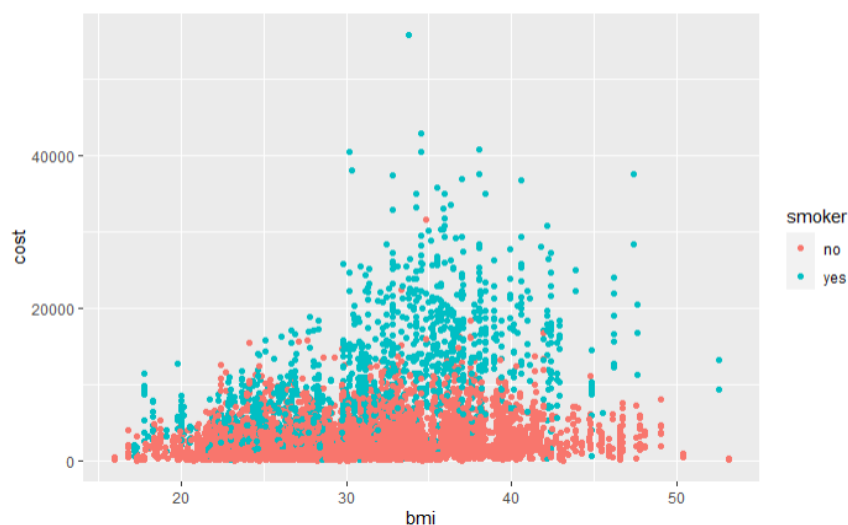
```
{r}
library(ggplot2)

#Using ggplot for scatter plot visualization
myPlot2 <- ggplot(hmoDF)

#using bmi on x-axis and cost on y-axis along with color as smoker
myPlot2 <- myPlot2 + aes(x=bmi, y=cost, color = smoker)
myPlot2 <- myPlot2 + geom_point()

#Displaying ggplot
myPlot2
```

#### Output:





## Inference:

The above scatter plot suggests that as the BMI of the person increases, the smoking plays an important role in increasing the cost of the patient. The more the person smokes, the more they smoke, the more they are at a risk of high cost. Hence, there is definitely a correlation between these 3 factors.

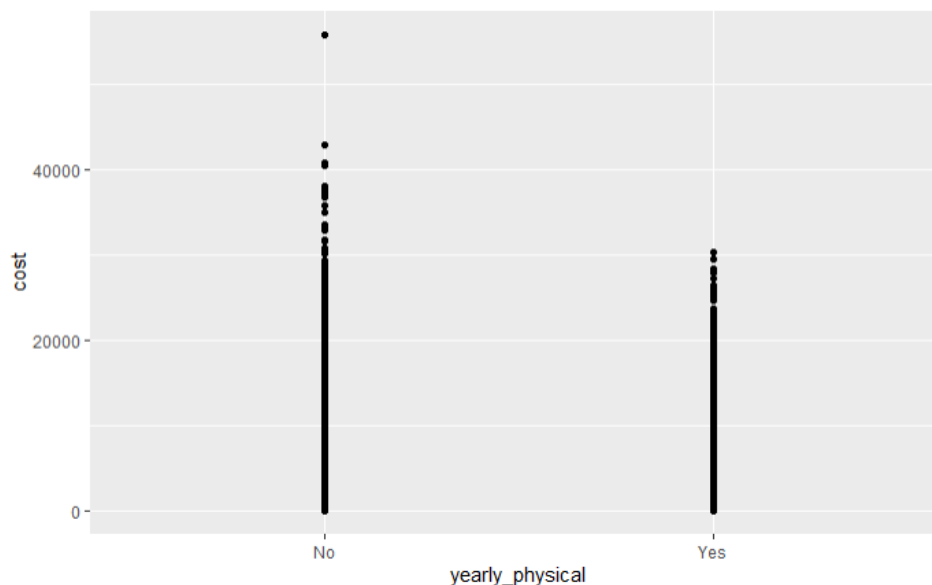
## Analyzing regular physical checkups

### Code:

```
{r}
library(ggplot2)
library(MASS)
ggplot(data=hmodf) + aes(x=yearly_physical, y=cost) + geom_point() +
geom_smooth(method="lm", se=FALSE)

#The yearly_physical has some significant correlation with cost factor and thus
#it could be a good attribute to be considered for prediction.
```

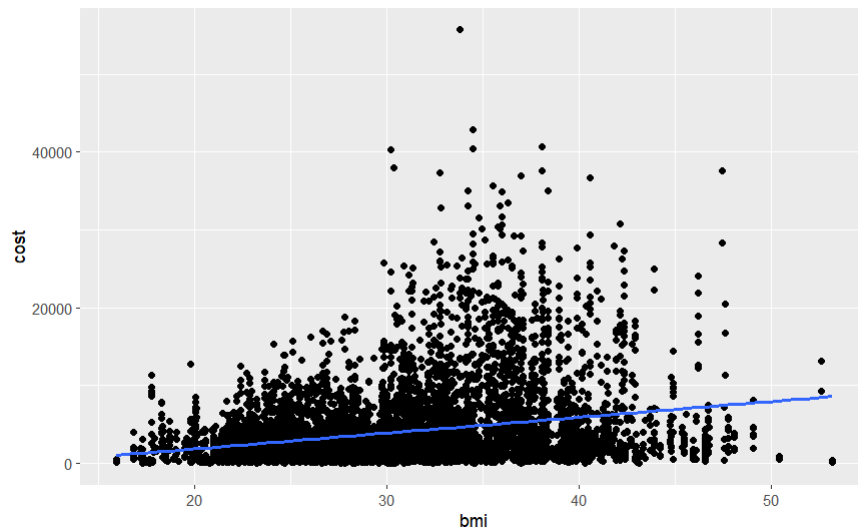
### Output:



We have created a scatter plot that shows us the comparison between those who take yearly physical tests and those who do not.

- The first observation is that those taking yearly physical tests have a lower healthcare cost.
- A person who does not take yearly physical tests will have higher healthcare costs.

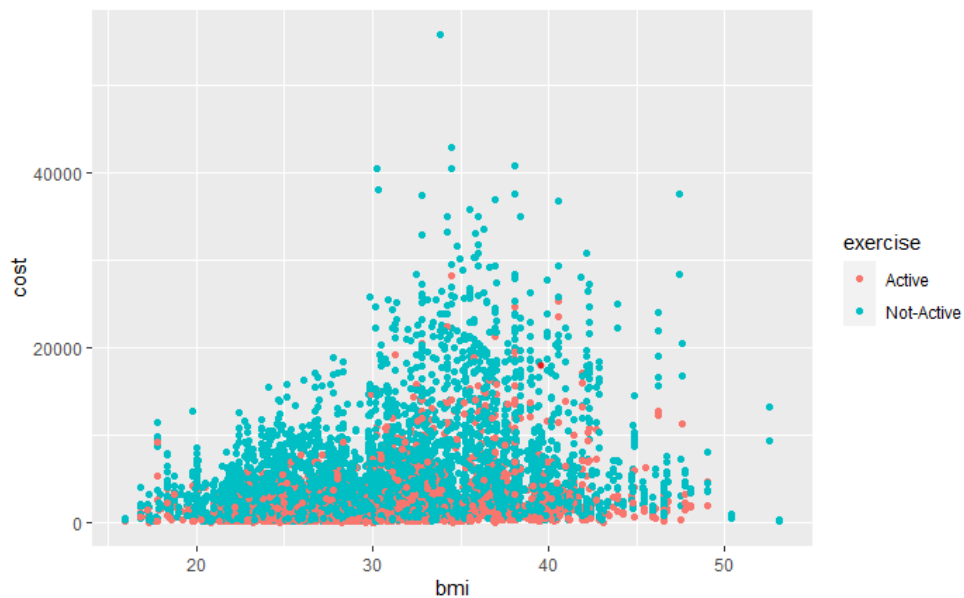
## Analyzing cost with smoker and bmi



This above image shows that there exists a positive correlation between cost and bmi too.

## Analyzing active exerciser

Taking the analysis a step further, we analyzed the exercise activity with the BMI attribute to check if there is any correlation between those two factors with the cost.

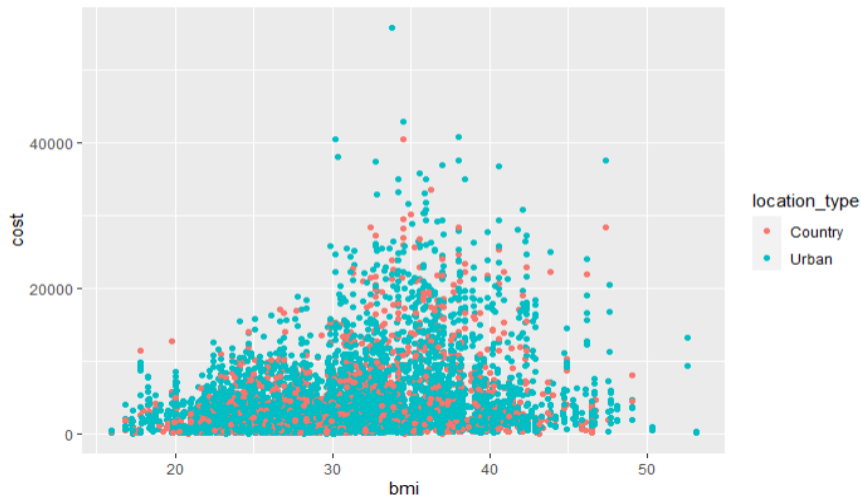


- The scatter plot pictured above suggests that as the individual person's BMI increases, the

exercise plays an important role in increasing the cost of the patient.

- The higher the bmi and less active the person is, the more they are at a risk of high cost. Hence, there is definitely a correlation between these 3 factors.
- It also suggests that less bmi and/or being active is very essential.

### Location affecting the cost



We have also created a scatter plot that shows the number of individuals living in the countryside and in urban areas, and their cost of healthcare based on their BMI. We can observe the following points:

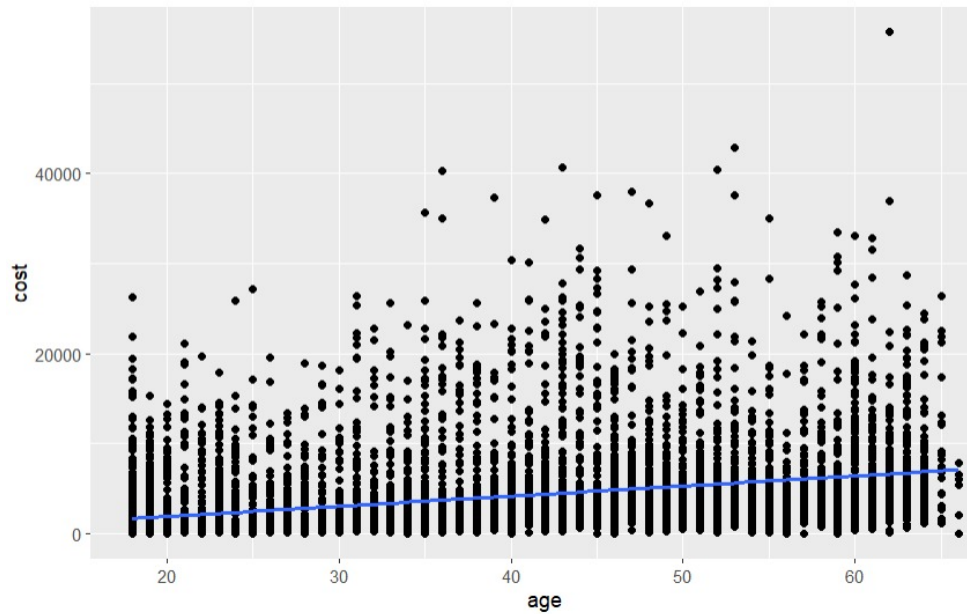
- An individual living in the city has a higher cost of healthcare
- An individual living in the country has a lower cost of living

## Analyzing Hypertension and Marriage attributes



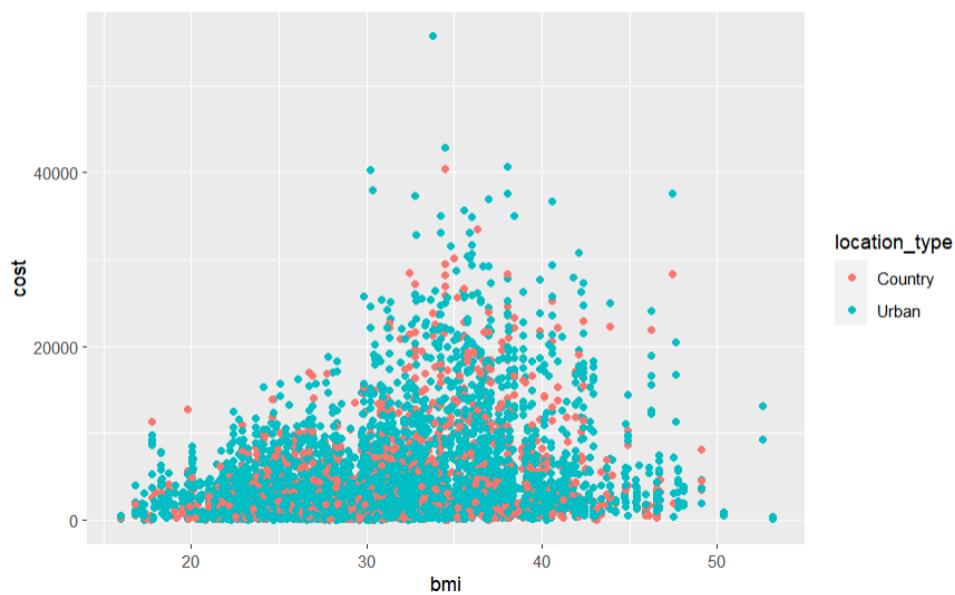
The above scatter plots suggest that even if a patient has high hypertension, and irrespective of marriage keeping the BMI below 30 could significantly reduce the cost. However, as bmi increases, the cost is going to increase irrespective of hypertension and marriage status. Hence, hypertension, marriage status has no major correlation with bmi and cost.

### Analyzing age and cost



From the scatter plot above, it can be understood that the cost of healthcare increase with Age. Hence, age and cost have a positive correlation.

### Analyzing cost and location\_type



The above scatter plot about cost, bmi and location\_type suggest that there is no significant

correlation of location\_type with bmi and cost. When the bmi increases, the cost is high for both county and urban population equally. The only one thing we can infer is there are comparatively more people from urban location who have high bmi above 40.

## 7. Modeling techniques

For accurate results of the information cleaned from the data collection, many models have been applied. These models are useful because they provide a clear representation of the real-world information in the data sets. The upcoming models have been put into practice.

### 7.1 Apriori Algorithm

The apriori algorithm is the method used to compute the association rules between things. It denotes the relationship between two or more items. In other words, the apriori algorithm is an association rule learning system that assesses whether consumers who purchased product A also purchased product B.

Therefore, we used the apriori association rule to find out the best combination that lead to less expensive by identifying the combinations that have the highest support and confidence factor.

Code

```
## {r}
rules1 <- apriori(datax,
  parameter=list(supp=0.3, conf=0.85),
  control=list(verbose=F),
  appearance=list(default="lhs",rhs=("isexpensive=FALSE")))
summary(rules1)
inspect(rules1)
inspectDT(rules1)
```

The datax dataset in the code above is as factored dataset.

Below are the combinations that have highest support and confidence

1	Non Smoker	Without Children	
2	Non Smoker	Without Hypertension	Having Bachelors Degree
3	Non Smoker	Married	Living in urban setting
4	Non Smoker	Married	Getting yearly Physical Test
5	Non Smoker	Having Bachelors Degree	Getting yearly Physical Test
6	Non Smoker	Hypertension	Married

## 7.2 Linear Modelling

The apriori algorithm is the method used to compute the association rules between things. It denotes the relationship between two or more items. In other words, the apriori algorithm is an association rule learning system that assesses whether consumers who purchased product A also purchased product B.

On our dataset, we first did linear modeling. We might analyze and explore relationships between two continuous variables in our dataset by using simple linear regression. Finding the line that best fits the data was the main goal. The line that has the lowest potential total prediction error is considered to be the best fit line.

After analyzing the datasets during the previous data analysis and visualization part, we came across few of the many attributes that have the impact on the cost factor. We used the combinations of those attributes and found the highest accuracy for linear model turned out to be 42.55% for age, bmi, children, smoker, hypertension, yearly\_physical, exercise, gender, married. These were the attributes that had the highest stars in the linear model output. Hence, used these attributes as a reference primarily for other models too. Here, we checking for a high coefficient of determination, or r squared, a low p value, and examining residual plots were all part of this approach.

Significant factors are :

Essential parameter : isexpensive

Coefficients : age, bmi, children, smoker, hypertension, yearly\_physical, exercise, gender, married.

p-value: < 2.2e-16

R-squared: 0.4255,

Accuracy : 42.55%

Code for the best combination of Linear Model:

```
```{r}
lmout <- lm(isexpensive~age+bmi+children+smoker+hypertension+yearly_physical+exercise+gender+married, data = hmoDF)
summary(lmout)
lmout
```
```

Output:

```
Call:
lm(formula = isexpensive ~ age + bmi + children + smoker + hypertension +
    yearly_physical + exercise + gender + married, data = hmoDF)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-0.95026 -0.20519 -0.05875  0.12966  1.14694
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.7050995   0.0234405  -30.080 < 2e-16 ***
age             0.0074128   0.0002679   27.667 < 2e-16 ***
bmi            0.0126274   0.0006350   19.884 < 2e-16 ***
children       0.0114130   0.0031051    3.676 0.000239 ***
smokeryes      0.5943609   0.0095534   62.215 < 2e-16 ***
hypertension   0.0344304   0.0094501    3.643 0.000271 ***
yearly_physicalYes 0.0219715   0.0087301    2.517 0.011864 *
exerciseNot-Active 0.1691899   0.0087241   19.393 < 2e-16 ***
gendermale     0.0143293   0.0075938    1.887 0.059202 .
marriedNot_Married 0.0082186   0.0080059    1.027 0.304656
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.3283 on 7572 degrees of freedom
Multiple R-squared:  0.4262,    Adjusted R-squared:  0.4255
F-statistic: 624.9 on 9 and 7572 DF, p-value: < 2.2e-16
```

```
Call:
lm(formula = isexpensive ~ age + bmi + children + smoker + hypertension +
    yearly_physical + exercise + gender + married, data = hmoDF)
```

```
Coefficients:
            (Intercept)              age              bmi              children
            -0.705100              0.007413              0.012627              0.011413
yearly_physicalYes exerciseNot-Active              gendermale marriedNot_Married
            0.021972              0.169190              0.014329              0.008219
smokeryes              hypertension
            0.594361              0.034430
```



### 7.3 Support Vector Machine

Support vector machines (SVM) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis.

For SVM, we executed multiple combinations of the attribute to predict the expensive factor. We found out the model with highest **accuracy of 87.6% and sensitivity 96.22%** had the attributes such as age, bmi, children, smoker, yearly\_physical, exercise, hypertension. These were the same attributes that linear model predicted to be most significant.

Code:

```
{r}
library(caret)
library(kernlab)
set.seed(123)

trainList <- createDataPartition(y=hmosvmDF$isexpensive ,p=.8 , list=FALSE)
trainset <- hmosvmDF[trainList,]
testset <- hmosvmDF[-trainList,]
svmModelOne <- ksvm(isexpensive ~ age + bmi + children + smoker + hypertension + yearly_physical + exercise,data=trainset,C=5,cross=3,prob.model = TRUE )
svmModelOne
```

In the above code, we first divided the dataset into training and testing dataset in the ratio of 80:20. Next, we used the training data to train the model using ksvm() method.

```
{r}
svmPredOne<- predict(svmModelOne,newdata=testset,type='response')
table(svmPredOne,testset$isexpensive)

confusionMatrix(svmPredOne,testset$isexpensive)
```

Now, the testing data using the 20% already set aside training dataset. The predict() function predicts the outcome of each of the unique entry. The confusionMatrix() calculates the accuracy and sensitivity further.

Output:

```
svmPredOne FALSE TRUE
  FALSE 1094 145
  TRUE   43 234
Confusion Matrix and Statistics

      Reference
Prediction FALSE TRUE
  FALSE 1094 145
  TRUE   43 234

      Accuracy : 0.876
      95% CI : (0.8583, 0.8922)
No Information Rate : 0.75
P-Value [Acc > NIR] : < 2.2e-16

      Kappa : 0.6367

McNemar's Test P-Value : 1.756e-13

      Sensitivity : 0.9622
      Specificity : 0.6174
  Pos Pred Value : 0.8830
  Neg Pred Value : 0.8448
    Prevalence : 0.7500
  Detection Rate : 0.7216
Detection Prevalence : 0.8173
Balanced Accuracy : 0.7898

      'Positive' Class : FALSE
```

Here the model shows how many values it has predicted right or wrong.

The table shows that there are 1094 False values which are accurately predicted as false and 234 values which are True and accurately predicted true.

## 7.4 Decision Tree

The flexible machine learning technique known as decision trees can carry out both classification and regression tasks. They are extremely strong algorithms that can successfully fit complicated datasets. Additionally, decision trees are essential parts of random forests, one of the most effective Machine Learning algorithms now in use. The decision tree algo generated the highest accuracy and sensitivity of 88.46% and 97.27% respectively.

Here we have used the following main attribute and coefficients.

Main attribute : isexpensive

Coefficients : age, bmi, children, smoker, hypertension, exercise, yearly\_physical,

location\_type.

The implementation of decision tree is similar to SVM Model where only difference for decision tree is, here we use rpart() function to train the model

Code:

```
library(rpart)
library(rpart.plot)

treeTrainList <- createDataPartition(y=hmosvmDF$isexpensive ,p=.8 , list=FALSE)
trainset <- hmosvmDF[trainList,]
testset <- hmosvmDF[-trainList,]
hmosvmDF$isexpensive <- as.factor(hmosvmDF$isexpensive)
tree <- rpart(isexpensive ~ age + bmi + children + smoker + hypertension + exercise + yearly_physical + location_type, data=trainset, method='class')
#tree<-train(isexpensive ~ age + bmi + children + smoker + hypertension + exercise + yearly_physical, data=hmosvmDF,method='rpart')
tree
```

```
treePred<- predict(tree,newdata=testset,type='class')
table(treePred,testset$isexpensive)

# str(treePred)
# str(as.factor(testset$isexpensive))

confusionMatrix(treePred,as.factor(testset$isexpensive))
View(testset)
```

Output:

```
treePred FALSE TRUE
FALSE 1106 144
TRUE 31 235
Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
FALSE 1106 144
TRUE 31 235

Accuracy : 0.8846
95% CI : (0.8674, 0.9002)
No Information Rate : 0.75
P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.6582

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

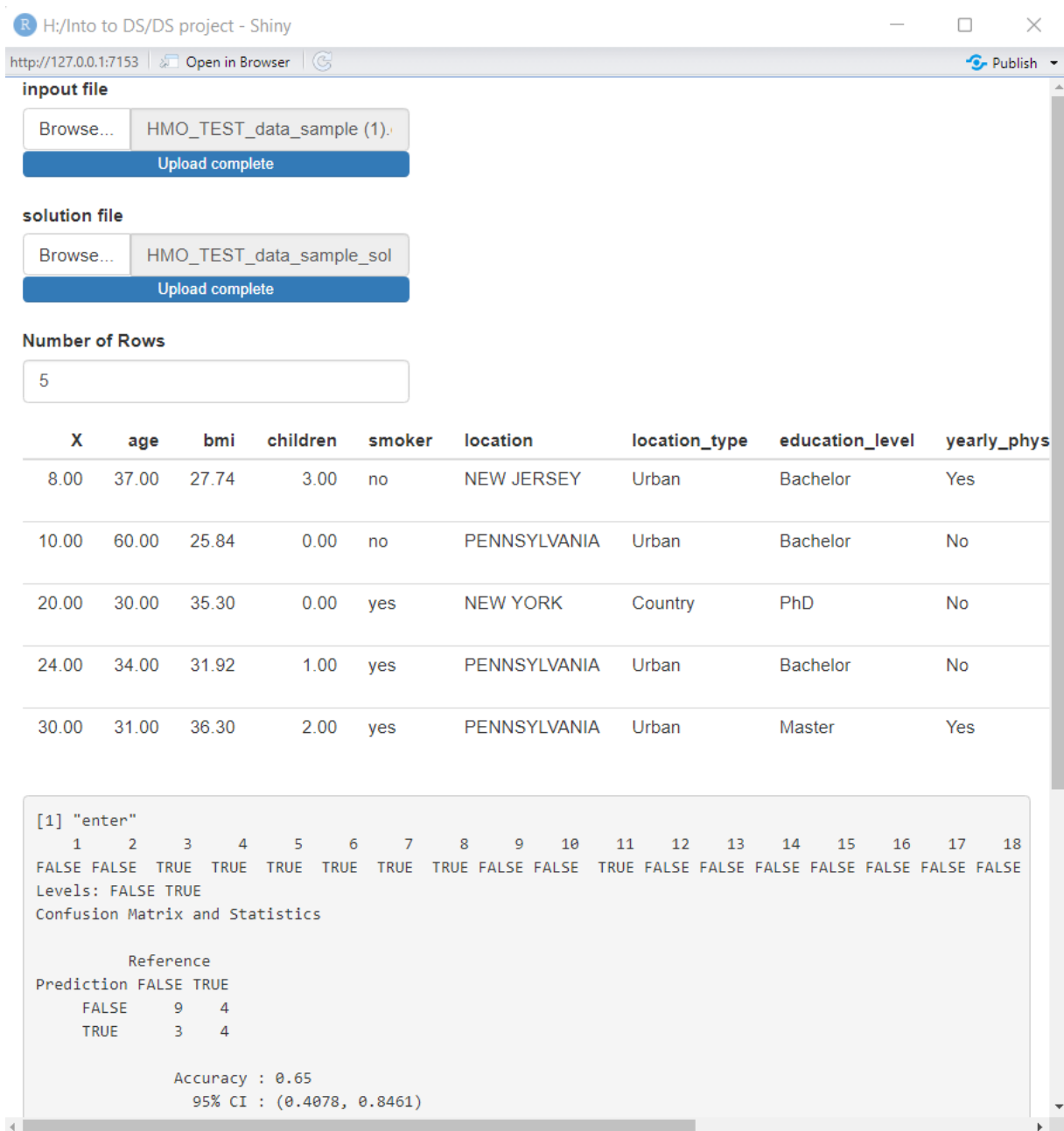
Sensitivity : 0.9727
Specificity : 0.6201
Pos Pred Value : 0.8848
Neg Pred Value : 0.8835
Prevalence : 0.7500
Detection Rate : 0.7296
Detection Prevalence : 0.8245
Balanced Accuracy : 0.7964

'Positive' Class : FALSE
```

## 8. Shiny App

We implemented an application that would predict the expense factor for an individual using dynamic database. The application would predict whether that particular customer would be expensive or not the next year and would also generate the accuracy and sensitivity based on the best model that we selected in previous steps which is Decision Tree.

The screenshot below shows the dashboard of our shiny application



**input file**

Browse... HMO\_TEST\_data\_sample (1).  
Upload complete

**solution file**

Browse... HMO\_TEST\_data\_sample\_sol  
Upload complete

**Number of Rows**

5

| X     | age   | bmi   | children | smoker | location     | location_type | education_level | yearly_phys |
|-------|-------|-------|----------|--------|--------------|---------------|-----------------|-------------|
| 8.00  | 37.00 | 27.74 | 3.00     | no     | NEW JERSEY   | Urban         | Bachelor        | Yes         |
| 10.00 | 60.00 | 25.84 | 0.00     | no     | PENNSYLVANIA | Urban         | Bachelor        | No          |
| 20.00 | 30.00 | 35.30 | 0.00     | yes    | NEW YORK     | Country       | PhD             | No          |
| 24.00 | 34.00 | 31.92 | 1.00     | yes    | PENNSYLVANIA | Urban         | Bachelor        | No          |
| 30.00 | 31.00 | 36.30 | 2.00     | yes    | PENNSYLVANIA | Urban         | Master          | Yes         |

```
[1] "enter"
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18
FALSE FALSE TRUE TRUE TRUE TRUE TRUE TRUE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE
Levels: FALSE TRUE
Confusion Matrix and Statistics

      Reference
Prediction FALSE TRUE
 FALSE      9      4
  TRUE      3      4

Accuracy : 0.65
95% CI : (0.4078, 0.8461)
```

Here, we have used two input files, which has a dataset that is displayed above. Each row is a data for every unique person and our application predicts whether that person would be expensive or not next year by displaying TRUE or FALSE just below the table database.

We can see from the above output that, row 1 ie patient 1 has been predicted False meaning that particular person won't be expensive for next year.

Next, the get the accuracy and sensitivity of our model and its predicted we have also given an input of solution csv file that has the expected out. Our model compares the predicted values with the values in the solution file and generates confusion Matrix as below

```
[1] "enter"
  1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17   18
FALSE FALSE TRUE TRUE TRUE TRUE TRUE TRUE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE
Levels: FALSE TRUE
Confusion Matrix and Statistics

          Reference
Prediction FALSE TRUE
   FALSE      9      4
   TRUE       3      4

      Accuracy : 0.65
      95% CI : (0.4078, 0.8461)
 No Information Rate : 0.6
 P-Value [Acc > NIR] : 0.4159

      Kappa : 0.2553

McNemar's Test P-Value : 1.0000

      Sensitivity : 0.7500
      Specificity : 0.5000
 Pos Pred Value : 0.6923
 Neg Pred Value : 0.5714
  Prevalence : 0.6000
Detection Rate : 0.4500
Detection Prevalence : 0.6500
Balanced Accuracy : 0.6250

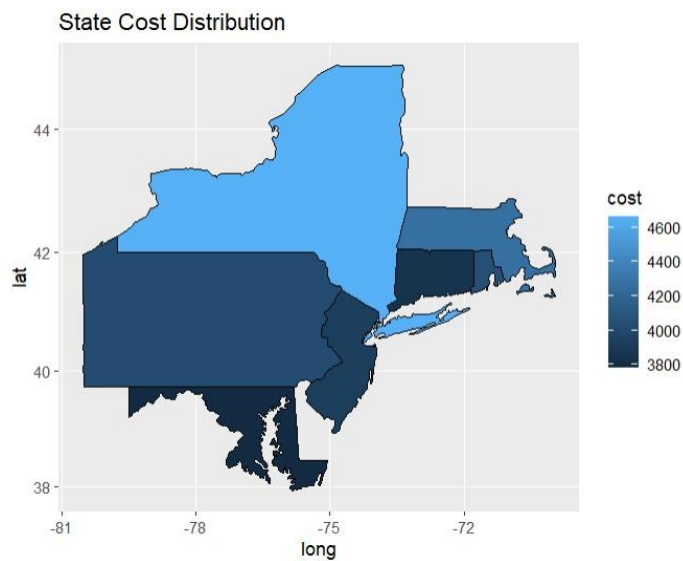
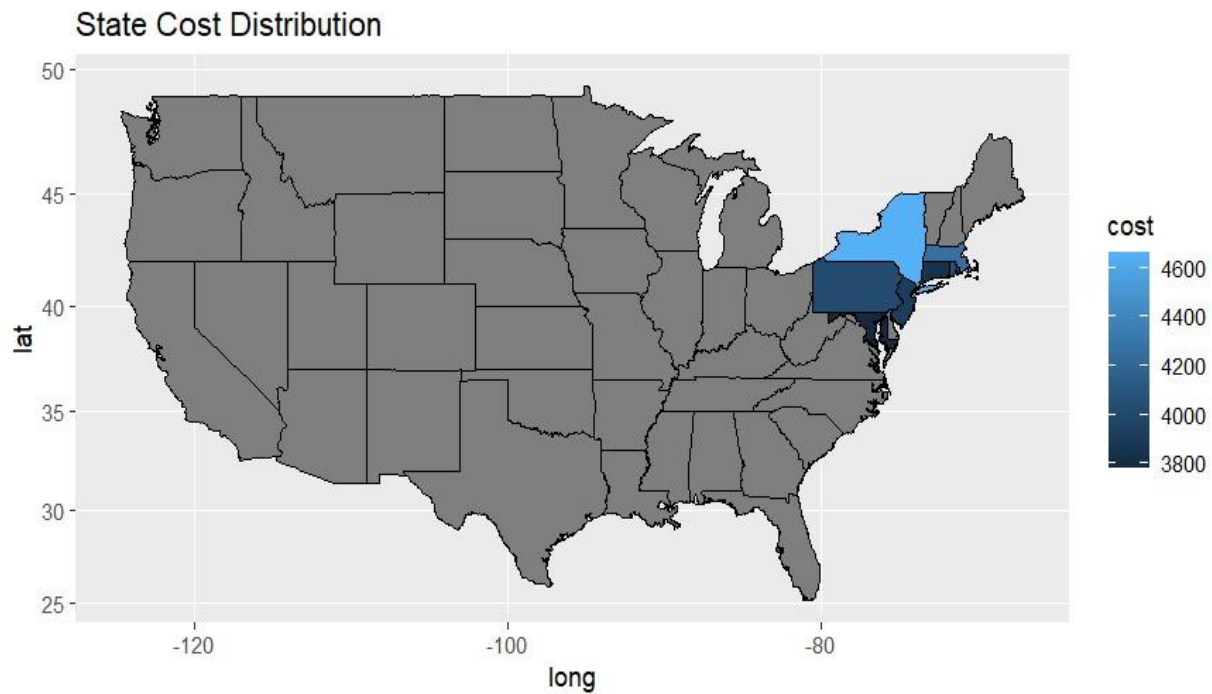
      'Positive' Class : FALSE
```

Here, our model has predicted the result with 65% accuracy and 75% Sensitivity which is very good.

## 9. Location Visualizations

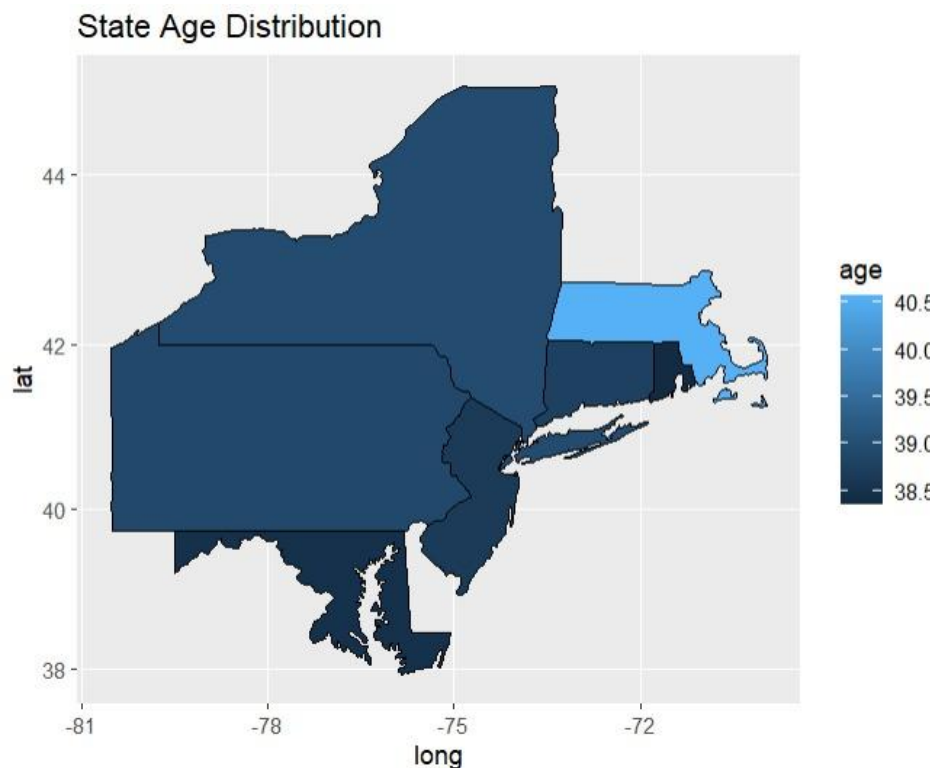
### Healthcare cost distribution based on geography:

This map shows the 48 contiguous United States, with the 7 states whose data we analyzed appearing in different shades of blue.



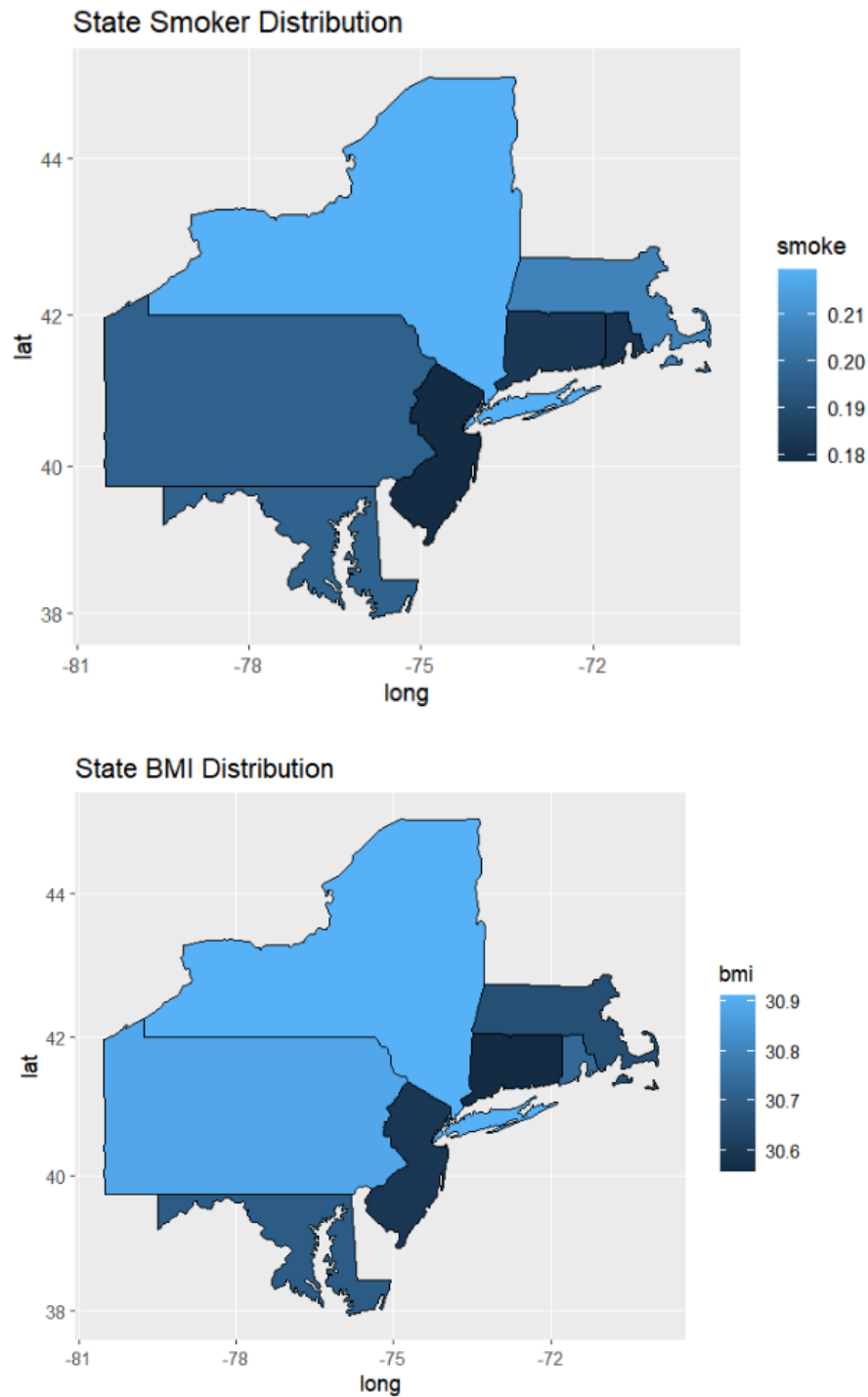
From this map above, we can see that the we states such as Pennsylvania, New York, Maryland, Connecticut, Massachusetts, Rhode Island, New Jersey. With New York paying the most for healthcare. We could implement various measures in those high cost states such as cheap insurance and hospital facilities, reduced taxes on services, health drives, etc.

### State Age Distribution



The distribution of patients based on their respective ages can be seen from the map pictured on the left, with Massachusetts having the most aged population.

## State Smoker Count and BMI Distribution



From the above map, we can conclude that New York has the highest amount of people who smoke and had highest BMI. Also, BMI and smoking were the factors that were most



highlighted in our analysis to increase the medical expense of the people. Hence, to reduce medical expenses in these areas we can initiate no smoking drives, spread awareness about no smoking among teenagers, higher taxes on smoking devices, promote health through opening gyms, free diet plans, good food habits awareness, etc.

## 10. Interpretation and recommendations

|   |                      | HIGH COST  | LOW COST   | ACTIONS  |
|---|----------------------|------------|------------|--|
| 1 | SMOKER STATUS        | ACTIVE     | NON-ACTIVE | <ul style="list-style-type: none"> <li>Smoking can be avoided to reduce the Healthcare cost.</li> <li>Maintaining a lower BMI will help Reduce the healthcare cost.</li> </ul>   |
| 2 | EXERCISER STATUS     | NON-ACTIVE | ACTIVE     | <ul style="list-style-type: none"> <li>An active exerciser can reduce their Healthcare cost so individual can exercise Reduce expenditure.</li> <li>Maintaining a lower BMI will help Reduce the healthcare cost.</li> </ul> |
| 3 | YEARLY PHYSICAL TEST | NOT TAKEN  | TAKEN      | <ul style="list-style-type: none"> <li>Yearly physical tests can be taken to reduce the healthcare cost</li> </ul>   |
| 4 | LOCATION             | CITY       | COUNTY     | <ul style="list-style-type: none"> <li>A person can consider relocation to a county from city to reduce the healthcare cost.</li> <li>Maintaining a lower BMI will help Reduce the healthcare cost.</li> </ul>               |