Example of PCA

bradl

2020-02-19

# Clear environment  
rm(list = ls())  
  
# Setting the random number generator seed so that our results are reproducible  
set.seed(101)  
  
#First, Read in the data  
usCrime <- read.table("C:/Users/bradl/OneDrive/Documents/uscrime.txt", stringsAsFactors = FALSE, header = TRUE)  
  
# optional check to make sure the data is read correctly  
head(usCrime)

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

summary(usCrime)

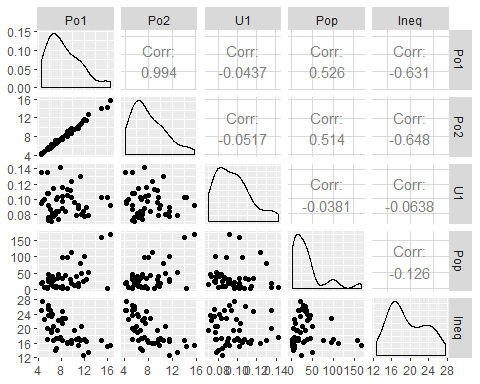
## M So Ed Po1   
## Min. :11.90 Min. :0.0000 Min. : 8.70 Min. : 4.50   
## 1st Qu.:13.00 1st Qu.:0.0000 1st Qu.: 9.75 1st Qu.: 6.25   
## Median :13.60 Median :0.0000 Median :10.80 Median : 7.80   
## Mean :13.86 Mean :0.3404 Mean :10.56 Mean : 8.50   
## 3rd Qu.:14.60 3rd Qu.:1.0000 3rd Qu.:11.45 3rd Qu.:10.45   
## Max. :17.70 Max. :1.0000 Max. :12.20 Max. :16.60   
## Po2 LF M.F Pop   
## Min. : 4.100 Min. :0.4800 Min. : 93.40 Min. : 3.00   
## 1st Qu.: 5.850 1st Qu.:0.5305 1st Qu.: 96.45 1st Qu.: 10.00   
## Median : 7.300 Median :0.5600 Median : 97.70 Median : 25.00   
## Mean : 8.023 Mean :0.5612 Mean : 98.30 Mean : 36.62   
## 3rd Qu.: 9.700 3rd Qu.:0.5930 3rd Qu.: 99.20 3rd Qu.: 41.50   
## Max. :15.700 Max. :0.6410 Max. :107.10 Max. :168.00   
## NW U1 U2 Wealth   
## Min. : 0.20 Min. :0.07000 Min. :2.000 Min. :2880   
## 1st Qu.: 2.40 1st Qu.:0.08050 1st Qu.:2.750 1st Qu.:4595   
## Median : 7.60 Median :0.09200 Median :3.400 Median :5370   
## Mean :10.11 Mean :0.09547 Mean :3.398 Mean :5254   
## 3rd Qu.:13.25 3rd Qu.:0.10400 3rd Qu.:3.850 3rd Qu.:5915   
## Max. :42.30 Max. :0.14200 Max. :5.800 Max. :6890   
## Ineq Prob Time Crime   
## Min. :12.60 Min. :0.00690 Min. :12.20 Min. : 342.0   
## 1st Qu.:16.55 1st Qu.:0.03270 1st Qu.:21.60 1st Qu.: 658.5   
## Median :17.60 Median :0.04210 Median :25.80 Median : 831.0   
## Mean :19.40 Mean :0.04709 Mean :26.60 Mean : 905.1   
## 3rd Qu.:22.75 3rd Qu.:0.05445 3rd Qu.:30.45 3rd Qu.:1057.5   
## Max. :27.60 Max. :0.11980 Max. :44.00 Max. :1993.0

# This is one way to examine some of the correlation in the data.  
# This is a visualization of a subset of the correlation matrix of the data.  
library(GGally)

## Loading required package: ggplot2

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

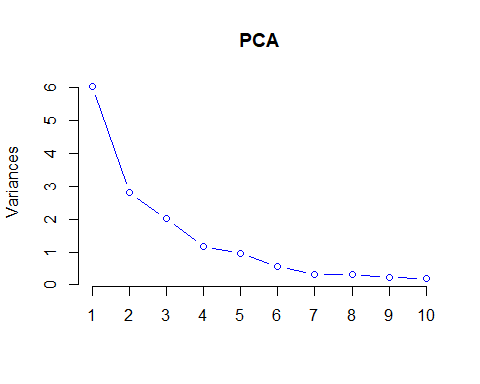
ggpairs(usCrime, columns = c("Po1", "Po2", "U1", "Pop", "Ineq"))



# Run PCA on the matrix of SCALED predictors.  
PCA <- prcomp(usCrime[,1:15], scale. = TRUE)  
summary(PCA)

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

# pca$rotation is the matrix of eigenvectors. Linear transformation = rotation  
rotation <- PCA$rotation  
  
# pca$center is the lateral shift required to have mean / center = 0 used for data scaling  
center <- PCA$center  
  
# calculate sdev for each factor  
sdev <- rbind(sd(usCrime[,1]),sd(usCrime[,2]),sd(usCrime[,3]),sd(usCrime[,4]),  
 sd(usCrime[,5]),sd(usCrime[,6]),sd(usCrime[,7]),sd(usCrime[,8]),  
 sd(usCrime[,9]),sd(usCrime[,10]),sd(usCrime[,11]),sd(usCrime[,12]),  
 sd(usCrime[,13]),sd(usCrime[,14]),sd(usCrime[,15]))  
  
# We can use the screeplot function to plot the variances of each of the principal  
# components (where variance = pca$sdev^2) to help us to decide on a number of principal  
# components to use. We are told to just use the first few principal components in this  
# homework though. Could we do better than this?  
screeplot(PCA,type="lines",col="blue")



# Select first 4 principal component as there is minimal variance gained for at >4 PC's for each additional PC added.  
PC1 <- PCA$x[,1:4]  
PC1

## PC1 PC2 PC3 PC4  
## [1,] -4.1992835 -1.09383120 -1.11907395 0.67178115  
## [2,] 1.1726630 0.67701360 -0.05244634 -0.08350709  
## [3,] -4.1737248 0.27677501 -0.37107658 0.37793995  
## [4,] 3.8349617 -2.57690596 0.22793998 0.38262331  
## [5,] 1.8392999 1.33098564 1.27882805 0.71814305  
## [6,] 2.9072336 -0.33054213 0.53288181 1.22140635  
## [7,] 0.2457752 -0.07362562 -0.90742064 1.13685873  
## [8,] -0.1301330 -1.35985577 0.59753132 1.44045387  
## [9,] -3.6103169 -0.68621008 1.28372246 0.55171150  
## [10,] 1.1672376 3.03207033 0.37984502 -0.28887026  
## [11,] 2.5384879 -2.66771358 1.54424656 -0.87671210  
## [12,] 1.0065920 -0.06044849 1.18861346 -1.31261964  
## [13,] 0.5161143 0.97485189 1.83351610 -1.59117618  
## [14,] 0.4265556 1.85044812 1.02893477 -0.07789173  
## [15,] -3.3435299 0.05182823 -1.01358113 0.08840211  
## [16,] -3.0310689 -2.10295524 -1.82993161 0.52347187  
## [17,] -0.2262961 1.44939774 -1.37565975 0.28960865  
## [18,] -0.1127499 -0.39407030 -0.38836278 3.97985093  
## [19,] 2.9195668 -1.58646124 0.97612613 0.78629766  
## [20,] 2.2998485 -1.73396487 -2.82423222 -0.23281758  
## [21,] 1.1501667 0.13531015 0.28506743 -2.19770548  
## [22,] -5.6594827 -1.09730404 0.10043541 -0.05245484  
## [23,] -0.1011749 -0.57911362 0.71128354 -0.44394773  
## [24,] 1.3836281 1.95052341 -2.98485490 -0.35942784  
## [25,] 0.2727756 2.63013778 1.83189535 0.05207518  
## [26,] 4.0565577 1.17534729 -0.81690756 1.66990720  
## [27,] 0.8929694 0.79236692 1.26822542 -0.57575615  
## [28,] 0.1514495 1.44873320 0.10857670 -0.51040146  
## [29,] 3.5592481 -4.76202163 0.75080576 0.64692974  
## [30,] -4.1184576 -0.38073981 1.43463965 0.63330834  
## [31,] -0.6811731 1.66926027 -2.88645794 -1.30977099  
## [32,] 1.7157269 -1.30836339 -0.55971313 -0.70557980  
## [33,] -1.8860627 0.59058174 1.43570145 0.18239089  
## [34,] 1.9526349 0.52395429 -0.75642216 0.44289927  
## [35,] 1.5888864 -3.12998571 -1.73107199 -1.68604766  
## [36,] 1.0709414 -1.65628271 0.79436888 -1.85172698  
## [37,] -4.1101715 0.15766712 2.36296974 -0.56868399  
## [38,] -0.7254706 2.89263339 -0.36348376 -0.50612576  
## [39,] -3.3451254 -0.95045293 0.19551398 -0.27716645  
## [40,] -1.0644466 -1.05265304 0.82886286 -0.12042931  
## [41,] 1.4933989 1.86712106 1.81853582 -1.06112429  
## [42,] -0.6789284 1.83156328 -1.65435992 0.95121379  
## [43,] -2.4164258 -0.46701087 1.42808323 0.41149015  
## [44,] 2.2978729 0.41865689 -0.64422929 -0.63462770  
## [45,] -2.9245282 -1.19488555 -3.35139309 -1.48966984  
## [46,] 1.7654525 0.95655926 0.98576138 1.05683769  
## [47,] 2.3125056 2.56161119 -1.58223354 0.59863946

# Build a linear regression model with the first 4 principal components.  
usCrimePC1 <- cbind(PC1, usCrime[,16])  
modelPCA1 <- lm(V5~., data = as.data.frame(usCrimePC1))  
modelPCA1

##   
## Call:  
## lm(formula = V5 ~ ., data = as.data.frame(usCrimePC1))  
##   
## Coefficients:  
## (Intercept) PC1 PC2 PC3 PC4   
## 905.09 65.22 -70.08 25.19 69.45

summary(modelPCA1)

##   
## Call:  
## lm(formula = V5 ~ ., data = as.data.frame(usCrimePC1))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -557.76 -210.91 -29.08 197.26 810.35   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 905.09 49.07 18.443 < 2e-16 \*\*\*  
## PC1 65.22 20.22 3.225 0.00244 \*\*   
## PC2 -70.08 29.63 -2.365 0.02273 \*   
## PC3 25.19 35.03 0.719 0.47602   
## PC4 69.45 46.01 1.509 0.13872   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 336.4 on 42 degrees of freedom  
## Multiple R-squared: 0.3091, Adjusted R-squared: 0.2433   
## F-statistic: 4.698 on 4 and 42 DF, p-value: 0.003178

AIC(modelPCA1)

## [1] 687.0241

BIC(modelPCA1)

## [1] 698.125

# Select first 6 principal component as there is minimal variance gained for at >6 PC's for each additional PC added.  
PC2 <- PCA$x[,1:6]  
PC2

## PC1 PC2 PC3 PC4 PC5 PC6  
## [1,] -4.1992835 -1.09383120 -1.11907395 0.67178115 0.055283376 0.30733835  
## [2,] 1.1726630 0.67701360 -0.05244634 -0.08350709 -1.173199821 -0.58323731  
## [3,] -4.1737248 0.27677501 -0.37107658 0.37793995 0.541345246 0.71872230  
## [4,] 3.8349617 -2.57690596 0.22793998 0.38262331 -1.644746496 0.72948841  
## [5,] 1.8392999 1.33098564 1.27882805 0.71814305 0.041590320 -0.39409015  
## [6,] 2.9072336 -0.33054213 0.53288181 1.22140635 1.374360960 -0.69225131  
## [7,] 0.2457752 -0.07362562 -0.90742064 1.13685873 0.718644387 -0.93107472  
## [8,] -0.1301330 -1.35985577 0.59753132 1.44045387 -0.222781388 0.04912052  
## [9,] -3.6103169 -0.68621008 1.28372246 0.55171150 -0.324292990 0.12683417  
## [10,] 1.1672376 3.03207033 0.37984502 -0.28887026 -0.646056610 0.33130781  
## [11,] 2.5384879 -2.66771358 1.54424656 -0.87671210 -0.324083561 0.44365740  
## [12,] 1.0065920 -0.06044849 1.18861346 -1.31261964 0.358087724 0.25696957  
## [13,] 0.5161143 0.97485189 1.83351610 -1.59117618 0.599881946 1.04761756  
## [14,] 0.4265556 1.85044812 1.02893477 -0.07789173 0.741887592 0.61569775  
## [15,] -3.3435299 0.05182823 -1.01358113 0.08840211 0.002969448 0.17074576  
## [16,] -3.0310689 -2.10295524 -1.82993161 0.52347187 -0.387454246 -0.20965321  
## [17,] -0.2262961 1.44939774 -1.37565975 0.28960865 1.337784608 -0.25633983  
## [18,] -0.1127499 -0.39407030 -0.38836278 3.97985093 0.410914404 0.09317136  
## [19,] 2.9195668 -1.58646124 0.97612613 0.78629766 1.356288600 -0.89044651  
## [20,] 2.2998485 -1.73396487 -2.82423222 -0.23281758 -0.653038858 0.68615337  
## [21,] 1.1501667 0.13531015 0.28506743 -2.19770548 0.084621572 0.45958300  
## [22,] -5.6594827 -1.09730404 0.10043541 -0.05245484 -0.689327990 0.13338054  
## [23,] -0.1011749 -0.57911362 0.71128354 -0.44394773 0.689939865 0.54002731  
## [24,] 1.3836281 1.95052341 -2.98485490 -0.35942784 -0.744371276 0.01453851  
## [25,] 0.2727756 2.63013778 1.83189535 0.05207518 0.803692524 1.52313508  
## [26,] 4.0565577 1.17534729 -0.81690756 1.66990720 -2.895110075 -0.47766314  
## [27,] 0.8929694 0.79236692 1.26822542 -0.57575615 1.830793964 -1.11656766  
## [28,] 0.1514495 1.44873320 0.10857670 -0.51040146 -1.023229895 -0.74149513  
## [29,] 3.5592481 -4.76202163 0.75080576 0.64692974 0.309946510 0.72486153  
## [30,] -4.1184576 -0.38073981 1.43463965 0.63330834 -0.254715638 -0.42316550  
## [31,] -0.6811731 1.66926027 -2.88645794 -1.30977099 -0.470913997 -0.45866080  
## [32,] 1.7157269 -1.30836339 -0.55971313 -0.70557980 0.331277622 1.30802615  
## [33,] -1.8860627 0.59058174 1.43570145 0.18239089 0.291863659 -0.13885903  
## [34,] 1.9526349 0.52395429 -0.75642216 0.44289927 0.723474420 -0.42036754  
## [35,] 1.5888864 -3.12998571 -1.73107199 -1.68604766 0.665406182 0.54144206  
## [36,] 1.0709414 -1.65628271 0.79436888 -1.85172698 0.020031154 -2.43356674  
## [37,] -4.1101715 0.15766712 2.36296974 -0.56868399 -2.469679496 0.07239996  
## [38,] -0.7254706 2.89263339 -0.36348376 -0.50612576 0.028157162 1.06465126  
## [39,] -3.3451254 -0.95045293 0.19551398 -0.27716645 0.487259213 -0.20571166  
## [40,] -1.0644466 -1.05265304 0.82886286 -0.12042931 -0.645884788 0.63320546  
## [41,] 1.4933989 1.86712106 1.81853582 -1.06112429 0.009855774 -1.03480444  
## [42,] -0.6789284 1.83156328 -1.65435992 0.95121379 2.115630145 -0.02332805  
## [43,] -2.4164258 -0.46701087 1.42808323 0.41149015 -0.867397522 -1.13982198  
## [44,] 2.2978729 0.41865689 -0.64422929 -0.63462770 -0.703116983 -0.65215040  
## [45,] -2.9245282 -1.19488555 -3.35139309 -1.48966984 0.806659622 -0.48157983  
## [46,] 1.7654525 0.95655926 0.98576138 1.05683769 0.542466034 0.71712602  
## [47,] 2.3125056 2.56161119 -1.58223354 0.59863946 -1.140712406 0.39563373

# Build a linear regression model with the first 6 principal components.  
usCrimePC2 <- cbind(PC2, usCrime[,16])  
modelPCA2 <- lm(V7~., data = as.data.frame(usCrimePC2))  
modelPCA2

##   
## Call:  
## lm(formula = V7 ~ ., data = as.data.frame(usCrimePC2))  
##   
## Coefficients:  
## (Intercept) PC1 PC2 PC3 PC4 PC5   
## 905.09 65.22 -70.08 25.19 69.45 -229.04   
## PC6   
## -60.21

summary(modelPCA2)

##   
## Call:  
## lm(formula = V7 ~ ., data = as.data.frame(usCrimePC2))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -377.15 -172.23 25.81 132.10 480.38   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 905.09 35.35 25.604 < 2e-16 \*\*\*  
## PC1 65.22 14.56 4.478 6.14e-05 \*\*\*  
## PC2 -70.08 21.35 -3.283 0.00214 \*\*   
## PC3 25.19 25.23 0.998 0.32409   
## PC4 69.45 33.14 2.095 0.04252 \*   
## PC5 -229.04 36.50 -6.275 1.94e-07 \*\*\*  
## PC6 -60.21 48.04 -1.253 0.21734   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 242.3 on 40 degrees of freedom  
## Multiple R-squared: 0.6586, Adjusted R-squared: 0.6074   
## F-statistic: 12.86 on 6 and 40 DF, p-value: 4.869e-08

AIC(modelPCA2)

## [1] 657.8925

BIC(modelPCA2)

## [1] 672.6936

#### CROSS VALIDATION LINEAR REGRESSION ####  
# Perform 10-fold CV with the linear model that was created earlier, load caret package  
library(caret)

## Loading required package: lattice

# Fit linear regression models using 10-fold cross validation  
train\_control <- trainControl(method="cv", number=10)  
  
# Predict the crime rate for test data point using 10-fold cv lm  
modelPCA2CV <- train(V7~.,  
 data = as.data.frame(usCrimePC2),  
 trControl=train\_control,  
 method = "lm")  
modelPCA2CV

## Linear Regression   
##   
## 47 samples  
## 6 predictor  
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 43, 43, 42, 42, 43, 42, ...   
## Resampling results:  
##   
## RMSE Rsquared MAE   
## 259.8227 0.5947805 220.1389  
##   
## Tuning parameter 'intercept' was held constant at a value of TRUE

summary(modelPCA2CV)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -377.15 -172.23 25.81 132.10 480.38   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 905.09 35.35 25.604 < 2e-16 \*\*\*  
## PC1 65.22 14.56 4.478 6.14e-05 \*\*\*  
## PC2 -70.08 21.35 -3.283 0.00214 \*\*   
## PC3 25.19 25.23 0.998 0.32409   
## PC4 69.45 33.14 2.095 0.04252 \*   
## PC5 -229.04 36.50 -6.275 1.94e-07 \*\*\*  
## PC6 -60.21 48.04 -1.253 0.21734   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 242.3 on 40 degrees of freedom  
## Multiple R-squared: 0.6586, Adjusted R-squared: 0.6074   
## F-statistic: 12.86 on 6 and 40 DF, p-value: 4.869e-08

coefficients <- modelPCA2CV$finalModel$coefficients[2:7]  
b0 <- modelPCA2CV$finalModel$coefficients[1]  
  
# calculate scaled regression coefficients factor by factor for k=1 to 6 principal components  
  
scaledCoeff1 <- coefficients\*rotation[1,1:6]  
scaledCoeff2 <- coefficients\*rotation[2,1:6]  
scaledCoeff3 <- coefficients\*rotation[3,1:6]  
scaledCoeff4 <- coefficients\*rotation[4,1:6]  
scaledCoeff5 <- coefficients\*rotation[5,1:6]  
scaledCoeff6 <- coefficients\*rotation[6,1:6]  
scaledCoeff7 <- coefficients\*rotation[7,1:6]  
scaledCoeff8 <- coefficients\*rotation[8,1:6]  
scaledCoeff9 <- coefficients\*rotation[9,1:6]  
scaledCoeff10 <- coefficients\*rotation[10,1:6]  
scaledCoeff11 <- coefficients\*rotation[11,1:6]  
scaledCoeff12 <- coefficients\*rotation[12,1:6]  
scaledCoeff13 <- coefficients\*rotation[13,1:6]  
scaledCoeff14 <- coefficients\*rotation[14,1:6]  
scaledCoeff15 <- coefficients\*rotation[15,1:6]  
  
scaledCoeffs <- rbind(scaledCoeff1,scaledCoeff2,scaledCoeff3,scaledCoeff4,  
 scaledCoeff5,scaledCoeff6,scaledCoeff7,scaledCoeff8,  
 scaledCoeff9,scaledCoeff10,scaledCoeff11,scaledCoeff12,  
 scaledCoeff13,scaledCoeff14,scaledCoeff15)  
  
a\_scaled <- rowSums(scaledCoeffs)  
a\_scaled

## scaledCoeff1 scaledCoeff2 scaledCoeff3 scaledCoeff4 scaledCoeff5   
## 87.838105 43.899723 20.463867 123.111917 118.647767   
## scaledCoeff6 scaledCoeff7 scaledCoeff8 scaledCoeff9 scaledCoeff10   
## 45.893299 112.612562 25.937634 94.987694 1.819916   
## scaledCoeff11 scaledCoeff12 scaledCoeff13 scaledCoeff14 scaledCoeff15   
## 29.445920 45.247340 5.724056 -51.712898 36.128815

a <- a\_scaled/sdev  
a

## [,1]  
## [1,] 6.989232e+01  
## [2,] 9.165344e+01  
## [3,] 1.829254e+01  
## [4,] 4.142536e+01  
## [5,] 4.243282e+01  
## [6,] 1.135641e+03  
## [7,] 3.821603e+01  
## [8,] 6.812930e-01  
## [9,] 9.237458e+00  
## [10,] 1.009450e+02  
## [11,] 3.486602e+01  
## [12,] 4.689284e-02  
## [13,] 1.434742e+00  
## [14,] -2.274397e+03  
## [15,] 5.097975e+00

a0\_subtractions <- a\*center  
a0 <- b0 - sum(a0\_subtractions)  
  
# Final note: Even though PCA estimates "proportion of variance" between the factors,  
# and R-squared estimates "proportion of variance explained" in the response,  
# they are not exactly the same thing -- so be careful.  
  
# Create the test datapoint manually  
test\_point <- 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 = 0.040, Time = 39.0)  
  
# Predict the crime rate for test data point  
prediction = a0 + sum(a\*test\_point)  
prediction

## (Intercept)   
## 1248.427

# Double Check Matrix Multiplication for predictiom  
prediction1 = a0 + a[1]\*test\_point[1] + a[2]\*test\_point[2] + a[3]\*test\_point[3] + + a[4]\*test\_point[4] +  
 a[5]\*test\_point[5] + a[6]\*test\_point[6] + + a[7]\*test\_point[7] + a[8]\*test\_point[8] + a[9]\*test\_point[9] +  
 a[10]\*test\_point[10] + a[11]\*test\_point[11] + a[12]\*test\_point[12] + a[13]\*test\_point[13] + a[14]\*test\_point[14] +  
 a[15]\*test\_point[15]  
prediction1

## M  
## 1 1248.427

# Cross-check result by instead scaling the test point  
test\_point\_scaled <- (test\_point-center)/sdev  
predictionCC = b0+sum(a\_scaled\*test\_point\_scaled)  
predictionCC

## (Intercept)   
## 1248.427

# Select all principal components for Linear Regression for analysis / comparison with last week's results  
PC3 <- PCA$x  
  
# Build a linear regression model with the 15 principal components.  
usCrimePC3 <- cbind(PC3, usCrime[,16])  
modelPCA3 <- lm(V16~., data = as.data.frame(usCrimePC3))  
modelPCA3

##   
## Call:  
## lm(formula = V16 ~ ., data = as.data.frame(usCrimePC3))  
##   
## Coefficients:  
## (Intercept) PC1 PC2 PC3 PC4 PC5   
## 905.09 65.22 -70.08 25.19 69.45 -229.04   
## PC6 PC7 PC8 PC9 PC10 PC11   
## -60.21 117.26 28.72 -37.18 56.32 30.59   
## PC12 PC13 PC14 PC15   
## 289.61 81.79 219.19 -622.21

summary(modelPCA3)

##   
## Call:  
## lm(formula = V16 ~ ., data = as.data.frame(usCrimePC3))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -395.74 -98.09 -6.69 112.99 512.67   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 905.09 30.50 29.680 < 2e-16 \*\*\*  
## PC1 65.22 12.56 5.191 1.24e-05 \*\*\*  
## PC2 -70.08 18.42 -3.806 0.000625 \*\*\*  
## PC3 25.19 21.77 1.157 0.255987   
## PC4 69.45 28.59 2.429 0.021143 \*   
## PC5 -229.04 31.49 -7.274 3.49e-08 \*\*\*  
## PC6 -60.21 41.44 -1.453 0.156305   
## PC7 117.26 54.34 2.158 0.038794 \*   
## PC8 28.72 55.60 0.517 0.609159   
## PC9 -37.18 63.57 -0.585 0.562890   
## PC10 56.32 68.95 0.817 0.420261   
## PC11 30.59 73.54 0.416 0.680272   
## PC12 289.61 86.09 3.364 0.002059 \*\*   
## PC13 81.79 117.06 0.699 0.489962   
## PC14 219.19 127.48 1.719 0.095517 .   
## PC15 -622.21 453.79 -1.371 0.180174   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 209.1 on 31 degrees of freedom  
## Multiple R-squared: 0.8031, Adjusted R-squared: 0.7078   
## F-statistic: 8.429 on 15 and 31 DF, p-value: 3.539e-07

AIC(modelPCA3)

## [1] 650.0291

BIC(modelPCA3)

## [1] 681.4816

# R^2 = 0.8031 & Adj R^2 = 0.7078 is equivalent to running basic Linear Regression as per last week for all 15 factors.  
  
# Select 12 principal components for Linear Regression for analysis / comparison with last week's results  
PC4 <- PCA$x[,1:12]  
  
# Build a linear regression model with the 15 principal components.  
usCrimePC4 <- cbind(PC4, usCrime[,16])  
modelPCA4 <- lm(V13~., data = as.data.frame(usCrimePC4))  
modelPCA4

##   
## Call:  
## lm(formula = V13 ~ ., data = as.data.frame(usCrimePC4))  
##   
## Coefficients:  
## (Intercept) PC1 PC2 PC3 PC4 PC5   
## 905.09 65.22 -70.08 25.19 69.45 -229.04   
## PC6 PC7 PC8 PC9 PC10 PC11   
## -60.21 117.26 28.72 -37.18 56.32 30.59   
## PC12   
## 289.61

summary(modelPCA4)

##   
## Call:  
## lm(formula = V13 ~ ., data = as.data.frame(usCrimePC4))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -491.36 -106.97 27.35 94.53 526.39   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 905.09 31.52 28.714 < 2e-16 \*\*\*  
## PC1 65.22 12.99 5.022 1.61e-05 \*\*\*  
## PC2 -70.08 19.03 -3.682 0.000797 \*\*\*  
## PC3 25.19 22.50 1.120 0.270704   
## PC4 69.45 29.55 2.350 0.024730 \*   
## PC5 -229.04 32.55 -7.037 4.01e-08 \*\*\*  
## PC6 -60.21 42.84 -1.406 0.168908   
## PC7 117.26 56.16 2.088 0.044385 \*   
## PC8 28.72 57.47 0.500 0.620493   
## PC9 -37.18 65.70 -0.566 0.575235   
## PC10 56.32 71.26 0.790 0.434854   
## PC11 30.59 76.01 0.402 0.689854   
## PC12 289.61 88.99 3.255 0.002572 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 216.1 on 34 degrees of freedom  
## Multiple R-squared: 0.7693, Adjusted R-squared: 0.6878   
## F-statistic: 9.446 on 12 and 34 DF, p-value: 1.15e-07

AIC(modelPCA4)

## [1] 651.4788

BIC(modelPCA4)

## [1] 677.3808