Factorial Analysis

```r
library(fpp)

library(fpp2)

library(psych)

adcm = data_clean[c(7,8,11,13,14)]

#first we see the correlation coefficient
cor(dcm, method = c("pearson", "kendall", "spearman"))

##             nrgy        dnce        val      acous        spch
## nrgy   1.0000000  0.16685024  0.4102908 -0.5625564  0.10711812
## dnce   0.1668502  1.00000000  0.5049296 -0.2413363 -0.02922118
## val    0.4102908  0.50492963  1.0000000 -0.2486811  0.12284677
## acous -0.5625564 -0.24133632 -0.2486811  1.0000000  0.00246410
## spch   0.1071181 -0.02922118  0.1228468  0.0024641  1.00000000

fit.pc <- principal(dcm, nfactors=3, rotate="varimax")
fit.pc

## Principal Components Analysis
## Call: principal(r = dcm, nfactors = 3, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##         RC1   RC3   RC2   h2    u2 com
## nrgy   0.87  0.17  0.14 0.80 0.201 1.1
## dnce   0.06  0.90 -0.12 0.82 0.180 1.0
## val    0.27  0.81  0.19 0.76 0.243 1.3
## acous -0.87 -0.12  0.10 0.79 0.214 1.1
## spch   0.02  0.02  0.98 0.96 0.039 1.0

##                        RC1  RC3  RC2
## SS loadings           1.59 1.50 1.04
## Proportion Var        0.32 0.30 0.21
## Cumulative Var        0.32 0.62 0.82
## Proportion Explained  0.39 0.36 0.25
## Cumulative Proportion 0.39 0.75 1.00
##
## Mean item complexity =  1.1
## Test of the hypothesis that 3 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.11
##  with the empirical chi square  136.72  with prob <  NA
##
## Fit based upon off diagonal values = 0.88

round(fit.pc$values, 3)

## [1] 2.085 1.036 1.002 0.543 0.335
```

```r
fit.pc$loadings
```

```
##
## Loadings:
##       RC1    RC3    RC2
## nrgy   0.865  0.172  0.143
## dnce          0.896 -0.121
## val    0.272  0.805  0.187
## acous -0.873 -0.123
## spch                 0.980
##
##                  RC1   RC3   RC2
## SS loadings    1.588 1.495 1.040
## Proportion Var 0.318 0.299 0.208
## Cumulative Var 0.318 0.617 0.825
```

```r
#loading for more digits
```

```r
for (i in c(1,3,2)) { print(fit.pc$loadings[[1,i]])}
```

```
## [1] 0.8653062
## [1] 0.1430084
## [1] 0.1720101
```

```r
#Checking for communalities
```

```r
fit.pc$communality
```

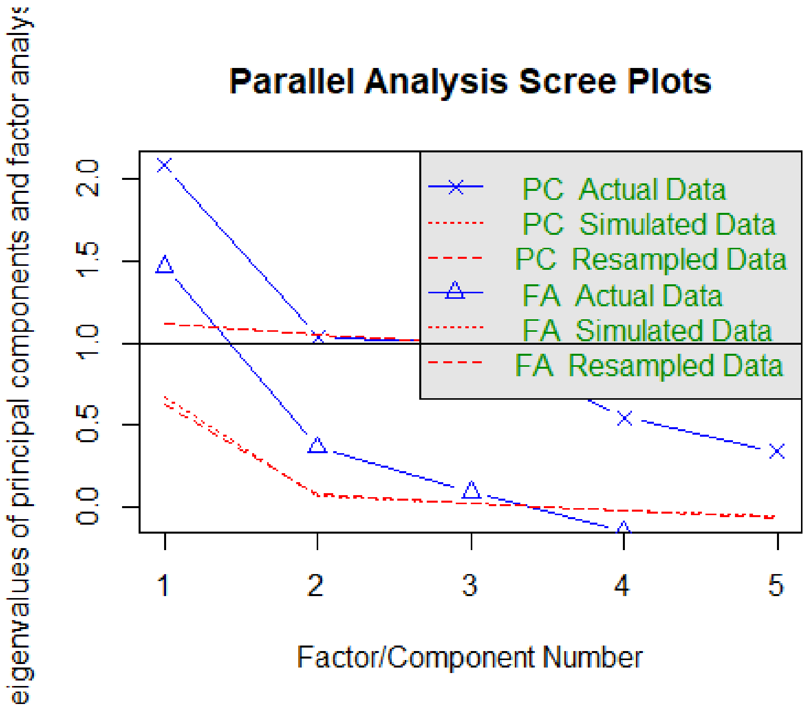
```
##      nrgy      dnce       val     acous      spch
## 0.7987937 0.8197369 0.7572138 0.7859786 0.9610781
```

```r
#Rotated factor scores and notice the column ordering RC1, RC3 RC2
```

```r
fit.pc$scores
```

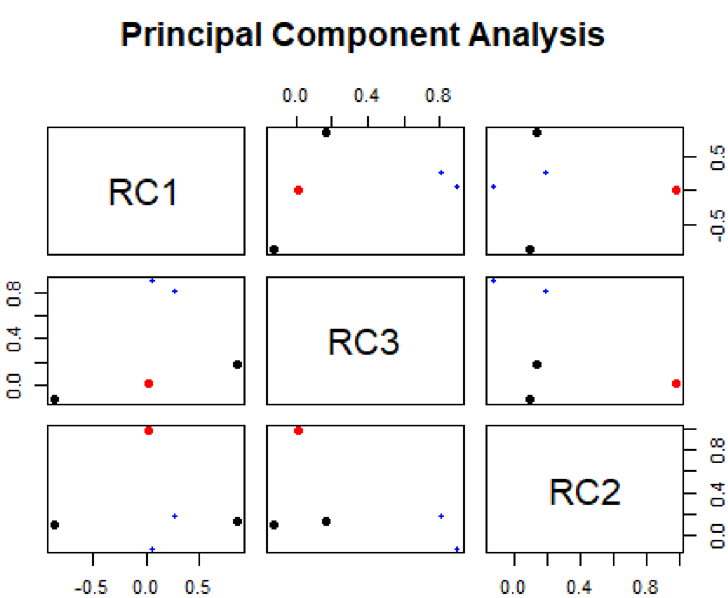
```
##              RC1          RC3           RC2
## 1    0.480146327  0.731330706 -0.2717584130
## 2    0.271778850  0.708318310  2.0013916128
## 3    0.395461562  0.915640929  0.7507024925
## 4    1.099274320  0.539066153 -0.4767320380
## 5    0.861779252 -0.373076834 -0.6191925696
```

```r
fa.parallel(dcm) # See factor recommendation
```

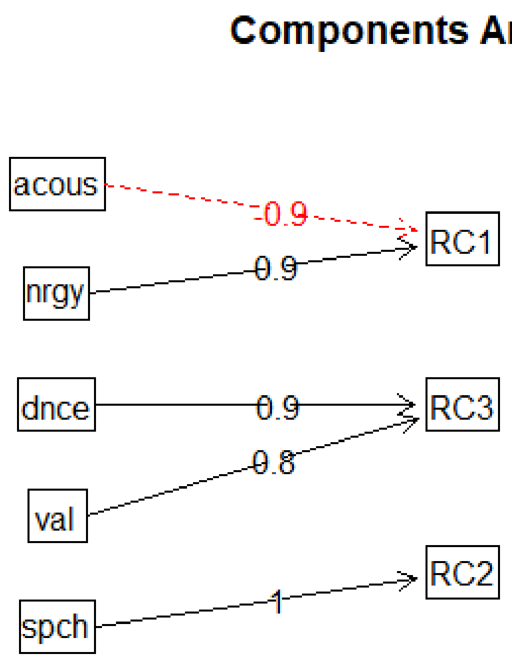



```
## Parallel analysis suggests that the number of factors = 3 and the number
of components = 1
```

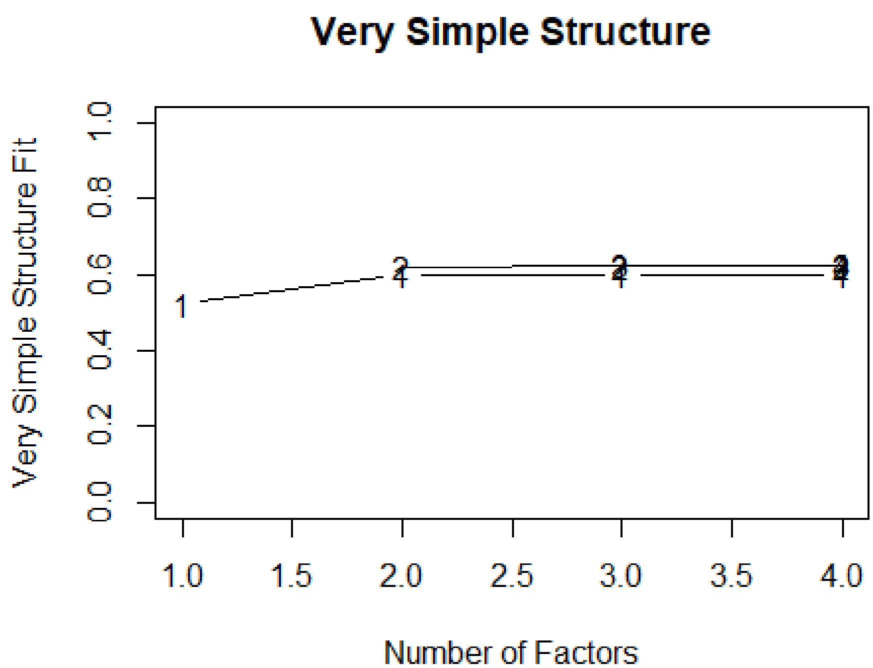
```r
fa.plot(fit.pc) # See Correlations within Factors
```



```r
fa.diagram(fit.pc) # Visualize the relationship
```




```r
vss(dcm[-1]) # See Factor recommendations for a simple structure
```




```
## Very Simple Structure
## Call: vss(x = dcm[-1])
## VSS complexity 1 achieves a maximum of 0.6  with  3  factors
## VSS complexity 2 achieves a maximum of 0.62  with  3  factors
##
## The Velicer MAP achieves a minimum of NA  with  1  factors
## BIC achieves a minimum of  NA  with  1  factors
## Sample Size adjusted BIC achieves a minimum of  NA  with  1
factors
##
## Statistics by number of factors
##   vss1 vss2  map dof   chisq    prob sqresid  fit RMSEA BIC SABIC
complex
## 1 0.52 0.00 0.13   2 1.4e+01 0.00093     2.3 0.52   0.1 1.2   7.5
1.0
## 2 0.60 0.62 0.37  -1 3.5e-08      NA     1.8 0.62    NA  NA    NA
1.1
## 3 0.60 0.62 1.00  -3 0.0e+00      NA     1.8 0.62    NA  NA    NA
1.1
## 4 0.60 0.62   NA  -4 0.0e+00      NA     1.8 0.62    NA  NA    NA
1.1
##    eChisq    SRMR eCRMS eBIC
## 1 1.4e+01 4.4e-02 0.076  1.1
## 2 2.6e-08 1.9e-06    NA   NA
```

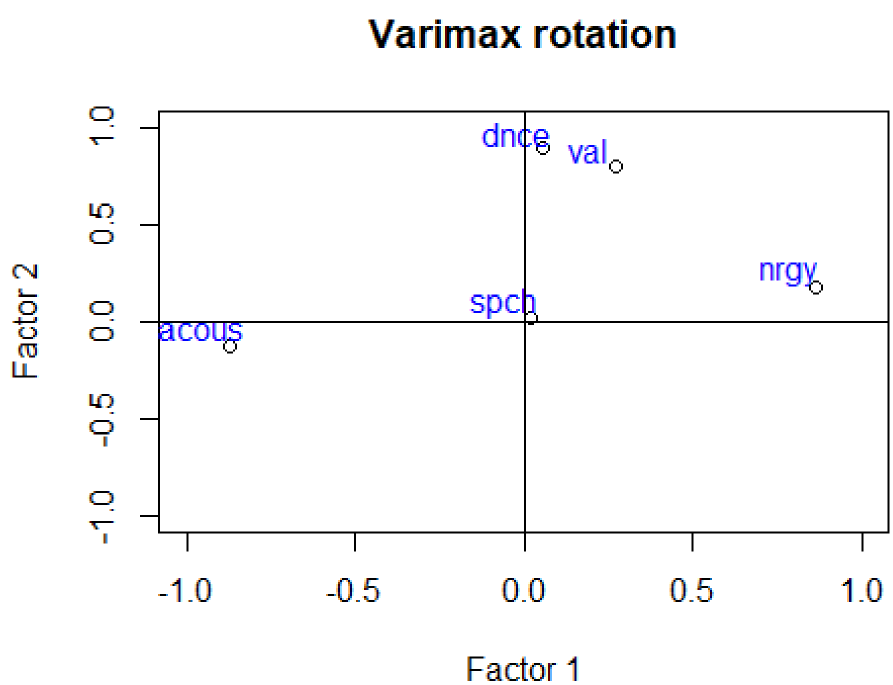
```
## 3 2.7e-15 6.1e-10    NA    NA
## 4 2.7e-15 6.1e-10    NA    NA
```

```
fit.pc$loadings[,1]
```

```
##        nrgy        dnce         val       acous        spch
##  0.86530620  0.05635188  0.27211868 -0.87273678  0.01956093
```

```
plot(fit.pc$loadings[,1],
     fit.pc$loadings[,2],
     xlab = "Factor 1",
     ylab = "Factor 2",
     ylim = c(-1,1),
     xlim = c(-1,1),
     main = "Varimax rotation")
abline(h = 0, v = 0)

text(fit.pc$loadings[,1]-0.08,
     fit.pc$loadings[,2]+0.08,
     colnames(dcm),
     col="blue")
abline(h = 0, v = 0)
```



As we can observe in component analysis that acoustic and energy have unique variance of 20% for RC1 component

And RC3 component has 30% of unique variance from danceability and valence, however speechability column forms a single component RC2 and we have not done any analysis on this column, but that column cannot be removed as it forms a whole component.

From Explanatory data analysis we found out that we will require all the audio properties and hence, we cannot delete any attribute related to it from our dataset.

Factorial analysis does not apply to our dataset, as there is very low correlation among the attributes and hence using factorial analysis, we cannot select any component which can reduce our attributes.