library("GGally")

## Loading required package: ggplot2

## Registered S3 methods overwritten by 'ggplot2':  
## method from   
## [.quosures rlang  
## c.quosures rlang  
## print.quosures rlang

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library("DAAG")

## Loading required package: lattice

library(tree)  
library(randomForest)

## randomForest 4.6-14

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

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':  
##   
## margin

library(pROC)

## Type 'citation("pROC")' for a citation.

##   
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':  
##   
## cov, smooth, var

set.seed(1234)

#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.

crime<- read.table("http://www.statsci.org/data/general/uscrime.txt",header=TRUE)  
head(crime)

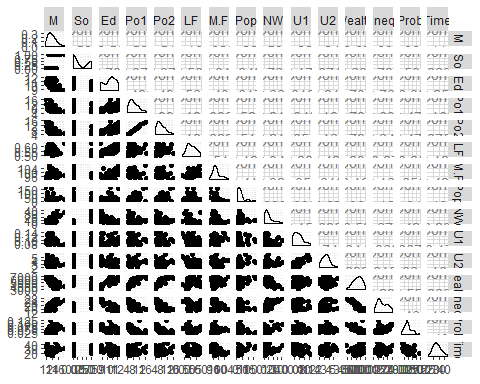
## M So Ed Po1 Po2 LF M.F Pop NW U1 U2 Wealth Ineq  
## 1 15.1 1 9.1 5.8 5.6 0.510 95.0 33 30.1 0.108 4.1 3940 26.1  
## 2 14.3 0 11.3 10.3 9.5 0.583 101.2 13 10.2 0.096 3.6 5570 19.4  
## 3 14.2 1 8.9 4.5 4.4 0.533 96.9 18 21.9 0.094 3.3 3180 25.0  
## 4 13.6 0 12.1 14.9 14.1 0.577 99.4 157 8.0 0.102 3.9 6730 16.7  
## 5 14.1 0 12.1 10.9 10.1 0.591 98.5 18 3.0 0.091 2.0 5780 17.4  
## 6 12.1 0 11.0 11.8 11.5 0.547 96.4 25 4.4 0.084 2.9 6890 12.6  
## Prob Time Crime  
## 1 0.084602 26.2011 791  
## 2 0.029599 25.2999 1635  
## 3 0.083401 24.3006 578  
## 4 0.015801 29.9012 1969  
## 5 0.041399 21.2998 1234  
## 6 0.034201 20.9995 682

#Check out if there are correlations between the predictors

names(crime)

## [1] "M" "So" "Ed" "Po1" "Po2" "LF" "M.F"   
## [8] "Pop" "NW" "U1" "U2" "Wealth" "Ineq" "Prob"   
## [15] "Time" "Crime"

ggpairs(crime,columns=c("M","So","Ed","Po1","Po2","LF","M.F","Pop","NW","U1","U2","Wealth","Ineq","Prob","Time"))



#There are correlations between Po1 vs Po2, Wealth vs Ed/Po1/Po2/Ineq , so PCA is a good choose.

# remove the response variable (it’s in the 16th column)

vars<-crime[-16]  
pca<-prcomp(vars, scale = TRUE)  
summary(pca)

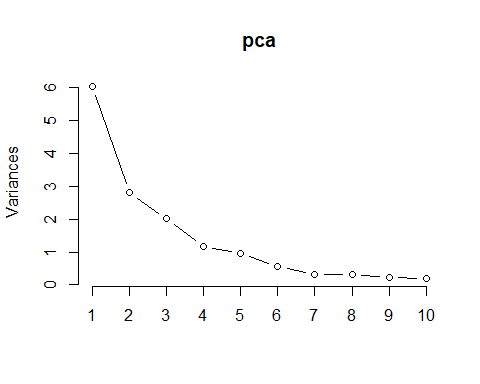
## Importance of components:  
## PC1 PC2 PC3 PC4 PC5 PC6  
## Standard deviation 2.4534 1.6739 1.4160 1.07806 0.97893 0.74377  
## Proportion of Variance 0.4013 0.1868 0.1337 0.07748 0.06389 0.03688  
## Cumulative Proportion 0.4013 0.5880 0.7217 0.79920 0.86308 0.89996  
## PC7 PC8 PC9 PC10 PC11 PC12  
## 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

#get the eigenvector of the matrix

eigen<-pca$rotation

#Use the screenplot to plot the variance of each princpal component

screeplot(pca,type="line",col="black")



#get the first 4 pc pc<-pca$x[,1:4]

#fit a linear regression model with the these 4 pc

crimepc1<-cbind(pc,crime$Crime)

``` ## Error in cbind(pc, crime$Crime): 鎵句笉鍒板璞