DSCI412FinalProject

Joey Matte

2025-03-09

# Final Course Project

## 1. Data Preperation

### a. load the insurance.csv dataset.

data = read.csv('insurance.csv')  
summary(data)

## age sex bmi children   
## Min. :18.00 Length:1338 Min. :15.96 Min. :0.000   
## 1st Qu.:27.00 Class :character 1st Qu.:26.30 1st Qu.:0.000   
## Median :39.00 Mode :character Median :30.40 Median :1.000   
## Mean :39.21 Mean :30.66 Mean :1.095   
## 3rd Qu.:51.00 3rd Qu.:34.69 3rd Qu.:2.000   
## Max. :64.00 Max. :53.13 Max. :5.000   
## smoker region charges   
## Length:1338 Length:1338 Min. : 1122   
## Class :character Class :character 1st Qu.: 4740   
## Mode :character Mode :character Median : 9382   
## Mean :13270   
## 3rd Qu.:16640   
## Max. :63770

### b. In the data frame, log transform the variable charges and name it as log\_charges.

data$log\_charges = log(data$charges)

### c. Use the sample () function with set.seed equal to 1 to generate row indexes for your training and tests sets, with 70% of the row indexes for your training set and 30% for your test set. Do not use any method other than the sample () function for splitting your data.

set.seed(1)  
index = sample(1:nrow(data),nrow(data)\*.7)  
train = data[index,]  
test = data[-index,]

## 2. Build a multiple regression model

### a. Perform multiple linear regression with log\_charges as the response and the predictors are age, sex, bmi, children, smoker, and region. Print out the results using the summary () function. Use the training dataset you created in #1 above.

mulRegressModel = lm(log\_charges~age+sex+bmi+children+smoker+region,data=train)  
summary(mulRegressModel)

##   
## Call:  
## lm(formula = log\_charges ~ age + sex + bmi + children + smoker +   
## region, data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.03524 -0.17125 -0.03964 0.06275 2.11914   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.052139 0.081384 86.653 < 2e-16 \*\*\*  
## age 0.034948 0.001005 34.785 < 2e-16 \*\*\*  
## sexmale -0.057167 0.027939 -2.046 0.041022 \*   
## bmi 0.011503 0.002394 4.804 1.81e-06 \*\*\*  
## children 0.099796 0.011625 8.585 < 2e-16 \*\*\*  
## smokeryes 1.540724 0.034743 44.347 < 2e-16 \*\*\*  
## regionnorthwest -0.090603 0.039661 -2.284 0.022571 \*   
## regionsoutheast -0.140780 0.039344 -3.578 0.000364 \*\*\*  
## regionsouthwest -0.112012 0.040208 -2.786 0.005449 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4246 on 927 degrees of freedom  
## Multiple R-squared: 0.7841, Adjusted R-squared: 0.7822   
## F-statistic: 420.8 on 8 and 927 DF, p-value: < 2.2e-16

### b. Is there a relationship between the predictors and the response?

There is a relationship between the predictors and the response. All predictors have a p-value less than 0.05, which shows a relationship between them and the response variable.

### c. Does sex have a statistically significant relationship to the response?

Sex has a p-value of 0.041, which is less than 0.05, meaning it has a statistically significant relationship to the response variable.

### d. Compare the test error of the model in #2a. Report the RMSE.

predictions1=predict(mulRegressModel,newdata=test)  
mse1 = mean((predictions1-test[,'log\_charges'])^2)  
rmse1=mse1\*\*.5  
rmse1

## [1] 0.4890933

The model has a RMSE of 0.4891.

## 3. Build a regression tree model

### a. Build a regression tree model using function tree (), where log\_charges is the response and the predictors are age, sex, bmi, children, smoker, and region.

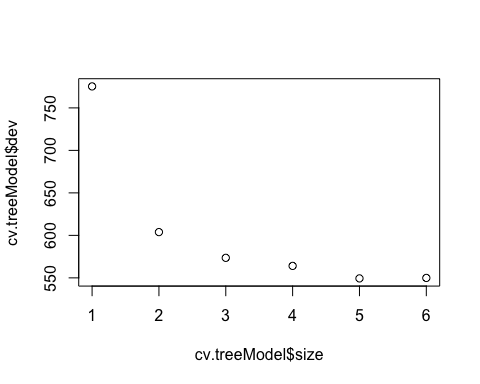
library(tree)  
treeModel = tree(log\_charges~age+sex+bmi+children+smoker+region,data=train)

## Warning in tree(log\_charges ~ age + sex + bmi + children + smoker + region, :  
## NAs introduced by coercion

### b. Find the optimal tree and display the results in a graphic. Report the best size.

cv.treeModel = cv.tree(treeModel)

plot(cv.treeModel$size,cv.treeModel$dev)



The best size for the model is 3.

### c. Justify the number you picked for the optimal tree with regard to the principle of the variance-bias trade-off.

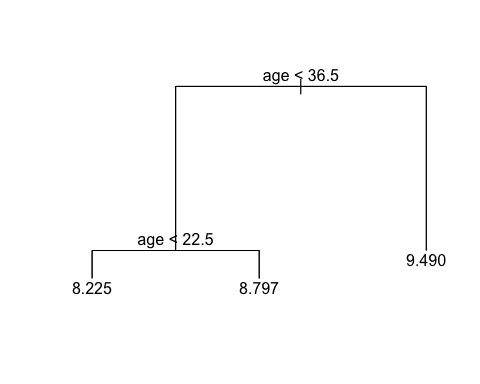
We want to keep the model as simple as possible while maintaining a high level of accuracy. When going from size 3 to 4, the accuracy does not change much, so the best option is size 3.

### d. Prune the tree using the optimal size found in 3.b.

prune.treeModel = prune.tree(treeModel,best=3)

### e. Plot the best tree model and give labels.

plot(prune.treeModel)  
text(prune.treeModel,pretty=0)



### f. Calculate the best RMSE for the best model.

predictions2=predict(prune.treeModel,newdata=test)

## Warning in pred1.tree(object, tree.matrix(newdata)): NAs introduced by coercion

mse2 = mean((predictions2-test[,'log\_charges'])^2)  
rmse2=mse2\*\*.5  
rmse2

## [1] 0.8425435

The RMSE for the model is 0.8425.

## 4. Build a random forest model

### a. Build a random forest model using function randomForest(), where log\_charges is the response and the predictors are age, sex, bmi, children, smoker, and region.

library(randomForest)

## randomForest 4.7-1.2

## Type rfNews() to see new features/changes/bug fixes.

rfModel = randomForest(log\_charges~age+sex+bmi+children+smoker+region,data=train)

### b. Compute the test error (using the test data set).

predictions3=predict(rfModel,newdata=test)  
mse3 = mean((predictions3-test[,'log\_charges'])^2)  
rmse3=mse3\*\*.5  
rmse3

## [1] 0.4309233

The RMSE of the model is 0.4309.

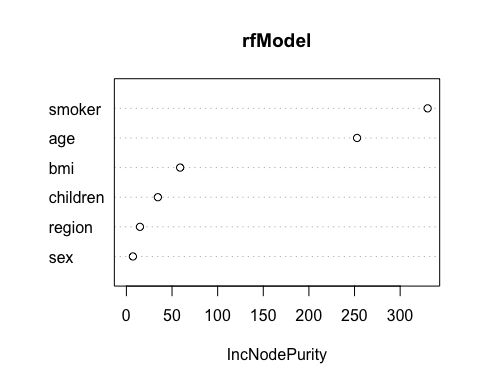
### c. Extract variable importance measure using the importance() function.

importance(rfModel)

## IncNodePurity  
## age 252.769292  
## sex 7.139561  
## bmi 58.900407  
## children 34.496996  
## smoker 330.114551  
## region 14.843826

### d. Plot the variable importance using the function, varImpPlot(). Which are the top 3 important predictors in this model?

varImpPlot(rfModel)



Based on the graph, the top 3 most important predictors in this model are smoker, age, and bmi.

## 5. Perform the k-means cluster analysis

### a. Remove the sex, smoker, and region, since they are not numerical values.

# also dropping charges here because I kept the column when making a new column for log\_charges.  
df = subset(data, select = -c(sex,smoker,region,charges))  
head(df)

## age bmi children log\_charges  
## 1 19 27.900 0 9.734176  
## 2 18 33.770 1 7.453302  
## 3 28 33.000 3 8.400538  
## 4 33 22.705 0 9.998092  
## 5 32 28.880 0 8.260197  
## 6 31 25.740 0 8.231275

### b. Determine the optimal number of clusters. Justify your answer.

library(factoextra)

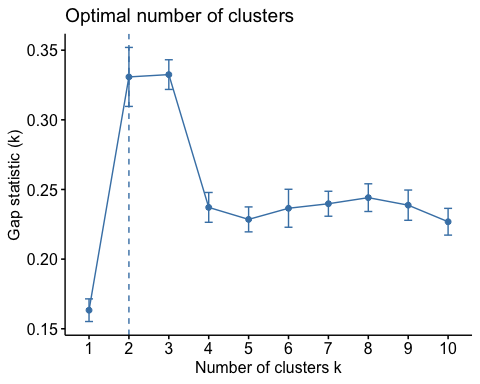
## Loading required package: ggplot2

##   
## Attaching package: 'ggplot2'

## The following object is masked from 'package:randomForest':  
##   
## margin

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

library(cluster)  
library(ggplot2)  
  
fviz\_nbclust(df,kmeans,method='gap\_stat')



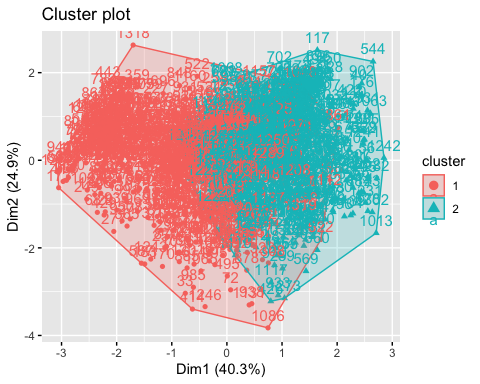
From the plot, we can see that the optimal number of clusters is 2.

### c. Perform k-means clustering using the optimal number of clusters from 5b.

df\_kmeans = kmeans(df,2,nstart=25)

### d. Visualize the clusters in different colors.

fviz\_cluster(df\_kmeans,data=df)



## 6. Putting it all together

### a. For predicting insurance charges, your supervisor asks you to choose the best model among the multiple regression, regression tree, and random forest. Compare their test RMSEs of the models generated above. Display the names for these types of these models, using these labels: Multiple Linear Regression, Regression Tree, and Random Forest and their corresponding test RMSEs in a data.frame. Label the column in your data frame with the labels as Model.Type, and label the column with the test RMSEs as Test.MSE and round the data in this column to 4 decimal places. Present the formatted data to your supervisor and recommend which model is best and why.

Model.Type=c('Multiple Linear Regression','Regression Tree','Random Forest')  
Test.MSE=round(c(rmse1,rmse2,rmse3),4)  
mseFrame = data.frame(Model.Type,Test.MSE)  
mseFrame

## Model.Type Test.MSE  
## 1 Multiple Linear Regression 0.4891  
## 2 Regression Tree 0.8425  
## 3 Random Forest 0.4309

I would recommended the Random Forest model because it has the lowest RMSE. This means it’s predictions are the most accurate.

### b. Another supervisor from the sales department has requested your help to create a predictive model that his sales representatives can use to explain to clients what the potential costs could be for different kinds of customers, and they need an easy and visual way of explaining it. What model would you recommend, and what are the benefits and disadvantages of your recommended model compared to other models?

I would recommend the Regression Tree Model. The benefits of this model are that it is very easy to explain visually, and can be interpreted by non experts. The disadvantage is it is not as accurate as the other two models.

### c. The supervisor from the sales department likes your regression tree model. But she says that the salespeople say the numbers in it are way too low and suggests that maybe the numbers on the leaf nodes predicting charges are log transformations of the actual charges. You realize that in step 1.b of this project that you had indeed transformed charges using the log function. And now you realize that you need to reverse the transformation in your final output. The solution you have is to reverse the log transformation of the variables in the regression tree model you created and re display the result.

copyPruneTreeModel = prune.tree(treeModel,best=3)  
copyPruneTreeModel$frame$yval = exp(copyPruneTreeModel$frame$yval)  
plot(copyPruneTreeModel)  
text(copyPruneTreeModel,pretty=0)

