Group10_HW2

Chenxuan He, Zejin Kong

5/19/2020

Problem 1: Perform principal component analysis on NHL.xlsx, which contains statistics of 30 teams in the National Hockey League. The description of the variables is provided in the 'Description' sheet of the file. Focus only on the variables 12 through 25, and create a new data frame.

- **Input the new data frame to fa.parallel() function to determine the number of components to extract**

```r
library(readxl)
NHL <- read_excel("NHL.xlsx")
```

```
## New names:
## * `` -> ...1
```

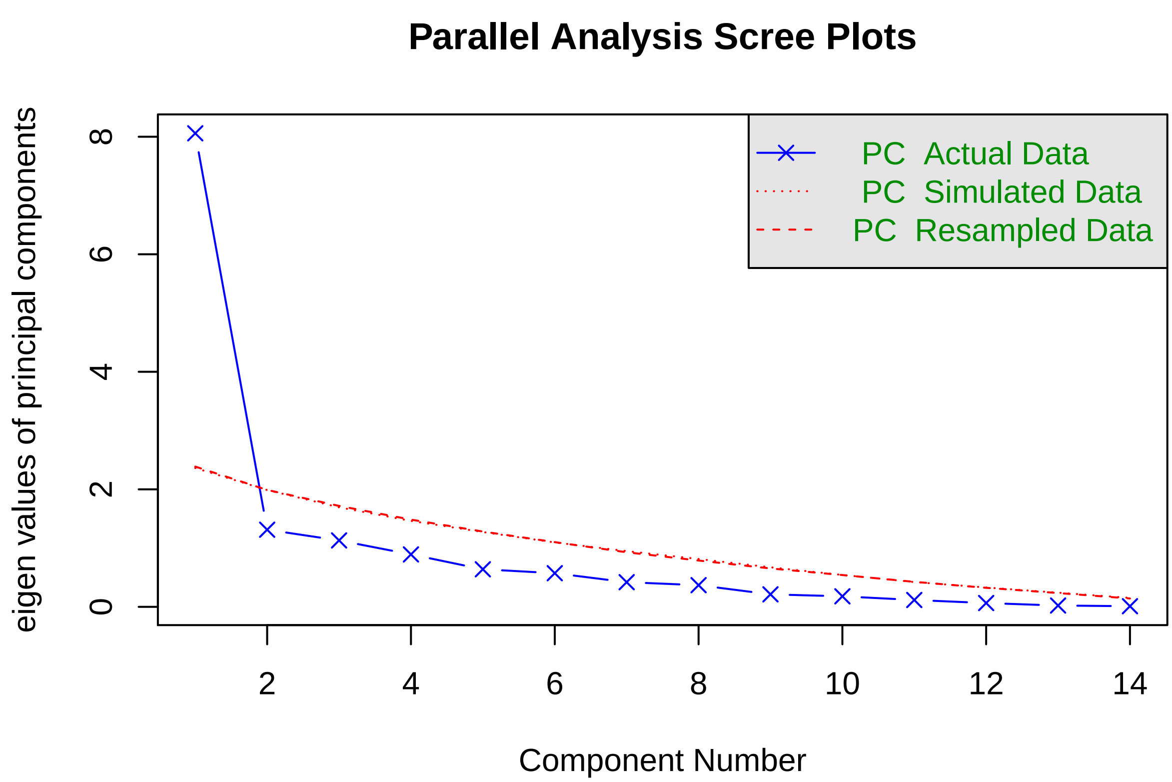
```r
str(NHL)
```

```
## tibble [30 x 26] (S3: tbl_df/tbl/data.frame)
##  $ ...1  : num [1:30] 1 2 3 4 5 6 7 8 9 10 ...
##  $ rank  : num [1:30] 1 2 3 4 5 6 7 8 9 10 ...
##  $ team  : chr [1:30] "NY RANGERS" "ST LOUIS" "ANAHEIM" "MONTREAL" ...
##  $ played: num [1:30] 82 82 82 82 82 82 82 82 82 82 ...
##  $ wins  : num [1:30] 53 51 51 50 50 48 48 47 47 46 ...
##  $ losses: num [1:30] 22 24 24 22 24 28 29 25 28 28 ...
##  $ OTL   : num [1:30] 7 7 7 10 8 6 5 10 7 8 ...
##  $ pts   : num [1:30] 113 109 109 110 108 102 101 104 101 100 ...
##  $ ROW   : num [1:30] 49 42 43 43 47 39 42 41 40 42 ...
##  $ HROW  : num [1:30] 23 23 21 25 30 19 21 24 22 20 ...
##  $ RROW  : num [1:30] 26 19 22 18 17 20 21 17 18 22 ...
##  $ ppc : num [1:30] 0.689 0.665 0.665 0.671 0.659 0.622 0.616 0.634 0.616 0.61 ...
##  $ gg    : num [1:30] 3.02 2.92 2.78 2.61 3.16 2.68 2.88 2.76 2.99 2.77 ...
##  $ gag   : num [1:30] 2.28 2.4 2.7 2.24 2.51 2.27 2.68 2.46 2.73 2.42 ...
##  $ five  : num [1:30] 1.32 1.18 1.04 1.18 1.28 1.19 0.96 1.26 1.08 1.14 ...
##  $ PPP   : num [1:30] 16.8 22.3 15.7 16.5 18.8 17.6 19.3 16.2 18.7 15.8 ...
##  $ PKP   : num [1:30] 84.3 83.7 81 83.7 83.7 83.4 85.7 80.8 78 86.3 ...
##  $ shots : num [1:30] 31.5 30.9 30 28.5 29.6 33.9 29.9 31.9 33.8 30.8 ...
##  $ sag   : num [1:30] 29.5 27.2 28.9 30.1 27.9 30.2 29.8 28.3 28.3 27.6 ...
##  $ sc1   : num [1:30] 0.82 0.783 0.766 0.821 0.761 0.761 0.771 0.711 0.592 0.778 ...
##  $ tr1   : num [1:30] 0.375 0.417 0.429 0.419 0.417 0.361 0.447 0.455 0.545 0.297 ...
```

```
##  $ lead1 : num [1:30] 0.853 0.808 0.813 0.714 0.833 0.897 0.773 0.741 0.667 0.844 ...
##  $ lead2 : num [1:30] 0.973 0.842 0.938 0.865 0.943 1 0.882 0.771 0.781 0.875 ...
##  $ wop   : num [1:30] 0.625 0.66 0.59 0.611 0.622 0.596 0.65 0.63 0.607 0.519 ...
##  $ wosp  : num [1:30] 0.656 0.556 0.643 0.585 0.606 0.586 0.524 0.529 0.476 0.667 ...
##  $ face  : num [1:30] 46.7 53.4 51.6 52.1 49.7 52 46.7 48.9 49.2 49.9 ...
```

```
NHL_new<-(NHL[,12:25])
library(psych)
fa.parallel(NHL_new,fa="pc",n.iter=100)
```

```
## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs = np.obs, :
## The estimated weights for the factor scores are probably incorrect. Try a
## different factor score estimation method.
```




```
## Parallel analysis suggests that the number of factors =  NA  and the number of components =  1
```

- **Input the new data frame to principal() function to extract the components. If raw data is input, the correlation matrix is automatically calculated by principal() function.**

```
principal(NHL_new,nfactors=1,rotate="none")
```

```
## Principal Components Analysis
## Call: principal(r = NHL_new, nfactors = 1, rotate = "none")
## Standardized loadings (pattern matrix) based upon correlation matrix
##         PC1   h2    u2 com
## p\rpc  0.97 0.94 0.057   1
## gg     0.83 0.69 0.308   1
## gag   -0.82 0.67 0.327   1
## five   0.92 0.84 0.162   1
## PPP    0.14 0.02 0.980   1
## PKP    0.69 0.48 0.519   1
## shots  0.59 0.34 0.656   1
## sag   -0.62 0.39 0.612   1
## sc1    0.81 0.66 0.338   1
## tr1    0.76 0.58 0.422   1
## lead1  0.81 0.65 0.351   1
## lead2  0.74 0.55 0.452   1
## wop    0.71 0.51 0.491   1
## wosp   0.86 0.73 0.267   1
##
##                 PC1
## SS loadings    8.06
## Proportion Var 0.58
##
## Mean item complexity =  1
## Test of the hypothesis that 1 component is sufficient.
##
## The root mean square of the residuals (RMSR) is  0.1
##  with the empirical chi square  52.29  with prob <  0.99
##
## Fit based upon off diagonal values = 0.97
```

- **Rotate the components**

```r
principal(NHL_new,nfactors=1,rotate = "varimax")
```

```
## Principal Components Analysis
## Call: principal(r = NHL_new, nfactors = 1, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##         PC1   h2    u2 com
## p\rpc  0.97 0.94 0.057   1
## gg     0.83 0.69 0.308   1
## gag   -0.82 0.67 0.327   1
## five   0.92 0.84 0.162   1
## PPP    0.14 0.02 0.980   1
## PKP    0.69 0.48 0.519   1
## shots  0.59 0.34 0.656   1
## sag   -0.62 0.39 0.612   1
## sc1    0.81 0.66 0.338   1
## tr1    0.76 0.58 0.422   1
## lead1  0.81 0.65 0.351   1
## lead2  0.74 0.55 0.452   1
## wop    0.71 0.51 0.491   1
```

```
## wosp   0.86 0.73 0.267   1
##
##                  PC1
## SS loadings    8.06
## Proportion Var 0.58
##
## Mean item complexity =  1
## Test of the hypothesis that 1 component is sufficient.
##
## The root mean square of the residuals (RMSR) is  0.1
##  with the empirical chi square  52.29  with prob <  0.99
##
## Fit based upon off diagonal values = 0.97
```

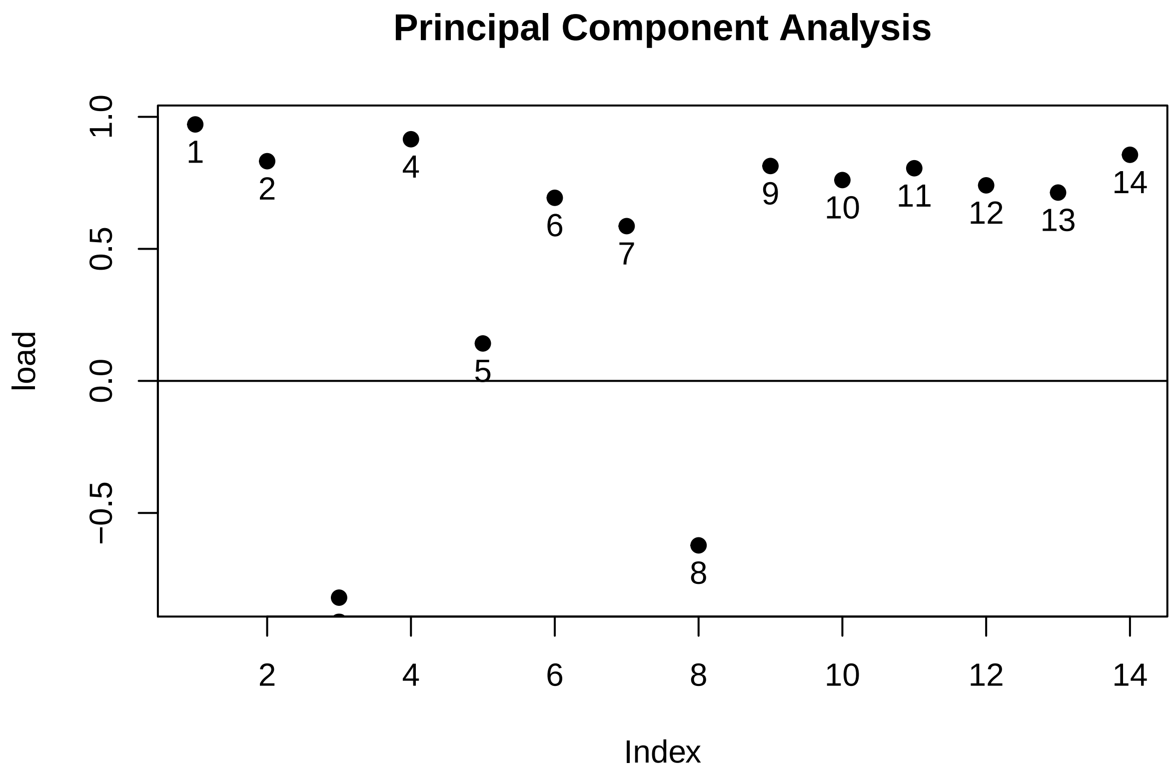
- **Compute component scores**

```r
pc_NHL<-principal(NHL_new,nfactors=1, score = TRUE)
head(pc_NHL$scores)
```

```
##            PC1
## [1,] 1.4163062
## [2,] 1.0620224
## [3,] 0.7363287
## [4,] 0.7558835
## [5,] 1.1940130
## [6,] 1.0912159
```

- **Graph an orthogonal solution using factor.plot()**

```r
factor.plot(pc_NHL)
```



- **Interpret the results**

```
cor(pc_NHL$scores,NHL_new)
```

```
##         p\rpc        gg        gag      five       PPP       PKP     shots
## PC1 0.9712334 0.8320467 -0.8205864 0.9151544 0.1419107 0.6934435 0.5862039
##            sag       sc1       tr1     lead1     lead2       wop      wosp
## PC1 -0.6226516 0.8138146 0.7605732 0.8053326 0.7405376 0.7132033 0.8562918
```

Problem 2: Perform principal component analysis on Glass Identification Data.xlsx

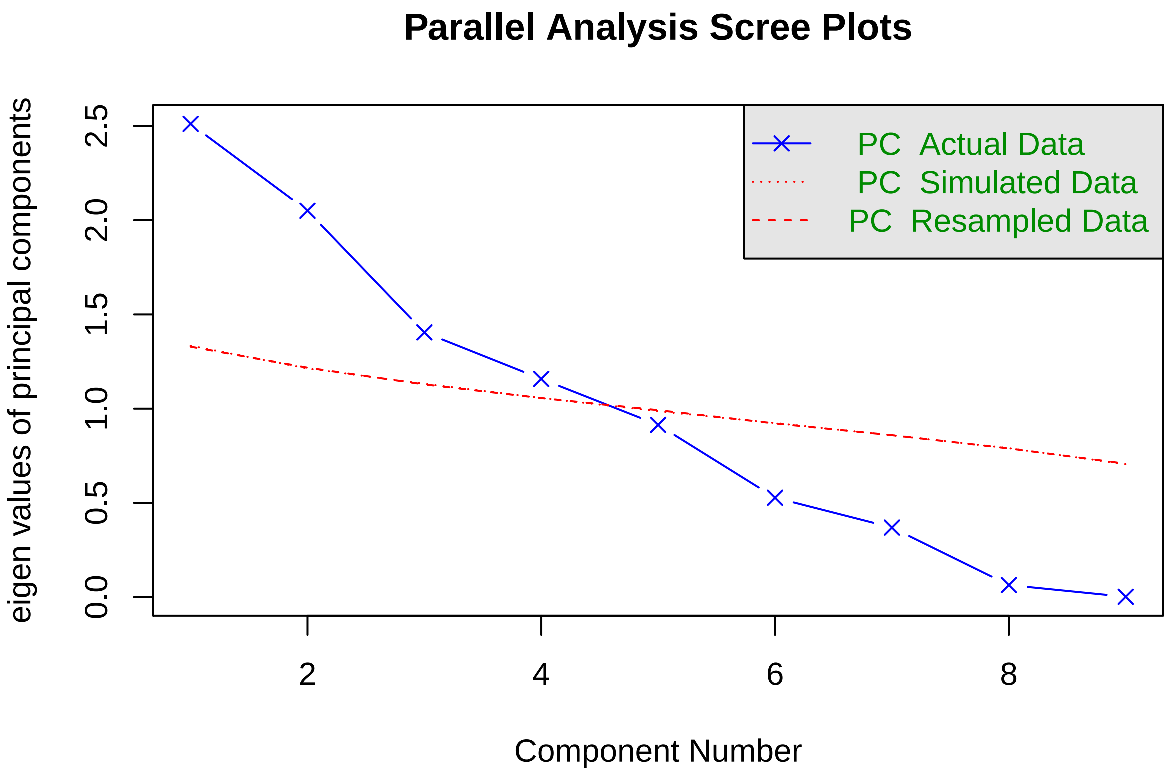
- **Input the raw data matrix to fa.parallel() function to determine the number of components to extract**

```
library(readxl)
Glass_Identification_Data <- read_excel("Glass Identification Data.xlsx")
fa.parallel(Glass_Identification_Data[,-c(1,11)],fa="pc",n.iter=100)
```

```
## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs = np.obs, :
```

```
## The estimated weights for the factor scores are probably incorrect. Try a
## different factor score estimation method.

## Warning in fac(r = r, nfactors = nfactors, n.obs = n.obs, rotate = rotate, : An
## ultra-Heywood case was detected. Examine the results carefully
```




```
## Parallel analysis suggests that the number of factors =  NA  and the number of components =  4
```

- **Input the raw data matrix to principal() function to extract the components. If raw data is input, the correlation matrix is automatically calculated by principal() function.**

```r
principal(Glass_Identification_Data[,-c(1,11)],nfactors=4,rotate="none")
```

```
## Principal Components Analysis
## Call: principal(r = Glass_Identification_Data[, -c(1, 11)], nfactors = 4,
##     rotate = "none")
## Standardized loadings (pattern matrix) based upon correlation matrix
##      PC1   PC2   PC3   PC4   h2    u2 com
## RI -0.86  0.41  0.10 -0.16 0.95 0.051 1.5
## Na  0.41  0.39 -0.46 -0.53 0.80 0.195 3.8
## Mg -0.18 -0.85  0.01 -0.41 0.92 0.081 1.5
```

```
## Al  0.68  0.42  0.39  0.15 0.81 0.186 2.5
## Si  0.36 -0.22 -0.54  0.70 0.97 0.031 2.7
## K   0.35 -0.22  0.79  0.04 0.79 0.212 1.6
## CA -0.78  0.49  0.00  0.30 0.94 0.058 2.0
## Ba  0.40  0.69  0.09 -0.14 0.67 0.333 1.7
## Fe -0.29 -0.09  0.34  0.25 0.27 0.730 3.0
##
##                        PC1  PC2  PC3  PC4
## SS loadings           2.51 2.05 1.40 1.16
## Proportion Var        0.28 0.23 0.16 0.13
## Cumulative Var        0.28 0.51 0.66 0.79
## Proportion Explained  0.35 0.29 0.20 0.16
## Cumulative Proportion 0.35 0.64 0.84 1.00
##
## Mean item complexity =  2.3
## Test of the hypothesis that 4 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.08
##  with the empirical chi square  102.53  with prob <  7.4e-20
##
## Fit based upon off diagonal values = 0.92
```

- **Rotate the components**

```
principal(Glass_Identification_Data[,-c(1,11)],nfactors=4,rotate = "varimax")
```

```
## Principal Components Analysis
## Call: principal(r = Glass_Identification_Data[, -c(1, 11)], nfactors = 4,
##     rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##      RC1   RC2   RC3   RC4   h2    u2 com
## RI  0.84 -0.07  0.15  0.47 0.95 0.051 1.7
## Na -0.06  0.22 -0.86  0.09 0.80 0.195 1.2
## Mg -0.35 -0.86  0.04  0.21 0.92 0.081 1.5
## Al -0.42  0.80  0.03  0.01 0.81 0.186 1.5
## Si -0.13  0.00 -0.02 -0.98 0.97 0.031 1.0
## K  -0.62  0.22  0.51  0.30 0.79 0.212 2.7
## CA  0.91  0.12  0.30  0.06 0.94 0.058 1.3
## Ba -0.01  0.72 -0.33  0.17 0.67 0.333 1.5
## Fe  0.12 -0.04  0.50  0.07 0.27 0.730 1.2
##
##                        RC1  RC2  RC3  RC4
## SS loadings           2.26 2.03 1.48 1.36
## Proportion Var        0.25 0.23 0.16 0.15
## Cumulative Var        0.25 0.48 0.64 0.79
## Proportion Explained  0.32 0.28 0.21 0.19
## Cumulative Proportion 0.32 0.60 0.81 1.00
##
## Mean item complexity =  1.5
## Test of the hypothesis that 4 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.08
```

```
##  with the empirical chi square  102.53  with prob <  7.4e-20
##
## Fit based upon off diagonal values = 0.92
```

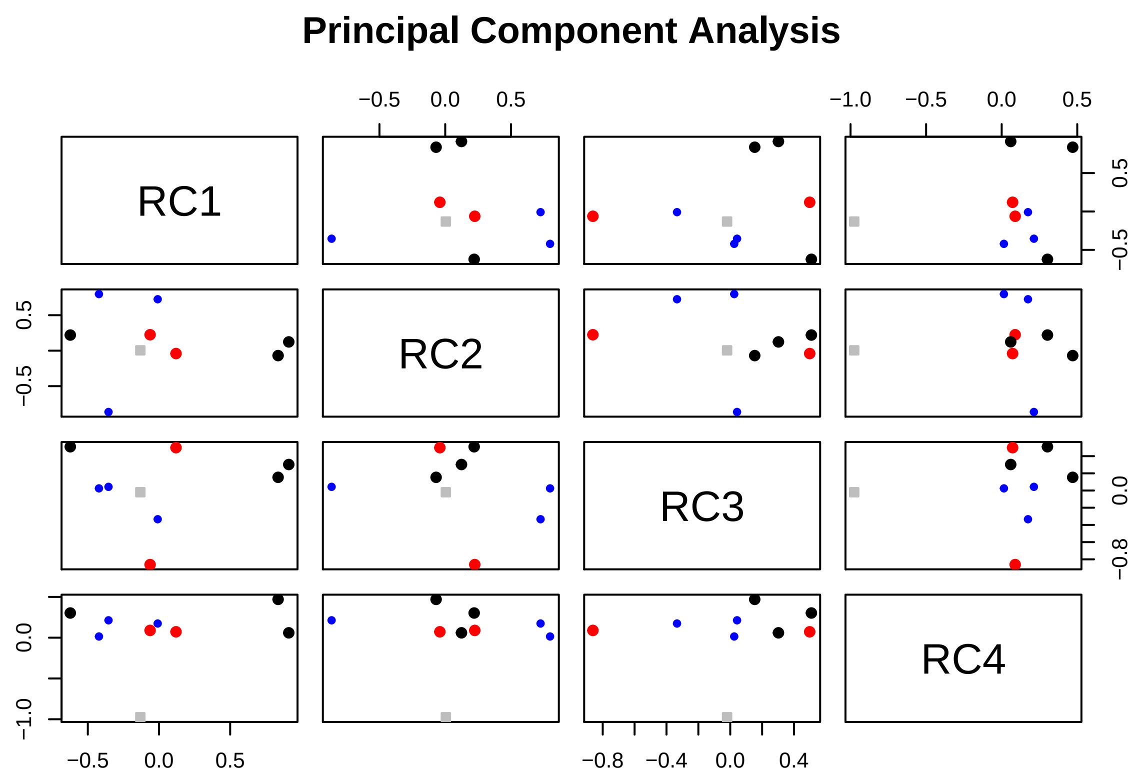
- **Compute component scores**

```
pc_Glass<-principal(Glass_Identification_Data[,-c(1,11)],nfactors=4,score=TRUE)
head(pc_Glass$scores)
```

```
##             RC1        RC2        RC3         RC4
## [1,]  0.2516834 -1.1257154 -0.8331376  1.14203433
## [2,] -0.5120556 -0.5823124 -0.7217195  0.07184681
## [3,] -0.6811108 -0.4417522 -0.4610237 -0.39146231
## [4,] -0.4363986 -0.6266048 -0.1520952  0.09532063
## [5,] -0.4446499 -0.6485935 -0.1947898 -0.37616223
## [6,] -0.7149524 -0.2237372  1.1926990 -0.41874608
```

- **Graph an orthogonal solution using factor.plot()**

```
factor.plot(pc_Glass)
```



• Interpret the results

```r
cor(pc_Glass$scores,Glass_Identification_Data)
```

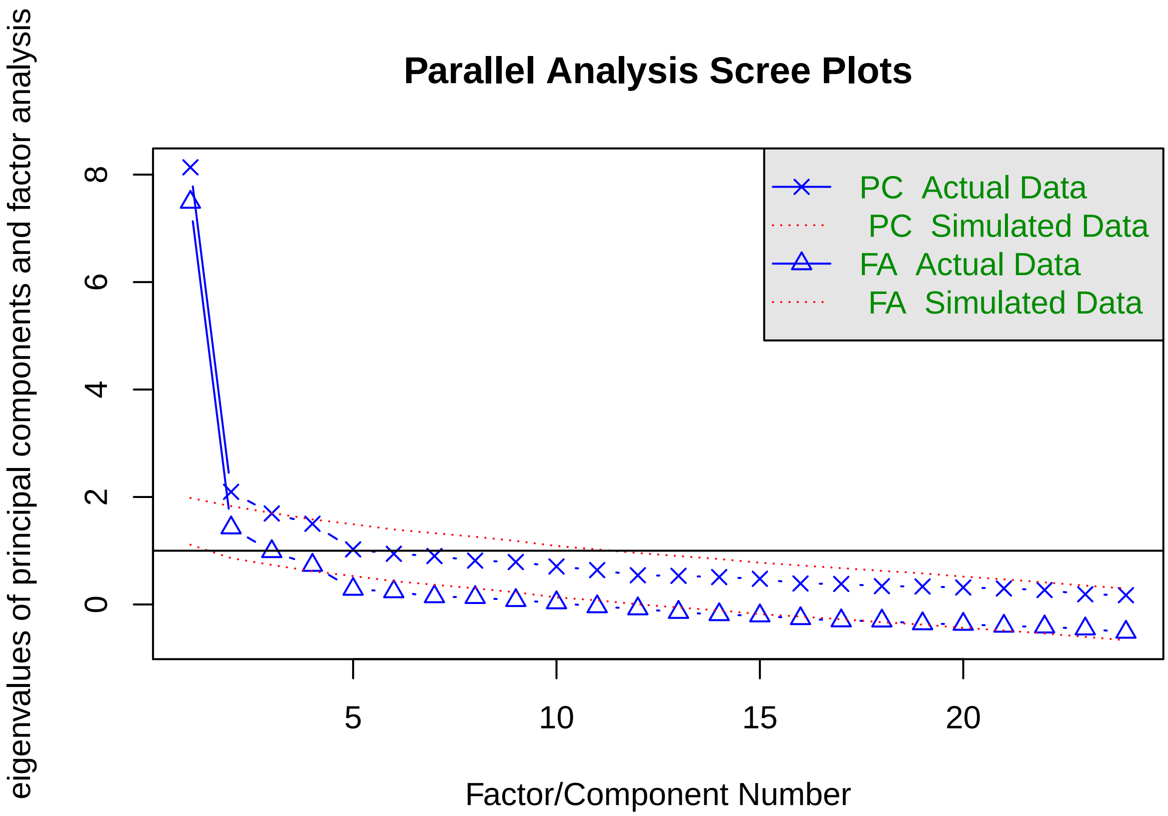
```
##              ID          RI          Na          Mg          Al           Si
## RC1  0.06604725  0.83681998 -0.06239306 -0.35461607 -0.42170176 -0.131078985
## RC2  0.65503208 -0.06920239  0.22477511 -0.86387729  0.79730117  0.004800603
## RC3 -0.19540235  0.15362211 -0.86159181  0.04285079  0.02505138 -0.019763568
## RC4 -0.07604637  0.46987606  0.08920043  0.21329071  0.01475206 -0.975624428
##              K        CA           Ba          Fe        Class
## RC1 -0.6231543 0.9117399 -0.008931497  0.11964638  0.005607427
## RC2  0.2201624 0.1236794  0.724901178 -0.04070424  0.769343218
## RC3  0.5092348 0.3025104 -0.333813764  0.49860648 -0.329641707
## RC4  0.3029136 0.0597812  0.174074595  0.07225133 -0.145061504
```

Problem 3: Perform factor analysis on Herman74.cor, which is a data structure available in the base installation (A correlation matrix of 24 psychological tests given to 145 seventh and eight-grade children in a Chicago suburb by Holzinger and Swineford).

- **Input the correlation matrix to fa.parallel() function to determine the number of components to extract**

```r
Herman<-Harman74.cor$cov
fa.parallel(Herman)
```

```
## Warning in fa.parallel(Herman): It seems as if you are using a correlation
## matrix, but have not specified the number of cases. The number of subjects is
## arbitrarily set to be 100
```




```
## Parallel analysis suggests that the number of factors =  4  and the number of components =  2
```

- **Input the correlation matrix to fa() function to extract the components. If raw data is input, the correlation matrix is automatically calculated by fa() function.**

```r
fa(Herman,nfactors = 4,rotate ="none")
```

```
## Factor Analysis using method =  minres
## Call: fa(r = Herman, nfactors = 4, rotate = "none")
## Standardized loadings (pattern matrix) based upon correlation matrix
##                          MR1   MR2   MR3   MR4   h2   u2 com
## VisualPerception        0.60  0.03  0.38 -0.22 0.55 0.45 2.0
## Cubes                   0.37 -0.03  0.26 -0.15 0.23 0.77 2.2
## PaperFormBoard          0.42 -0.12  0.36 -0.13 0.34 0.66 2.3
## Flags                   0.48 -0.11  0.26 -0.19 0.35 0.65 2.0
## GeneralInformation      0.69 -0.30 -0.27 -0.04 0.64 0.36 1.7
## PargraphComprehension   0.69 -0.40 -0.20  0.08 0.68 0.32 1.8
## SentenceCompletion      0.68 -0.41 -0.30 -0.08 0.73 0.27 2.1
## WordClassification      0.67 -0.19 -0.09 -0.11 0.51 0.49 1.3
## WordMeaning             0.70 -0.45 -0.23  0.08 0.74 0.26 2.0
## Addition                0.47  0.53 -0.48 -0.10 0.74 0.26 3.1
## Code                    0.56  0.36 -0.16  0.09 0.47 0.53 2.0
## CountingDots            0.47  0.50 -0.14 -0.24 0.55 0.45 2.6
```

```
## StraightCurvedCapitals 0.60  0.26  0.01 -0.29 0.51 0.49 1.9
## WordRecognition        0.43  0.06  0.01  0.42 0.36 0.64 2.0
## NumberRecognition      0.39  0.10  0.09  0.37 0.31 0.69 2.2
## FigureRecognition      0.51  0.09  0.35  0.25 0.45 0.55 2.3
## ObjectNumber           0.47  0.21 -0.01  0.39 0.41 0.59 2.4
## NumberFigure           0.52  0.32  0.16  0.14 0.41 0.59 2.1
## FigureWord             0.44  0.10  0.10  0.13 0.23 0.77 1.4
## Deduction              0.62 -0.13  0.14  0.04 0.42 0.58 1.2
## NumericalPuzzles       0.59  0.21  0.07 -0.14 0.42 0.58 1.4
## ProblemReasoning       0.61 -0.10  0.12  0.03 0.40 0.60 1.1
## SeriesCompletion       0.69 -0.06  0.15 -0.10 0.51 0.49 1.2
## ArithmeticProblems     0.65  0.17 -0.19  0.00 0.49 0.51 1.3
##
##                        MR1  MR2  MR3  MR4
## SS loadings           7.65 1.69 1.22 0.92
## Proportion Var        0.32 0.07 0.05 0.04
## Cumulative Var        0.32 0.39 0.44 0.48
## Proportion Explained  0.67 0.15 0.11 0.08
## Cumulative Proportion 0.67 0.81 0.92 1.00
##
## Mean item complexity =  1.9
## Test of the hypothesis that 4 factors are sufficient.
##
## The degrees of freedom for the null model are  276  and the objective function was  11.44
## The degrees of freedom for the model are 186  and the objective function was  1.72
##
## The root mean square of the residuals (RMSR) is  0.04
## The df corrected root mean square of the residuals is  0.05
##
## Fit based upon off diagonal values = 0.98
## Measures of factor score adequacy
##                                                    MR1  MR2  MR3  MR4
## Correlation of (regression) scores with factors   0.97 0.91 0.87 0.79
## Multiple R square of scores with factors          0.94 0.82 0.75 0.62
## Minimum correlation of possible factor scores     0.89 0.65 0.50 0.24
```

- **Rotate the factors**

```
fa(Herman,nfactors = 4,rotate = "varimax")
```

```
## Factor Analysis using method =  minres
## Call: fa(r = Herman, nfactors = 4, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##                         MR1   MR3   MR2  MR4   h2   u2 com
## VisualPerception       0.15  0.68  0.20 0.15 0.55 0.45 1.4
## Cubes                  0.11  0.45  0.08 0.08 0.23 0.77 1.3
## PaperFormBoard         0.15  0.55 -0.01 0.11 0.34 0.66 1.2
## Flags                  0.23  0.53  0.09 0.07 0.35 0.65 1.5
## GeneralInformation     0.73  0.19  0.22 0.14 0.64 0.36 1.4
## PargraphComprehension  0.76  0.21  0.07 0.23 0.68 0.32 1.4
## SentenceCompletion     0.81  0.19  0.15 0.07 0.73 0.27 1.2
## WordClassification     0.57  0.34  0.23 0.14 0.51 0.49 2.2
```

```
## WordMeaning            0.81  0.20  0.05 0.22 0.74 0.26 1.3
## Addition               0.17 -0.11  0.82 0.16 0.74 0.26 1.2
## Code                   0.18  0.11  0.54 0.37 0.47 0.53 2.1
## CountingDots           0.02  0.20  0.71 0.09 0.55 0.45 1.2
## StraightCurvedCapitals 0.18  0.42  0.54 0.08 0.51 0.49 2.2
## WordRecognition        0.21  0.05  0.08 0.56 0.36 0.64 1.3
## NumberRecognition      0.12  0.12  0.08 0.52 0.31 0.69 1.3
## FigureRecognition      0.07  0.42  0.06 0.52 0.45 0.55 2.0
## ObjectNumber           0.14  0.06  0.22 0.58 0.41 0.59 1.4
## NumberFigure           0.02  0.31  0.34 0.45 0.41 0.59 2.7
## FigureWord             0.15  0.25  0.18 0.35 0.23 0.77 2.8
## Deduction              0.38  0.42  0.10 0.29 0.42 0.58 2.9
## NumericalPuzzles       0.18  0.40  0.43 0.21 0.42 0.58 2.8
## ProblemReasoning       0.37  0.41  0.13 0.29 0.40 0.60 3.0
## SeriesCompletion       0.37  0.52  0.23 0.22 0.51 0.49 2.7
## ArithmeticProblems     0.36  0.19  0.49 0.29 0.49 0.51 2.9
##
##                        MR1  MR3  MR2  MR4
## SS loadings           3.64 2.93 2.67 2.23
## Proportion Var        0.15 0.12 0.11 0.09
## Cumulative Var        0.15 0.27 0.38 0.48
## Proportion Explained  0.32 0.26 0.23 0.19
## Cumulative Proportion 0.32 0.57 0.81 1.00
##
## Mean item complexity =  1.9
## Test of the hypothesis that 4 factors are sufficient.
##
## The degrees of freedom for the null model are  276  and the objective function was  11.44
## The degrees of freedom for the model are 186  and the objective function was  1.72
##
## The root mean square of the residuals (RMSR) is  0.04
## The df corrected root mean square of the residuals is  0.05
##
## Fit based upon off diagonal values = 0.98
## Measures of factor score adequacy
##                                                    MR1  MR3  MR2  MR4
## Correlation of (regression) scores with factors   0.93 0.87 0.91 0.82
## Multiple R square of scores with factors          0.87 0.76 0.83 0.68
## Minimum correlation of possible factor scores     0.74 0.52 0.65 0.36
```

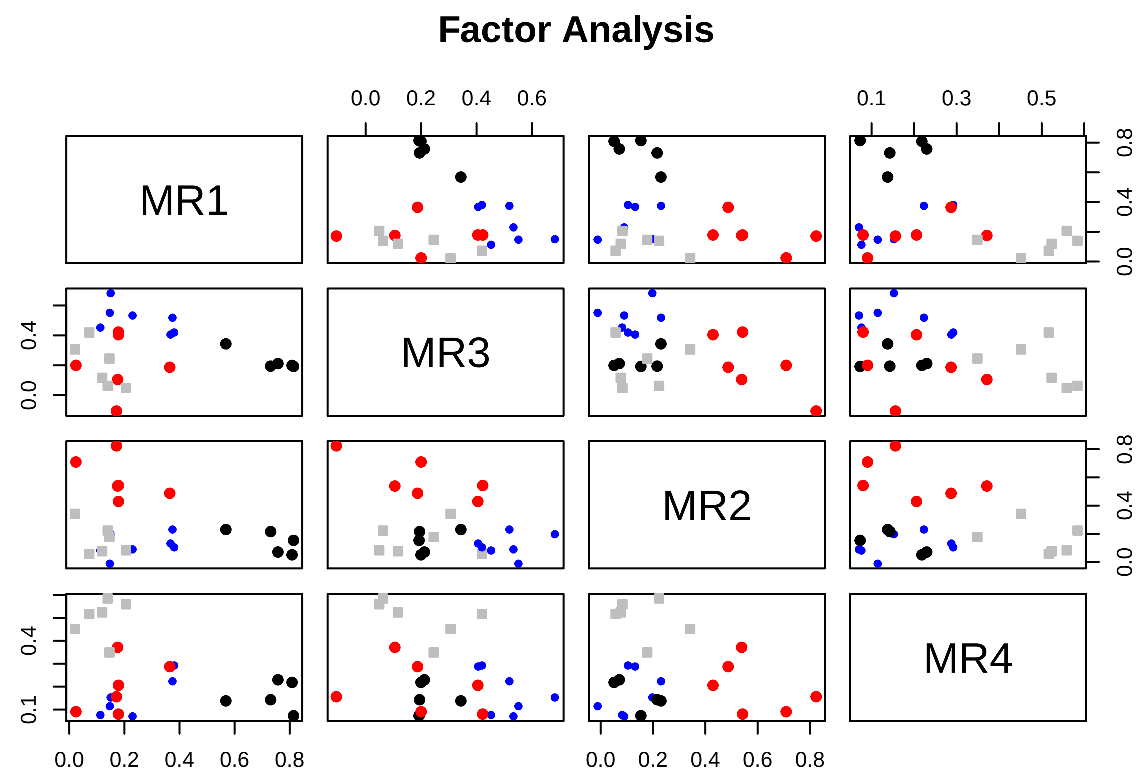
- **Compute factor scores**

```r
fa_Herman<-fa(Herman,nfactors = 4,rotate = "varimax",scores="regression")
fa_Herman$score.cor
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,] 1.0000000 0.6063244 0.4793939 0.4265543
## [2,] 0.6063244 1.0000000 0.5092860 0.5309466
## [3,] 0.4793939 0.5092860 1.0000000 0.5275798
## [4,] 0.4265543 0.5309466 0.5275798 1.0000000
```

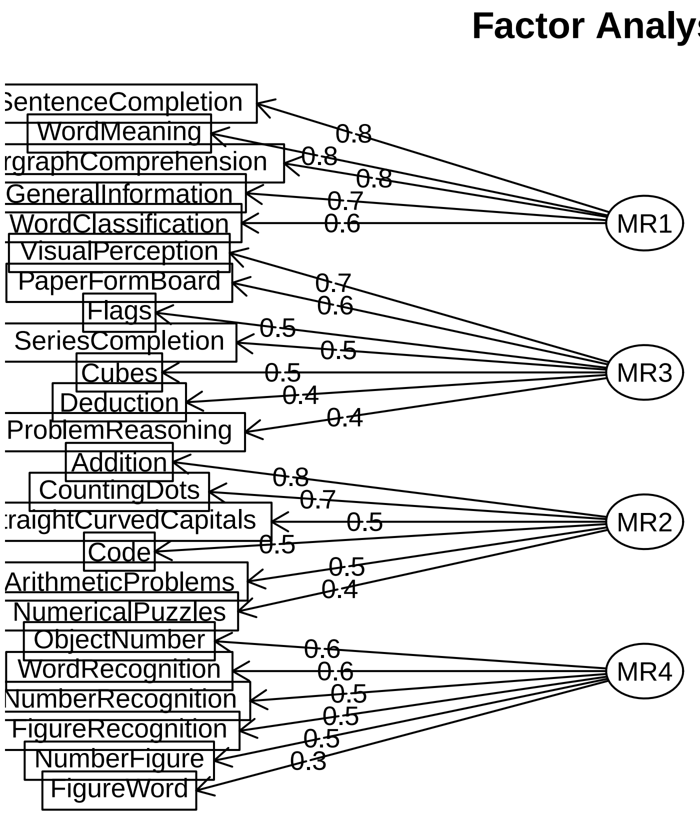
- **Graph an orthogonal solution using factor.plot()**

```
factor.plot(fa_Herman)
```



• Graph an oblique solutions using fa.diagram()

```
fa.diagram(fa_Herman)
```



• Interpret the results

Problem 4: Perform factor analysis on breast-cancer-wisconsin.xlsx, is a multivariate dataset that is used to predict whether a cancer is malignant or benign from biopsy details of 699 patients with 11 attributes. Create a new data frame by removing the variable "BN".

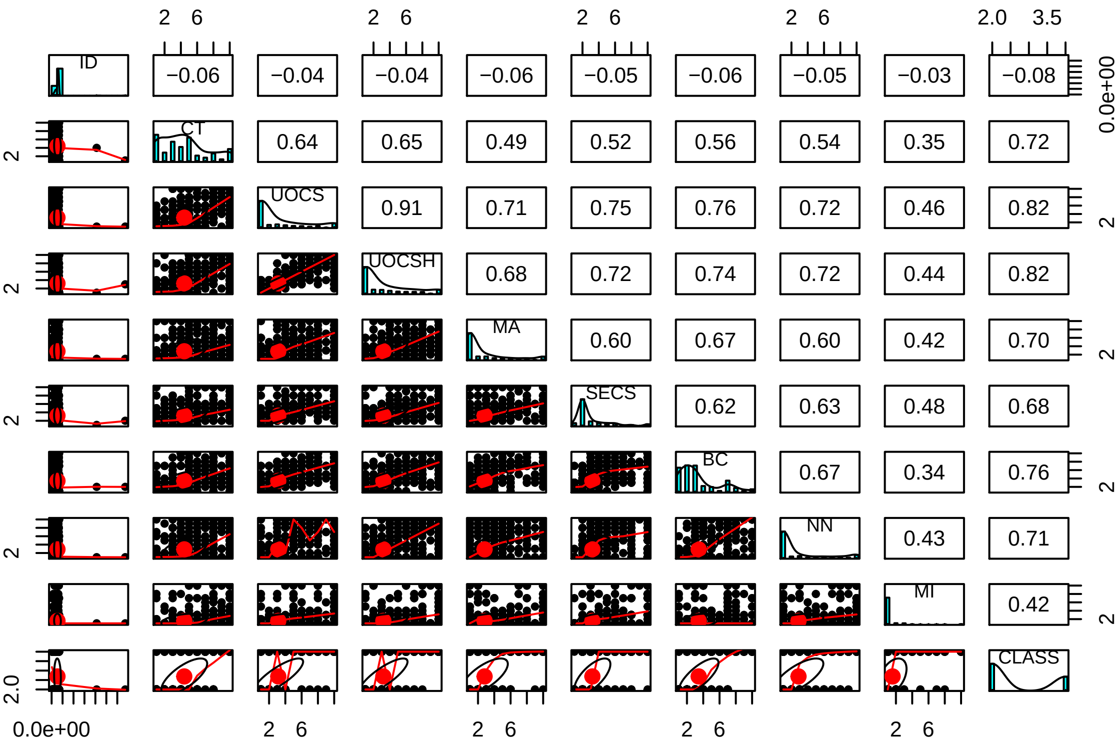
- **Calculate the correlation matrix from the new data frame. Visualize the correlation matrix using pairs.panels function of the "psych" package. How would you interpret the result in terms of correlation among the variables?**

```r
breast_cancer_wisconsin <- read_excel("breast-cancer-wisconsin.xlsx")
bcw_new<- data.frame(breast_cancer_wisconsin[-7])
cor(bcw_new)
```

```
##                ID          CT        UOCS       UOCSH          MA        SECS
## ID     1.00000000 -0.05530844 -0.04160334 -0.04157607 -0.06487808 -0.04552828
## CT    -0.05530844  1.00000000  0.64491250  0.65458908  0.48635624  0.52181622
## UOCS  -0.04160334  0.64491250  1.00000000  0.90688191  0.70558181  0.75179913
## UOCSH -0.04157607  0.65458908  0.90688191  1.00000000  0.68307920  0.71966844
## MA    -0.06487808  0.48635624  0.70558181  0.68307920  1.00000000  0.59959907
## SECS  -0.04552828  0.52181622  0.75179913  0.71966844  0.59959907  1.00000000
## BC    -0.06005053  0.55842816  0.75572098  0.73594845  0.66671533  0.61610184
```

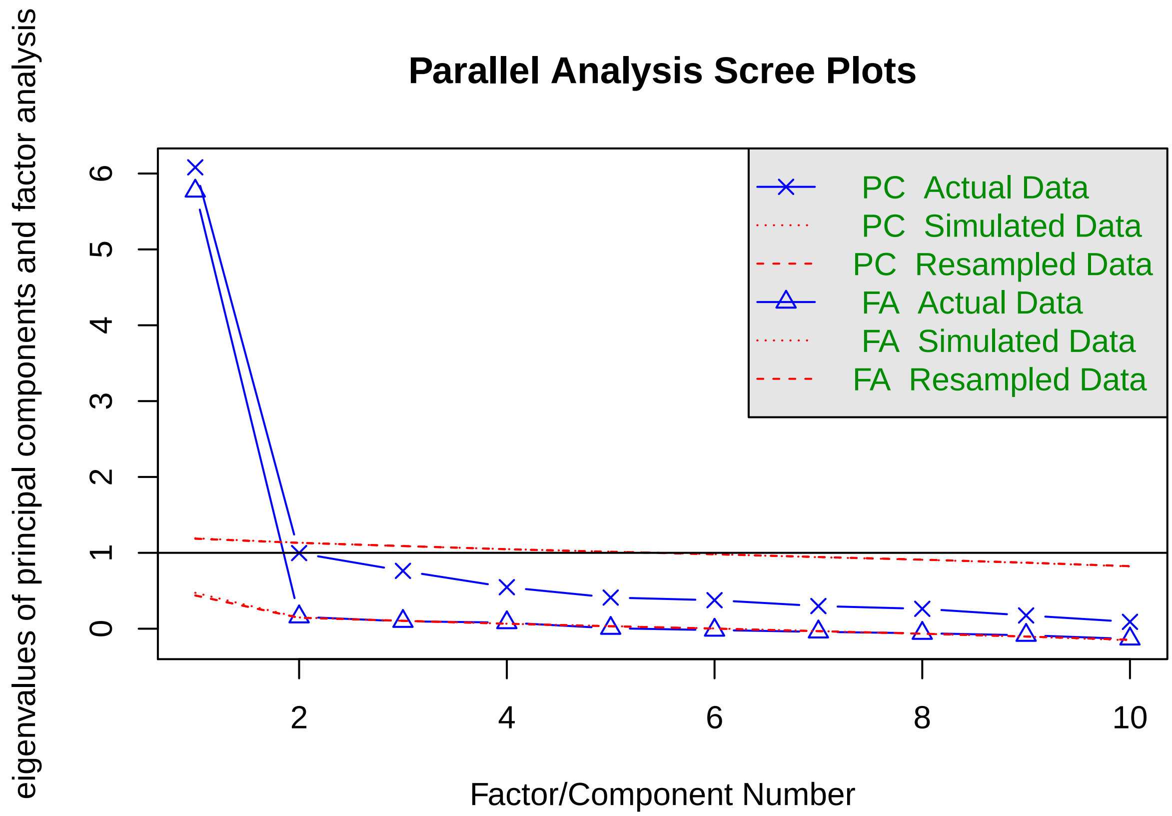
```
## NN    -0.05207195  0.53583455  0.72286482  0.71944632  0.60335241  0.62888069
## MI    -0.03490066  0.35003386  0.45869315  0.43891093  0.41763278  0.47910148
## CLASS -0.08022565  0.71600136  0.81790374  0.81893374  0.69680021  0.68278453
##                BC          NN          MI       CLASS
## ID    -0.06005053 -0.05207195 -0.03490066 -0.08022565
## CT     0.55842816  0.53583455  0.35003386  0.71600136
## UOCS   0.75572098  0.72286482  0.45869315  0.81790374
## UOCSH  0.73594845  0.71944632  0.43891093  0.81893374
## MA     0.66671533  0.60335241  0.41763278  0.69680021
## SECS   0.61610184  0.62888069  0.47910148  0.68278453
## BC     1.00000000  0.66587781  0.34416950  0.75661615
## NN     0.66587781  1.00000000  0.42833575  0.71224362
## MI     0.34416950  0.42833575  1.00000000  0.42317026
## CLASS  0.75661615  0.71224362  0.42317026  1.00000000
```

```
pairs.panels(bcw_new)
```



• Input the correlation matrix to fa.parallel() function to determine the number of components to extract

```
fa.parallel(bcw_new,n.iter=100)
```




```
## Parallel analysis suggests that the number of factors =  1  and the number of components =  1
```

- **Input the correlation matrix to fa() function to extract the components. If raw data is input, the correlation matrix is automatically calculated by fa() function.**

```
fa(bcw_new,nfactors = 1,rotate ="none")
```

```
## Factor Analysis using method =  minres
## Call: fa(r = bcw_new, nfactors = 1, rotate = "none")
## Standardized loadings (pattern matrix) based upon correlation matrix
##         MR1     h2   u2 com
## ID    -0.07 0.0044 1.00   1
## CT     0.70 0.4885 0.51   1
## UOCS   0.93 0.8722 0.13   1
## UOCSH  0.92 0.8435 0.16   1
## MA     0.76 0.5837 0.42   1
## SECS   0.79 0.6168 0.38   1
## BC     0.81 0.6614 0.34   1
## NN     0.79 0.6263 0.37   1
## MI     0.50 0.2532 0.75   1
## CLASS  0.91 0.8200 0.18   1
##
##                 MR1
```

```
## SS loadings    5.77
## Proportion Var 0.58
##
## Mean item complexity =  1
## Test of the hypothesis that 1 factor is sufficient.
##
## The degrees of freedom for the null model are  45  and the objective function was  7.64 with Chi Squa
## The degrees of freedom for the model are 35  and the objective function was  0.36
##
## The root mean square of the residuals (RMSR) is  0.03
## The df corrected root mean square of the residuals is  0.03
##
## The harmonic number of observations is  699 with the empirical chi square  50.36  with prob <  0.045
## The total number of observations was  699  with Likelihood Chi Square =  247.9  with prob <  6.9e-34
##
## Tucker Lewis Index of factoring reliability =  0.948
## RMSEA index =  0.093  and the 90 % confidence intervals are  0.083 0.104
## BIC =  18.66
## Fit based upon off diagonal values = 1
## Measures of factor score adequacy
##                                                    MR1
## Correlation of (regression) scores with factors   0.98
## Multiple R square of scores with factors          0.96
## Minimum correlation of possible factor scores     0.92
```

- **Rotate the factors**

```
fa(bcw_new,nfactors = 1,rotate ="varimax")
```

```
## Factor Analysis using method =  minres
## Call: fa(r = bcw_new, nfactors = 1, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##         MR1     h2   u2 com
## ID    -0.07 0.0044 1.00   1
## CT     0.70 0.4885 0.51   1
## UOCS   0.93 0.8722 0.13   1
## UOCSH  0.92 0.8435 0.16   1
## MA     0.76 0.5837 0.42   1
## SECS   0.79 0.6168 0.38   1
## BC     0.81 0.6614 0.34   1
## NN     0.79 0.6263 0.37   1
## MI     0.50 0.2532 0.75   1
## CLASS  0.91 0.8200 0.18   1
##
##                 MR1
## SS loadings    5.77
## Proportion Var 0.58
##
## Mean item complexity =  1
## Test of the hypothesis that 1 factor is sufficient.
##
## The degrees of freedom for the null model are  45  and the objective function was  7.64 with Chi Squa
```

```
## The degrees of freedom for the model are 35  and the objective function was  0.36
##
## The root mean square of the residuals (RMSR) is  0.03
## The df corrected root mean square of the residuals is  0.03
##
## The harmonic number of observations is  699 with the empirical chi square  50.36  with prob <  0.045
## The total number of observations was  699  with Likelihood Chi Square =  247.9  with prob <  6.9e-34
##
## Tucker Lewis Index of factoring reliability =  0.948
## RMSEA index =  0.093  and the 90 % confidence intervals are  0.083 0.104
## BIC =  18.66
## Fit based upon off diagonal values = 1
## Measures of factor score adequacy
##                                                    MR1
## Correlation of (regression) scores with factors   0.98
## Multiple R square of scores with factors          0.96
## Minimum correlation of possible factor scores     0.92
```

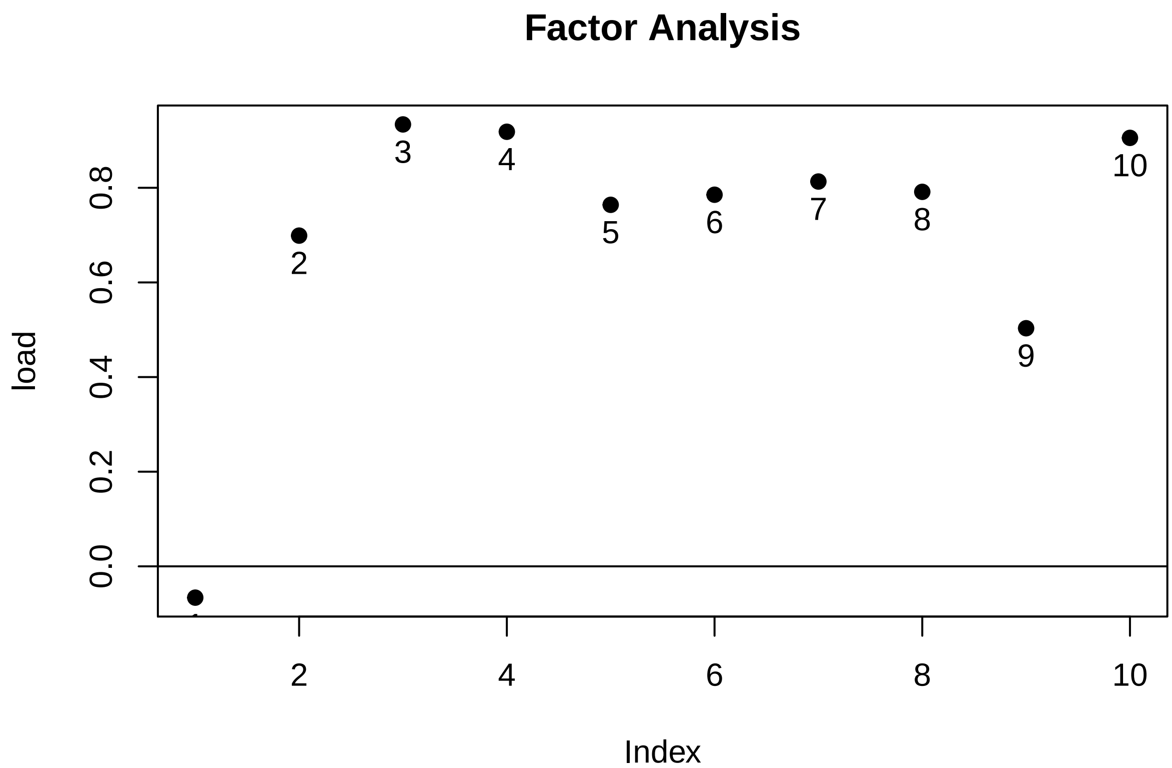
- **Compute factor scores**

```
fa_bcw<-fa(bcw_new,nfactors = 1,rotate ="varimax",scores="regression")
head(fa_bcw$scores)
```

```
##              MR1
## [1,] -0.68482609
## [2,]  0.09740335
## [3,] -0.70518849
## [4,]  0.62466302
## [5,] -0.64382383
## [6,]  2.01615740
```

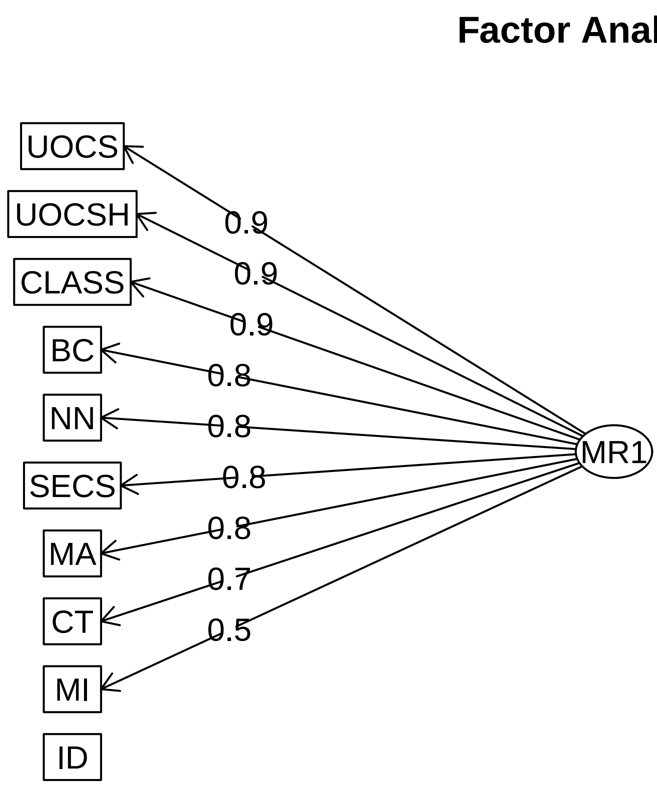
- **Graph an orthogonal solution using factor.plot()**

```
factor.plot(fa_bcw)
```



• Graph an oblique solutions using fa.diagram()

```
fa.diagram(fa_bcw)
```

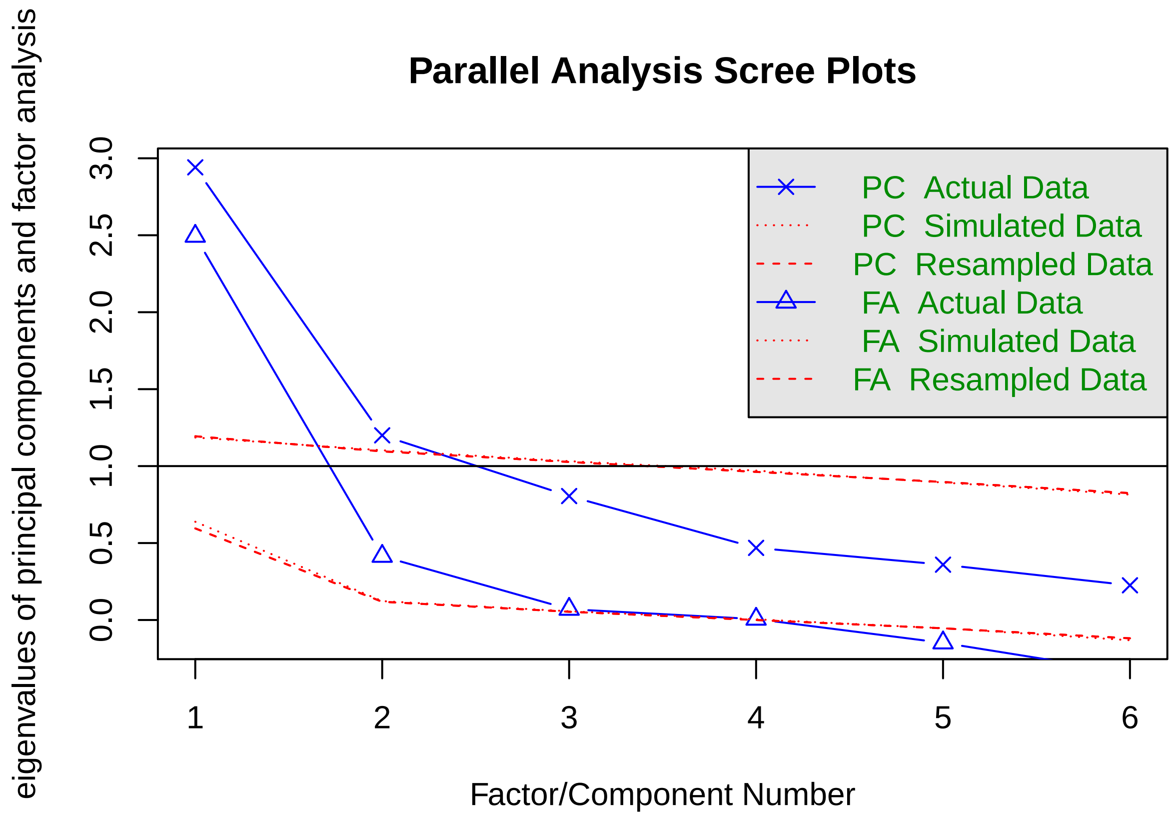


• Interpret the results

Problem 5. Perform multidimensional scaling on Vertebral Column Data.xlsx

• Input the raw data matrix to fa.parallel() function to determine the number of components to extract

```
Vertebral_Column_Data <- read_excel("Vertebral Column Data.xlsx")
vcd<-data.matrix(Vertebral_Column_Data)
fa.parallel(vcd[,-1],n.iter=100)
```




```
## Parallel analysis suggests that the number of factors =  2  and the number of components =  2
```

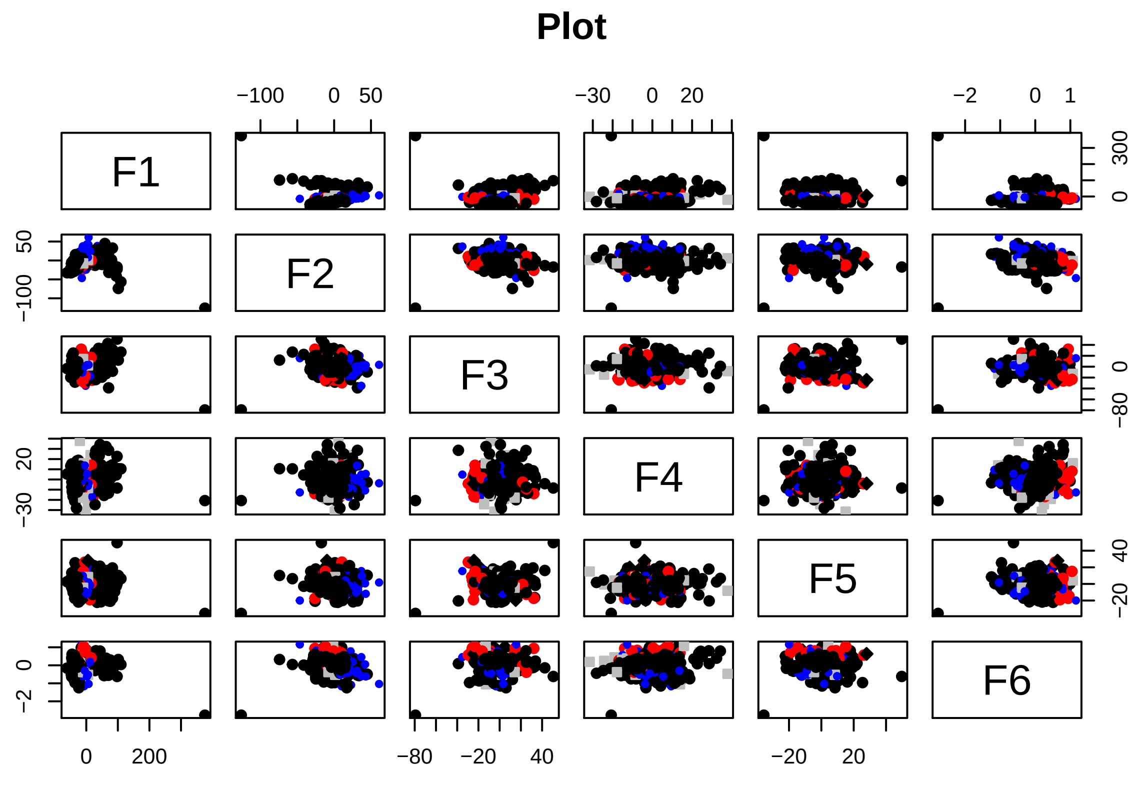
- **Input the raw data matrix to cmdscale() function to perform multidimensional scaling.cmdscale() function which is available in the base installation performs a classical multidimensional scaling.**

```
dis<-dist(vcd)
scl<-cmdscale(dis,k=6)
head(scl)
```

```
##           [,1]       [,2]       [,3]      [,4]      [,5]       [,6]
## [1,] -25.22624  13.193646 -15.893402 14.105517  1.902267 -0.8762699
## [2,] -37.56373 -18.958030 -11.839644  2.127067  2.317858 -0.6112231
## [3,] -21.96435  23.052560  -6.320437  8.965257 -2.519092 -1.0921009
## [4,] -10.86172  13.905335 -12.973047 13.920430 -1.139001 -1.0558190
## [5,] -27.74691  -7.597454 -18.436229  1.456963 -2.934478 -0.8296567
## [6,] -39.76141 -22.965722   2.545123  3.753434 -4.883766 -0.5188790
```

- **Graph an orthogonal solution using factor.plot()**

```r
factor.plot(scl)
```



• Interpret the results