ISyE 6501-HOMEWORK 6

Group Members:
Jinyu, Huang | jhuang472@gatech.edu | GTID: 903522245
Chengqi, Huang | chengqihuang@gatech.edu | GTID: 903534690
Yuefan, Hu | yuefanhu@gatech.edu | GTID:903543027
Jingyu, Li | alanli@gatech.edu | GTID: 903520148

Qusetion 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

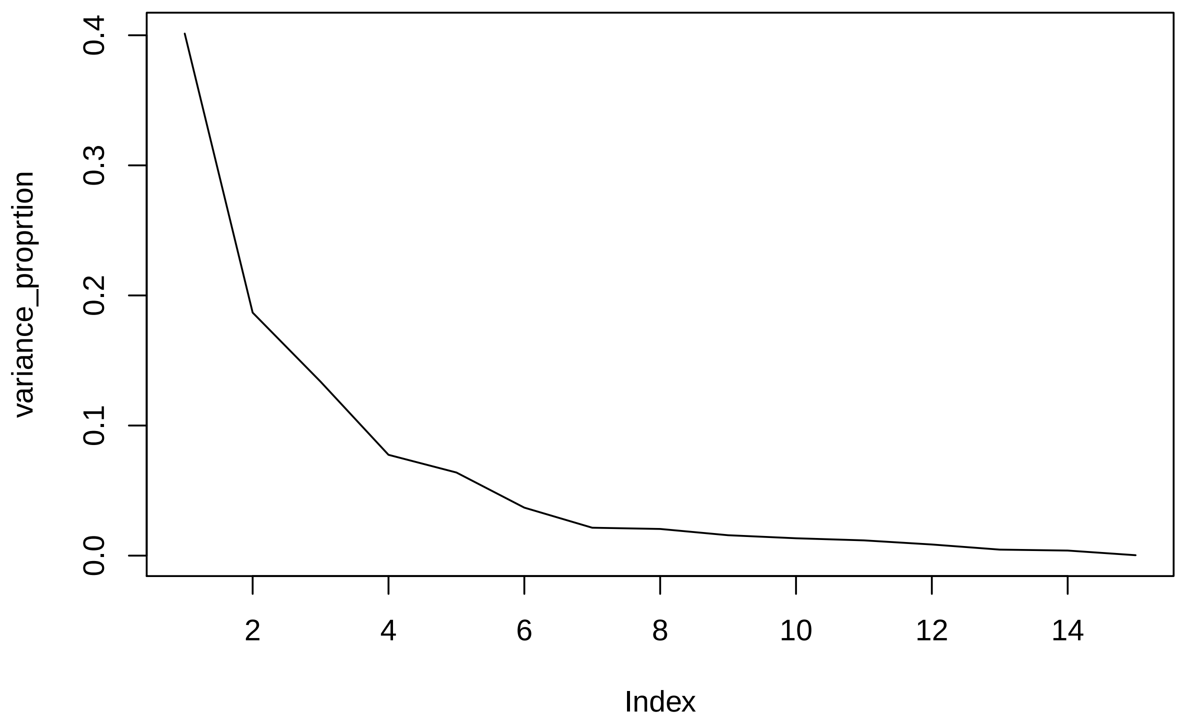
After we import the data, we do PCA to the original data. In fact, there is a binary variable in our dataset(So). Whether or not a binary feature can be include in a PCA model is widely debated. For this homework, we choose put it in. Since there are 15 predictors, we will have 15 principal components.

```
> data = read.table("uscrime.txt", header=TRUE)  # import data
> crime_features = data[,-16]
> pca_model = prcomp(crime_features, scale.=TRUE)
> summary(pca_model)
```

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

According to the decreasing line of variance proportion from each principal components, we find the breaking point, which is 7. We include the first seven PCs in our regression model.

```
> variance_proprtion = c(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)
> plot(variance_proprtion, type='l')
```



We extract the value of PC1 to PC7 for each data point and run the regressiong model. The results show that the coefficients of PC1, PC2, PC4 and PC5 are significant and PC7 is marginally significant. The *adj.* R^2 equals to 0.63, which is less than the model we get in Homework5(Crime ~ M + Ed + Po1 + M.F + U1 + U2 + Ineq + Prob, with *adj.* $R^2 = 0.74$). It's in our expectation, because every principal component is a linear combination of all the original features. That means: 1)Beside features that contribute more to the variance of response variable, other features with lower explanatory power are also include in the model. 2) Some information from the high explanatory power features are missing because we omit some of the principal components(PC8 to PC15).

```
> pc_seven = pca_model$x[,1:7] # get the PC1 to PC7 values for each data point
> crime_pc = data.frame((cbind(pc_seven, data[,16])))
> colnames(crime_pc)[8] = 'Crime'
> model_1 = lm(Crime~., data=crime_pc)
> summary(model_1)
```

```
Call:
lm(formula = Crime ~ ., data = crime_pc)

Residuals:
    Min      1Q  Median      3Q     Max 
-475.41 -141.65   34.73  137.25  412.32 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   905.09      34.21  26.454  < 2e-16 ***
```

```
PC1            65.22      14.10   4.626 4.04e-05 ***
PC2           -70.08      20.66  -3.392   0.0016 **
PC3            25.19      24.42   1.032   0.3086
PC4            69.45      32.08   2.165   0.0366 *
PC5          -229.04      35.33  -6.483 1.11e-07 ***
PC6           -60.21      46.50  -1.295   0.2029
PC7           117.26      60.96   1.923   0.0617 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 234.6 on 39 degrees of freedom
Multiple R-squared:  0.6882,    Adjusted R-squared:  0.6322
F-statistic:  12.3 on 7 and 39 DF,  p-value: 3.513e-08
```

To calculate the coefficients in terms of the original features, we need to use the coefficient of the regression with principal components and the loading matrix from PCA model. The coefficient for the scaled original features equals to $L.C$, in which L represent the loading matrix and C is the coefficient matrix from Y~PCs(exclude intercept). The regression function for scaled original features we get is:
$Crime = 69.42M + 66.94So - 7.61Ed + 132.51Po1 + 129.81Po2 + 27.21LF + 130.84M.F + 36.54Pop + 58.46NW - 18.53U1 + 20.62U2 + 27.82Wealth + 49.68Ineq - 117.56Prob - 15.70Time + 905.09$

```
> loading = pca_model$rotation[,1:7] # loading matrix from PCA model
> coe_pc = model_1$coefficients[-1] # coefficients of regression Y~PCs
> coe_origin = loading%*%coe_pc # coefficients of regression Y~scaled original features
> coe_origin
```

```
             [,1]
M       69.420279
So      66.940187
Ed      -7.611451
Po1    132.506149
Po2    129.808521
LF      27.212545
M.F    130.843740
Pop     36.544822
NW      58.457563
U1     -18.528807
U2      20.620319
Wealth  27.823790
Ineq    49.675121
Prob  -117.563087
Time   -15.698148
```

```
> intercept_origin = model_1$coefficients[1]
> intercept_origin
```

```
(Intercept)
   905.0851
```

Then in order to do prediction to a given city, we need to transform the regression coefficients to match unscaled predictors:
$coefficient\ in\ unscaled\ data = coefficient\ in\ unscaled\ data / sd$

$intercept\ in\ unscaled\ data = intercept\ in\ scaled\ data - \sum(coefficient\ in\ unscaled\ data * \mu_i)$

The regression function with unscaled predictors is:
$Crime = 55.24M + 139.76So - 6.80Ed + 44.59Po1 + 46.42Po2 + 673.38LF + 44.40M.F + 0.96Pop + 5.68NW -$

$1027.74U1 + 24.42U2 + 0.03Wealth + 12.45Ineq - 5170.57Prob - 2.22Time - 5498.458$

```
> sd = pca_model$scale
> coe_unscale = coe_origin / sd
> coe_unscale
```

```
                [,1]
M       5.523735e+01
So      1.397571e+02
Ed     -6.803836e+00
Po1     4.458638e+01
Po2     4.642432e+01
LF      6.733809e+02
M.F     4.440293e+01
Pop     9.599076e-01
NW      5.684940e+00
U1     -1.027735e+03
U2      2.441589e+01
Wealth  2.883565e-02
Ineq    1.245113e+01
Prob   -5.170569e+03
Time   -2.215095e+00
```

```
> intercept_unscale = intercept_origin - sum(pca_model$center * coe_unscale)
> intercept_unscale
```

```
(Intercept)
  -5498.458
```

Based on the model, the prediction is 1230.

```
> new_data_point = c(14, 0, 10, 12, 15.5,
+                    0.64, 94, 150, 1.1, 0.12,
+                    3.6, 3200, 20.1, 0.04, 39)
> new_data_point%*%coe_unscale+intercept_unscale
```

```
         [,1]
[1,] 1230.418
```