Module 6: Factor Analysis

Weixing Song

August 15, 2018

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

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

1. Factor Analysis Based on Covariance Matrix

Example 6.1 (Factor analysis of consumer-preference data) In a consumer-preference study, a random sample of customers were asked to rate several attributes of a new product. The responses, on a 7-point semantic differential scale, were tabulated and the attribute correlation matrix constructed. The correlation matrix is presented next:

$$\text{Attribute (Variables)} \begin{cases} \text{Taste} \\ \text{Good buy for money} \\ \text{Flavor} \\ \text{Suitable for snack} \\ \text{Provides lots of energy} \end{cases} \begin{bmatrix} 1.00 & 0.02 & 0.96 & 0.42 & 0.01 \\ 0.02 & 1.00 & 0.13 & 0.71 & 0.85 \\ 0.96 & 0.13 & 1.00 & 0.50 & 0.11 \\ 0.42 & 0.71 & 0.50 & 1.00 & 0.79 \\ 0.01 & 0.85 & 0.11 & 0.79 & 1.00 \end{bmatrix}$$

Principal Component Method

```r
R=matrix(c(1.00,0.02,0.96,0.42,0.01,
           0.02,1.00,0.13,0.71,0.85,
           0.96,0.13,1.00,0.50,0.11,
           0.42,0.71,0.50,1.00,0.79,
           0.01,0.85,0.11,0.79,1.00),nrow=5)
# Customized Codes
R.eigen=eigen(R);
R.eigenvalue=R.eigen$values;
R.eigenvector=R.eigen$vectors;
m=2;
L=R.eigenvector[,1:m]%*%diag(sqrt(R.eigenvalue[1:m]))
h2=rowSums(L^2)
psi=diag(R)-h2
cum.prop=cumsum(R.eigenvalue)/sum(diag(R))
result1=cbind(L,h2,psi)
dimnames(result1)=list(NULL,c("F1","F2","Community","Specific Variance"))
result1
```

```
##              F1         F2 Community Specific Variance
## [1,] -0.5598618  0.8160981 0.9794614        0.02053865
## [2,] -0.7772594 -0.5242021 0.8789200        0.12107998
## [3,] -0.6453364  0.7479464 0.9758829        0.02411712
## [4,] -0.9391057 -0.1049187 0.8929275        0.10707250
```

```
## [5,] -0.7982069 -0.5432281 0.9322311          0.06776888
```

Maximum Likelihood Method

Example 6.2 (Factor Analysis of Stock Price Data Using MLE) The following dat set contains stock price data listed on the New York Stock Exchange were determined for the period January 2004 through December 2005. The estimated factor loadings, communalities, specific variances, and proportion of total (standardized) sample variance explained by each factor are obtained using the following codes. The corresponding figures for the $m = 2$ factor solution obtained by the principal component method are also provided.

```r
stock=read.table("C:/Users/weixi/Dropbox/Teaching/Stat730/Supplements/MVA_data/T8-4.DAT")

# Principal Component Method: Eigen Decomposition
R=cor(stock)
R.eigen=eigen(R);
R.eigenvalue=R.eigen$values;
R.eigenvector=R.eigen$vectors;
m=2;
L=R.eigenvector[,1:m]%*%diag(sqrt(R.eigenvalue[1:m]))
h2=rowSums(L^2)
psi=diag(R)-h2
cum.prop=cumsum(R.eigenvalue)/sum(diag(R))
result2=cbind(L,h2,psi)
dimnames(result2)=list(NULL,c("F1","F2","Community","Specific Variance"))
result2
```
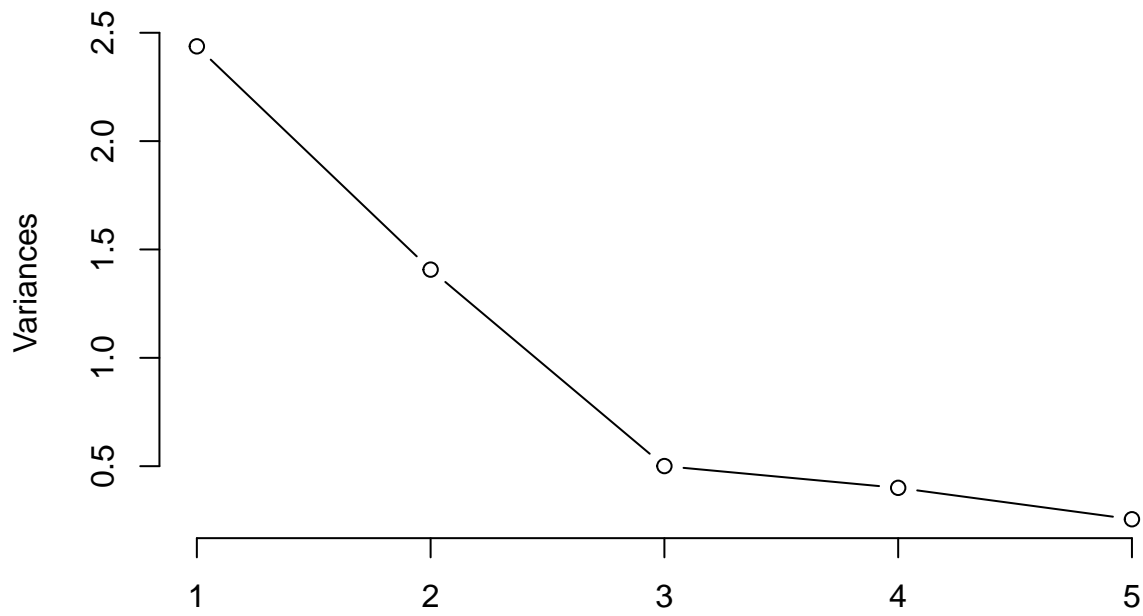
```
##              F1         F2 Community Specific Variance
## [1,] 0.7323218  0.4365209 0.7268458         0.2731542
## [2,] 0.8311791  0.2804859 0.7695311         0.2304689
## [3,] 0.7262022  0.3738582 0.6671396         0.3328604
## [4,] 0.6047155 -0.6939569 0.8472571         0.1527429
## [5,] 0.5630885 -0.7186401 0.8335122         0.1664878
```

```r
# Principal Component Method:Using prcomp function

R.prcomp=prcomp(stock,scale=T)
screeplot(R.prcomp,type="l")
library(ggfortify)
```
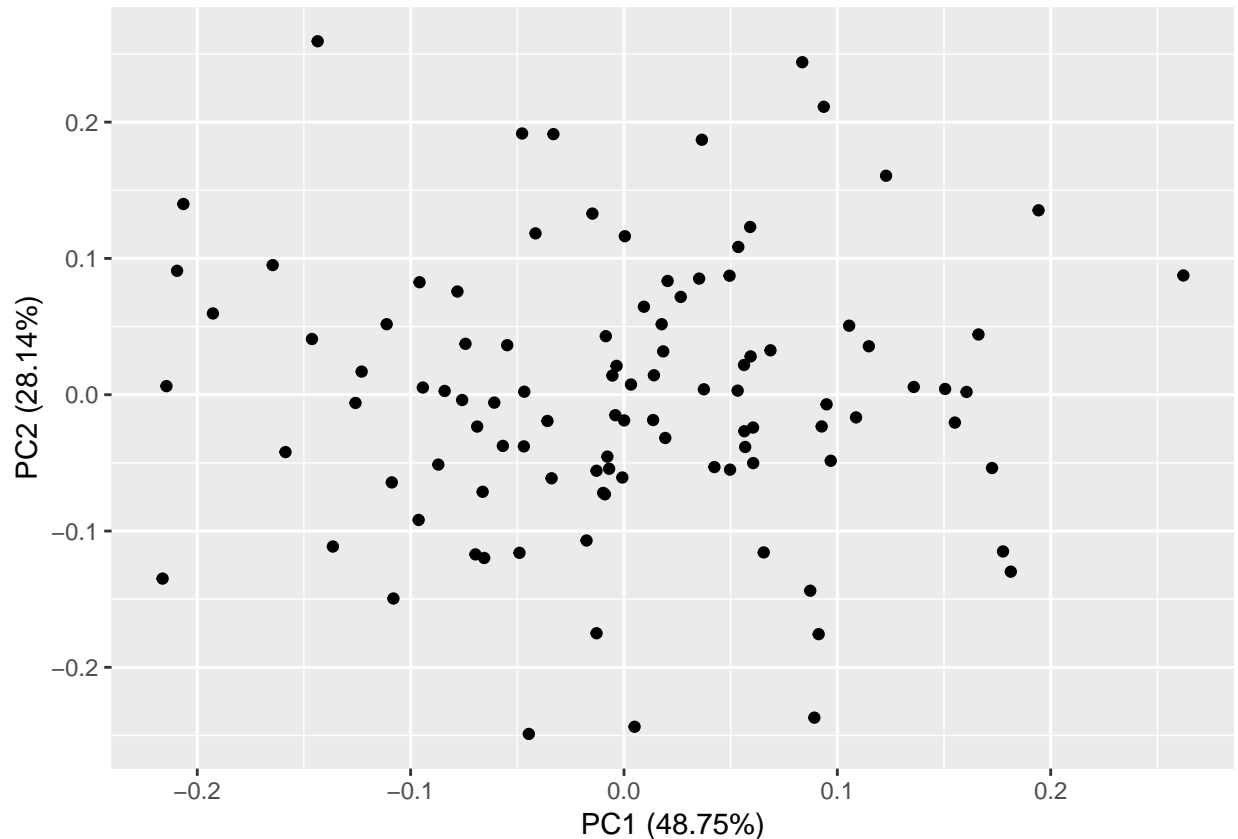
```
## Loading required package: ggplot2
```

R.prcomp


```
autoplot(R.prcomp)
```

```
m=2;
L=R.prcomp$rotation[,1:m]%*%diag(R.prcomp$sdev[1:m])
h2=rowSums(L^2)
psi=diag(R)-h2
result3=cbind(L,h2,psi)
dimnames(result3)=list(NULL,c("F1","F2","Community","Specific Variance"))
result3
```

```
##            F1         F2 Community Specific Variance
## [1,] -0.7323218  0.4365209 0.7268458         0.2731542
## [2,] -0.8311791  0.2804859 0.7695311         0.2304689
## [3,] -0.7262022  0.3738582 0.6671396         0.3328604
## [4,] -0.6047155 -0.6939569 0.8472571         0.1527429
## [5,] -0.5630885 -0.7186401 0.8335122         0.1664878
```

```
# MLE method: factanal is a function in package stats

result=factanal(~V1+V2+V3+V4+V5, factors=2,data=stock,rotation="none")

psi=result$uniquenesses
L=result$loadings
h2=rowSums(L^2)
result4=cbind(L,h2,psi)
dimnames(result4)=list(NULL,c("F1","F2","Community","Specific Variance"))
result4
```

```
##           F1          F2 Community Specific Variance
## [1,] 0.1205972  0.754267060 0.5834625         0.4165374
```

```
## [2,] 0.3284924   0.785749642 0.7253097          0.2746902
## [3,] 0.1876017   0.650217058 0.4579766          0.5420233
## [4,] 0.9974724  -0.007103505 0.9950016          0.0050000
## [5,] 0.6851746   0.026317443 0.4701569          0.5298429
```

```
# MLE method+Rotation: factanal is a function in package stats

result=factanal(~V1+V2+V3+V4+V5, factors=2,data=stock,rotation="varimax")

Psi=result$uniquenesses
L=result$loadings
h2=rowSums(L^2)
result5=cbind(L,h2,psi)
dimnames(result5)=list(NULL,c("F1","F2","Community","Specific Variance"))
result5
```

```
##            F1          F2 Community Specific Variance
## [1,] 0.7632891 0.02919252 0.5834625          0.4165374
## [2,] 0.8194973 0.23180593 0.7253097          0.2746902
## [3,] 0.6680337 0.10820147 0.4579766          0.5420233
## [4,] 0.1126721 0.99111380 0.9950016          0.0050000
## [5,] 0.1083670 0.67706236 0.4701569          0.5298429
```

Example 6.3 Examination Scores

This example shows how to do factor analysis by inputing covariance/correlation matrix.

MLE method

```
R=matrix(c(1, 0.439, 0.410, 0.288, 0.329, 0.248,
           0.439, 1, 0.351, 0.354, 0.320, 0.329,
           0.410, 0.351, 1, 0.164, 0.190, 0.181,
           0.288, 0.354, 0.164, 1, 0.595, 0.470,
           0.329, 0.320, 0.190, 0.595, 1, 0.464,
           0.248, 0.329, 0.181, 0.470, 0.464, 1),nrow=6);

# No rotation
result=factanal(factors=2,covmat=R,rotation="none")
Psi=result$uniquenesses
L=result$loadings
h2=rowSums(L^2)
result6=cbind(L,h2,Psi)
dimnames(result6)=list(NULL,c("F1","F2","Community","Specific Variance"))
result6
```

```
##            F1          F2 Community Specific Variance
## [1,] 0.5533154  0.4285621 0.4898234          0.5101781
## [2,] 0.5681558  0.2883194 0.4059291          0.5940703
## [3,] 0.3921770  0.4499695 0.3562753          0.6437238
## [4,] 0.7404275 -0.2728035 0.6226546          0.3773440
## [5,] 0.7238666 -0.2113024 0.5686316          0.4313708
## [6,] 0.5953587 -0.1316867 0.3717933          0.6282071
```

```
# Rotation with varimax criterion
result=factanal(factors=2,covmat=R,rotation="varimax")
Psi=result$uniquenesses
L=result$loadings
h2=rowSums(L^2)
```

```
result7=cbind(L,h2,Psi)
dimnames(result7)=list(NULL,c("F1","F2","Community","Specific Variance"))
result7
```

```
##             F1         F2 Community Specific Variance
## [1,] 0.23474047 0.6593332 0.4898234         0.5101781
## [2,] 0.32288574 0.5492485 0.4059291         0.5940703
## [3,] 0.08750805 0.5904385 0.3562753         0.6437238
## [4,] 0.77062380 0.1696868 0.6226546         0.3773440
## [5,] 0.72350422 0.2125399 0.5686316         0.4313708
## [6,] 0.57234951 0.2102603 0.3717933         0.6282071
```

```
# Rotation with certain degree

library(psych)
```
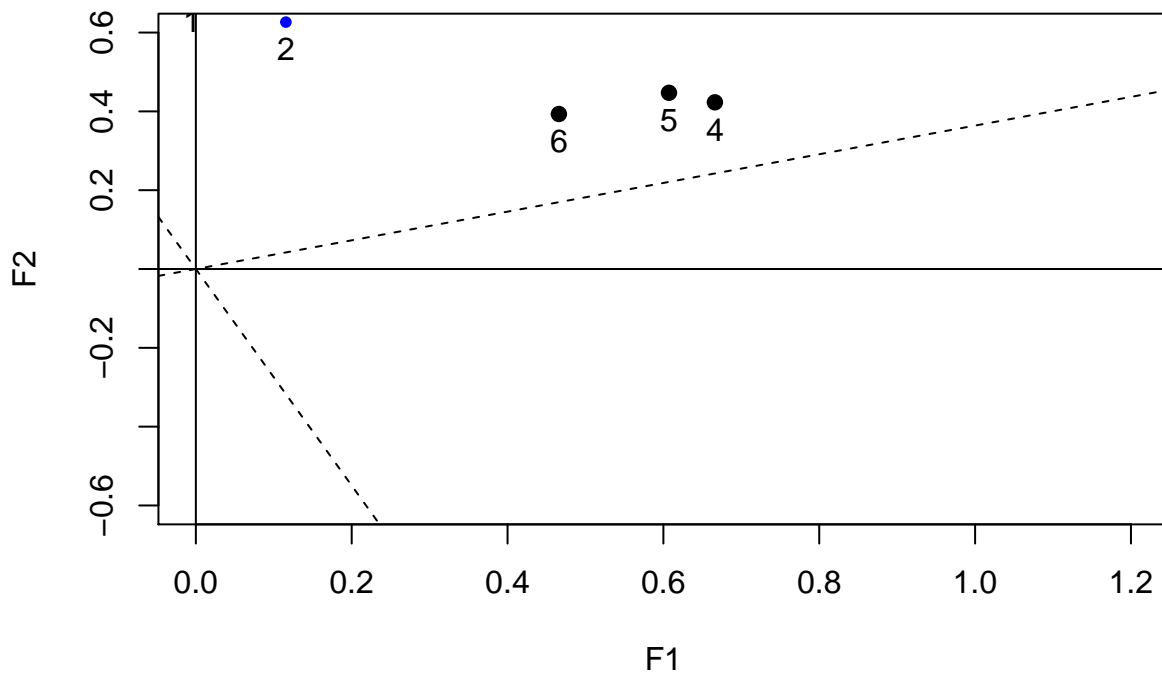
```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
result8=fac(R,nfactors=2,rotate="none",fm="ml")
result9=factor.rotate(result,angle=-20,plot=T,xlim=c(0,1.2),
                ylim=c(-0.6,0.6),title="Rotated -20 degrees")
```

Rotated –20 degrees

```
# The rotation angle is in degrees counter clockwise.
# The dashed line are the original axes.
```