ISYE-6501
Fall 2024

Homework 6

Question 9.1

Using the same 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 principal components), and compare its quality to that of you solution to Question 8.2. You can use the R function prcomp for PCA.

ANALYSIS:

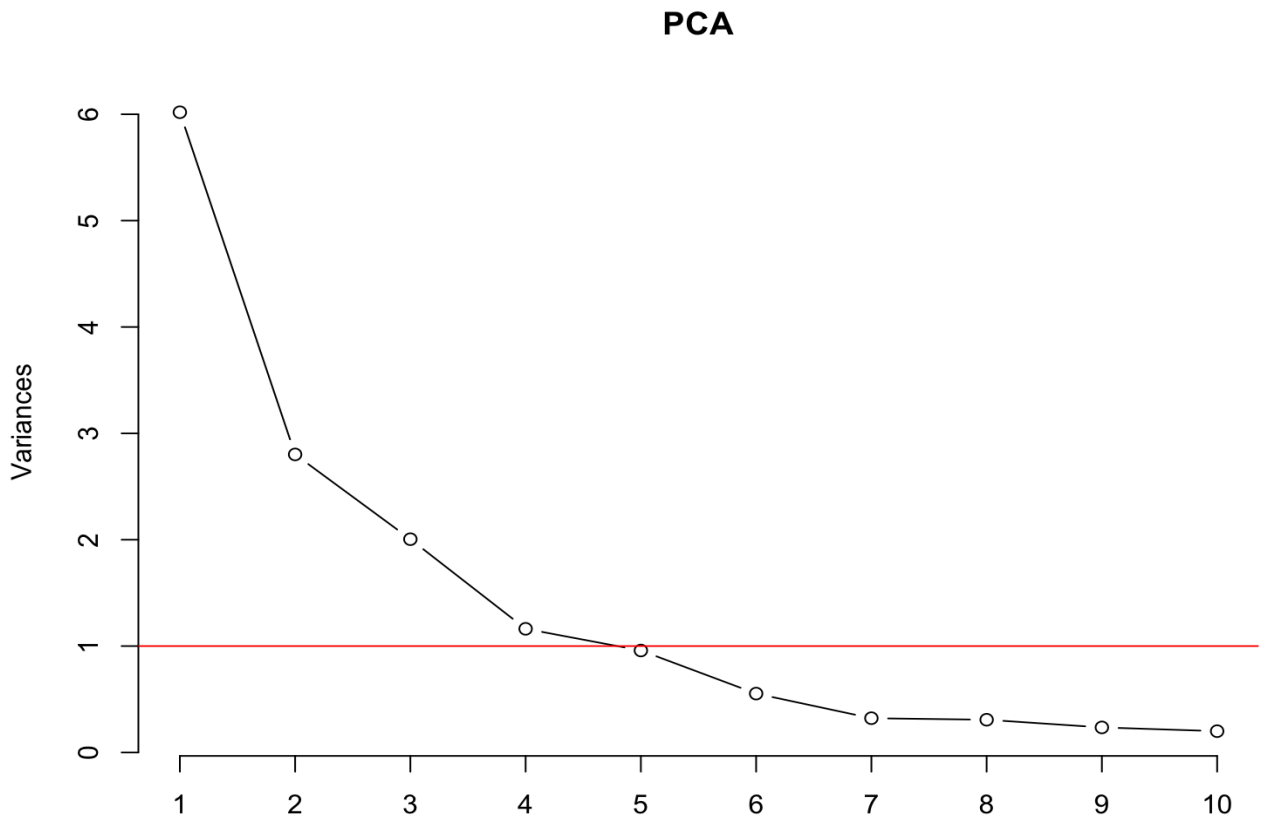
In question 8.2 it was found that the crime prediction was 1304 with an R-squared of 0.638 and an adjusted R-squared of 0.584.

I ran the prcomp function and plotted the findings in a graph with a red line at the 1 SD mark. I decided to use the first 5 principal components because they are almost all at or under 1 standard deviation. (I know PC 5 is at a standard deviation of ~.98 but I decided to add it anyway). After running the regression model on the new PCA data, I was able to predict a crime rate of about 1388 compared to 1304 from question 8.2. It calculated a new R-squared of about 0.645 and a new adjusted R-squared of about 0.602. The new prediction does not seem out of the norm and the new model seems to fit slightly better with an increase in the adjusted R-squared of about 0.018.

Below is the code/ R markdown, find the input in black, the comments in green, and the output in blue.

CODE:

```
> #load the data uscrime data into a table named data.
> data <- read.table("uscrime.txt", header = TRUE)
> #below I load the test/prediction parameters into a data frame
> crime_test <- data.frame(M = 14.0, So = 0, Ed = 10.0, Po1 = 12.0, Po2 = 15.5, LF = 0.640, M.F = 94.0,
Pop = 150, NW = 1.1, U1 = 0.120, U2 = 3.6, Wealth = 3200, Ineq = 20.1, Prob = 0.04, Time = 39.0)
> #Run Principal Component Analysis on matrix of scaled predictors
> PCA=prcomp(data[,1:15], scale. = TRUE, center = TRUE)
> #Plot the PCA values on a graph to find the best number of Principal Components to use
> screeplot(PCA, type= "line")
> abline(h=1, col = "red")
```




```
> summary(PCA)
Importance of components:
                          PC1    PC2    PC3     PC4     PC5     PC6     PC7     PC8     PC9    PC10    PC11    PC12    PC13
PC14    PC15
Standard deviation     2.4534 1.6739 1.4160 1.07806 0.97893 0.74377 0.56729 0.55444 0.48493 0.44708
0.41915 0.35804 0.26333 0.2418 0.06793
Proportion of Variance 0.4013 0.1868 0.1337 0.07748 0.06389 0.03688 0.02145 0.02049 0.01568
0.01333 0.01171 0.00855 0.00462 0.0039 0.00031
Cumulative Proportion  0.4013 0.5880 0.7217 0.79920 0.86308 0.89996 0.92142 0.94191 0.95759
0.97091 0.98263 0.99117 0.99579 0.9997 1.00000
> #I now add the 5 first principal components to the original dataset because as we can see principal
components 1-5 are near above the 1 SD line in red.
> PCAdata = cbind(PCA$x[,1:5], data[,16])
> #I will then create a linear regression model with the new dataset
> newmodel <- lm(V6~., data= as.data.frame(PCAdata))
> summary(newmodel)

Call:
lm(formula = V6 ~ ., data = as.data.frame(PCAdata))

Residuals:
```

```
   Min     1Q  Median     3Q    Max
-420.79 -185.01   12.21  146.24  447.86

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

> #Now we can predict the crime in the new city
> prediction <- data.frame(predict(PCA, crime_test))
> predictionmodel <- predict(newmodel, prediction)
> predictionmodel
       1
1388.926
```