IST 687 Group Project

2022-11-30

1. Install and Library Needed Packages

```
#Library Needed Packages
library(tidyverse)
library(ggplot2)
library(caret)
library(kernlab)
library(imputeTS)
library(e1071)
library(rpart)
library(rpart.plot)
```

2. Load Dataset

```
data <- read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/HMO_data.csv")
```

```
## Rows: 7582 Columns: 14
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (8): smoker, location, location_type, education_level, yearly_physical, ...
## dbl (6): X, age, bmi, children, hypertension, cost
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

3. Change Character Variables to Factor Variables for Analysis

```
str(data)
```

```
## 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" "PENNSY
LVANIA" ...
##  $ 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 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   X = col_double(),
##   ..   age = col_double(),
##   ..   bmi = col_double(),
##   ..   children = col_double(),
##   ..   smoker = col_character(),
##   ..   location = col_character(),
##   ..   location_type = col_character(),
##   ..   education_level = col_character(),
##   ..   yearly_physical = col_character(),
##   ..   exercise = col_character(),
##   ..   married = col_character(),
##   ..   hypertension = col_double(),
##   ..   gender = col_character(),
##   ..   cost = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
data$smoker = as.factor(data$smoker)
data$location = as.factor(data$location)
data$location_type = as.factor(data$location_type)
data$education_level = as.factor(data$education_level)
data$yearly_physical = as.factor(data$yearly_physical)
data$exercise = as.factor(data$exercise)
data$married = as.factor(data$married)
data$hypertension = as.factor(data$hypertension)
data$gender = as.factor(data$gender)
```

4. Fix NA values and remove ones that can’t be interpolated

```
sum(is.na(data)) #158 Null Values
```

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

```
data = na_interpolation(data)
```

```
## Warning: na_interpolation: No imputation performed for column 12 of the input datase
t.
##                  Reason: Input x is not numeric.
```

```
sum(is.na(data)) #80 Null Values remain. Can't be changed because character value
```

```
## [1] 80
```

```
data = drop_na(data) #Remove these values
sum(is.na(data)) #0 Null values left
```

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

5. Linear Model to see which variables are predictive of Cost

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

```
##
## Call:
## lm(formula = cost ~ ., data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -12012  -1482   -356   1015  41741
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       -9.166e+03  2.712e+02 -33.801  < 2e-16 ***
## X                                  1.183e-05  6.921e-06   1.710 0.087339 .
## age                                1.022e+02  2.648e+00  38.597  < 2e-16 ***
## bmi                                1.817e+02  6.274e+00  28.963  < 2e-16 ***
## children                           2.325e+02  3.071e+01   7.571 4.16e-14 ***
## smokeryes                          7.699e+03  9.445e+01  81.513  < 2e-16 ***
## locationMARYLAND                  -1.164e+02  1.771e+02  -0.657 0.511143
## locationMASSACHUSETTS              3.818e+00  2.005e+02   0.019 0.984810
## locationNEW JERSEY                 1.356e+02  1.959e+02   0.692 0.489028
## locationNEW YORK                   4.927e+02  1.918e+02   2.569 0.010212 *
## locationPENNSYLVANIA               2.888e+01  1.413e+02   0.204 0.838073
## locationRHODE ISLAND               1.368e+02  1.797e+02   0.761 0.446456
## location_typeUrban                -1.662e+01  8.612e+01  -0.193 0.846955
## education_levelMaster             -9.984e+01  9.575e+01  -1.043 0.297114
## education_levelNo College Degree   3.940e+01  1.271e+02   0.310 0.756592
## education_levelPhD                -2.329e+02  1.307e+02  -1.782 0.074808 .
## yearly_physicalYes                 1.396e+02  8.630e+01   1.618 0.105702
## exerciseNot-Active                 2.273e+03  8.614e+01  26.384  < 2e-16 ***
## marriedNot_Married                 1.348e+02  7.913e+01   1.704 0.088422 .
## hypertension1                      3.377e+02  9.310e+01   3.628 0.000288 ***
## gendermale                         2.414e+01  7.506e+01   0.322 0.747758
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3225 on 7481 degrees of freedom
## Multiple R-squared:  0.5754, Adjusted R-squared:  0.5743
## F-statistic: 506.9 on 20 and 7481 DF,  p-value: < 2.2e-16
```

```
#Age, BMI, Children, Being a Smoker, Not Exercising, living in New York, and having Hype
rtension raise costs of healthcare
```

6. Create variable for Expensive Healthcare

```
data$expensive[data$cost >= 5000] <-  "Expensive"
```

```
## Warning: Unknown or uninitialised column: `expensive`.
```

```
#anything greater than or equal to 5000 is expensive
data$expensive[is.na(data$expensive)] <-  "Normal"
data$expensive = as.factor(data$expensive) #change to factor variable
```

7. Partition data into training and test set

```
new_data = data [-14] #removing "cost" variable from data.
trainList <- createDataPartition(y=new_data$expensive,p=.4,list=FALSE) #40% train data
train <- new_data[trainList,] #create train data
test <- new_data[-trainList,] #create test data
```

8. Create SVM Classification Model

```
#build the model
svm.model <- train(expensive ~ ., data = train, method = "svmRadial",
                   trControl=trainControl(method = "none"),
                       preProcess = c("center", "scale"))
svm.model
```

```
## Support Vector Machines with Radial Basis Function Kernel
##
## 3001 samples
##   13 predictor
##    2 classes: 'Expensive', 'Normal'
##
## Pre-processing: centered (20), scaled (20)
## Resampling: None
```

```
svmPred <- predict(svm.model, newdata = test) #predict using test data
```

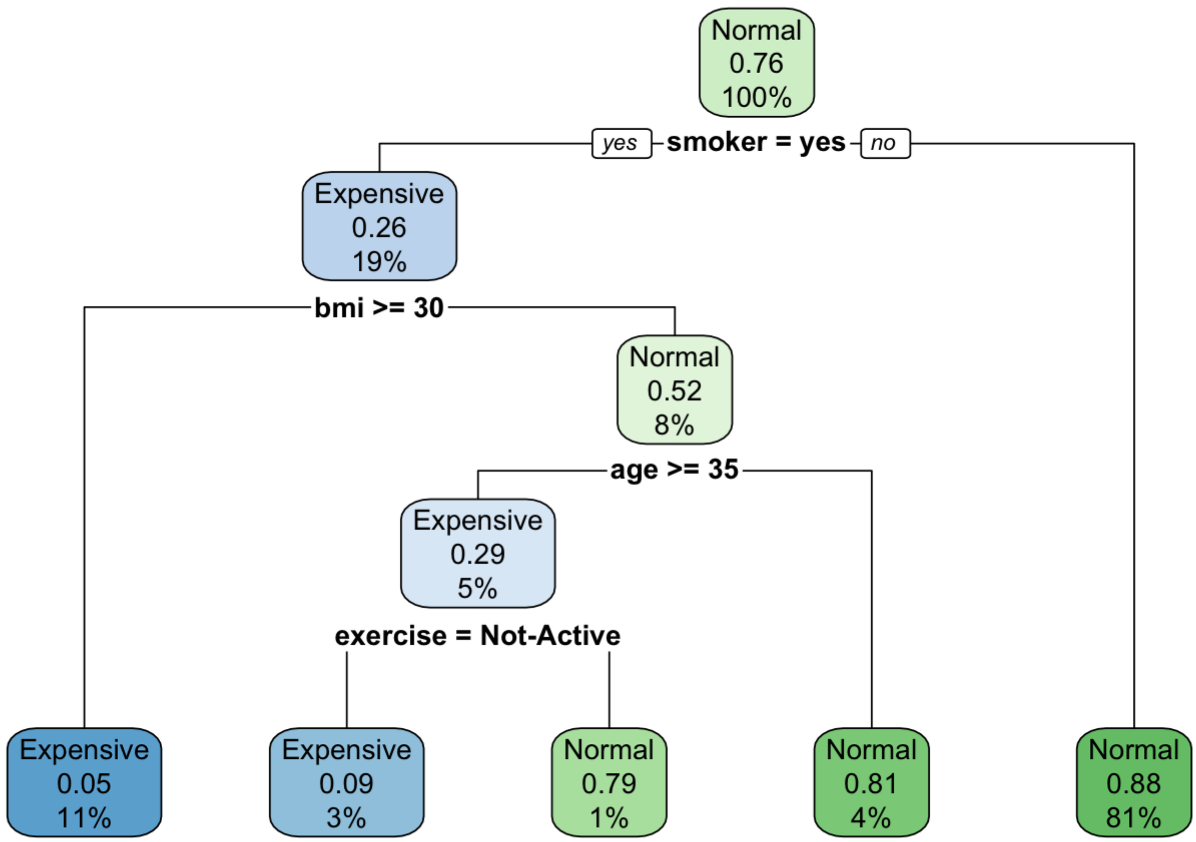
9. Look at Accuracy of Model

```
confusionMatrix(svmPred, test$expensive)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  Expensive Normal
##   Expensive       620    206
##   Normal          451   3224
##
##                Accuracy : 0.854
##                  95% CI : (0.8434, 0.8642)
##     No Information Rate : 0.7621
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5631
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.5789
##             Specificity : 0.9399
##          Pos Pred Value : 0.7506
##          Neg Pred Value : 0.8773
##              Prevalence : 0.2379
##          Detection Rate : 0.1377
##    Detection Prevalence : 0.1835
##       Balanced Accuracy : 0.7594
##
##        'Positive' Class : Expensive
##
```

10. Rpart Model

```
rpartmodel <- rpart(expensive ~ ., data = train)
rpart.plot(rpartmodel)
```




```
predictValues <- predict(rpartmodel, newdata = test, type = "class")
table(predictValues)
```

```
## predictValues
## Expensive    Normal
##       647      3854
```

11. Look at Rpart Accuracy

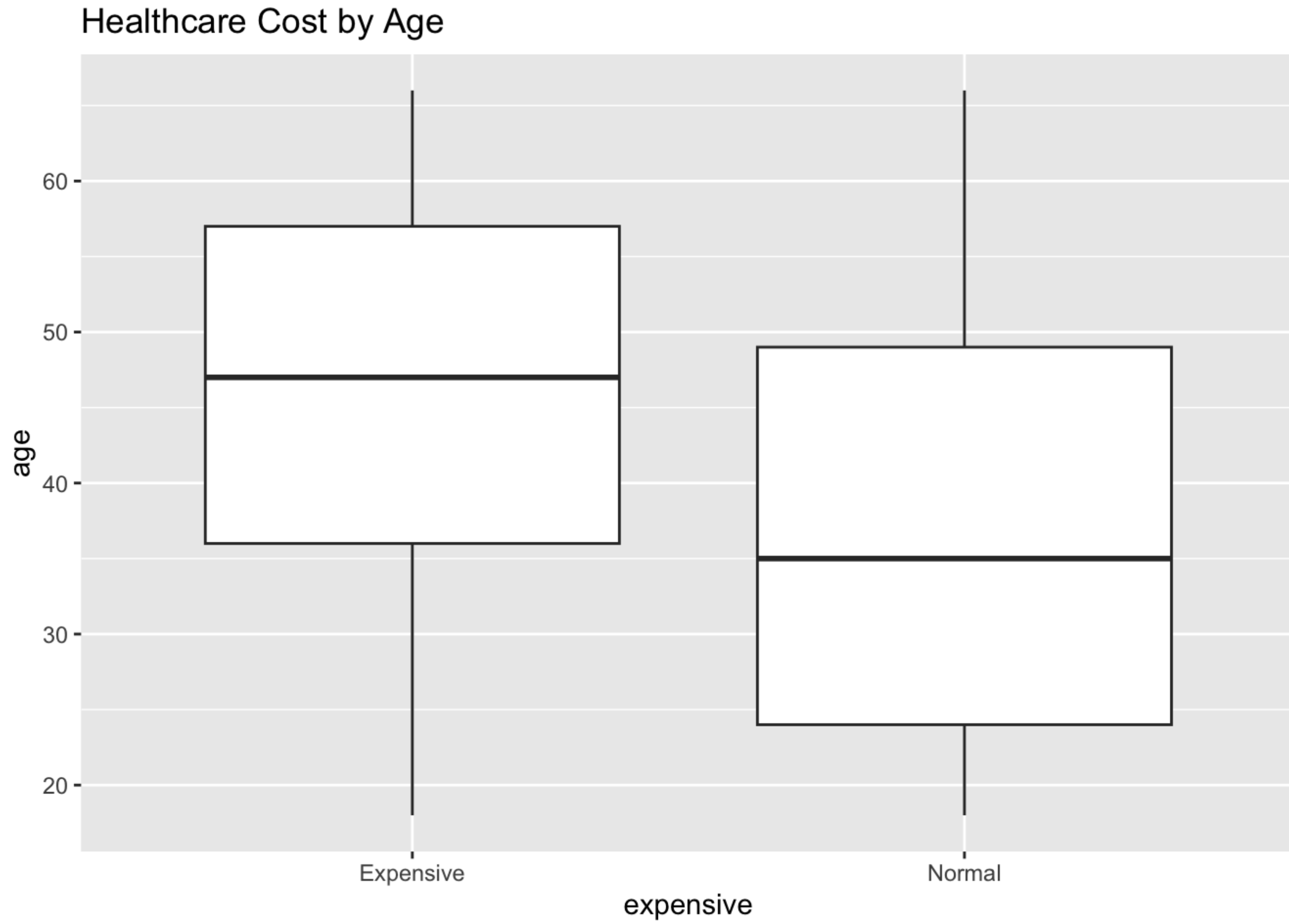
```
confusionMatrix(predictValues, test$expensive)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  Expensive Normal
##   Expensive       587     60
##   Normal          484   3370
##
##                Accuracy : 0.8791
##                  95% CI : (0.8693, 0.8885)
##     No Information Rate : 0.7621
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6142
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.5481
##             Specificity : 0.9825
##          Pos Pred Value : 0.9073
##          Neg Pred Value : 0.8744
##              Prevalence : 0.2379
##          Detection Rate : 0.1304
##    Detection Prevalence : 0.1437
##       Balanced Accuracy : 0.7653
##
##        'Positive' Class : Expensive
##
```

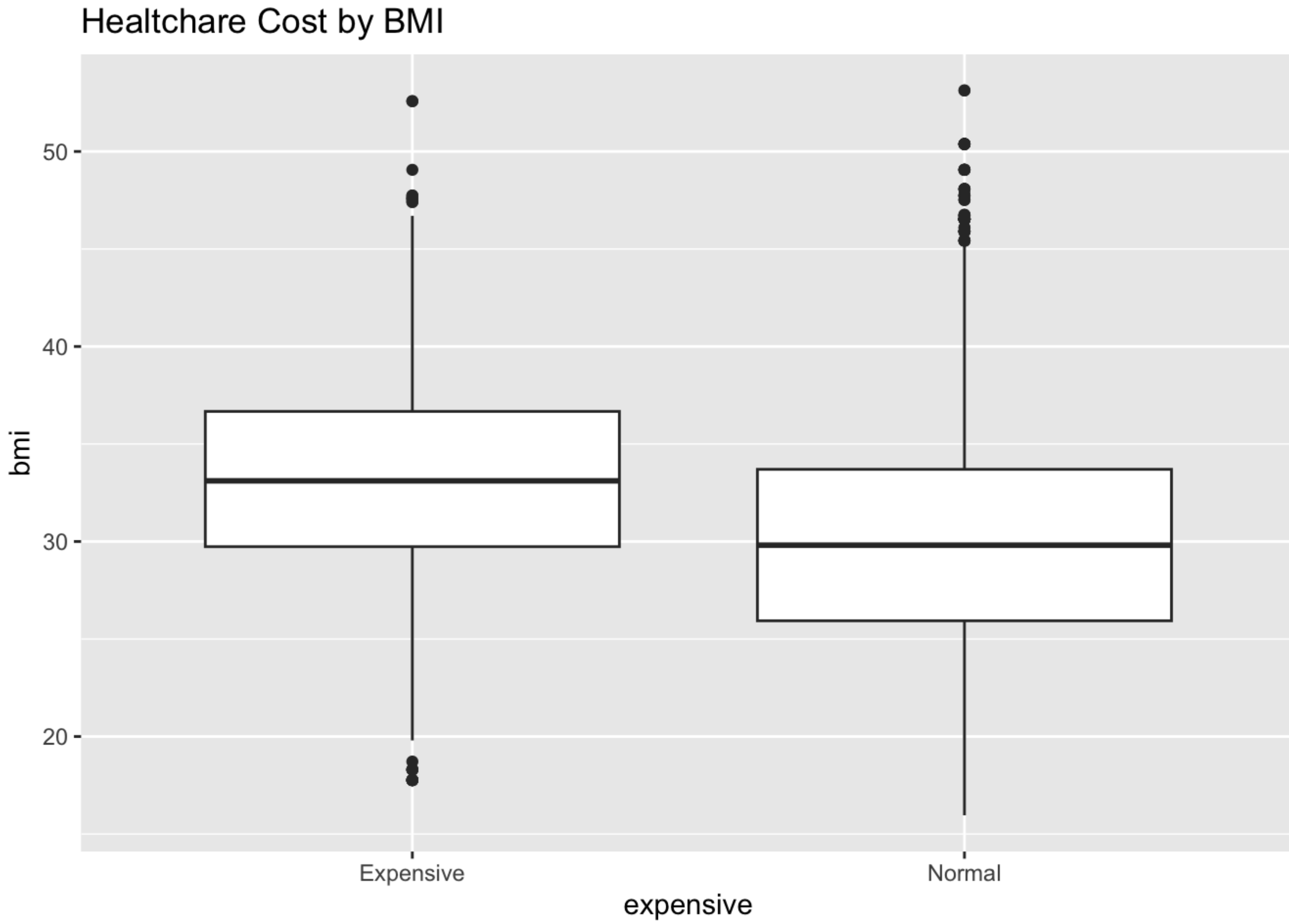
```
#less sensitive than SVM
```

12. Visualizations

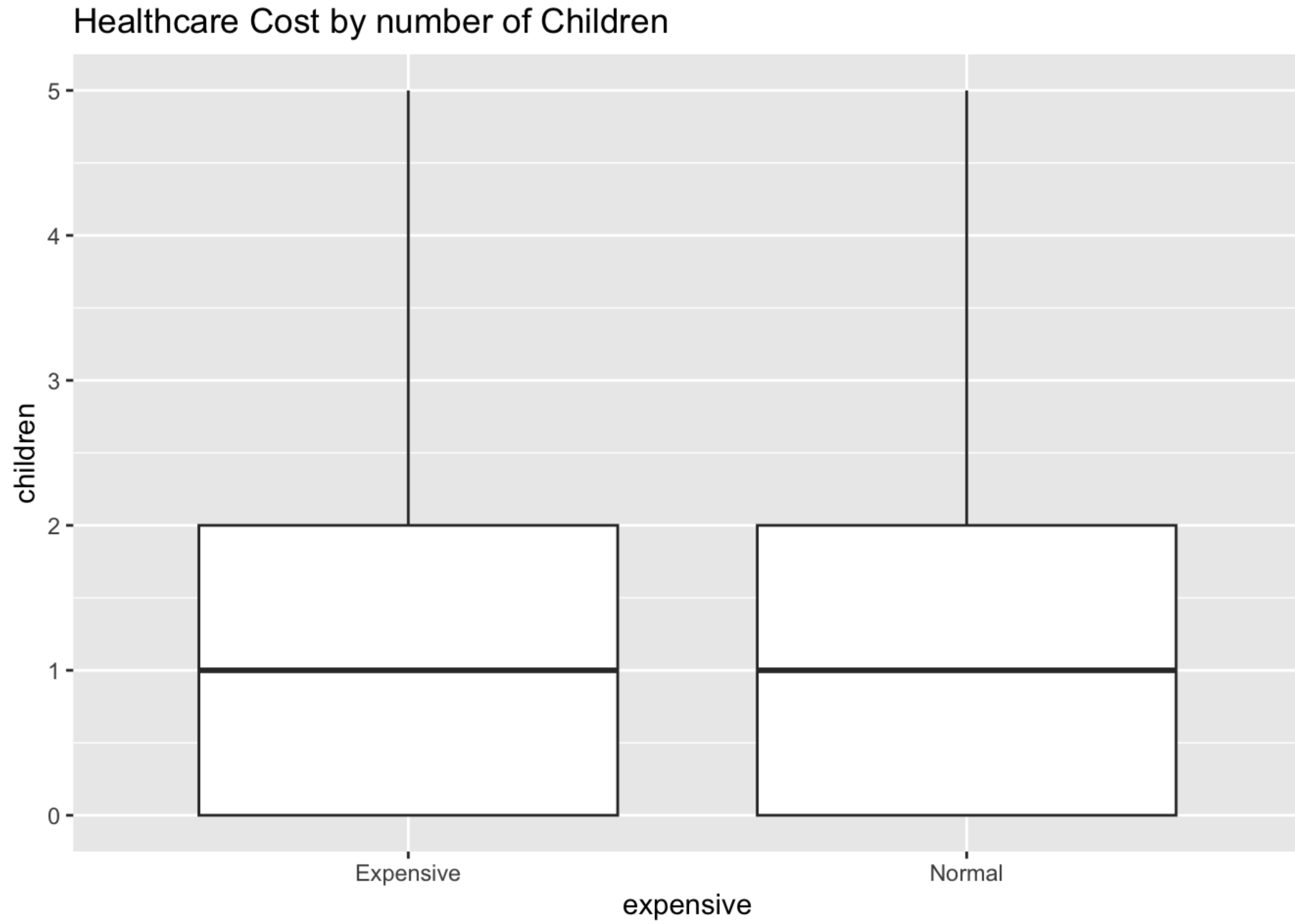
```
ggplot(data, aes(expensive, age)) + geom_boxplot() + ggtitle("Healthcare Cost by Age")
```




```
#For those who cost more in healthcare, they are typically older.
ggplot(data, aes(expensive, bmi)) + geom_boxplot() + ggtitle("Healtchare Cost by BMI")
```



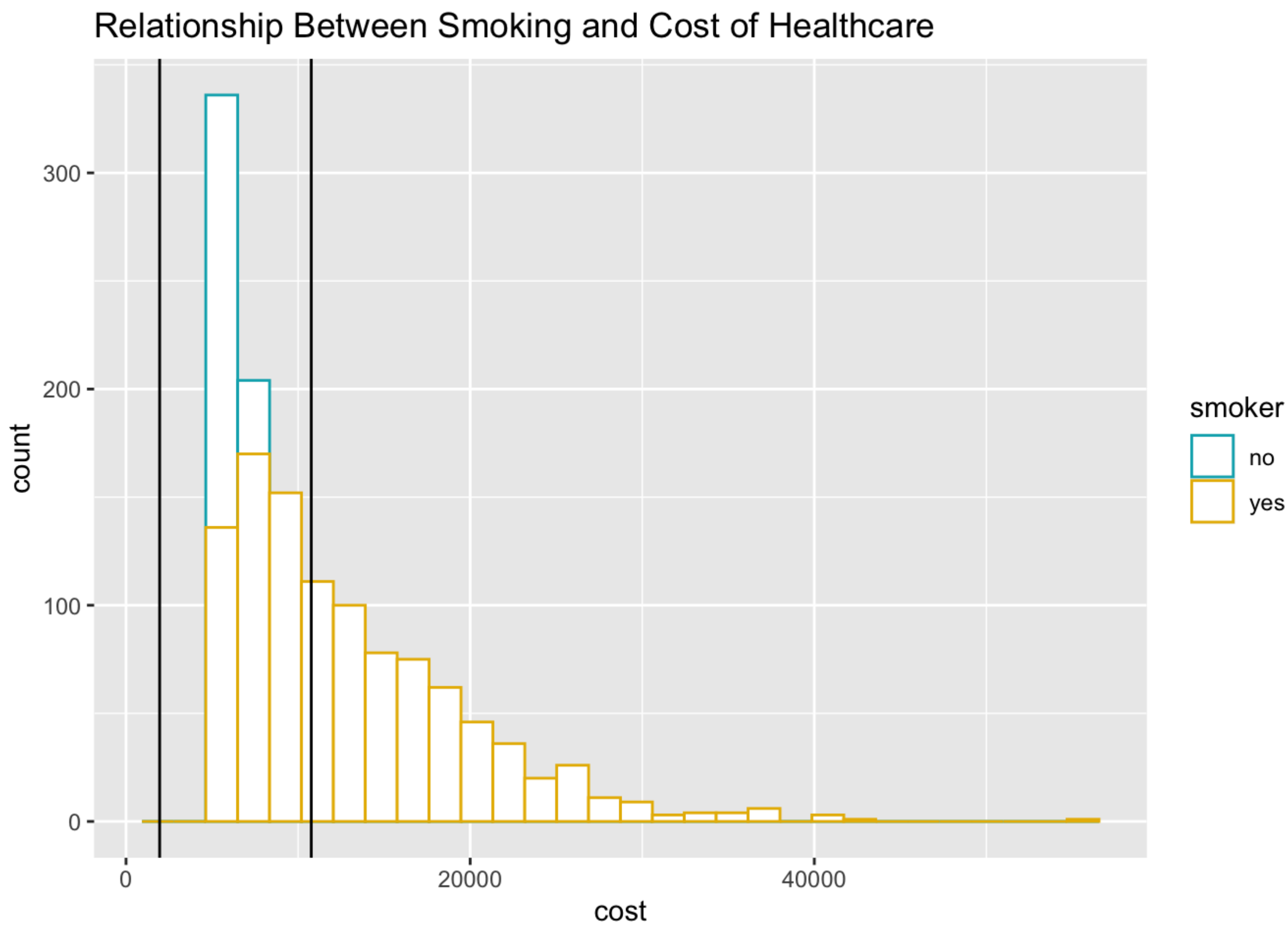

```
#Those who cost more in healthcare typically have a higher BMI.
ggplot(data, aes(expensive, children)) + geom_boxplot() + ggtitle("Healthcare Cost by nu
mber of Children")
```



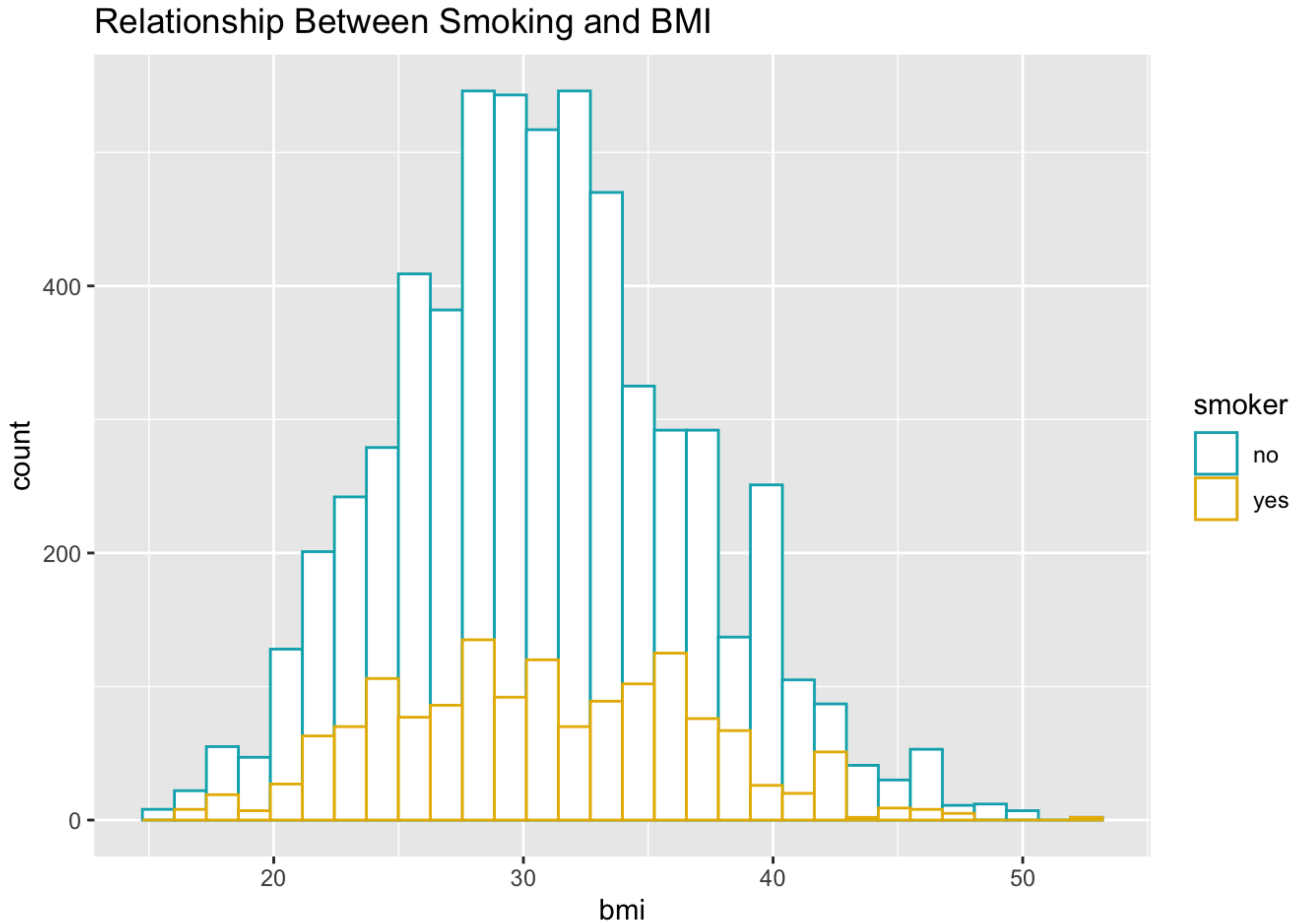

```
#Cost of healthcare doesn't appear to change with number of children.
```

13. Subset Data into Expensive and Normal

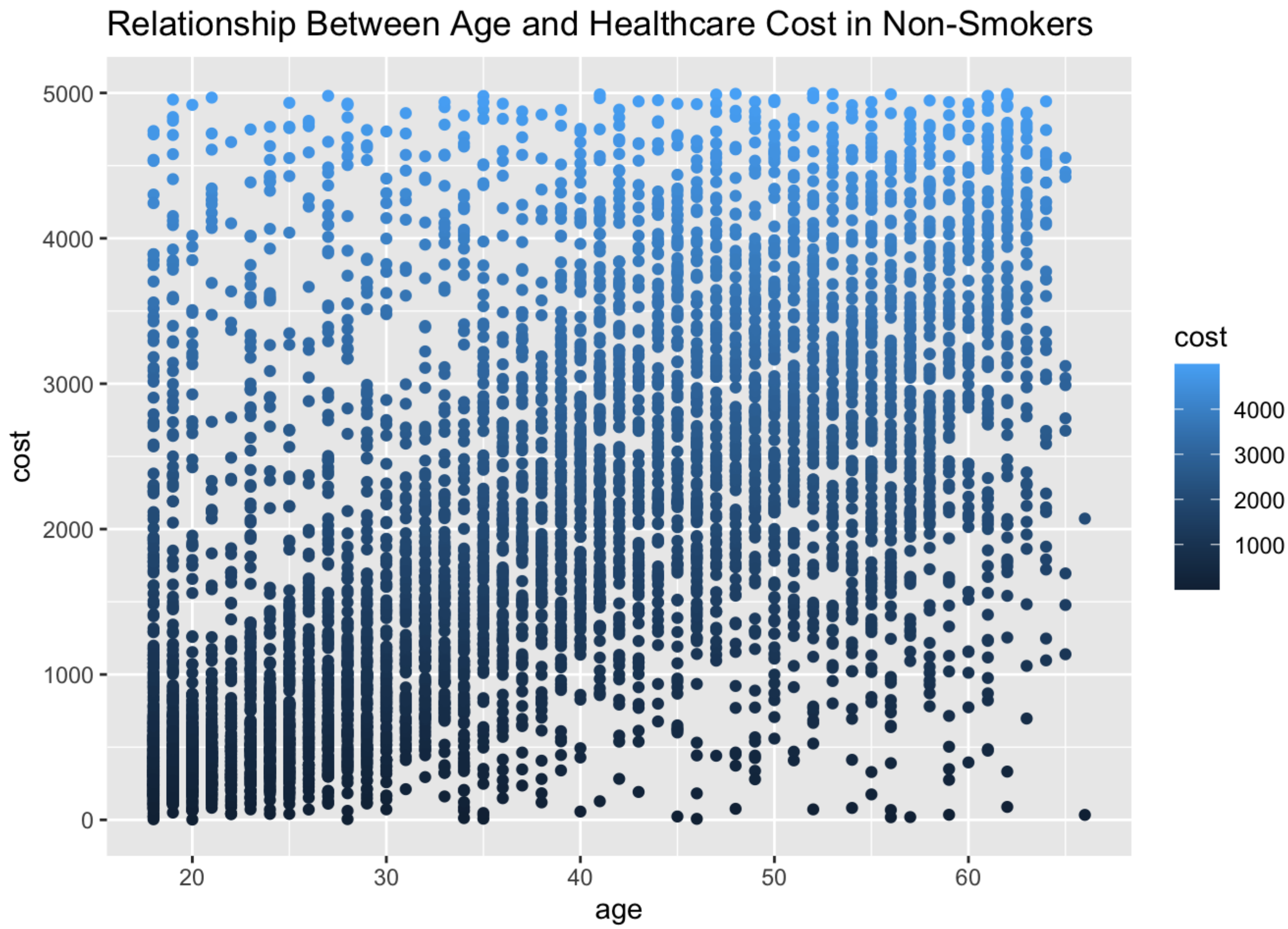
```
expensive = filter(data, expensive == "Expensive")
normal = filter(data, expensive == "Normal")
data$smokeryes <-ifelse(data$smoker == 'yes', 1,0) #if they smoke, assign 1
data$smokerno <- ifelse(data$smoker == 'no',1,0)

#Inference: Smokers shown to have higher cost of healthcare as the data is skewed
#heavily to the right. The non-smokers are concentrated toward the lower end of
#the cost scale, despite having a greater count
ggplot(expensive, aes(x = cost)) +
geom_histogram(aes(color = smoker), fill = "white",position = "identity", bins = 30) +
scale_color_manual(values = c("#00AFBB", "#E7B800")) +geom_vline(xintercept = mean(expen
sive$cost)) +geom_vline(xintercept = mean(normal$cost)) + ggtitle("Relationship Between
Smoking and Cost of Healthcare ")
```



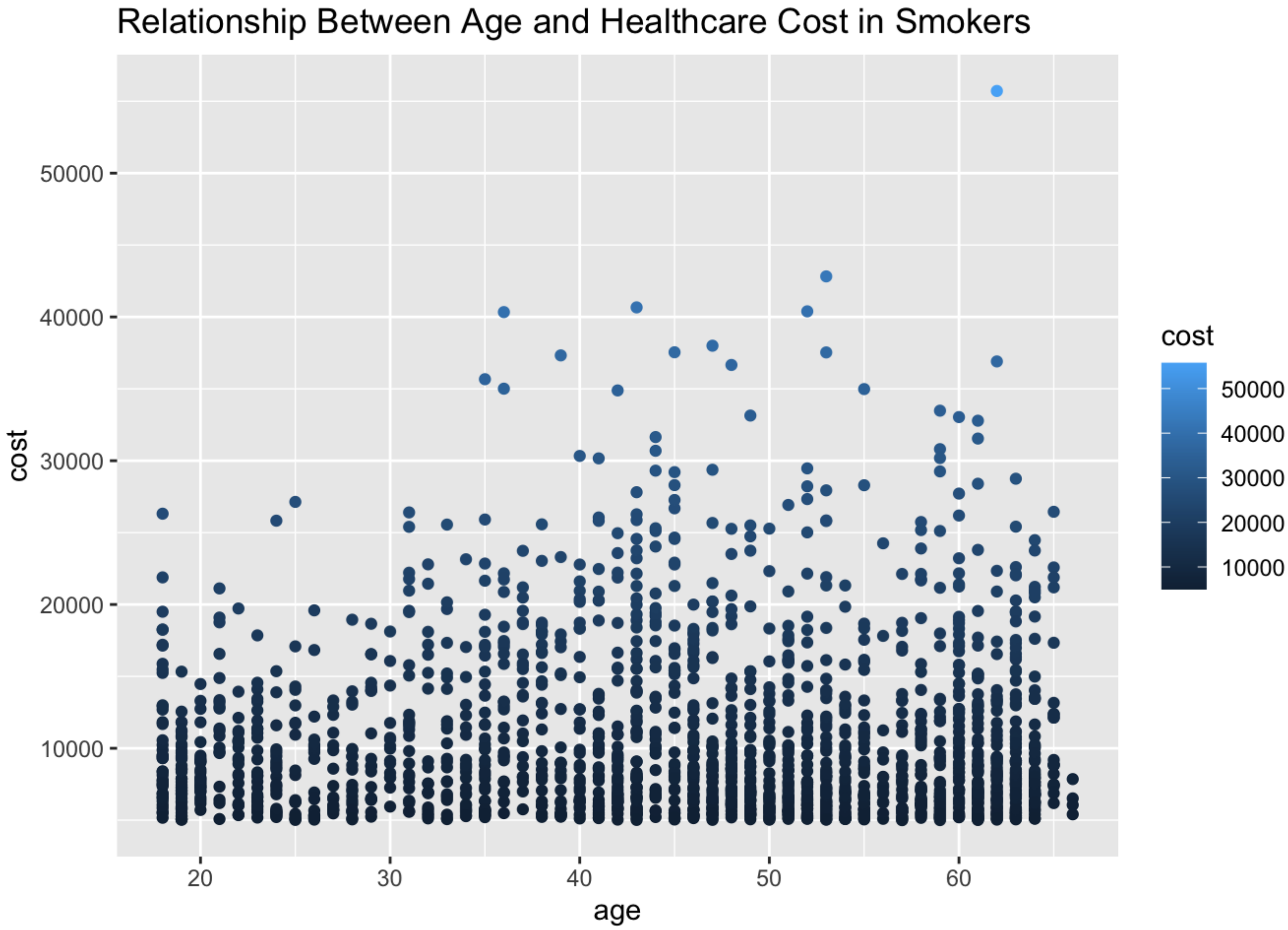

```
#Inference: Appears that being a smoker has no effect on a person's BMI
ggplot(data, aes(x = bmi)) +
geom_histogram(aes(color =smoker), fill = "white",position = "identity", bins = 30) +
scale_color_manual(values = c("#00AFBB", "#E7B800"))+ ggtitle("Relationship Between Smok
ing and BMI")
```




```
#Inference: There is a linear relationship between age and cost in non-smokers.
#This is notable as for people who are non-smokers, age is a main factor that
#will increase the cost of healthcare.
ggplot(normal, aes(x=age, y=cost, color=cost))+geom_point() +
  ggtitle("Relationship Between Age and Healthcare Cost in Non-Smokers ")
```




```
#Inference: The relationship between age and cost in smokers is non-linear, likely no
#relationship at all. Intuitively, this makes sense. Those who smoke already have
#a much higher cost of healthcare, so age shouldn't increase it as its already
#increased
ggplot(expensive, aes(x=age, y=cost, color=cost))+geom_point() +
  ggtitle("Relationship Between Age and Healthcare Cost in Smokers")
```



14. US Map plotting for costs for each state in US

```
library(ggplot2)
library(maps)
```

```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

```
library(ggmap)
```

```
## ℹ Google's Terms of Service: < ]8;;https://mapsplatform.google.com https://mapsplatfor
m.google.com ]8;; >
```

```
## ℹ Please cite ggmap if you use it! Use `citation("ggmap")` for details.
```

```
newDF <- data %>% group_by(location) %>% summarise(avgcost = mean(cost))
#Load the pre-defined dataset 'state' is loaded in us dataframe
us<- map_data("state")
#Change state_name column values to lowercase
us$state_name <- tolower(us$region)
#check the structure of the us dataframe
str(us)
```

```
## 'data.frame':    15537 obs. of  7 variables:
##  $ long      : num  -87.5 -87.5 -87.5 -87.5 -87.6 ...
##  $ lat       : num  30.4 30.4 30.4 30.3 30.3 ...
##  $ group     : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ order     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ region    : chr  "alabama" "alabama" "alabama" "alabama" ...
##  $ subregion : chr  NA NA NA NA ...
##  $ state_name: chr  "alabama" "alabama" "alabama" "alabama" ...
```

```
newDF$state_name <- tolower(newDF$location)
str(newDF)
```

```
## tibble [7 × 3] (S3: tbl_df/tbl/data.frame)
##  $ location  : Factor w/ 7 levels "CONNECTICUT",..: 1 2 3 4 5 6 7
##  $ avgcost   : num [1:7] 3823 3773 4285 3943 4676 ...
##  $ state_name: chr [1:7] "connecticut" "maryland" "massachusetts" "new jersey" ...
```

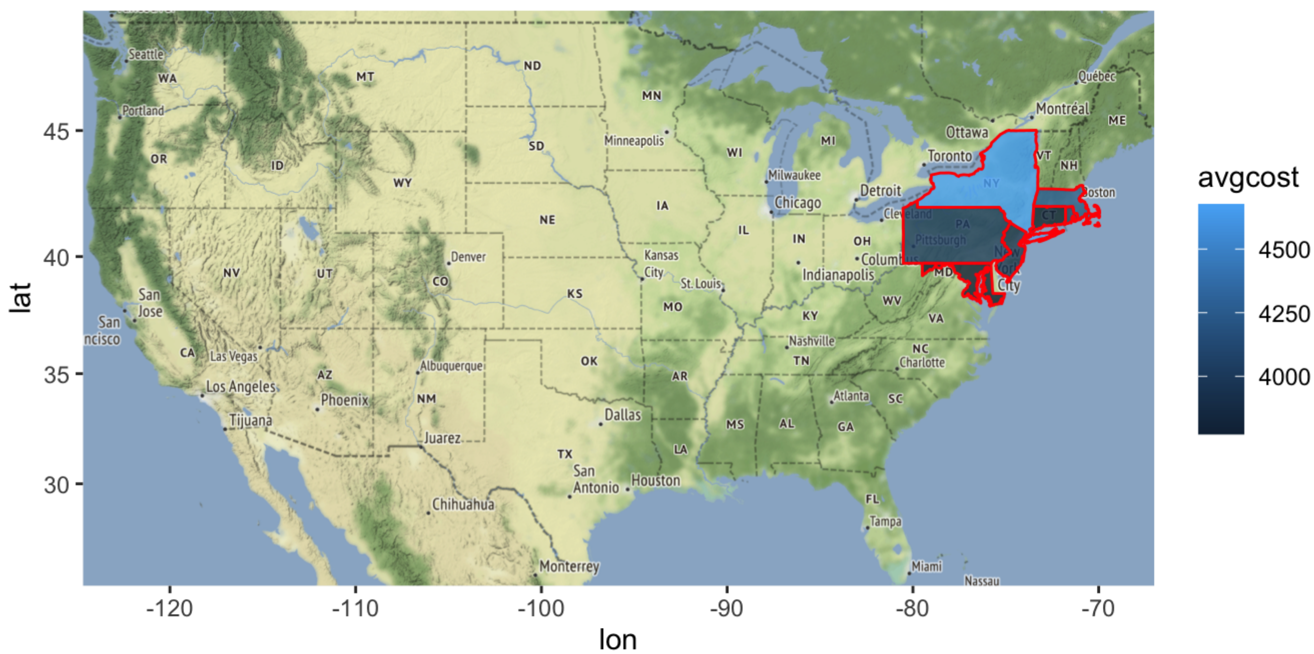
```
#view(newDF)
#merge() us and dfSimple dataframe, the merge is done based on state_name
#column in both dataframes
farewithgeom <- inner_join(us,newDF,by="state_name")
#arrange the order of the popwithgeom dataframe
#structure of the popwithgeom
str(farewithgeom)
```

```
## 'data.frame':    1881 obs. of  9 variables:
##  $ long      : num  -73.5 -73 -73 -72.8 -72.8 ...
##  $ lat       : num  42 42 42 42 42 ...
##  $ group     : num  6 6 6 6 6 6 6 6 6 6 ...
##  $ order     : int  1264 1265 1266 1267 1268 1269 1270 1271 1272 1273 ...
##  $ region    : chr  "connecticut" "connecticut" "connecticut" "connecticut" ...
##  $ subregion : chr  NA NA NA NA ...
##  $ state_name: chr  "connecticut" "connecticut" "connecticut" "connecticut" ...
##  $ location  : Factor w/ 7 levels "CONNECTICUT",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ avgcost   : num  3823 3823 3823 3823 3823 ...
```

```
#Calculate the bounding box to define the us states
bb <- c(left = min(us$long), bottom = min(us$lat),right = max(us$long), top = max(us$la
t))
map <- get_stamenmap(bbox = bb, zoom=5)
```

```
## ℹ Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
```

```
#plot map using ggmap and add the color shading based on the Pop
#column of dfNew dataframe
ggmap(map) + geom_polygon(data=farewithgeom,color="red", alpha=0.8,aes(x=long,y=lat,grou
p=group,fill=avgcost))
```



15.

Zoomed in US Map:

```
#Since data is concentrated in the Northeast, it's helpful to zoom in on the
#map to get a better look at the visualization

library(ggplot2)
library(maps)
library(ggmap)
newDF <- data %>% group_by(location) %>% summarise(avgcost = mean(cost))
#Load the pre-defined dataset 'state' is loaded in us dataframe
us<- map_data("state")
#Change state_name column values to lowercase
us$state_name <- tolower(us$region)
#check the structure of the us dataframe
str(us)
```

```
## 'data.frame':    15537 obs. of  7 variables:
##  $ long      : num  -87.5 -87.5 -87.5 -87.5 -87.6 ...
##  $ lat       : num  30.4 30.4 30.4 30.3 30.3 ...
##  $ group     : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ order     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ region    : chr  "alabama" "alabama" "alabama" "alabama" ...
##  $ subregion : chr  NA NA NA NA ...
##  $ state_name: chr  "alabama" "alabama" "alabama" "alabama" ...
```

```
newDF$state_name <- tolower(newDF$location)
str(newDF)
```

```
## tibble [7 × 3] (S3: tbl_df/tbl/data.frame)
##  $ location  : Factor w/ 7 levels "CONNECTICUT",..: 1 2 3 4 5 6 7
##  $ avgcost   : num [1:7] 3823 3773 4285 3943 4676 ...
##  $ state_name: chr [1:7] "connecticut" "maryland" "massachusetts" "new jersey" ...
```

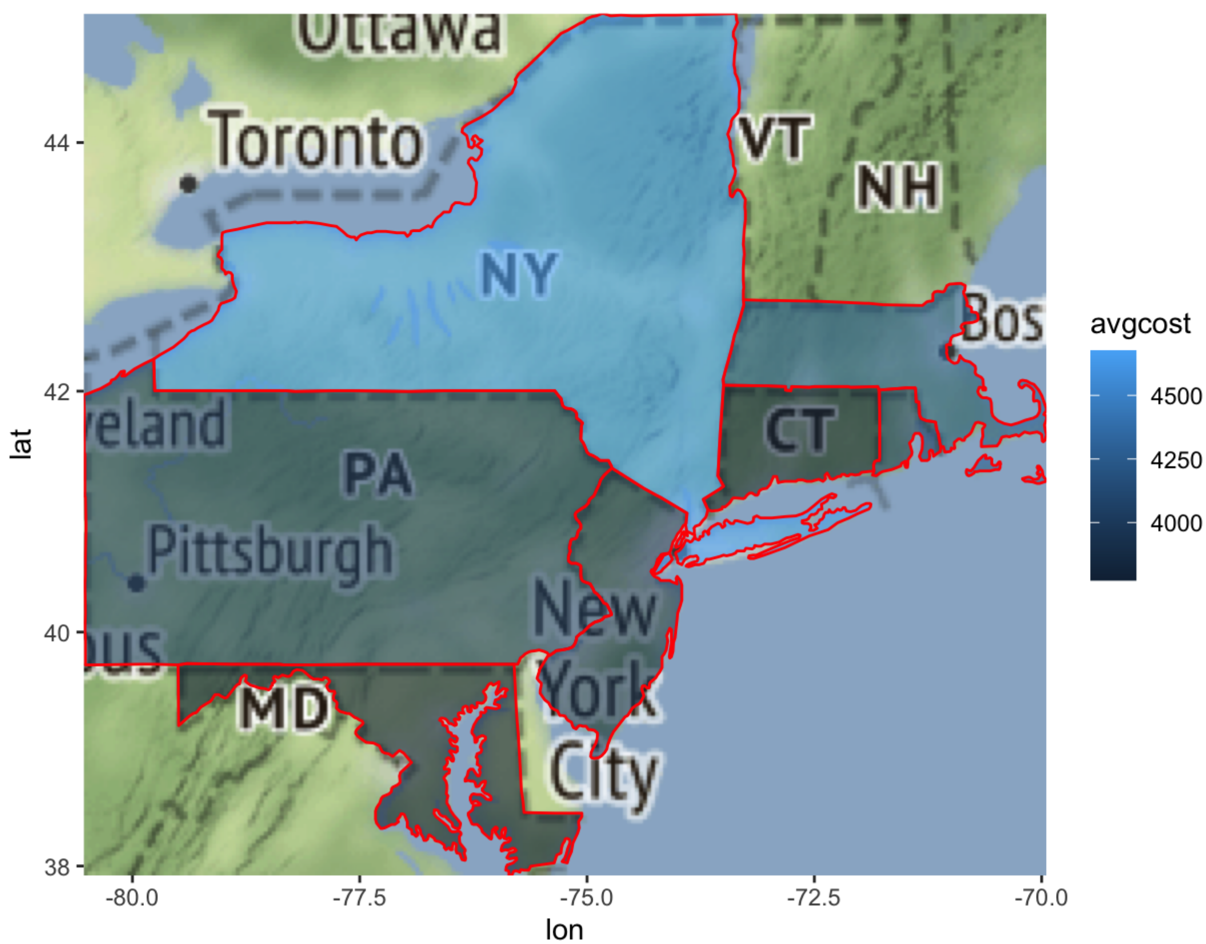
```
#view(newDF)
#merge() us and dfSimple dataframe, the merge is done based on state_name
#column in both dataframes
farewithgeom <- inner_join(us,newDF,by="state_name")
#arrange the order of the popwithgeom dataframe
#structure of the popwithgeom
str(farewithgeom)
```

```
## 'data.frame':    1881 obs. of  9 variables:
##  $ long      : num  -73.5 -73 -73 -72.8 -72.8 ...
##  $ lat       : num  42 42 42 42 42 ...
##  $ group     : num  6 6 6 6 6 6 6 6 6 6 ...
##  $ order     : int  1264 1265 1266 1267 1268 1269 1270 1271 1272 1273 ...
##  $ region    : chr  "connecticut" "connecticut" "connecticut" "connecticut" ...
##  $ subregion : chr  NA NA NA NA ...
##  $ state_name: chr  "connecticut" "connecticut" "connecticut" "connecticut" ...
##  $ location  : Factor w/ 7 levels "CONNECTICUT",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ avgcost   : num  3823 3823 3823 3823 3823 ...
```

```
#Calculate the bounding box to define the us states
bb <- c(left = min(farewithgeom$long), bottom = min(farewithgeom$lat),right = max(farewi
thgeom$long), top = max(farewithgeom$lat))
map <- get_stamenmap(bbox = bb, zoom=5)
```

```
## ℹ Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
```

```
#plot map using ggmap and add the color shading based on the Pop
#column of dfNew dataframe
ggmap(map) + geom_polygon(data=farewithgeom,color="red", alpha=0.6,aes(x=long,y=lat,grou
p=group,fill=avgcost))
```



```
#The map shows that out of the states in the Northeast represented in the data set
#healthcare is most expensive in New York.
```

16. Save and writing files

```
save(svm.model, file = "svm.rda")
#save svm model as rda file for shiny
write_csv(train, "train.csv")
#save train csv for shiny
write_csv(test, "test.csv")
#save test csv for shiny
```