R package for simultaneous Q and R mode Factor analysis

The qrfactor package simultaneously estimates Q-and R-mode factor analysis loadings and scores. It also offers summary statistics, printing of returning values and annotated plots of first loadings. It is designed with R S3 object oriented principles.

After installing R and qrfactor package, we can proceed with tutorials on how to use the package.

qrfactor usage

First we can call the main function with data and pass it to an object 'model' from R command line as:

```
library(qrfactor)
model=qrfactor(data,obs_start=NULL,mod_type= “sd”)
```

Table 1 Input data and arguments for qrfactor package

Argument	Descriptions
data	A numeric design matrix for the model. All records must be numeric; it also accepts continuous data. Avoid using categorical variables and characters.
obs_start	a numeric record where the row number starts. For example, in time series analysis a record can start from 1960. Insert 1 if your record starts from 1. Set it to NULL, the default, if you do not want specific row numbers assigned to loadings and scores.
mod_type	Standardisation method that you want to use. Set it to "data" if you do not desire data transformation or scaling; set it to "sd" if you want the data to be standardised after centring it; set it to "n" if you want to divide the centred data by square root of the number of observations. The default is "sd".
object	an object of class "qrfactor", i.e., a fitted model.
x	an object of class "qrfactor", i.e., a fitted model.
...	any other R parameters can be added

We can then get the summary of the results of the object 'model' from R prompt as:

```
summary(model)
```

You can get detailed results of the object 'model' as:

```
print(model)
```

You can plot the first two factor axes of the object 'model' as:

```
 plot(model)
```

Returning Values

The package returns the results in the table 2

Table 2 Output or returned values the main function

data	Original data for the model. All records must be numeric. It also accepts continuous data
x.standard	it is the scale matrix of the original data
correlation	The correlation matrix for the data
eigen.value	eigen value of correlation matrix of the data
eigen.vector	eigen vector of correlation matrix of the data
diagonal.matrix	diagonal matrix of eigen vector
r.loading	R-mode loadings
q.loading	Q-mode loadings
loadings	combined loadings of R and Q on the same axis
q.scores	computed Q-mode scores
scores	combined R-mode and Q-mode scores on the same axis
rownames	row names of the loadings
variables	variables names of the loadings, of the original data

A user can assess the values of the object in table 2. For instance, we can assess R-mode loadings as:

```
Rloadings= model$r.loading
```

Assess Q-mode loadings as:

```
Qloadings= model$q.loading
```

Assess all loadings as:

```
loadings= model$loadings
```

We can get documentation from the package as

```
 ?qrfactor
```

We can get citation of the package as:

```
citation("qrfactor")
```

You are encouraged to cite the author if you use the package as:

George Owusu (2011). qrfactor: Simultaneous simulation of Q and R mode factor analyses. R package version 1.0.

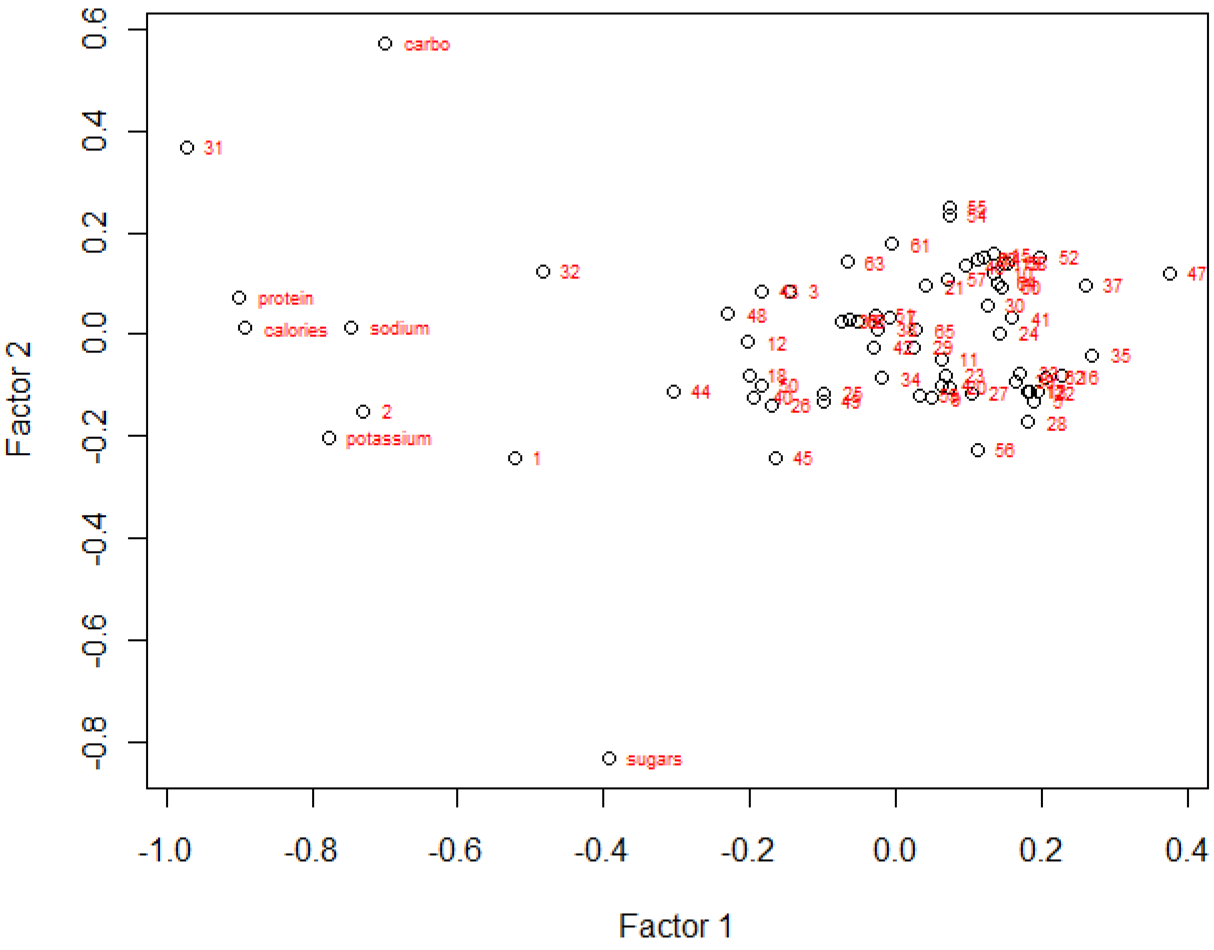
Tutorials

Example 1: Us cereal

In this example we are going to measure similarity between and among US nutritional and marketing Information. The data is imported from the MASS package. And because the data contains other variables, count and categorical, we only need to select what we want, the continuous data. Therefore we shall model the following nutrients: calories, protein, sodium, carbo, sugars, and potassium as:

```
>library(qrfactor)
 >data(UScereal, package="MASS")
>variables=c("calories","protein","sodium","carbo","sugars","potassium")
>data=UScereal[variables]
#Then we can create an object from 'qrfactor' class as:
mod1 <- qrfactor(data)
#We output the results to the screen as:
>mod1
#We can extract only the loadings as:
>loadings=mod1$loadings
>loadings
#We summarize our results as:
>summary(mod1)
#Finally we plot the first two axes of the loadings as:
>plot(mod1)
```

Figure 1 Plotting mod1



Example 2: Plotting other loadings

In the second example we want to plot only the 3rd and 4th axes. From R file menu click on ‘New script’ and type the following code and save it as ‘uscereal.R’. Run the script: from edit menu click ‘Run all’.

```
mod2<-qrfactor(data)
#print object 2
mod2
#extract the loadings
loadings=mod2$loadings
#summary of the loadings
summary(mod2)
#plotting other axes for mod2 for axes 3 and 4
plot(mod2$loadings[,3],mod2$loadings[,4])
```

Example 3: Model Evaluation

Using Davis, J. C. (2002). Statistics and Data Analysis in Geology: Wiley

In this example an attempt is made to reproduce Davis (2002) figure 6-45 in page 571 data on Table 6-33 on page 570. The data consists of contents of airborne radiometric intensity (AERO), Uranium (U), thorium (TH) and potassium (K) in Berea, Virginia. Fortunately the data is also online from the student companion website; we shall read the online data into R as:

```
#you can substitute this with a local path
>file_path="http://higheredbcs.wiley.com/legacy/college/davis/0471172758/datafiles/ascii/radio.txt"
> data= read.table(file_path)
#We select the four needed variables as:
>data= data[c("V2", "V3", "V4","V5")]
```

Let us assign variable names to the data according to the structure on Table 6-33 in (Davis 2002) as:

```
>names(data) <- c("AERO","U","TH", "K")
```

We can now plot our first two loading axes as:

```
> davis= qrfactor(data)
```

Print the content of the model as

```
> davis
```

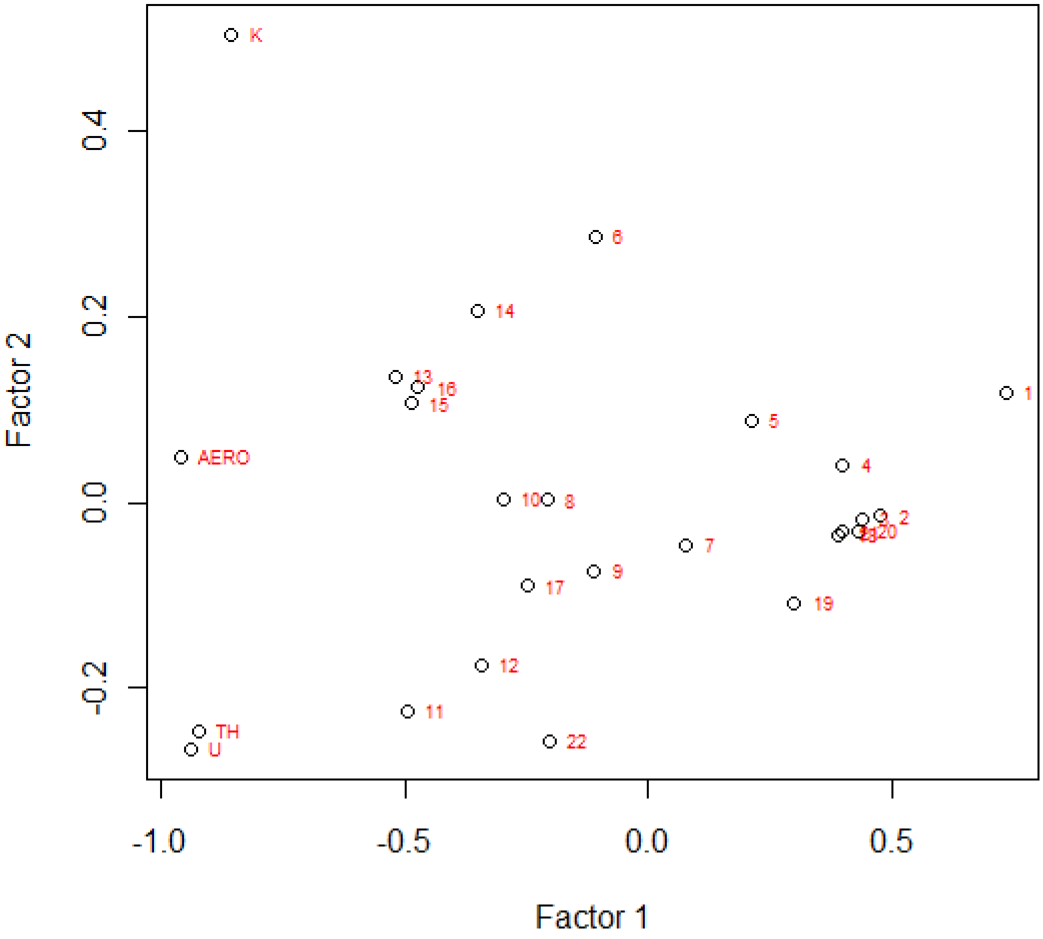
Assess loadings as

```
> davis$loadings
```

```
>plot(davis)
```

Our plot (figure 2) looks very similar to Davis (2002) figure 6-45, though loading one is inverted. We can clearly see from both figures that variables TH and U look similar while observations 13, 15, and 16 as well as observations 2, 3, 18, 20 and 21 also looking similar on all the two plots. There is huge dissimilarity between K on one hand and TH and U on the other hand. The only difference between the two plots is the inversion of the first loading axis, which Davis (2002) acknowledged that it does not affect similarity measurement and the loading can be inverted back by multiplying it by -1.

Figure 2 Similarity plot of first two loading axes of the evaluated model; compare this plot to Davis (2002) figure 6-45.



References

George Owusu (2011). qrfactor: Simultaneous simulation of Q and R mode factor analyses. R package version 1.0 Avaialable from: cran.r-project.org/web/packages/qrfactor/