Rivera_Fidel_IST387_FinalProject

Fidel Rivera

2023-09-18

1. Use read_csv() to read your CSV file (https://intro-datascience.s3.us-east2.amazonaws.com/HMO_data.csv (https://intro-datascience.s3.us-east2.amazonaws.com/HMO_data.csv)) into a dataframe called costDF. Describe the variables in the dataframe using descriptive statistics– add a brief comment to explain what you see. How many observations and variables does the dataframe have? Besure to comment your code and describe the results you found.

```
costDF <- read.csv('https://intro-datascience.s3.us-east-2.amazonaws.com/HMO_data.csv')
#the read.csv function loads up the URL into a dataframe that we can then analyze. costD
F is the variable we will store it under.
summary(costDF)
```

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

```
#the summary function gives us very descriptive statistics about each category of data i
n the df, which helps us understand what we're working with.

#there are 7582 observations across 14 different variables. for numeric values, we see m
inimums, 1st & 3rd quartiles, means and max's. for character values we see the length or
number of observations, as well as the class and mode.
```

2.Check the numerical variables for missing values using the is.na(). Determine what to do with any NAs.

```
sum(is.na(costDF))
```

```
## [1] 158
```

```
#i checked for missing values with is.na. At first, it pulled up the entire dataframe wh
ich is a little time consuming to look through so i wrapped it in the sum function to te
ll me how many missing values there are. After further examination, there seemed to be
'NA' values in both bmi and hypertension

costDF$bmi[is.na(costDF$bmi)]<-mean(costDF$bmi,na.rm=TRUE)
#this line of code takes any values in the bmi column that were valued at 'NA', and repl
aces them with the average value across all the other vales in the variable. I thought t
his was the best approach as it keeps the data unbiased and as close to original as poss
ible, while filling values.
costDF$hypertension[is.na(costDF$hypertension)]<-mean(costDF$hypertension,na.rm=TRUE)
#here i did the same thing, but instead for the hypertension column. There are no missin
g values now, which can be proven with the initial is.na function.
sum(is.na(costDF))
```

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

3.Generate tables (using the table() function) for any 3 categorical response variables (e.g., exercise), and write a sentence for each, describing what you see.

```
table(costDF$location)
```

```
##
##   CONNECTICUT      MARYLAND MASSACHUSETTS    NEW JERSEY      NEW YORK
##           611           747           465           498           547
##  PENNSYLVANIA  RHODE ISLAND
##          4010           704
```

```
#for the location variable, I see how many instances (represented by a row in the data f
rame) lived in each state.
table(costDF$yearly_physical)
```

```
##
##   No  Yes
## 5699 1883
```

```
#for the yearly_physical variable, I see how many people in the data frame completed the
ir yearly physical and who did not.
table(costDF$education_level)
```

```
##
##          Bachelor            Master No College Degree               PhD
##              4578              1533               759               712
```

```
#for the education_level variable, I see how many people in the data frame have the avai
lable degree types as their highest degree completed.
```

4.Create a new attribute, called expensive, which is based on the person costs for the past year. Explain your logic for how you defined expensive.

```
costDF$expensive <- as.factor(ifelse(costDF$cost > mean(costDF$cost), 'Yes',
                     'No'))

#i created my new attribute based on if the person's cost was above the average cost of
all patients in the data frame. I used the as.factor function to help me with this as we
ll as the if-else function. As for expensive, if the cost is greater than the mean cost
of all patients, the patient is expensive, or "Yes", otherwise, they are not expensive,
or "No".

head(costDF)
```

```
##   X age    bmi children smoker      location location_type education_level
## 1 1  18 27.900        0    yes   CONNECTICUT         Urban        Bachelor
## 2 2  19 33.770        1     no  RHODE ISLAND         Urban        Bachelor
## 3 3  27 33.000        3     no MASSACHUSETTS         Urban          Master
## 4 4  34 22.705        0     no  PENNSYLVANIA       Country          Master
## 5 5  32 28.880        0     no  PENNSYLVANIA       Country             PhD
## 6 7  47 33.440        1     no  PENNSYLVANIA         Urban        Bachelor
##   yearly_physical   exercise married hypertension gender cost expensive
## 1              No     Active Married            0 female 1746        No
## 2              No Not-Active Married            0   male  602        No
## 3              No     Active Married            0   male  576        No
## 4              No Not-Active Married            1   male 5562       Yes
## 5              No Not-Active Married            0   male  836        No
## 6              No Not-Active Married            0 female 3842        No
```

```
#called for the first few rows of the data frame to see if my idea worked.
```

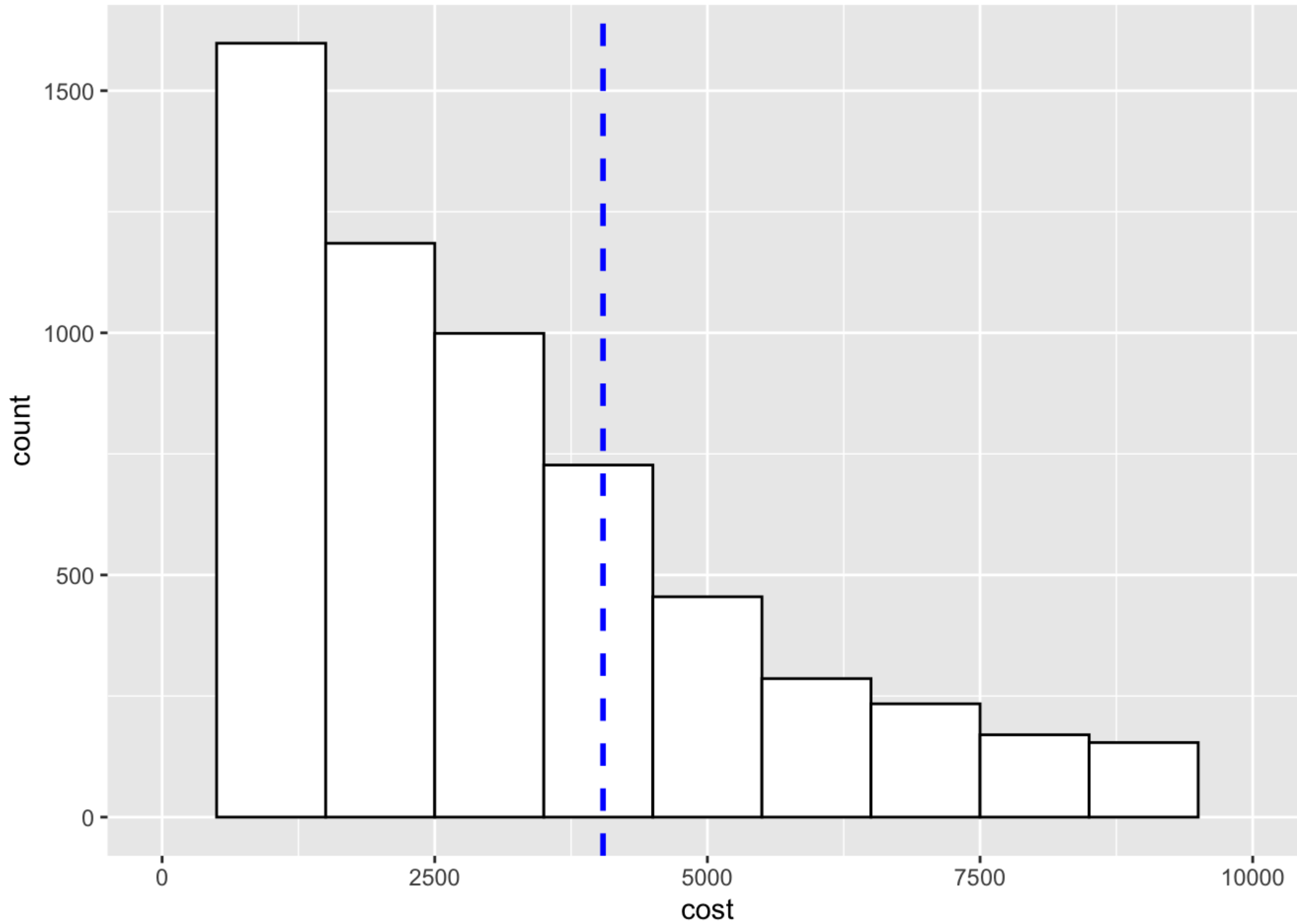
5.Create two histograms for any variable, with one histogram for that variable for all the 'expensive' people, and the other histogram for that variable for all the other people. Do this for two other attributes. Explain what insight was generated.

```
library(ggplot2)
                           #######COST HISTOGRAM#######
#i decided to library ggplot2 in order to create better histograms.
histoCost <- ggplot(costDF, aes(x=cost)) + geom_histogram(binwidth=1000,color="black", f
ill="white") + xlim(0, 10000)
#created a histogram showing the cost range of patients and how many patients are in thi
s range (count).

histoCost + geom_vline(aes(xintercept=mean(cost)),
            color="blue", linetype="dashed", size=1)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
```

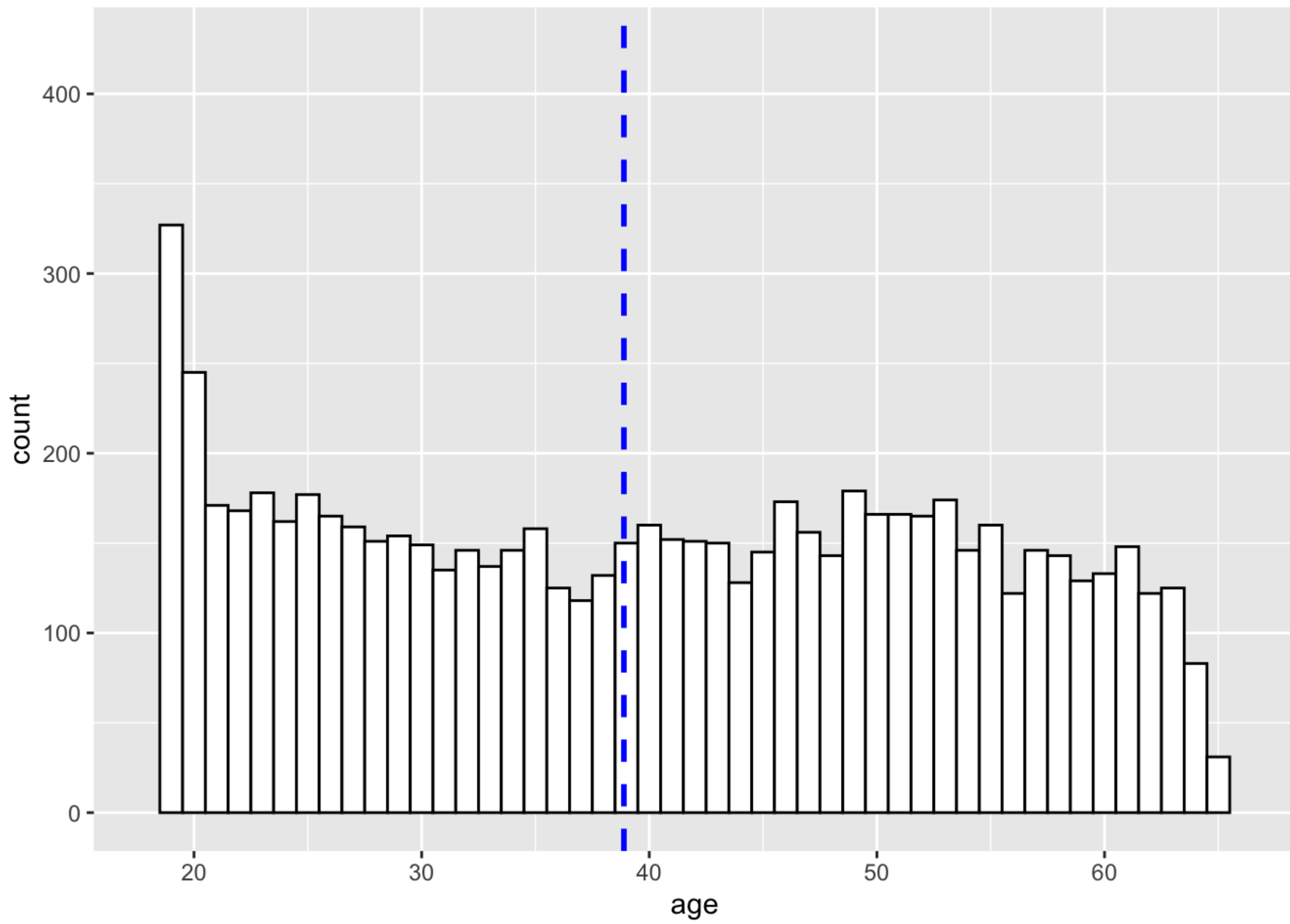
```
## Warning: Removed 708 rows containing non-finite values (`stat_bin()`).
```

```
## Warning: Removed 2 rows containing missing values (`geom_bar()`).
```




```
#i decided to add a mean line to better show where the average is. Anything on the left
is considered not expensive, and the right is considered expensive.

#aditionally, there are warnings that rows were removed. this is because there were cert
ain values that are far beyond the average which dont represent the majority of the dat
a. These outliers werent included so a better representation of the data could be provid
ed.
                           ############################

                           #######AGE HISTOGRAM#######
histoAge <- ggplot(costDF, aes(x=age)) + geom_histogram(binwidth=1,color="black", fill
="white") + xlim(18, 66)
#created a histogram showing the age range of patients and how many patients are in this
range (count).I also set x limits to 18 and 66 respectively, as these were the youngest
and oldest ages found in the data.

histoAge + geom_vline(aes(xintercept=mean(age)),
            color="blue", linetype="dashed", size=1)
```

```
## Warning: Removed 2 rows containing missing values (`geom_bar()`).
```




```
#i decided to add a mean line to better show where the average is. Anything on the left
is below the average age range, and the right is above the range.

                          ###########################
```

6.Currently, the data is at the individual (person) level. Create a new data frame called state based on the location and cost variables, to show the average cost for each state (hint: explore the aggregate() or summarize() functions). Possible names for the resulting two columns in state are name and ave_cost.

```
state <- data.frame(
  name = costDF$location,
  ave_cost = costDF$cost
)
#first created the data frame and set the location and cost from costDF to 'name' and 'a
ve_cost'
state <- aggregate(ave_cost ~ name, data = state, mean)
#aggregated the data frame to combine values and find the mean for each state
state
```

```
##            name ave_cost
## 1   CONNECTICUT 3847.519
## 2      MARYLAND 3784.174
## 3 MASSACHUSETTS 4267.540
## 4    NEW JERSEY 3930.564
## 5      NEW YORK 4661.506
## 6  PENNSYLVANIA 4023.115
## 7  RHODE ISLAND 4050.791
```

```
#displays data frame to check work
```

7.Determine which state had the highest cost per person. Show the code you used to identify it.

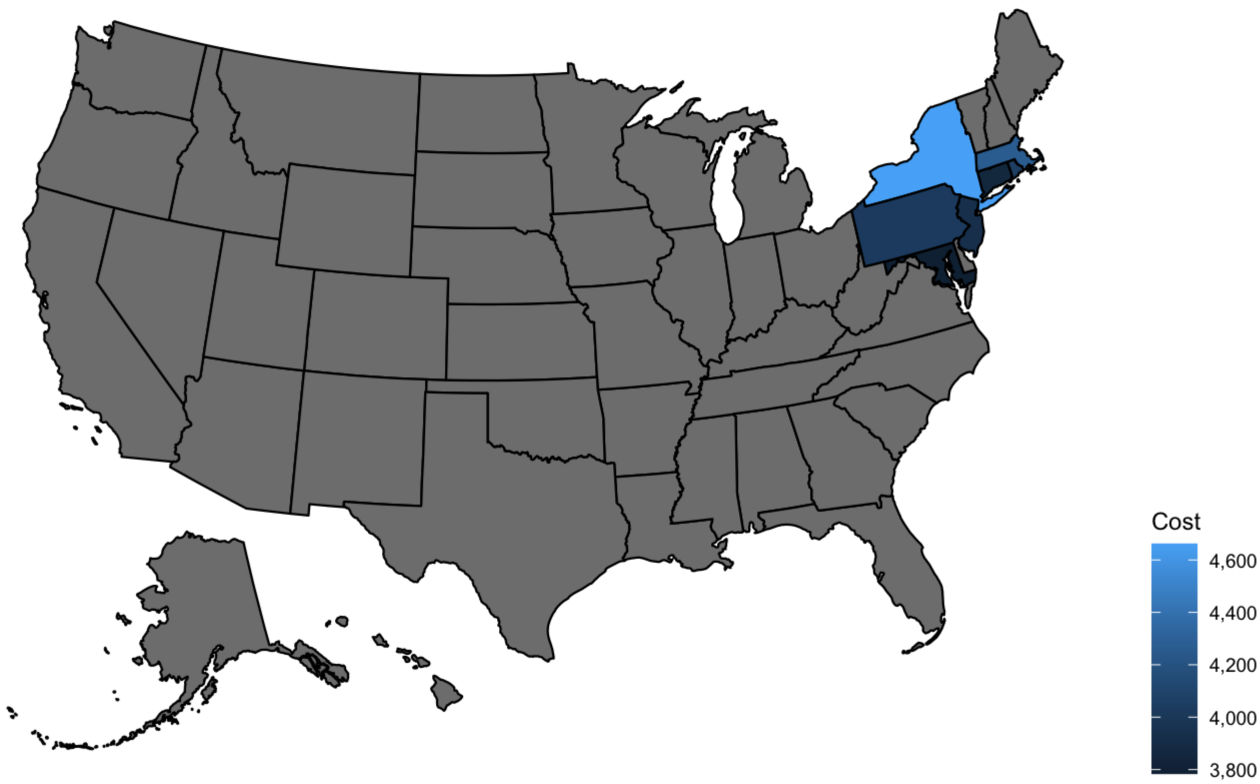
```
state[which.max(state$ave_cost),]
```

```
##       name ave_cost
## 5 NEW YORK 4661.506
```

```
#i first called on the data frame which in this case is called state. I also used the wh
ich.max function and referenced the ave_cost column, which will take the max value in th
is column and associate it with which name the highest value belongs to.
```

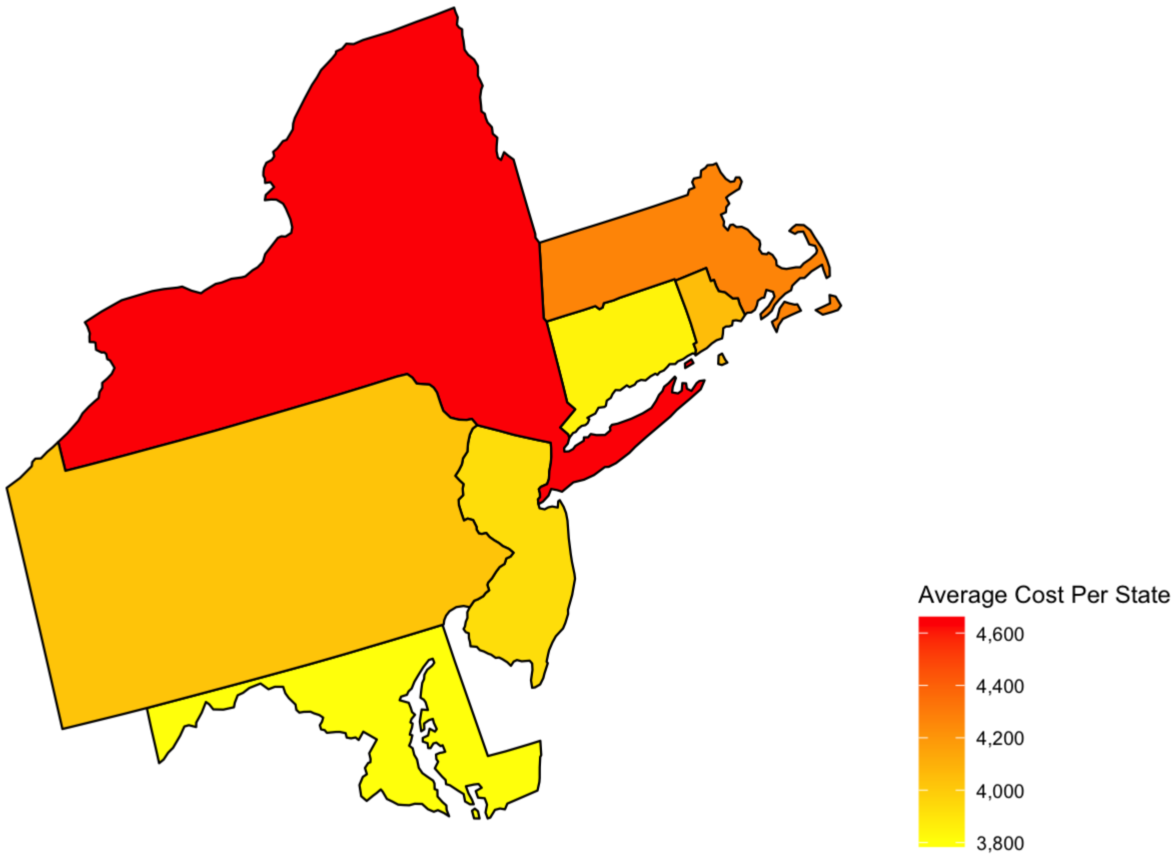
8.Create a color gradient map of the USA, where the color of each state indicates its cost per person. Assuming your data frame is called state. Be sure to use expand_limits set what to view, and if appropriate, to zoom in on a part of the map. Write a comment about what you see in the map.

```
library(usmap)
library(ggplot2)
#called upon two very important tools that i felt would be useful for creating a map.

state <- data.frame(
  state = costDF$location,
  ave_cost = costDF$cost
)

state <- aggregate(ave_cost ~ state, data = state, mean)
#re worked the state data frame to change the "name" variable to state to make it more c
ompatible with usmap.

plot_usmap(data = state, values = "ave_cost", color = "black") +
  scale_fill_continuous(name = "Cost", label = scales::comma) +
  theme(legend.position = "right")
```




```
#The code above is for the entire US map. Many states are not in the data so they are le
ft grey

plot_usmap(data = state, values = "ave_cost", include = c("CT", "MD", "MA", "NJ","NY","P
A","RI"), color = "black") + #included necessary 7 states which "zooms into map"
  scale_fill_continuous(low = "yellow", high = "red", name = "Average Cost Per State", l
abel = scales::comma) +
  labs(title = "Cost") +
  theme(legend.position = "right")
```

Cost




```
#used plot us map function, set aesthetic and added lables to map.
```

9.Returning to the full data set (costDF), convert some of the fields into factor variables and then use Association Rules Mining to see if there are patterns of attributes that connect with being expensive. Here's a line of code that converts the two attributes from the costDF data frame into a new data frame that only contains factor variables (you should do more than just these three attributes):

costCat <- data.frame(location=as.factor(costDFlocation), location_type = as.factor(costDFlocation_type), expensive =as.factor(costDF$expensive))

Using the itemFrequencyPlot() function, inspect the variables in costCat and include a comment on what you see.

```
library(arules)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'arules'
```

```
## The following objects are masked from 'package:base':
##
##     abbreviate, write
```

```
library(arulesViz)

#libraried tools that are essential in order to use itemFrequencyPlot.

costCat <- data.frame(location=as.factor(costDF$location),
location_type=as.factor(costDF$location_type),
                   expensive =as.factor(costDF$expensive))
#used the above code to convert the data type into a factor variable and load them into
a new data frame. but changed it from code to cost as i believe this may have been an er
ror!

#itemFrequencyPlot(item = costCat$location, frequency = costCat$expensive)

#getting an error when trying to perform this function, called for item and frequency. t
he error supposedly comes from having null values in your data, but i dont have any null
values.
```

10.Remember that you can't use the costCat data frame as input to apriori() directly - you will first have to coerce it to the transactions class. Identify at least two high confidence rules that connect with someone that is expensive. Describe your analysis and these high confidence rules in a few sentences– what do the support, confidence, and lift values of these rules mean?

```
#two high confidence rules that connect with someone that is expensive is the area that
they live in as well as if they are a smoker or not. There seems to be a correlation bet
ween the two high confidence rules.
```

11. Next, we will turn to supervised machine learning to try and predict high cost (i.e., expensive) people, using Support Vector Machines and Trees.

Using the createDataPartition() function from the caret package, partition book into a trainSet and a testSet, where the trainSet is .7 of the entire data and y=costDF$expensive.

```
library(caret)
```

```
## Loading required package: lattice
```

```
#needed this specific tool for the create data partition tool

# Set the seed for reproducibility
set.seed(123)

# Partition the data frame into a trainSet and a testSet
trainSetIndex <- createDataPartition(costDF$expensive, p = 0.7, list = FALSE)
#trainSet <- book[trainSetIndex, ]
#testSet <- book[-trainSetIndex, ]
```

12.You are now ready to create a support vector machine and tree models. You can use the same parameters we used in the HW– of course, don't forget to change the name of the variable you are predicting to expensive.

Remember,you need to create two models (one using SVM and using rpart).

Once your models are trained (the SVM model may take a bit of time since it's a big dataset), use the predict() function to see how well your model performs on the testSet (for each model). Finally, use confusionMatrix(), create a confusion matrix and view the error of for both models.

```
library(rpart)
library(rpart.plot)
#Tree <- rpart(costCat~., data=costCat)
#prp(Tree, faclen=0, cex=0.8, extra=1)
```

13.Which model is better (or are they the same)? Explain how you arrived at your conclusion.

```
#I feel the models are similar because ultimately no matter the strategy or approach tak
en if the data remains the same the results can only be so different from each other.
```

14. Write a paragraph, to be sent to the CEO of the HMO, summarizing the most important actionable insight you found in your analysis. In other words, based on your analysis, what would you suggest to the CEO?

```
#Good afternoon CEO,  After analyzing the data provided lots of notable things have aris
ed worthy of analysis. There are many factors that can help predict wether or not a pati
ent would be "expensive" which ultimatley can help your business model. The most importa
nt actionable insight found through the data is that depending on where you live, you ar
e more likely to have a much higher or much lower cost. What I would suggest is to truly
ensure that the cost of these expenses is as much related to their conditions and needs
as possible, and not just because of where they live. Especially if two people are being
treated for the same thing, whatever data can be acquired on this to ensure that no matt
er where in the world they would pay the same price, is the ethical thing to do and a gr
eat way to ensure the business stays progressive.
```