Module 5: Principal Component Analysis

Weixing Song

August 15, 2018

First, we use the following R-codes to install all necessary packages.

```
list.of.packages=c("ggfortify")
if(length(which(!list.of.packages %in% installed.packages()))){
  install.packages(list.of.packages[!list.of.packages %in% installed.packages()])
}
```

setwd("C:/Users/dujuan/Documents/workshop_MV/workshop_MV/workshopcode")

1. Population Principal Components

Example 5.1 Find the population principal components and the correlation coefficients between each variable and principal component of the following covariance matrix

$$\begin{bmatrix} 1 & -2 & 0 \\ -2 & 5 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

```
pop.prcomp=function(A)
 {
  eigvv=eigen(A);
  Lam=eigvv$value;
  Ev=eigvv$vectors;
  R=diag(sqrt(1/diag(A)))%*%(Ev)%*%diag(sqrt(Lam))
  ord=order(Lam);
  cat("There are", nrow(A), "PCs\n\n")
  cat("The variances of PCs are\n")
  cat(rev(sort(Lam)),"\n\n")
  cat("The PC coefficients are\n")
  Pcc=Ev[,rev(ord)]
  dimnames(Pcc)=list(NULL,paste("PC",1:nrow(A),sep=""))
  print(Pcc)
  cat("\n\n")
  cat("The Correlation Matrix between Variables and PCs\n")
  dimnames(R)=list(paste("X",1:nrow(A),sep=""),paste("PC",1:nrow(A),sep=""))
  print(R)
 }

Sigma=matrix(c(1,-2,0,-2,5,0,0,0,2),nrow=3)
pop.prcomp(Sigma)
```

```
## There are 3 PCs
##
## The variances of PCs are
## 5.828427 2 0.1715729
##
## The PC coefficients are
```

```
##             PC1 PC2       PC3
## [1,] -0.3826834   0 0.9238795
## [2,]  0.9238795   0 0.3826834
## [3,]  0.0000000   1 0.0000000
##
##
## The Correlation Matrix between Variables and PCs
##           PC1 PC2        PC3
## X1 -0.9238795   0 0.38268343
## X2  0.9974842   0 0.07088902
## X3  0.0000000   1 0.00000000
```

2. Sample Principal Components

Example 5.2 A census provided information, by tract, on five socioeconomic variables for the Madison, Wisconsin, area. The data from 61 tracts are listed in Table 8.5. Can the sample variation be summarized by one or two principal components?

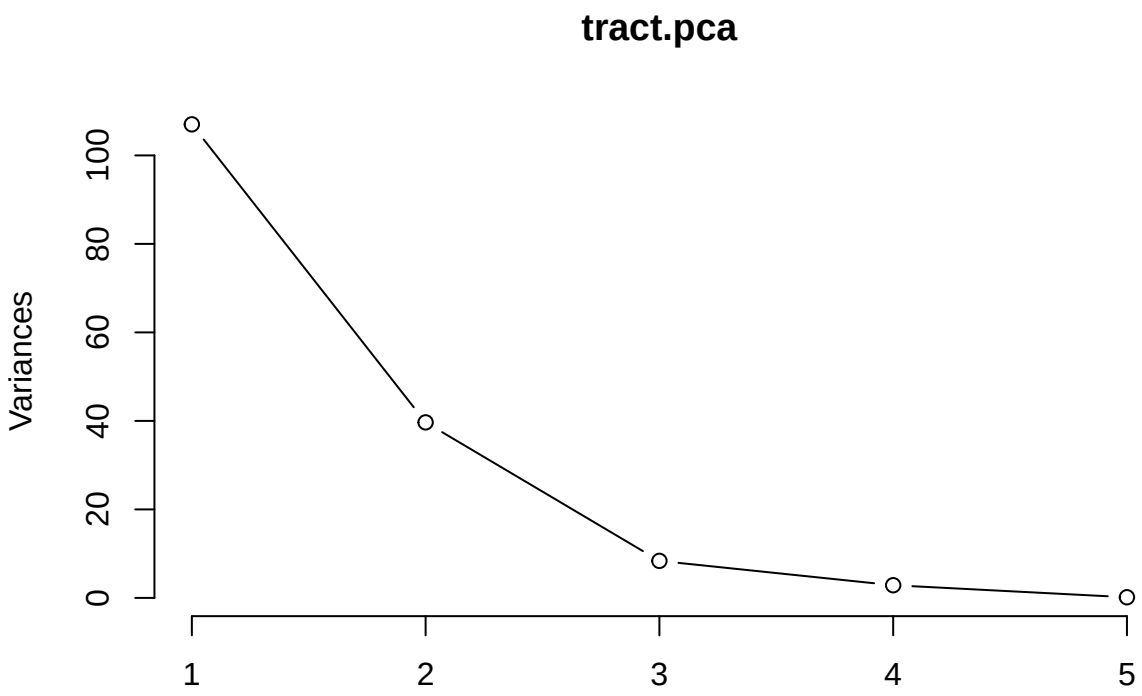
We shall use the function **prcomp** from the package **stats** in R to do PCA. We also use the function **autoplot** from the package **ggfortify** to draw scores from the the first and second PCs.

```
tract=read.table("T8-5.DAT")
S=cov(tract)
tract.pca=prcomp(tract,scale=F)
tract.R=diag(sqrt(1/diag(S)))%*%(tract.pca$rotation)%*%diag(tract.pca$sdev)
dimnames(tract.R)=list(paste("X",1:nrow(S),sep=""),paste("PC",1:nrow(S),sep=""))
print(tract.R)
```

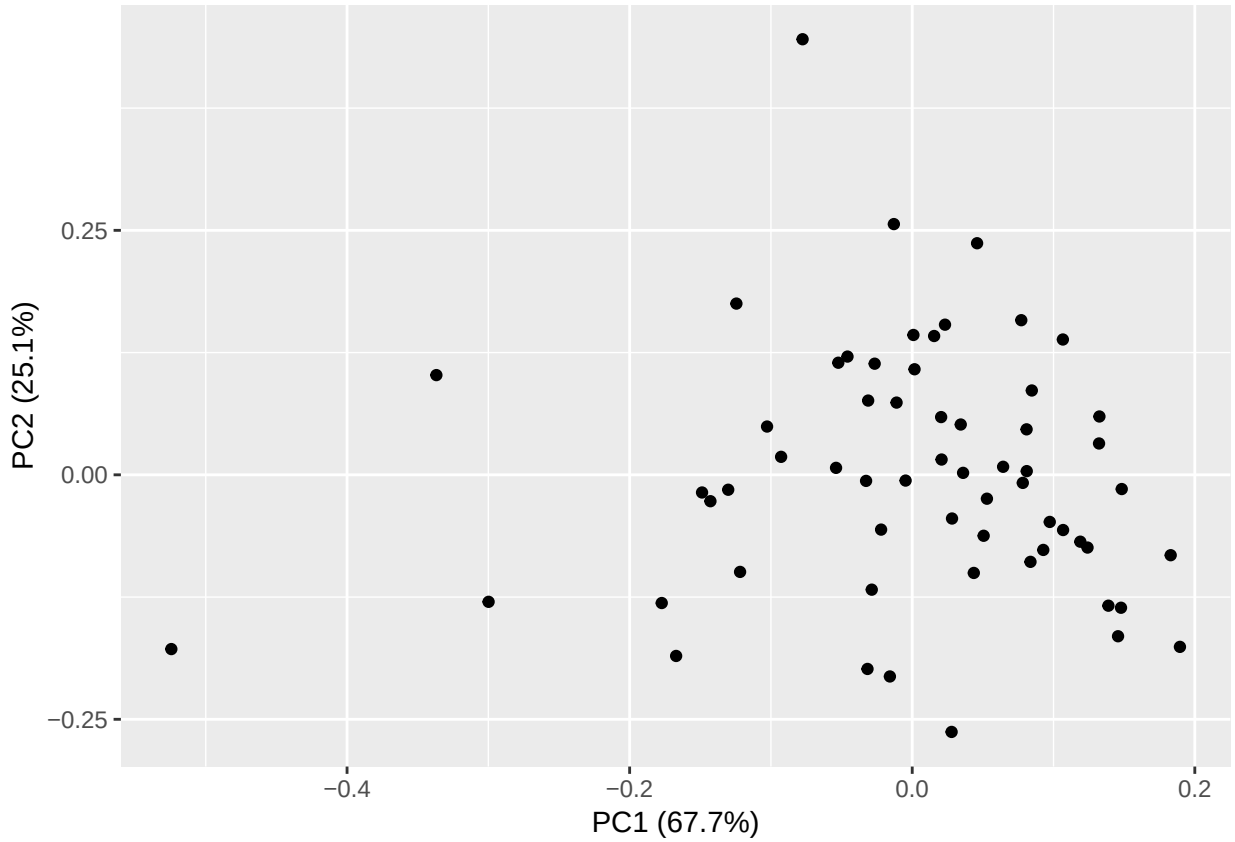
```
##           PC1        PC2         PC3          PC4           PC5
## X1  0.2182675 -0.2431339  0.29495348  0.897828503 -0.0123131431
## X2 -0.3503211 -0.2627741 -0.89398689  0.093302482 -0.0175218224
## X3  0.6829211 -0.7299834  0.01776590 -0.020672417  0.0002618833
## X4 -0.9460440 -0.3205728  0.04696183 -0.005326855  0.0002788823
## X5 -0.1671804 -0.1645133 -0.64060409  0.245114136  0.6888624942
```

```
cumvar.pca=cumsum(tract.pca$sdev^2)/sum(tract.pca$sdev^2)
screeplot(tract.pca,type="l")
library(ggfortify)
```

```
## Loading required package: ggplot2
```




```
autoplot(tract.pca)
```



Example 5.3 The weekly rates of return for five stocks (JP Morgan, Citibank, Wells Fargo, Royal Dutch Shell, and ExxonMobil) listed on the New York Stock Exchange were determined for the period January 2004 through December 2005. The weekly rates of return are defined as (current Friday closing price - previous Friday closing price)/(previous Friday closing price), adjusted for stock splits and dividends. The data are listed in Table 8.4 in the Exercises. The observations in 103 successive weeks appear to be independently distributed, but the rates of return across stocks are correlated, since, as one expects, stocks tend to move together in response to general economic conditions.

Let x_1, x_2, x_3, x_4, x_5 denote the observed weekly rates of return for JP Morgan, Citibank, Wells Fargo, Royal Dutch Shell, and ExxonMobil, respectively. Find the PCs based on the sample correlation coefficient.

```
stock=read.table("T8-4.DAT")
p=dim(stock)[2]
stock.pca=prcomp(stock,scale=T)
print(stock.pca)
```

```
## Standard deviations (1, .., p=5):
## [1] 1.5611768 1.1861756 0.7074693 0.6324805 0.5051434
##
## Rotation (n x k) = (5 x 5):
##           PC1        PC2         PC3        PC4         PC5
## V1 -0.4690832  0.3680070 -0.60431522  0.3630228  0.38412160
## V2 -0.5324055  0.2364624 -0.13610618 -0.6292079 -0.49618794
## V3 -0.4651633  0.3151795  0.77182810  0.2889658  0.07116948
## V4 -0.3873459 -0.5850373  0.09336192 -0.3812515  0.59466408
## V5 -0.3606821 -0.6058463 -0.10882629  0.4934145 -0.49755167
```

```r
stock.R=(stock.pca$rotation)%*%diag(stock.pca$sdev)
dimnames(stock.R)=list(paste("X",1:p,sep=""),paste("PC",1:p,sep=""))
print(stock.R)
```

```
##           PC1        PC2         PC3        PC4         PC5
## X1 -0.7323218  0.4365209 -0.42753444  0.2296048  0.19403650
## X2 -0.8311791  0.2804859 -0.09629094 -0.3979617 -0.25064608
## X3 -0.7262022  0.3738582  0.54604465  0.1827653  0.03595079
## X4 -0.6047155 -0.6939569  0.06605069 -0.2411341  0.30039065
## X5 -0.5630885 -0.7186401 -0.07699125  0.3120751 -0.25133495
```

```r
cumvar.pca=cumsum(stock.pca$sdev^2)/sum(stock.pca$sdev^2)
screeplot(stock.pca,type="l")
```