Homework 4

Question 9.1 Using the same crime data set uscrime.txt as in Question 8.2, apply Principal Component Analysis and then create a regression model using the first few principal components. Specify your new model in terms of the original variables (not the principal components), and compare its quality to that of your solution to Question 8.2. You can use the R function prcomp for PCA. (Note that to first scale the data, you can include scale. = TRUE to scale as part of the PCA function. Don't forget that, to make a prediction for the new city, you'll need to unscale the coefficients (i.e., do the scaling calculation in reverse)!)

Answer: First, we'll load the dataset "5.1uscrimeSummer2018.txt". Running PCA on the dataset may yield over-fitting since the number of column vectors is too high. After we load our data set, we'll exclude the column vector we are trying to predict which is the last column (Crime) . Hence it will look like this:

```
data2<- read.table("5.1uscrimeSummer2018.txt",stringsAsFactors = FALSE,header = TRUE)
b2<- prcomp(data2[,1:15],scale. = TRUE)
```

```
> summary(b2)
Importance of components:
                          PC1    PC2    PC3     PC4     PC5     PC6     PC7     PC8     PC9
Standard deviation     2.4534 1.6739 1.4160 1.07806 0.97893 0.74377 0.56729 0.55444 0.48493
Proportion of Variance 0.4013 0.1868 0.1337 0.07748 0.06389 0.03688 0.02145 0.02049 0.01568
Cumulative Proportion  0.4013 0.5880 0.7217 0.79920 0.86308 0.89996 0.92142 0.94191 0.95759
                          PC10    PC11    PC12    PC13   PC14    PC15
Standard deviation     0.44708 0.41915 0.35804 0.26333 0.2418 0.06793
Proportion of Variance 0.01333 0.01171 0.00855 0.00462 0.0039 0.00031
Cumulative Proportion  0.97091 0.98263 0.99117 0.99579 0.9997 1.00000
```

This will return column vectors of each being a PCA, PC1 to PC15 ordering from most significant to least significant predictors. For my model, I will only consider the first 5 PCA or the first 5 columns. Since the first 5th accounts for 88+ percent of the variability .

```
combinedata<-cbind(b2$x[,1:5],data2[,16])
linearmodel1<-lm(combinedata[,6]~.,data=as.data.frame(combinedata[,1:5]))
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   905.09      35.59  25.428  < 2e-16 ***
PC1            65.22      14.67   4.447 6.51e-05 ***
PC2           -70.08      21.49  -3.261  0.00224 **
PC3            25.19      25.41   0.992  0.32725
PC4            69.45      33.37   2.081  0.04374 *
PC5          -229.04      36.75  -6.232 2.02e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 244 on 41 degrees of freedom
Multiple R-squared:  0.6452,    Adjusted R-squared:  0.6019
F-statistic: 14.91 on 5 and 41 DF,  p-value: 2.446e-08
```

As one can see, PC3 has a high p-value, which means it contributes little to predicting Crime. The difficult part for me was how to obtain, or how to scale back to the original dataset. Because we have to do this to compare this model to the previous. Luckily the videos help a lot. I got the R^2 value and R^2 ad-

justed value for the new model.

```
SSE = sum((estimates - data2[,16])^2)
SStot = sum((data2[,16] - mean(data2[,16]))^2)
R2= 1-SSE/SStot
R2adjusted = 1-(1-R2)*(nrow(data2)-1)/nrow((data2)-5-1)
```

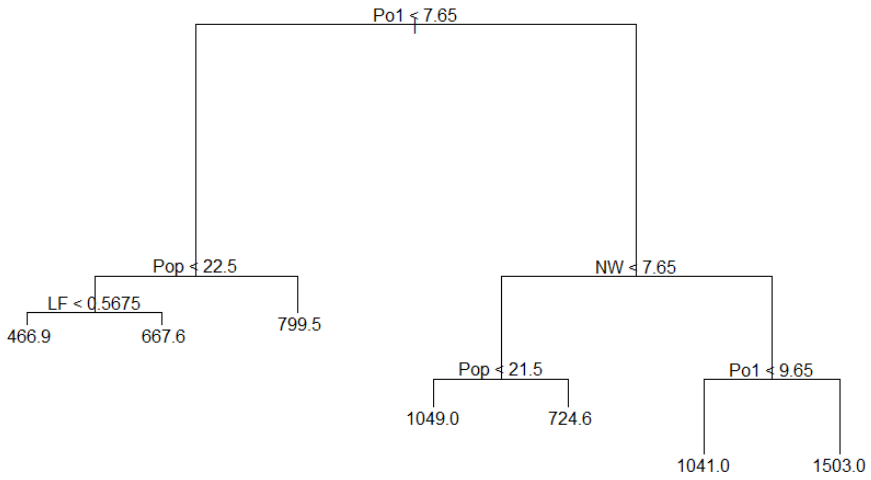
Which turned out to be 0.6451941 and 0.6527431, and since they are close to each other. Hence this model good at predicting Crime.

Question 10.1 Using the same crime data set uscrime.txt as in Questions 8.2 and 9.1, find the best model you can using (a) a regression tree model, and (b) a random forest model. In R, you can use the tree package or the rpart package, and the randomForest package. For each model, describe one or two qualitative takeaways you get from analyzing the results (i.e., don't just stop when you have a good model, but interpret it too).

Answer: For this question, we have the same dataset, we are building a regression tree model and randomforest. insert images. As you can see above, only a 4 predictors are in the decision tree. Po1,Pop,LF and NW. And when trying to find the R^2 value, we get 0.7244962, which is better higher than our previous model of 0.6451941, meaning that this is a better model. And for random forest, Number of variable tried at each split equaling to 3 yield the best result amongst all as seen below. As the highest R^2 I got from randomforest model was 0.446523, after many iteration.

```
data2<- read.table("5.1uscrimeSummer2018.txt",stringsAsFactors = FALSE,header = TRUE)
treemodel<-tree(Crime~.,data=data2)
summary(treemodel)
```

```
Regression tree:
tree(formula = Crime ~ ., data = data2)
Variables actually used in tree construction:
[1] "Po1" "Pop" "LF"  "NW"
Number of terminal nodes:  7
Residual mean deviance:  47390 = 1896000 / 40
Distribution of residuals:
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-573.900  -98.300   -1.545    0.000  110.600  490.100
```




```
randomforestmodel <- randomForest(Crime~.,data=data2,mtry=3,importance=TRUE)
Call:
 randomForest(formula = Crime ~ ., data = data2, mtry = 3, importance = TRUE
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 3

          Mean of squared residuals: 80749.29
                    % Var explained: 44.84
> |
```

Question 10.2

Describe a situation or problem from your job, everyday life, current events, etc., for which a logistic regression model would be appropriate. List some (up to 5) predictors that you might use.

Answer: Let assume we are interested in knowing if a person is interested in trying electronic cigarette or not. Here are some possible predictors that we may be interested in. Person's Sex Person's Age Educational background Current or past status on whether they smoke tobacco cigarette The amount of sex partner that person has had.

Question 10.3

1. Using the GermanCredit data set germancredit.txt from use logistic regression to find a good predictive model for whether credit applicants are good credit risks or not. Show your model (factors used and their coefficients), the software output, and the quality of fit. You can use the glm function in R. To get a logistic regression (logit) model on data where the response is either zero or one, use family=binomial(link="logit") in your glm function call.

2. Because the model gives a result between 0 and 1, it requires setting a threshold probability to separate between "good" and "bad" answers. In this data set, they estimate that incorrectly identifying a bad customer as good, is

5 times worse than incorrectly classifying a good customer as bad. Determine a good threshold probability based on your model.

Answer:

Like usual, we'll load the data, and change the V21 to '0' if equals to 1 and '1' if it's equal to 2. This is to make it a binary response, since logistic regression deals dependent variable being binary.

```
rm(list=ls())
data3<- read.table("10.3germancreditSummer2018.txt",sep = " ")
data3$V21[data3$V21==1]<-0
data3$V21[data3$V21==2]<-1
indexes <-sample(1:nrow(data3),size=0.2*nrow(data3))
test= data3[indexes,]
dim(test)
train=data3[-indexes,]
dim(train)

logitmodel2 = glm(V21~ V1+V3+V5+V20,data=train, family=binomial(link = "logit"))
summary(logitmodel2)

pred<-predict(logitmodel2,test)


fitted.results <- predict(logitmodel2 ,test,type='response')
fitted.results <- ifelse(fitted.results > 0.4,1,0)
misClasificError <- mean(fitted.results != test$V21)
print(paste('Accuracy',1-misClasificError))
```

```
pred<-predict(logitmodel2,test)


fitted.results <- predict(logitmodel2 ,test,type='response')
fitted.results <- ifelse(fitted.results > 0.4,1,0)
misClasificError <- mean(fitted.results != test$V21)
print(paste('Accuracy',1-misClasificError))
] "Accuracy 0.745"
```

And I found 0.4 with a loop of testing each boundary line, and in result, 0.4 being the boundary line for this logistic regression model seemed to work the best, and 0.745 accurately guessing V21 in our test dataset isn't too bad.