Exploratory Factor Analysis in R for MBA Students

Jovita Monteiro*
Nikitha Fernandes**
Mohsin Ahmed***

Abstract

In this paper we show how exploratory factor analysis using R can be used as educational language and research tool. Despite the use of exploratory factor analysis in research, researchers make questionable decisions when conducting analysis. This article provides some practical usage patterns and insights as to how analysis can be conducted using R. R can ease the data analysis work and also it can handle complex data. Survey data of students set is used to illustrate factor analysis with R.

Keywords: Factor analysis, pedagogy, R, statistics, Survey analysis.

INTRODUCTION

R is a powerful language for statistical computing and graphics. R is much used as an educational language and research tool. It is a very flexible and powerful language that many data analyst use. One of the main reasons why R has been gaining prominence is because of its free and flexible nature. One of the reason as to why R should be used when compared with other data analysis packages is that it is very powerful programming language that is able to conduct a wide range of analysis. Thus for many projects all analysis can be conducted within the same program. This is particularly useful for factor analytic work as one can examine the data's, properties, conduct factor analysis without having to export data. Once the data has been entered into R it will be available for use until the R programme is closed. Second reason to use R is that it allows the users to submit their own modifications to the R server for anyone to use and it is open source programming language available for most operating systems and it can be transportable from one system to another. This article gives a general introduction to using R in conducting factor analysis from an exploratory approach.

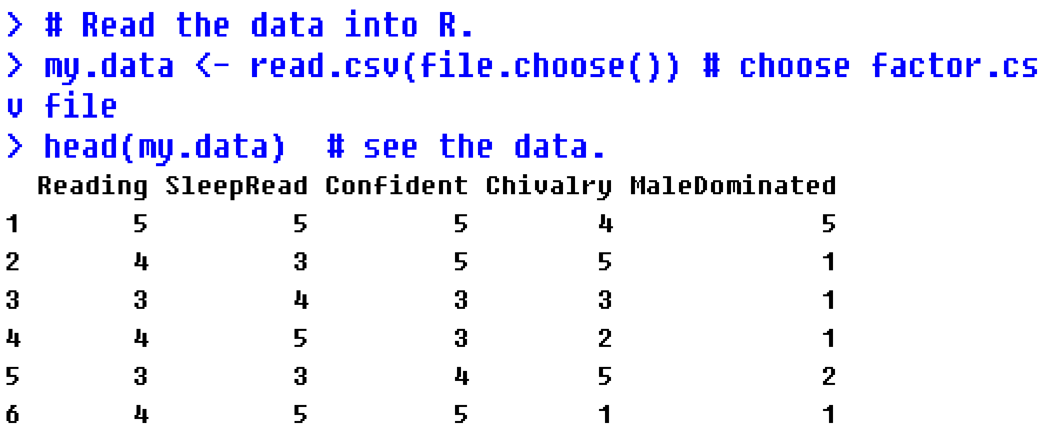
Factor analysis is a statistical method used to describe variability among observed, correlated variables in terms of a potentially lower number of unobserved variables called factors. Factor analysis is used for reducing the data to a smaller set and it can be used for exploring an empirically testing a factor. It can also be used to determine the number of factors. R allows complete control on its factor analysis methods. Factor analysis involves two types among which exploratory factor analysis is a common technique for explaining the variance between several measured variables as a smaller set of latent variables. It is often used to consolidate survey data. Some of the examples where factor analysis can be practically applicable are, in the field of advertising it can be used to better understand media habits of various customers, it can help to identify the characteristics of price sensitive customers, Factor analysis can also be used to identify brand attributes that influence customer choice, it can also be used to better understand channel selection criteria among distribution channel members.

*Student, Justice K S Hegde Institute of Management, NITTE, India jovita.monteiro3@gmail.com
**Student, Justice K S Hegde Institute of Management, NITTE, India jacklineniks@gmail.com
***Professor, Justice K S Hegde Institute of Management, NITTE, India moshahmed@gmail.coom

DETAILS SECTION

A practical example of conducting an exploratory factor analysis using R is illustrated below. This data set contains 78 student's responses on 55 items from a survey of college students learning habits. The data is from Kavya et al (2014). The response range is the likert scale from 1-5, representing a scale from *strongly dislike* to *strongly like*, with 3 being *neutral* response. To begin we need to read data set into R and store its contents in variables.

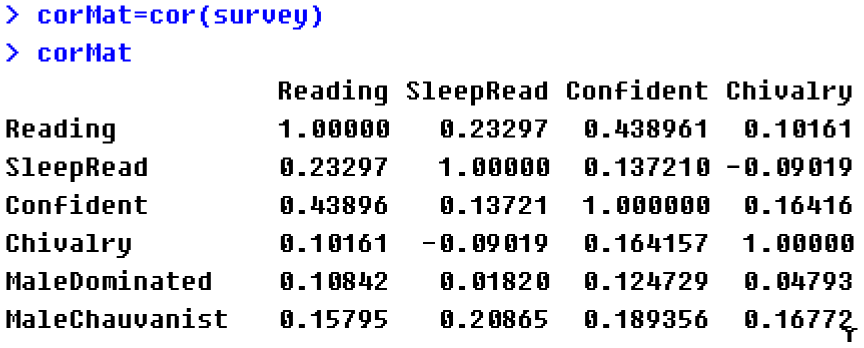
# read the dataset into R variable using the read.csv(file) function
data <- read.csv("survey.csv")



```
> # Read the data into R.
> my.data <- read.csv(file.choose()) # choose factor.cs
v file
> head(my.data)  # see the data.
  Reading SleepRead Confident Chivalry MaleDominated
1       5         5         5        4             5
2       4         3         5        5             1
3       3         4         3        3             1
4       4         5         3        2             1
5       3         3         4        5             2
6       4         5         5        1             1
```

# Large amount of data now shown

Correlation
# calculate the correlation matrix

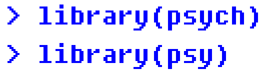


```
> corMat=cor(survey)
> corMat
                 Reading SleepRead Confident Chivalry
Reading          1.00000   0.23297  0.438961  0.10161
SleepRead        0.23297   1.00000  0.137210 -0.09019
Confident        0.43896   0.13721  1.000000  0.16416
Chivalry         0.10161  -0.09019  0.164157  1.00000
MaleDominated    0.10842   0.01820  0.124729  0.04793
MaleChauvanist   0.15795   0.20865  0.189356  0.16772
```

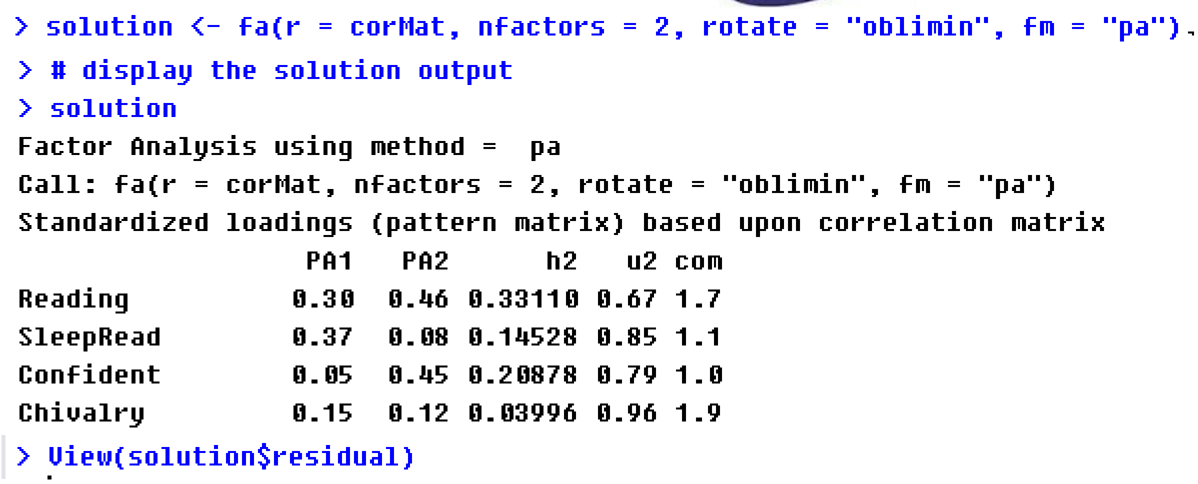
# Large amount of data now shown

Correlation Matrix

# use fa() to conduct an oblique principal-axis
# exploratory factor analysis
# save the solution to an R variable

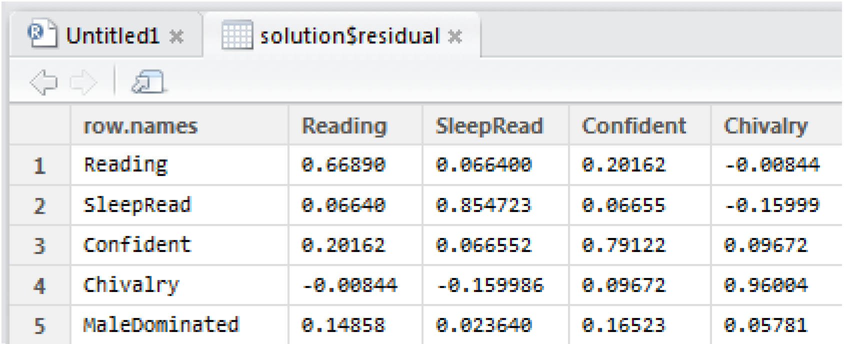


```
> library(psych)
> library(psy)
```



```
> solution <- fa(r = corMat, nfactors = 2, rotate = "oblimin", fm = "pa")
> # display the solution output
> solution
Factor Analysis using method =  pa
Call: fa(r = corMat, nfactors = 2, rotate = "oblimin", fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
                  PA1   PA2      h2   u2 com
Reading          0.30  0.46 0.33110 0.67 1.7
SleepRead        0.37  0.08 0.14528 0.85 1.1
Confident        0.05  0.45 0.20878 0.79 1.0
Chivalry         0.15  0.12 0.03996 0.96 1.9
> View(solution$residual)
```

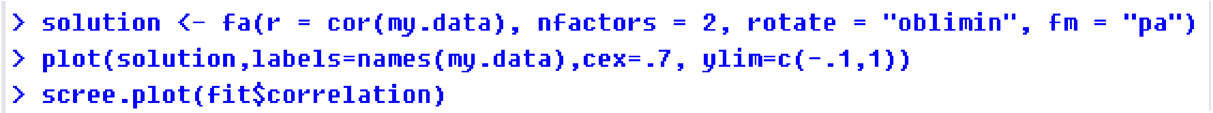
# Large amount of data now shown



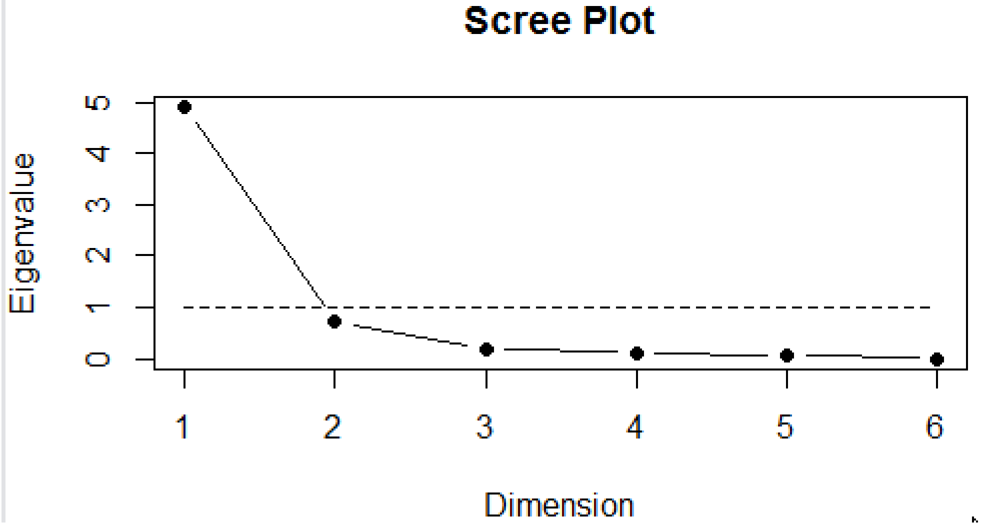
	row.names	Reading	SleepRead	Confident	Chivalry
1	Reading	0.66890	0.066400	0.20162	-0.00844
2	SleepRead	0.06640	0.854723	0.06655	-0.15999
3	Confident	0.20162	0.066552	0.79122	0.09672
4	Chivalry	-0.00844	-0.159986	0.09672	0.96004
5	MaleDominated	0.14858	0.023640	0.16523	0.05781

# Large amount of data now shown

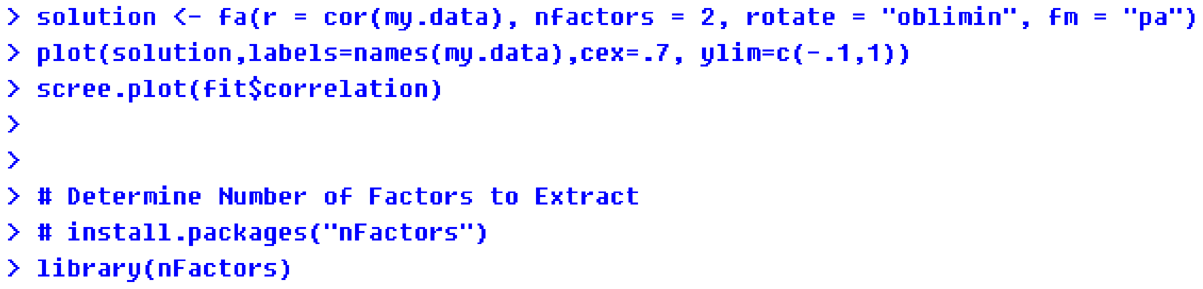
Scree plots



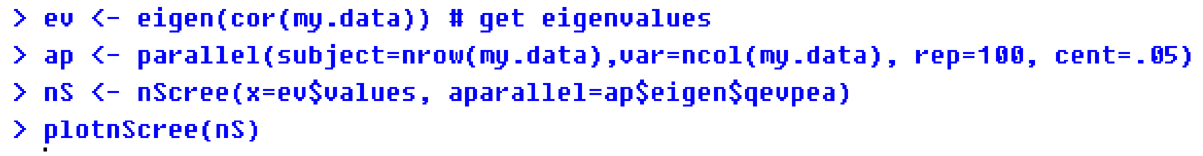
```
> solution <- fa(r = cor(my.data), nfactors = 2, rotate = "oblimin", fm = "pa")
> plot(solution,labels=names(my.data),cex=.7, ylim=c(-.1,1))
> scree.plot(fit$correlation)
```



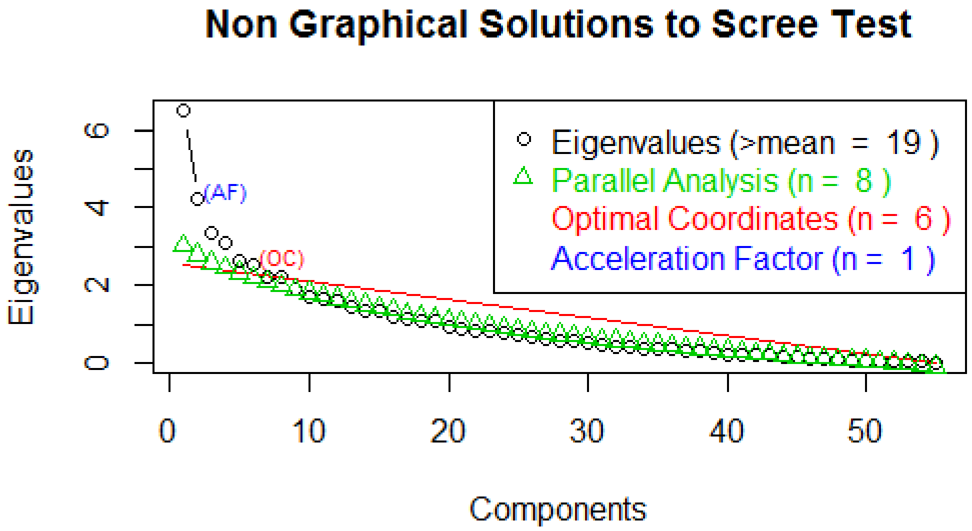
With nfactor=2, we find that all questions have factor loadings around 0.7 or less on the first factor (PA1). So we use automatic scree plot to determine the number of factors needed.



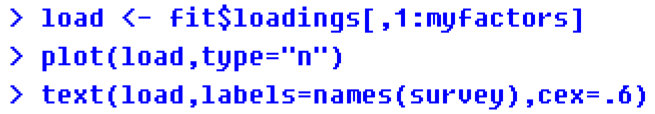
```
> solution <- fa(r = cor(my.data), nfactors = 2, rotate = "oblimin", fm = "pa")
> plot(solution,labels=names(my.data),cex=.7, ylim=c(-.1,1))
> scree.plot(fit$correlation)
>
>
> # Determine Number of Factors to Extract
> # install.packages("nFactors")
> library(nFactors)
```



```
> ev <- eigen(cor(my.data)) # get eigenvalues
> ap <- parallel(subject=nrow(my.data),var=ncol(my.data), rep=100, cent=.05)
> nS <- nScree(x=ev$values, aparallel=ap$eigen$qevpea)
> plotnScree(nS)
```

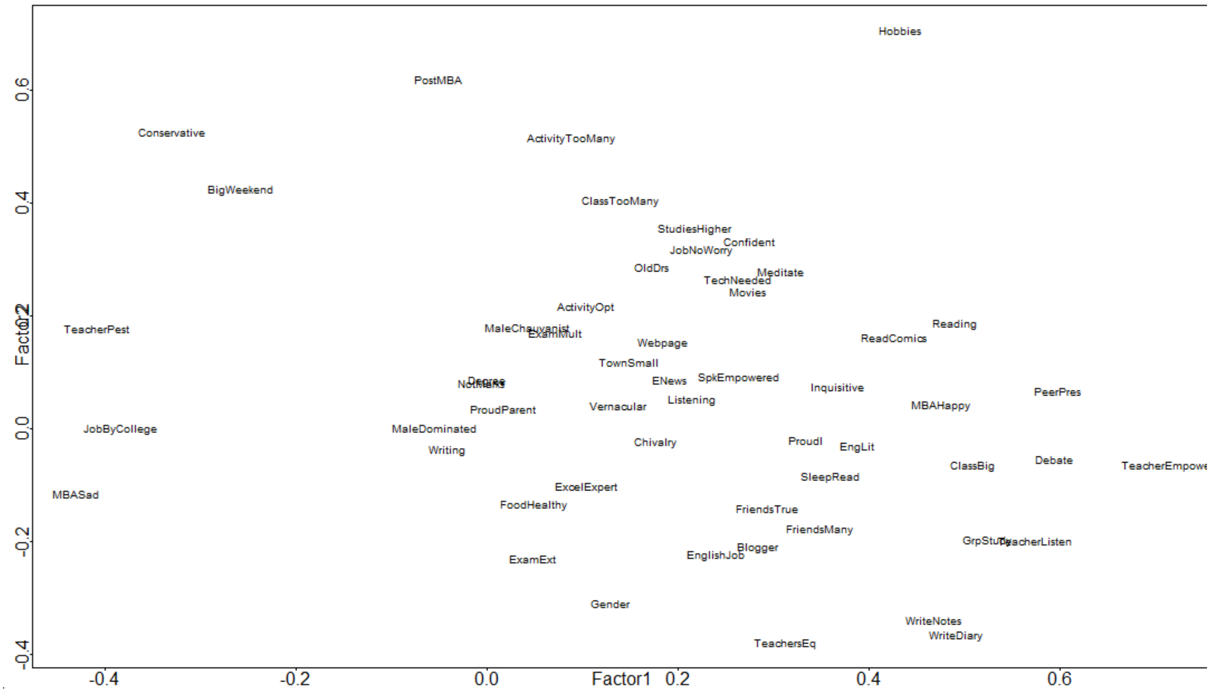


We need at least 6 factors to proceed.



```
> load <- fit$loadings[,1:myfactors]
> plot(load,type="n")
> text(load,labels=names(survey),cex=.6)
```

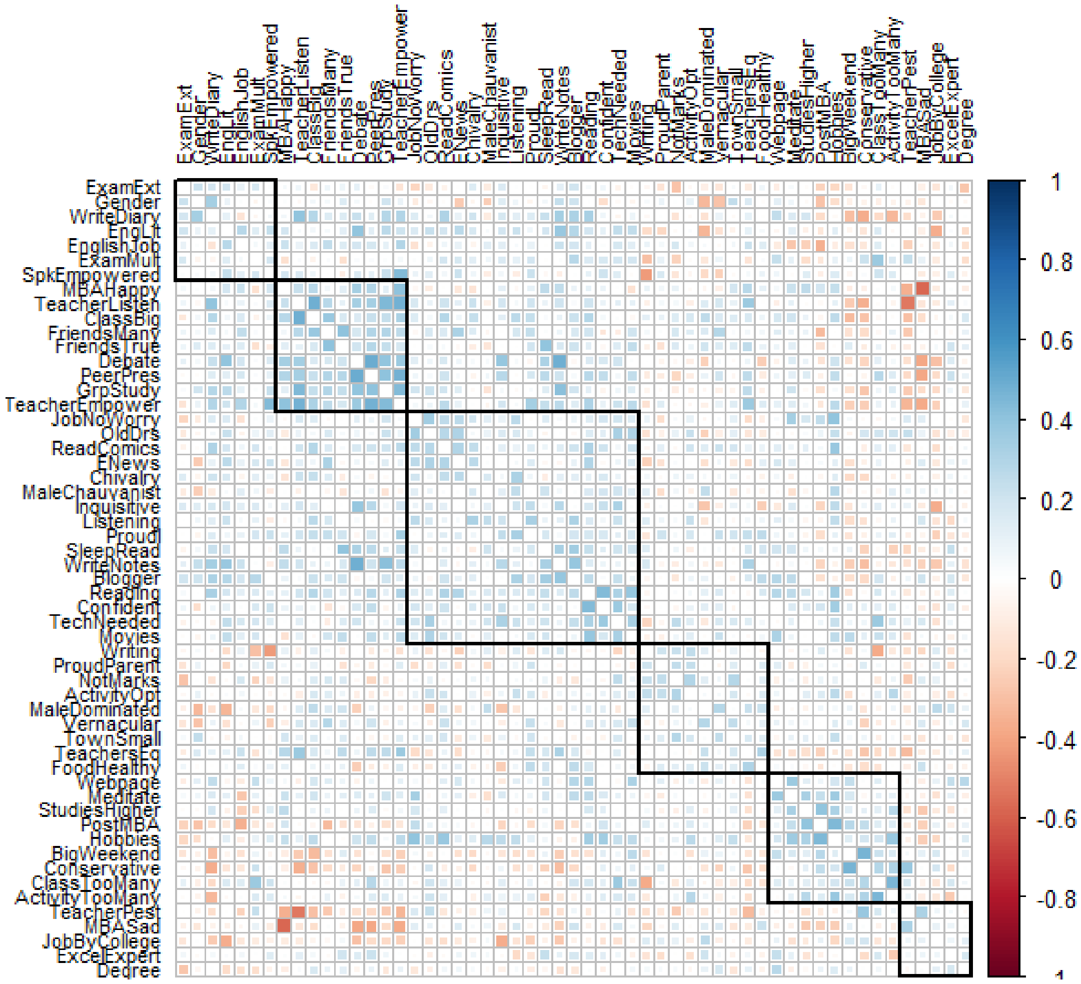
Now we can look at the distribution of factors in the following plot. However it still doesn’t provide a suitable category of questions, so next we resort to correlation analysis.


Hobbies
PostMBA
Conservative
ActivityTooMany
BigWeekend
ClassTooMany
StudiesHigher
Confident
JobNoWorry
OldDrs
Meditate
TechNeeded
Movies
ActivityOpt
TeacherPest
MaleChauvanist
ExamMult
Reading
Webpage
ReadComics
TownSmall
SpkEmpowered
ENews
Degree
NotMarks
Inquisitive
PeerPres
Listening
MBAHappy
Vernacular
ProudParent
JobByCollege
MaleDominated
Chivalry
ProudI
EngLit
Writing
Debate
ClassBig
TeacherEmpowe
SleepRead
MBASad
ExcelExpert
FoodHealthy
FriendsTrue
FriendsMany
GrpStudy
TeacherListen
Blogger
EnglishJob
ExamExt
Gender
WriteNotes
WriteDiary
TeachersEq
0.6
0.4
0.2
0.0
-0.2
-0.4
Factor2
-0.4
-0.2
0.0
Factor1
0.2
0.4
0.6

Correlating the questions

We try correlation again with 6 factors. We get strongly correlated questions boxed together in the corrplot.

```
> corrplot(cormat, method = "square", tl.cex=.7, tl.col='black', diag
=F,type='full',order="hclust", addrect=myfactors)
```



This plot suggests we can group the questions into several categories: Studies, Extra curricular activities, Placements, Listening and speaking, Reading and writing. We see that many questions could not be grouped cleanly as textbook example would suggest. For example, gender sensitivity seems to be an outlier, not falling cleanly into one category. We hypothesize that more questions are needed for a clearer insight into the student learning habits.

CONCLUSION

With the many steps involved in factor analysis it can be difficult finding a program that does everything that one may desire. While R does not have a function for everything involved in factor analysis, it does have many including those for determination of number of factors as well as confirmatory methods. It has a large library of packages and online community to do such analysis. This alone sets it apart from many other programming options currently available.

Acknowledgements
We thank our Director Prof. Sankaran and Prof. Krishna Prasad for the encouragement and help in preparing the paper. We thank our colleagues who collected and processed the student survey data (Kavya 2014).

References

Kavya M N et al, "*A Statistical Approach to Modernize the Indian Higher Education System for Rural and Vernacular Students*", NITTE International Conference, India, 2014.

Factor Analysis, Retrieved from Wikipedia on Oct 2, 2014, http://en.wikipedia.org/wiki/Factor_analysis

PCA, Retrieved from Statsoft on Oct 22014 from http://www.statsoft.com/ Textbook/Principal-Components-Factor-Analysis

SPSS factor analysis, Retrieved from UCLA on Oct 2, 2014, http://www.ats.ucla.edu/stat/spss/output/factor1.htm

Differences between PCA and Factor Analysis, Retrieved Oct 2, 2014, from Stackexchange, http://stats.stackexchange.com/questions/1576/what-are-the-differences-between-factor-analysis-and-principal-component-analysis