Identifying or Verifying the Number of Factors to Extract using Very Simple Structure.

As published in Benchmarks RSS Matters, December 2014

http://web3.unt.edu/benchmarks/issues/2014/12/rss-matters

Jon Starkweather, PhD

Jon Starkweather, PhD
`jonathan.starkweather@unt.edu`
Consultant
Research and **S**tatistical **S**upport

UNT UNIVERSITY OF NORTH TEXAS A green light to greatness.™

http://www.unt.edu

RSS
Research and Statistical Support

http://www.unt.edu/rss

RSS hosts a number of “Short Courses”.
A list of them is available at:
http://www.unt.edu/rss/Instructional.htm

Those interested in learning more about R, or how to use it, can find information here:
`http://www.unt.edu/rss/class/Jon/R_SC`

Identifying or Verifying the Number of Factors to Extract using Very Simple Structure.

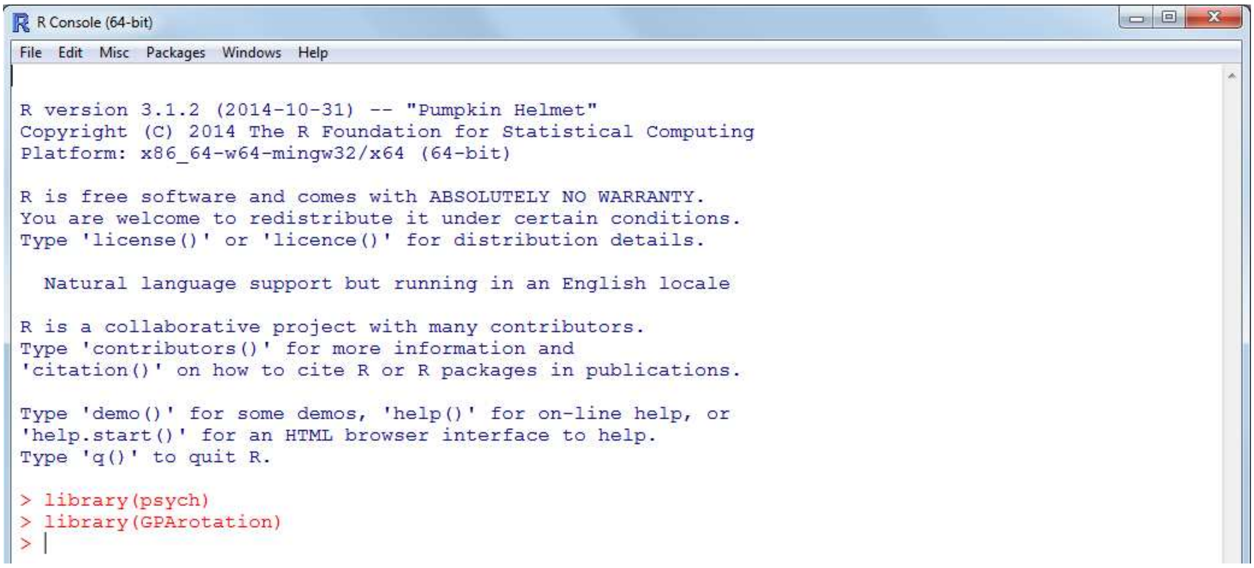
Factor analysis is perhaps one of the most frequently used analyses. It is versatile and flexible; meaning, it can be applied to a variety of data situations and types, and it can be applied in a variety of ways. However, conducting factor analysis generally requires the data analyst to make several decisions. Analysts often run several factor analyses, even when attempting to confirm an established factor structure; in order to assess the fit of the data to several factor models (e.g. one factor model, two factor model, three factor model, etc.). Over the 100 years since Spearman (1904) developed factor analysis there have been many, many criteria proposed for determining the number of factors to extract (e.g. eigenvalues greater than one, Horn's [1965] parallel analysis, Cattell's [1966] scree plot or test, Velicer's [1976] Minimum Average Partial [MAP] criterion, etc.). Each of these proposed criteria have strengths and weaknesses; and they occasionally conflict with one another, which makes using one criterion over another a risky proposition. This month's article demonstrates a very handy method for comparing multiple criteria in the pursuit of choosing to extract the appropriate number of factors during factor analysis.

In popular culture it is not uncommon to hear someone say, "There's an *app* for that." The phrase generally refers to the idea that an *application* exists (for a smart phone) which does the task being discussed. Likewise, here at RSS we very frequently find "There's a *pack* for that." This phrase refers to the virtual certainty of finding an R *package* which has a function devoted to some analysis or technique we are discussing. The primary package we will be using here is one package which contains a great many useful functions and as a result is very often *the* package we end up using for a variety of analyses. The primary package we will be using here is the 'psych' package (Revelle, 2014). The 'psych' package has grown substantially over the last few years and includes many very useful functions — if you have not taken a look at it recently, you might want to check it out.

Our examples below will actually require two packages, the 'psych' package and the 'GPArotation' package (Bernaards & Jennrich, 2014). The 'GPArotation' package should be familiar to anyone with experience doing factor analysis — it provides functions for several rotation strategies. The primary function we demonstrate below is the 'vss' function from the 'psych' package. The *Very Simple Structure* (VSS; Revell & Rocklin, 1979) function provides a nice output of criteria for varying levels of factor model complexity (i.e. number of factors to extract). The Very Simple Structure (VSS) terminology is used to refer to the idea that all loadings which are less than the maximum loading (of an item to a factor) are suppressed to zero — thus forcing a particular factor model to be much more interpretable or more clearly distinguished. Then, fit of several models of increasing rank complexity (i.e. more and more factors specified) can be assessed using the residual matrix of each model (i.e. original matrix minus the reproduced matrix of the models). We will also be using both the 'fa' function (from the 'psych' package) and the 'factanal' function (from the 'stats' package — included with all installations of R) to fit factor analysis models to the data structures.

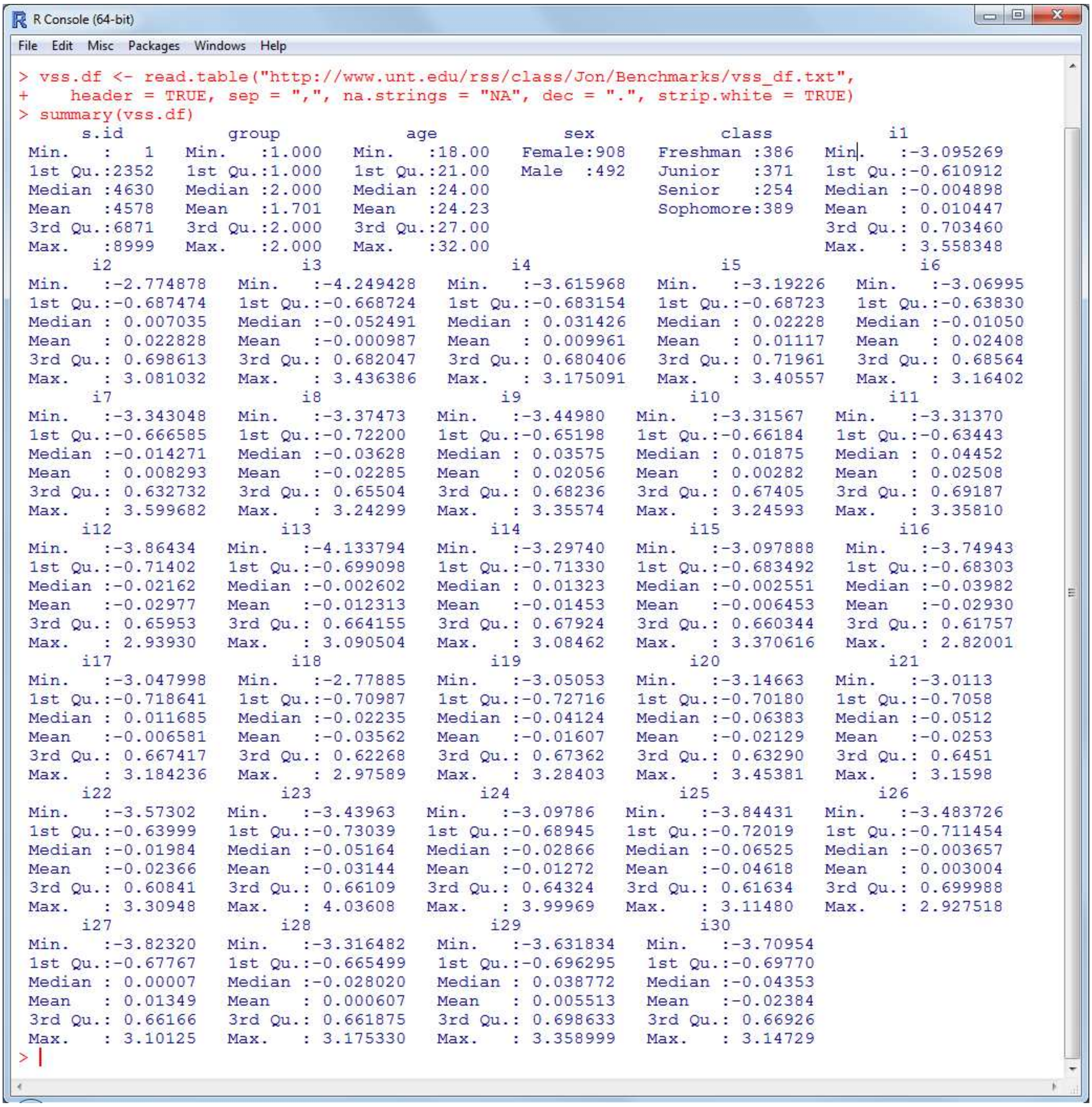
1 Examples

The first two examples used here can easily be duplicated using the scripts provided below (i.e. the data file is available at the URL in the script / screen capture image). The third example is the example contained in the help file of the 'vss' function and can be accessed using the script below. First, load the two packages we will be using.



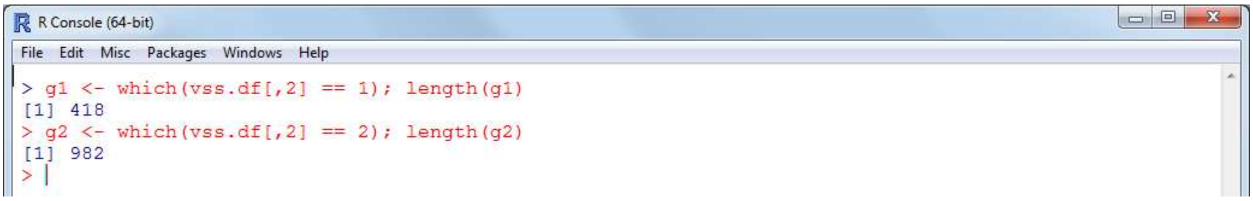
```
R version 3.1.2 (2014-10-31) -- "Pumpkin Helmet"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(psych)
> library(GPArotation)
> 
```

Next, we will import the comma delimited text (.txt) file from the RSS server using the URL and file name (vss_df.txt) contained in the script / image below. We also run a simple‘summary’ on the data frame to make sure it was imported correctly.



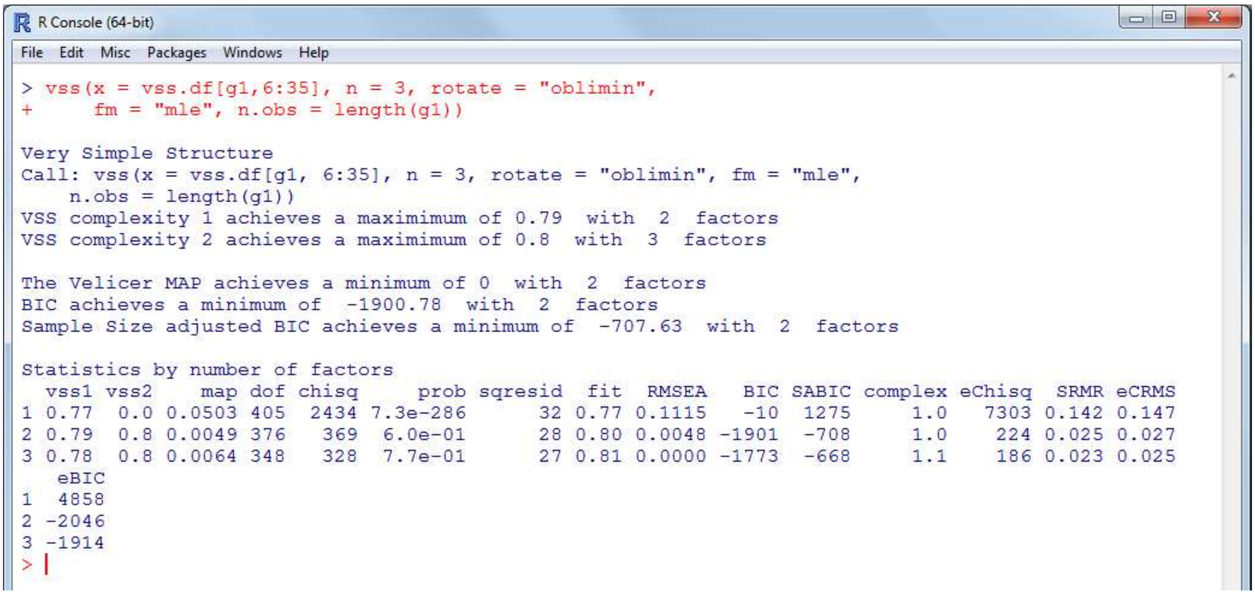
```
> vss.df <- read.table("http://www.unt.edu/rss/class/Jon/Benchmarks/vss_df.txt",
+    header = TRUE, sep = ",", na.strings = "NA", dec = ".", strip.white = TRUE)
> summary(vss.df)
      s.id          group            age             sex             class            i1            
 Min.   :   1   Min.   :1.000   Min.   :18.00   Female:908   Freshman :386   Min.   :-3.095269  
 1st Qu.:2352   1st Qu.:1.000   1st Qu.:21.00   Male  :492   Junior   :371   1st Qu.:-0.610912  
 Median :4630   Median :2.000   Median :24.00                Senior   :254   Median :-0.004898  
 Mean   :4578   Mean   :1.701   Mean   :24.23                Sophomore:389   Mean   : 0.010447  
 3rd Qu.:6871   3rd Qu.:2.000   3rd Qu.:27.00                                3rd Qu.: 0.703460  
 Max.   :8999   Max.   :2.000   Max.   :32.00                                Max.   : 3.558348  
       i2                 i3                  i4                 i5                i6          
 Min.   :-2.774878   Min.   :-4.249428   Min.   :-3.615968   Min.   :-3.19226   Min.   :-3.06995  
 1st Qu.:-0.687474   1st Qu.:-0.668724   1st Qu.:-0.683154   1st Qu.:-0.68723   1st Qu.:-0.63830  
 Median : 0.007035   Median :-0.052491   Median : 0.031426   Median : 0.02228   Median :-0.01050  
 Mean   : 0.022828   Mean   :-0.000987   Mean   : 0.009961   Mean   : 0.01117   Mean   : 0.02408  
 3rd Qu.: 0.698613   3rd Qu.: 0.682047   3rd Qu.: 0.680406   3rd Qu.: 0.71961   3rd Qu.: 0.68564  
 Max.   : 3.081032   Max.   : 3.436386   Max.   : 3.175091   Max.   : 3.40557   Max.   : 3.16402  
       i7                 i8                 i9                 i10                i11          
 Min.   :-3.343048   Min.   :-3.37473   Min.   :-3.44980   Min.   :-3.31567   Min.   :-3.31370  
 1st Qu.:-0.666585   1st Qu.:-0.72200   1st Qu.:-0.65198   1st Qu.:-0.66184   1st Qu.:-0.63443  
 Median :-0.014271   Median :-0.03628   Median : 0.03575   Median : 0.01875   Median : 0.04452  
 Mean   : 0.008293   Mean   :-0.02285   Mean   : 0.02056   Mean   : 0.00282   Mean   : 0.02508  
 3rd Qu.: 0.632732   3rd Qu.: 0.65504   3rd Qu.: 0.68236   3rd Qu.: 0.67405   3rd Qu.: 0.69187  
 Max.   : 3.599682   Max.   : 3.24299   Max.   : 3.35574   Max.   : 3.24593   Max.   : 3.35810  
      i12                i13                 i14                i15                i16          
 Min.   :-3.86434   Min.   :-4.133794   Min.   :-3.29740   Min.   :-3.097888   Min.   :-3.74943  
 1st Qu.:-0.71402   1st Qu.:-0.699098   1st Qu.:-0.71330   1st Qu.:-0.683492   1st Qu.:-0.68303  
 Median :-0.02162   Median :-0.002602   Median : 0.01323   Median :-0.002551   Median :-0.03982  
 Mean   :-0.02977   Mean   :-0.012313   Mean   :-0.01453   Mean   :-0.006453   Mean   :-0.02930  
 3rd Qu.: 0.65953   3rd Qu.: 0.664155   3rd Qu.: 0.67924   3rd Qu.: 0.660344   3rd Qu.: 0.61757  
 Max.   : 2.93930   Max.   : 3.090504   Max.   : 3.08462   Max.   : 3.370616   Max.   : 2.82001  
      i17                 i18                i19                i20                i21          
 Min.   :-3.047998   Min.   :-2.77885   Min.   :-3.05053   Min.   :-3.14663   Min.   :-3.0113  
 1st Qu.:-0.718641   1st Qu.:-0.70987   1st Qu.:-0.72716   1st Qu.:-0.70180   1st Qu.:-0.7058  
 Median : 0.011685   Median :-0.02235   Median :-0.04124   Median :-0.06383   Median :-0.0512  
 Mean   :-0.006581   Mean   :-0.03562   Mean   :-0.01607   Mean   :-0.02129   Mean   :-0.0253  
 3rd Qu.: 0.667417   3rd Qu.: 0.62268   3rd Qu.: 0.67362   3rd Qu.: 0.63290   3rd Qu.: 0.6451  
 Max.   : 3.184236   Max.   : 2.97589   Max.   : 3.28403   Max.   : 3.45381   Max.   : 3.1598  
      i22                i23                i24                i25                i26           
 Min.   :-3.57302   Min.   :-3.43963   Min.   :-3.09786   Min.   :-3.84431   Min.   :-3.483726  
 1st Qu.:-0.63999   1st Qu.:-0.73039   1st Qu.:-0.68945   1st Qu.:-0.72019   1st Qu.:-0.711454  
 Median :-0.01984   Median :-0.05164   Median :-0.02866   Median :-0.06525   Median :-0.003657  
 Mean   :-0.02366   Mean   :-0.03144   Mean   :-0.01272   Mean   :-0.04618   Mean   : 0.003004  
 3rd Qu.: 0.60841   3rd Qu.: 0.66109   3rd Qu.: 0.64324   3rd Qu.: 0.61634   3rd Qu.: 0.699988  
 Max.   : 3.30948   Max.   : 4.03608   Max.   : 3.99969   Max.   : 3.11480   Max.   : 2.927518  
      i27                i28                 i29                 i30          
 Min.   :-3.82320   Min.   :-3.316482   Min.   :-3.631834   Min.   :-3.70954  
 1st Qu.:-0.67767   1st Qu.:-0.665499   1st Qu.:-0.696295   1st Qu.:-0.69770  
 Median : 0.00007   Median :-0.028020   Median : 0.038772   Median :-0.04353  
 Mean   : 0.01349   Mean   : 0.000607   Mean   : 0.005513   Mean   :-0.02384  
 3rd Qu.: 0.66166   3rd Qu.: 0.661875   3rd Qu.: 0.698633   3rd Qu.: 0.66926  
 Max.   : 3.10125   Max.   : 3.175330   Max.   : 3.358999   Max.   : 3.14729  
> 
```

The simulated data includes a sample identification number for each participant (s.id), a grouping variable (group 1 or group 2), age of each participant (age in years), sex of each participant (female or male), class standing of each participant (freshman, sophomore, junior, or senior), and 30 item scores. Next, we will identify which participants belong to group 1 and which belong to group 2; as well as the number of participants in each group.



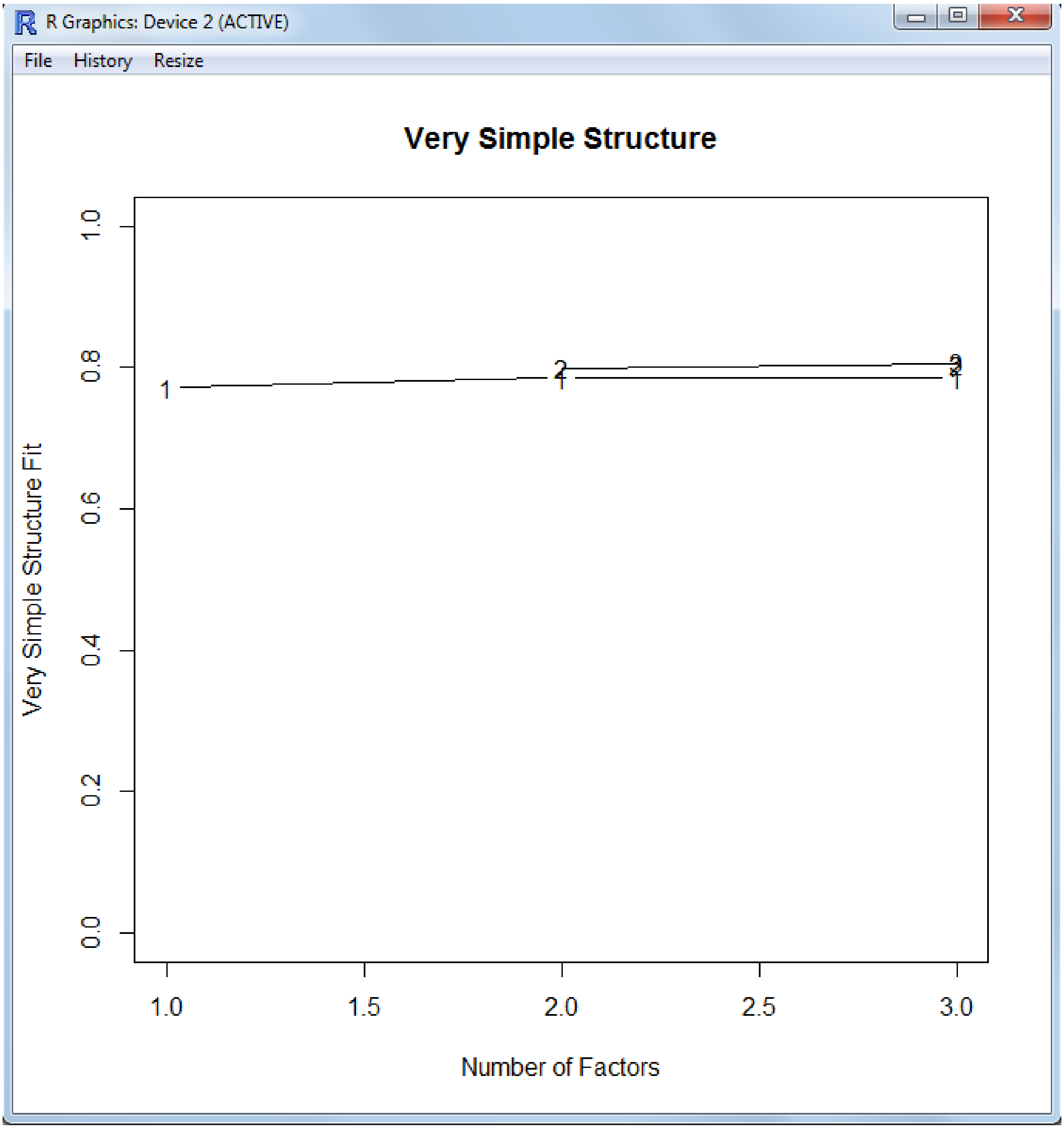
```
> g1 <- which(vss.df[,2] == 1); length(g1)
[1] 418
> g2 <- which(vss.df[,2] == 2); length(g2)
[1] 982
> 
```

So, we have 418 participants in group 1 and 982 participants in group 2. Generally when analysts intend to do factor analysis they have an idea of how many factors they believe the appropriate factor model contains; and often they have an idea of whether an orthogonal or oblique rotation strategy is warranted. For this first example (i.e. group 1) looking at the 30 item scores (i.e. columns 6 through 35), we believe there are two factors and therefore; we specify 3 factors (*n* = 3) in the ‘vss’ function. We also believe the factors are likely to be meaningfully related and consequently, we specify an oblimin rotation strategy. Next, we apply the ‘vss’ function to group 1. Also note, we specified Maximum Likelihood Estimation as the Factor Method (fm = “mle”) because this is the method used by default with the ‘factanal’ (i.e. factor analysis) function of the ‘stats’ package. We specified the number of observations (i.e. number of rows, cases, or participants) using the length of the group 1 vector (g1). Recall from above, the group 1 vector contains the row numbers of all the participants from group 1.



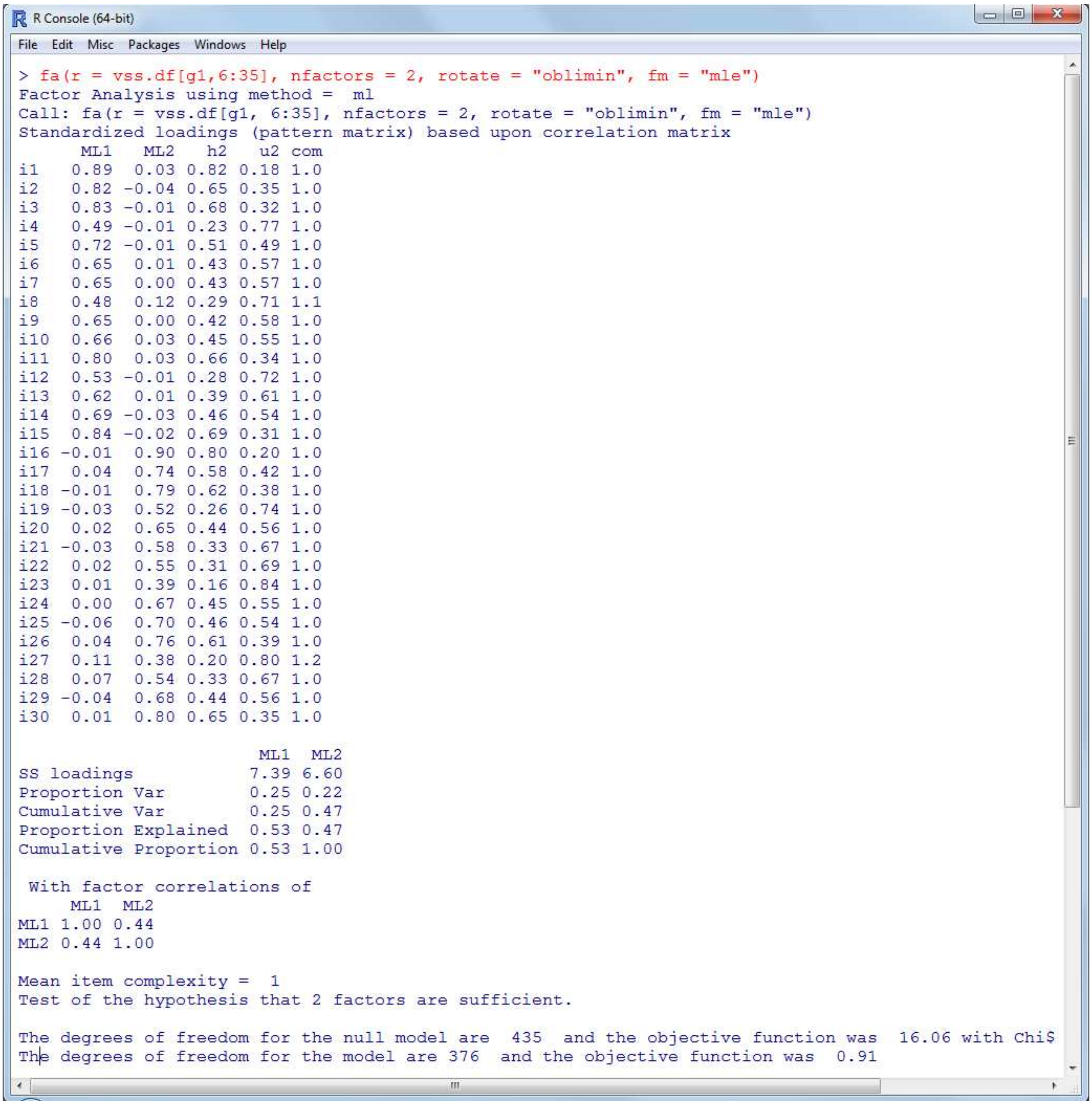
```
> vss(x = vss.df[g1,6:35], n = 3, rotate = "oblimin",
+     fm = "mle", n.obs = length(g1))

Very Simple Structure
Call: vss(x = vss.df[g1, 6:35], n = 3, rotate = "oblimin", fm = "mle",
    n.obs = length(g1))
VSS complexity 1 achieves a maximimum of 0.79  with  2  factors
VSS complexity 2 achieves a maximimum of 0.8  with  3  factors

The Velicer MAP achieves a minimum of 0  with  2  factors
BIC achieves a minimum of  -1900.78  with  2  factors
Sample Size adjusted BIC achieves a minimum of  -707.63  with  2  factors

Statistics by number of factors
  vss1 vss2    map dof chisq     prob sqresid  fit  RMSEA   BIC SABIC complex eChisq  SRMR eCRMS
1 0.77  0.0 0.0503 405  2434 7.3e-286      32 0.77 0.1115   -10  1275     1.0   7303 0.142 0.147
2 0.79  0.8 0.0049 376   369  6.0e-01      28 0.80 0.0048 -1901  -708     1.0    224 0.025 0.027
3 0.78  0.8 0.0064 348   328  7.7e-01      27 0.81 0.0000 -1773  -668     1.1    186 0.023 0.025
   eBIC
1  4858
2 -2046
3 -1914
> |
```

The first few rows of output (i.e. “Very Simple Structure” table) show the function called and the *maximum* complexity values. This is a good example because the VSS complexity rows are conflicting; VSS complexity 1 shows a 2–factor model is best while VSS complexity 2 indicates a 3–factor model is best. The VSS complexity 2 is a bit misleading because both the 2–factor model and 3–factor model display a VSS complexity 2 of 0.80; as can be seen in the first column of output under the “Statistics by number of factors” table. So, in fact both complexity 1 and complexity 2 are in agreement. Furthermore, the Velicer MAP *minimum* is reached with the 2–factor model; which can also be seen in the third column of the “Statistics by number of factors” table. The Bayesian Information Criterion (BIC) *minimum* is reached with the 2–factor model; as well as the Sample Size adjusted BIC (SABIC) — shown in columns 10 and 11 respectively of the “Statistics by number of factors” table. The ‘vss’ function also produces a plot (by default) which shows the number of factors on the x–axis and the VSS (complexity) Fit along the y–axis with lines and numbers in the Cartesian plane representing the (3) different factor models (see below).



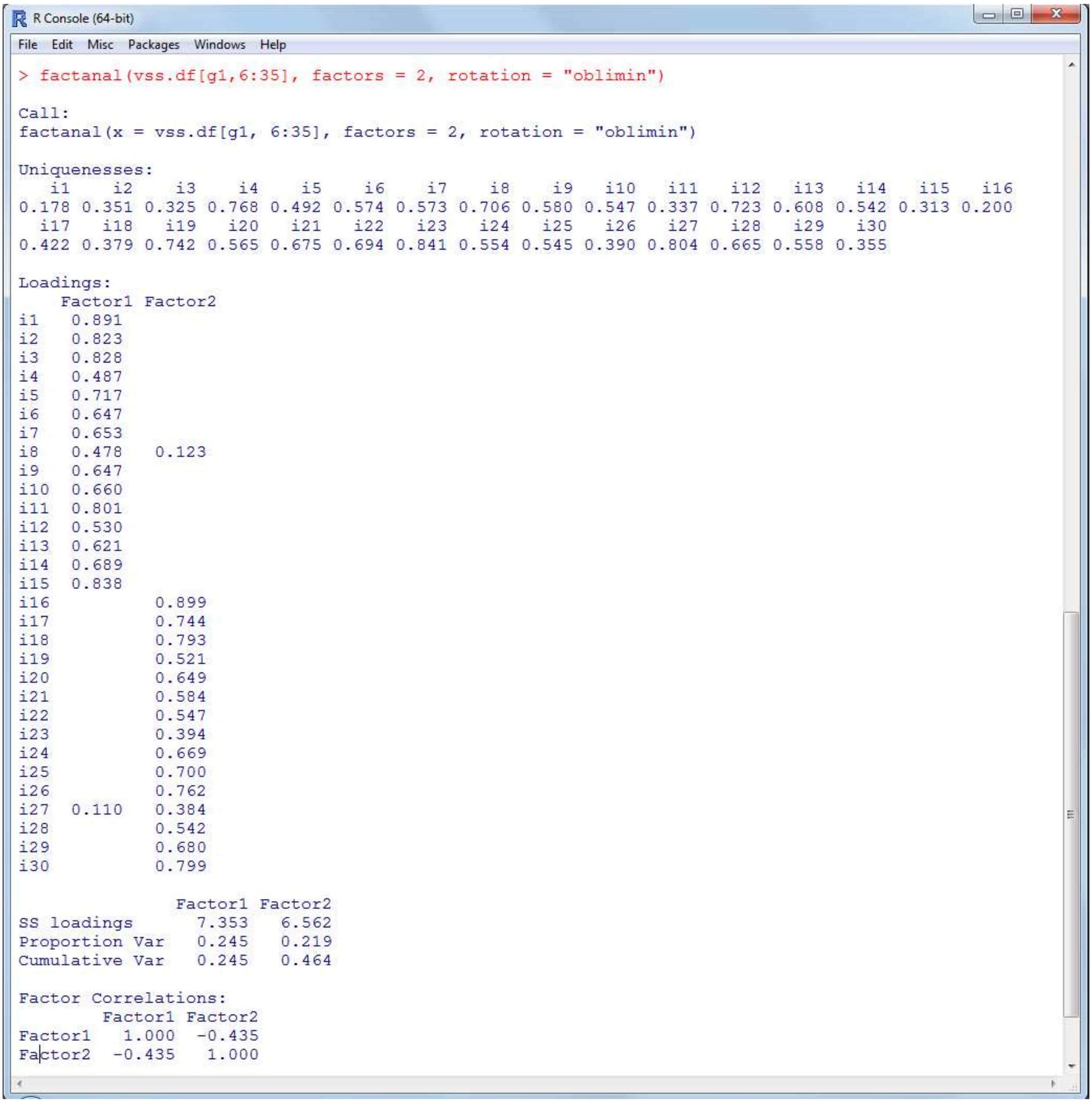
To interpret the graph, focus on the model (1, 2, or 3 factor models) which has the highest line (and numerals) in relation to the y-axis; but also note any transitions of the model lines. In this example, the transitions are all very nearly flat but a later example will better demonstrate the utility of this type of plot.

Next, we can verify the fit of our 2–factor model using either the ‘fa’ function (from the ‘psych’ package) and / or the ‘factanal’ function (of the ‘stats’ package).



```
> fa(r = vss.df[g1,6:35], nfactors = 2, rotate = "oblimin", fm = "mle")
Factor Analysis using method =  ml
Call: fa(r = vss.df[g1, 6:35], nfactors = 2, rotate = "oblimin", fm = "mle")
Standardized loadings (pattern matrix) based upon correlation matrix
      ML1   ML2   h2   u2 com
i1   0.89  0.03 0.82 0.18 1.0
i2   0.82 -0.04 0.65 0.35 1.0
i3   0.83 -0.01 0.68 0.32 1.0
i4   0.49 -0.01 0.23 0.77 1.0
i5   0.72 -0.01 0.51 0.49 1.0
i6   0.65  0.01 0.43 0.57 1.0
i7   0.65  0.00 0.43 0.57 1.0
i8   0.48  0.12 0.29 0.71 1.1
i9   0.65  0.00 0.42 0.58 1.0
i10  0.66  0.03 0.45 0.55 1.0
i11  0.80  0.03 0.66 0.34 1.0
i12  0.53 -0.01 0.28 0.72 1.0
i13  0.62  0.01 0.39 0.61 1.0
i14  0.69 -0.03 0.46 0.54 1.0
i15  0.84 -0.02 0.69 0.31 1.0
i16 -0.01  0.90 0.80 0.20 1.0
i17  0.04  0.74 0.58 0.42 1.0
i18 -0.01  0.79 0.62 0.38 1.0
i19 -0.03  0.52 0.26 0.74 1.0
i20  0.02  0.65 0.44 0.56 1.0
i21 -0.03  0.58 0.33 0.67 1.0
i22  0.02  0.55 0.31 0.69 1.0
i23  0.01  0.39 0.16 0.84 1.0
i24  0.00  0.67 0.45 0.55 1.0
i25 -0.06  0.70 0.46 0.54 1.0
i26  0.04  0.76 0.61 0.39 1.0
i27  0.11  0.38 0.20 0.80 1.2
i28  0.07  0.54 0.33 0.67 1.0
i29 -0.04  0.68 0.44 0.56 1.0
i30  0.01  0.80 0.65 0.35 1.0

                       ML1  ML2
SS loadings           7.39 6.60
Proportion Var        0.25 0.22
Cumulative Var        0.25 0.47
Proportion Explained  0.53 0.47
Cumulative Proportion 0.53 1.00

 With factor correlations of
     ML1  ML2
ML1 1.00 0.44
ML2 0.44 1.00

Mean item complexity =  1
Test of the hypothesis that 2 factors are sufficient.

The degrees of freedom for the null model are  435  and the objective function was  16.06 with Chi$
The degrees of freedom for the model are 376  and the objective function was  0.91
```

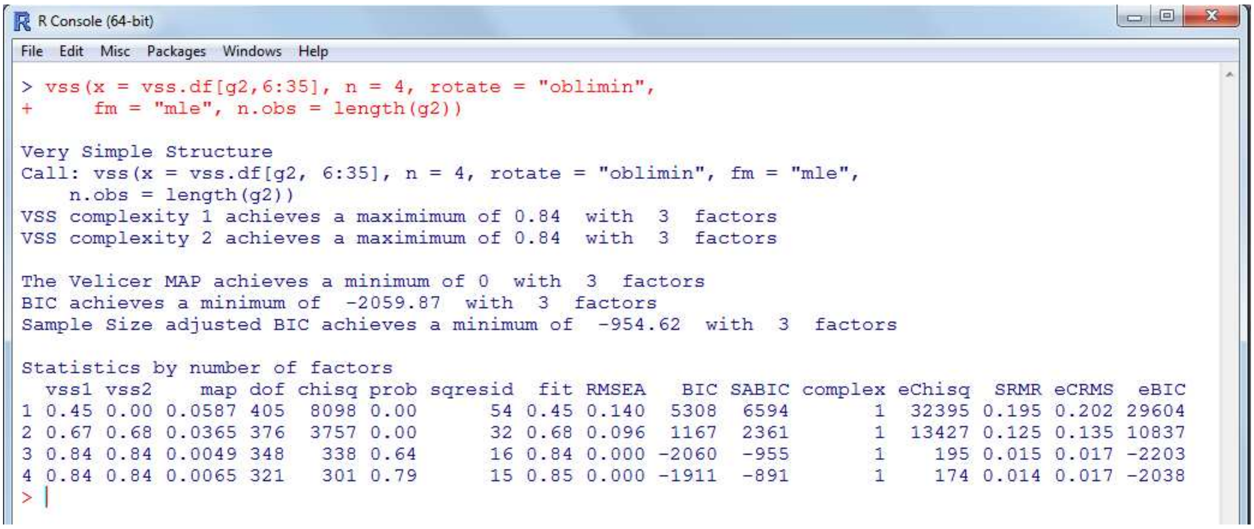
Note: the last few lines of output from the ‘fa’ function are cut off (i.e. not shown).



```
> factanal(vss.df[g1,6:35], factors = 2, rotation = "oblimin")

Call:
factanal(x = vss.df[g1, 6:35], factors = 2, rotation = "oblimin")

Uniquenesses:
   i1    i2    i3    i4    i5    i6    i7    i8    i9   i10   i11   i12   i13   i14   i15   i16 
0.178 0.351 0.325 0.768 0.492 0.574 0.573 0.706 0.580 0.547 0.337 0.723 0.608 0.542 0.313 0.200 
  i17   i18   i19   i20   i21   i22   i23   i24   i25   i26   i27   i28   i29   i30 
0.422 0.379 0.742 0.565 0.675 0.694 0.841 0.554 0.545 0.390 0.804 0.665 0.558 0.355 

Loadings:
    Factor1 Factor2
i1   0.891         
i2   0.823         
i3   0.828         
i4   0.487         
i5   0.717         
i6   0.647         
i7   0.653         
i8   0.478   0.123 
i9   0.647         
i10  0.660         
i11  0.801         
i12  0.530         
i13  0.621         
i14  0.689         
i15  0.838         
i16          0.899 
i17          0.744 
i18          0.793 
i19          0.521 
i20          0.649 
i21          0.584 
i22          0.547 
i23          0.394 
i24          0.669 
i25          0.700 
i26          0.762 
i27  0.110   0.384 
i28          0.542 
i29          0.680 
i30          0.799 

               Factor1 Factor2
SS loadings      7.353   6.562
Proportion Var   0.245   0.219
Cumulative Var   0.245   0.464

Factor Correlations:
        Factor1 Factor2
Factor1   1.000  -0.435
Factor2  -0.435   1.000
```

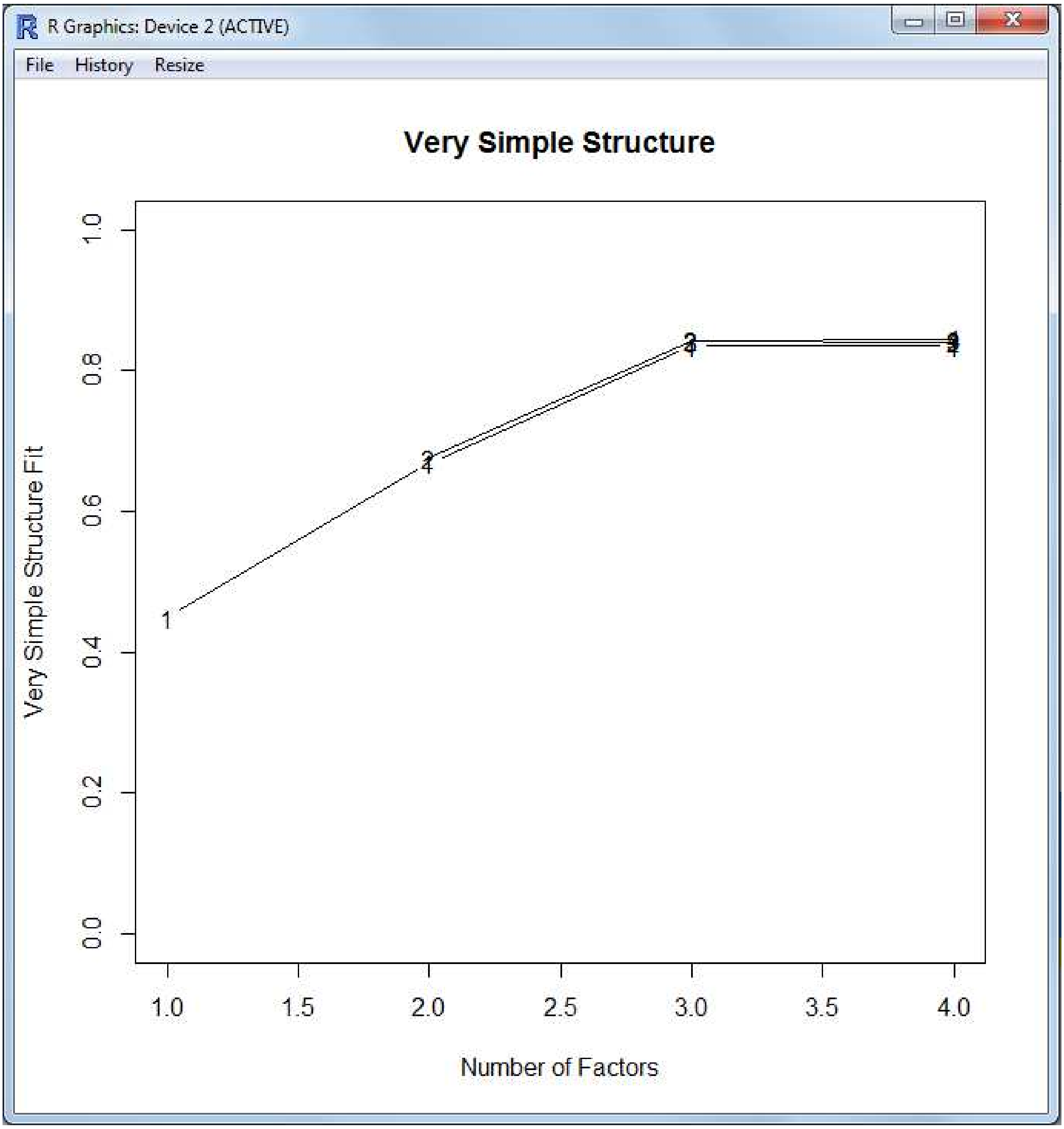
Note: last few lines of output from the ‘factanal’ function are cut off (i.e. not shown).

We will now assess the group 2 (g2) data. This group is believed to be best served with a 3–factor model; so we specify 4 factors (*n* = 4) in the ‘vss’ function call; again with the factor method set to Maximum Likelihood Estimation (fm = “mle”) and an oblique rotation strategy (rotate = “oblimin”).

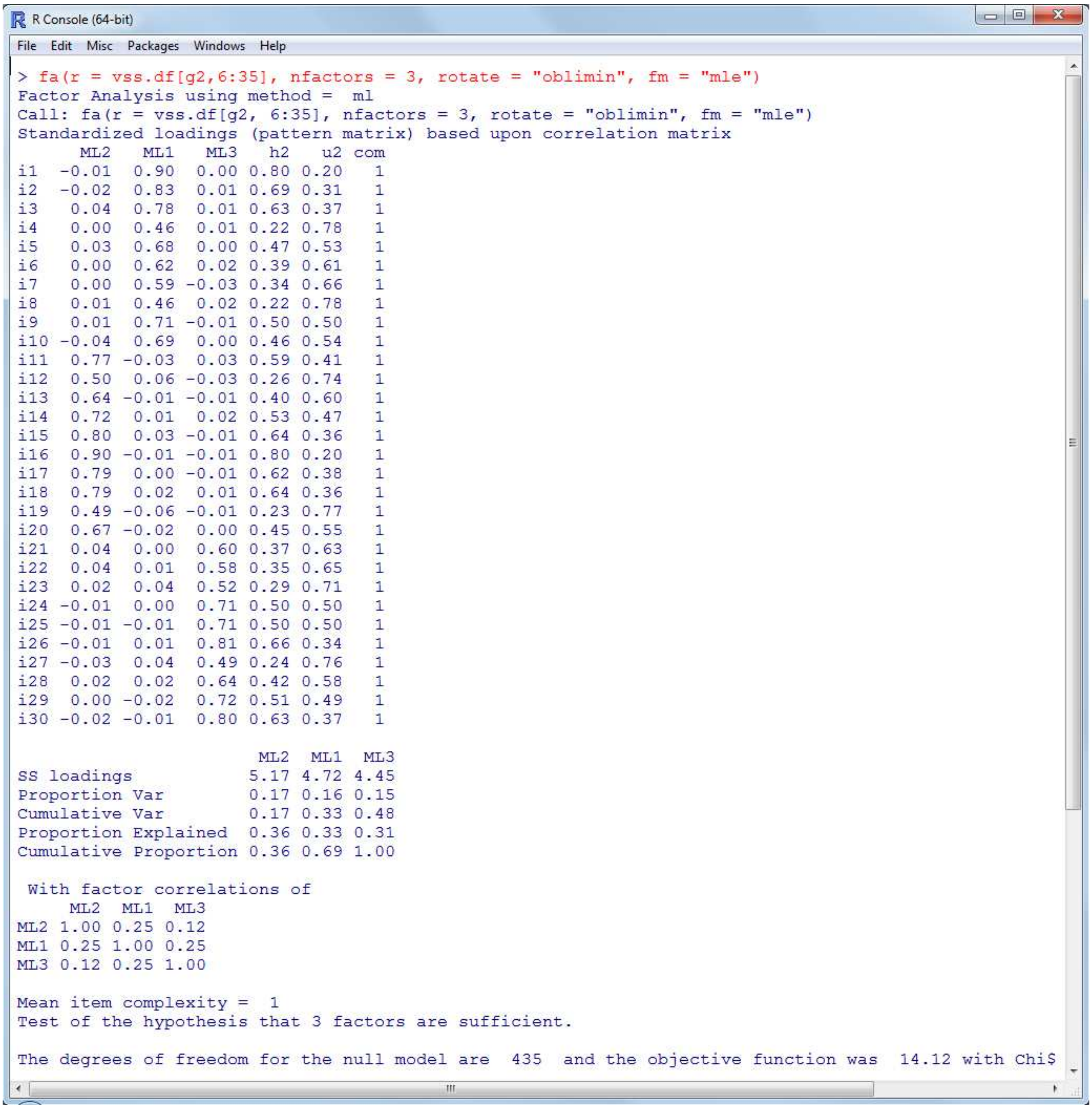


```
> vss(x = vss.df[g2,6:35], n = 4, rotate = "oblimin",
+     fm = "mle", n.obs = length(g2))

Very Simple Structure
Call: vss(x = vss.df[g2, 6:35], n = 4, rotate = "oblimin", fm = "mle",
    n.obs = length(g2))
VSS complexity 1 achieves a maximimum of 0.84  with  3  factors
VSS complexity 2 achieves a maximimum of 0.84  with  3  factors

The Velicer MAP achieves a minimum of 0  with  3  factors
BIC achieves a minimum of  -2059.87  with  3  factors
Sample Size adjusted BIC achieves a minimum of  -954.62  with  3  factors

Statistics by number of factors
  vss1 vss2    map dof chisq prob sqresid  fit RMSEA   BIC SABIC complex eChisq  SRMR eCRMS  eBIC
1 0.45 0.00 0.0587 405  8098 0.00      54 0.45 0.140  5308  6594       1  32395 0.195 0.202 29604
2 0.67 0.68 0.0365 376  3757 0.00      32 0.68 0.096  1167  2361       1  13427 0.125 0.135 10837
3 0.84 0.84 0.0049 348   338 0.64      16 0.84 0.000 -2060  -955       1    195 0.015 0.017 -2203
4 0.84 0.84 0.0065 321   301 0.79      15 0.85 0.000 -1911  -891       1    174 0.014 0.017 -2038
> |
```

In this example all of the indices in the top table (“Very Simple Structure”) are in agreement; although both VSS complexity metrics display the same *maximum* for a 3–factor model and a 4–factor model. Looking at the first two columns of the “Statistics by number of factors” table shows the identical complexity *maximums* (0.84) for both the 3–factor model (row 3) and the 4–factor model (row 4) with both complexities 1 and 2 (columns 1 and 2). But, given the other indices agreement in support of the 3–factor model, that would be the model most appropriate. The plot (below) reinforces the interpretation of the tabular output above.

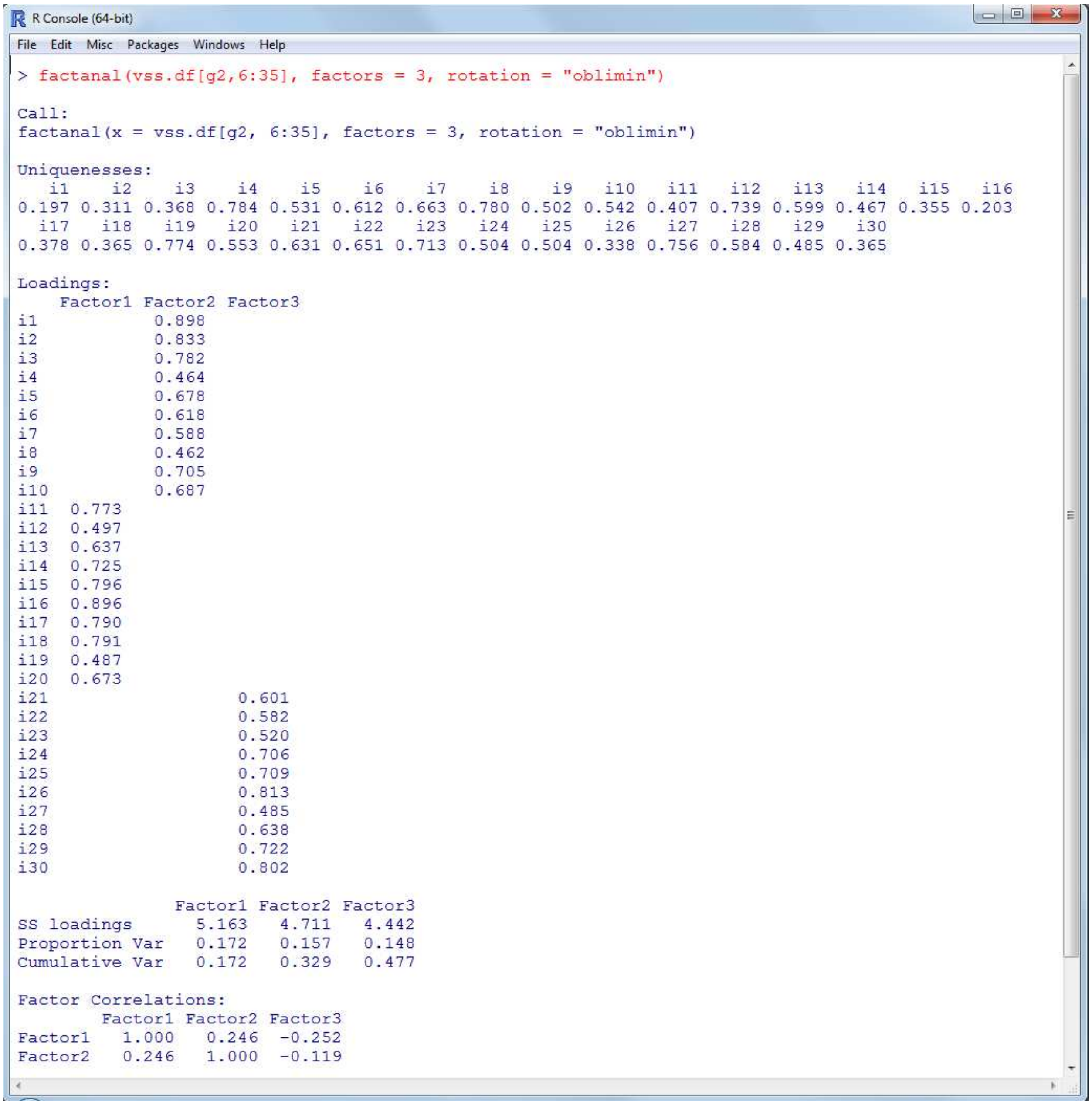


The plot (above) shows that the 3–factor model is meaningfully better than the 1–factor or 2–factor models and the 4–factor model does not show any improvement over the 3–factor model — which is evident because the number 4 in the plot is not [further] above the line associated with the 3–factor model (i.e. no gain or transition upward; as is the case from 1–factor to 2–factors and to 3–factors). Therefore, we fit the 3–factor model to our data using the ‘fa’ function (of the ‘psych’ package) and / or the ‘factanal’ function of the ‘stats’ package.



```
> fa(r = vss.df[g2,6:35], nfactors = 3, rotate = "oblimin", fm = "mle")
Factor Analysis using method =  ml
Call: fa(r = vss.df[g2, 6:35], nfactors = 3, rotate = "oblimin", fm = "mle")
Standardized loadings (pattern matrix) based upon correlation matrix
      ML2   ML1   ML3   h2   u2 com
i1  -0.01  0.90  0.00 0.80 0.20   1
i2  -0.02  0.83  0.01 0.69 0.31   1
i3   0.04  0.78  0.01 0.63 0.37   1
i4   0.00  0.46  0.01 0.22 0.78   1
i5   0.03  0.68  0.00 0.47 0.53   1
i6   0.00  0.62  0.02 0.39 0.61   1
i7   0.00  0.59 -0.03 0.34 0.66   1
i8   0.01  0.46  0.02 0.22 0.78   1
i9   0.01  0.71 -0.01 0.50 0.50   1
i10 -0.04  0.69  0.00 0.46 0.54   1
i11  0.77 -0.03  0.03 0.59 0.41   1
i12  0.50  0.06 -0.03 0.26 0.74   1
i13  0.64 -0.01 -0.01 0.40 0.60   1
i14  0.72  0.01  0.02 0.53 0.47   1
i15  0.80  0.03 -0.01 0.64 0.36   1
i16  0.90 -0.01 -0.01 0.80 0.20   1
i17  0.79  0.00 -0.01 0.62 0.38   1
i18  0.79  0.02  0.01 0.64 0.36   1
i19  0.49 -0.06 -0.01 0.23 0.77   1
i20  0.67 -0.02  0.00 0.45 0.55   1
i21  0.04  0.00  0.60 0.37 0.63   1
i22  0.04  0.01  0.58 0.35 0.65   1
i23  0.02  0.04  0.52 0.29 0.71   1
i24 -0.01  0.00  0.71 0.50 0.50   1
i25 -0.01 -0.01  0.71 0.50 0.50   1
i26 -0.01  0.01  0.81 0.66 0.34   1
i27 -0.03  0.04  0.49 0.24 0.76   1
i28  0.02  0.02  0.64 0.42 0.58   1
i29  0.00 -0.02  0.72 0.51 0.49   1
i30 -0.02 -0.01  0.80 0.63 0.37   1

                       ML2  ML1  ML3
SS loadings           5.17 4.72 4.45
Proportion Var        0.17 0.16 0.15
Cumulative Var        0.17 0.33 0.48
Proportion Explained  0.36 0.33 0.31
Cumulative Proportion 0.36 0.69 1.00

 With factor correlations of
     ML2  ML1  ML3
ML2 1.00 0.25 0.12
ML1 0.25 1.00 0.25
ML3 0.12 0.25 1.00

Mean item complexity =  1
Test of the hypothesis that 3 factors are sufficient.

The degrees of freedom for the null model are  435  and the objective function was  14.12 with Chi$
```

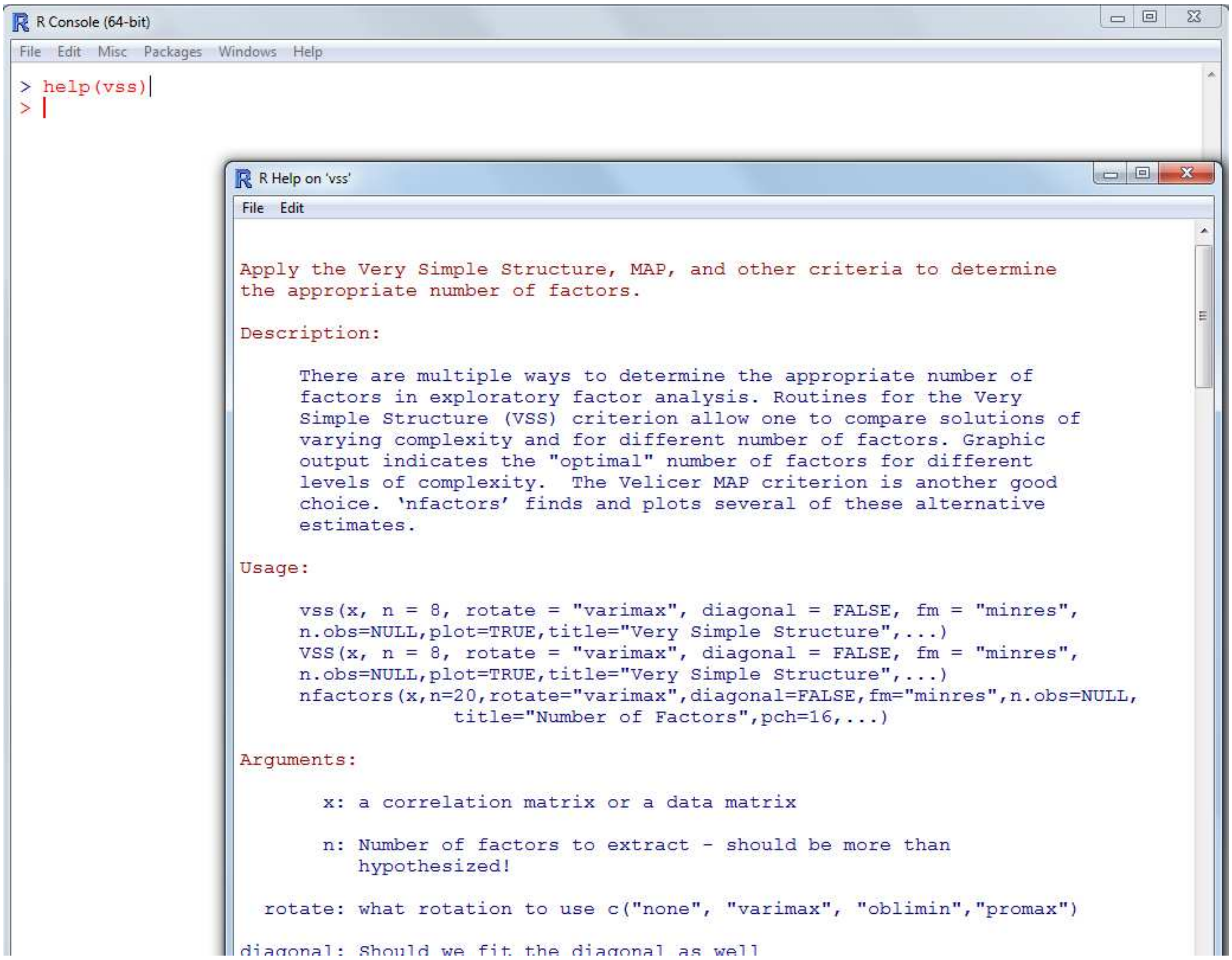
Note: the last few lines of output from the ‘fa’ function are cut off (i.e. not shown).



```
> factanal(vss.df[g2,6:35], factors = 3, rotation = "oblimin")

Call:
factanal(x = vss.df[g2, 6:35], factors = 3, rotation = "oblimin")

Uniquenesses:
   i1    i2    i3    i4    i5    i6    i7    i8    i9   i10   i11   i12   i13   i14   i15   i16 
0.197 0.311 0.368 0.784 0.531 0.612 0.663 0.780 0.502 0.542 0.407 0.739 0.599 0.467 0.355 0.203 
  i17   i18   i19   i20   i21   i22   i23   i24   i25   i26   i27   i28   i29   i30 
0.378 0.365 0.774 0.553 0.631 0.651 0.713 0.504 0.504 0.338 0.756 0.584 0.485 0.365 

Loadings:
    Factor1 Factor2 Factor3
i1           0.898         
i2           0.833         
i3           0.782         
i4           0.464         
i5           0.678         
i6           0.618         
i7           0.588         
i8           0.462         
i9           0.705         
i10          0.687         
i11  0.773                 
i12  0.497                 
i13  0.637                 
i14  0.725                 
i15  0.796                 
i16  0.896                 
i17  0.790                 
i18  0.791                 
i19  0.487                 
i20  0.673                 
i21                  0.601 
i22                  0.582 
i23                  0.520 
i24                  0.706 
i25                  0.709 
i26                  0.813 
i27                  0.485 
i28                  0.638 
i29                  0.722 
i30                  0.802 

               Factor1 Factor2 Factor3
SS loadings      5.163   4.711   4.442
Proportion Var   0.172   0.157   0.148
Cumulative Var   0.172   0.329   0.477

Factor Correlations:
        Factor1 Factor2 Factor3
Factor1   1.000   0.246  -0.252
Factor2   0.246   1.000  -0.119
```

Note: last few lines of output from the ‘factanal’ function are cut off (i.e. not shown).

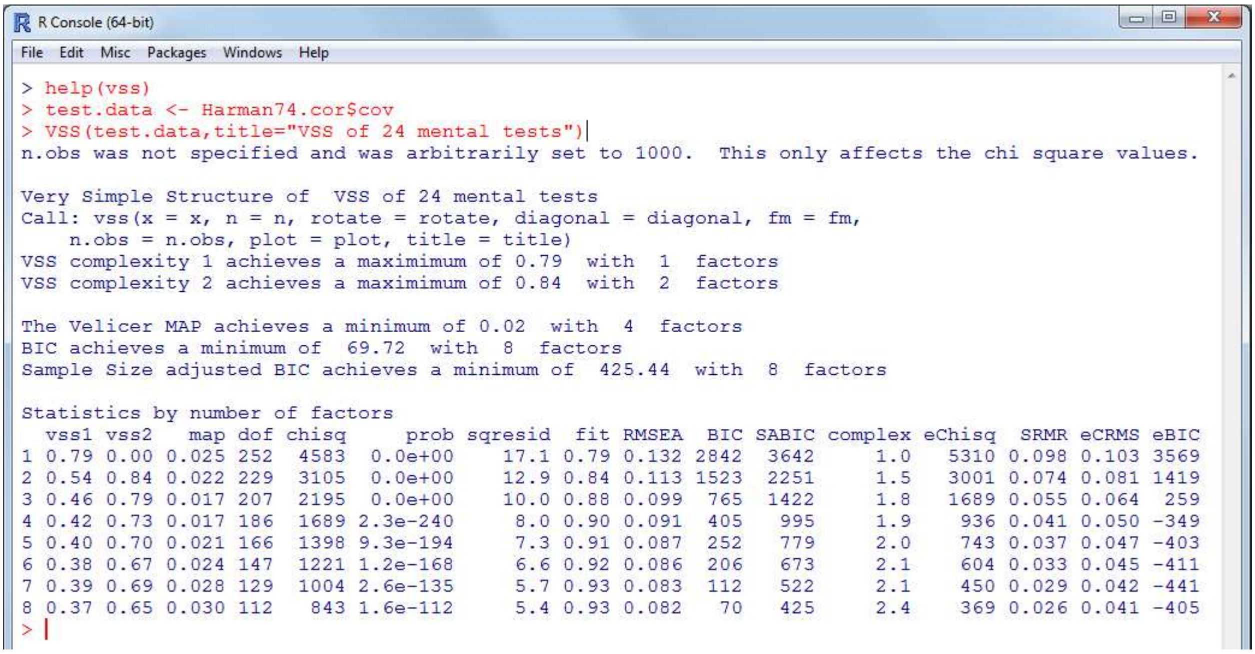
The next example is straight from the help file of the ‘vss’ function and is discussed here because it demonstrates a situation when the tables of output from the ‘vss’ function are not in agreement. When this situation occurs, one must rely upon the plot produced by the ‘vss’ function rather than the textual output. First, open the help file (here the plain text version is shown).



```
> help(vss)
>
```

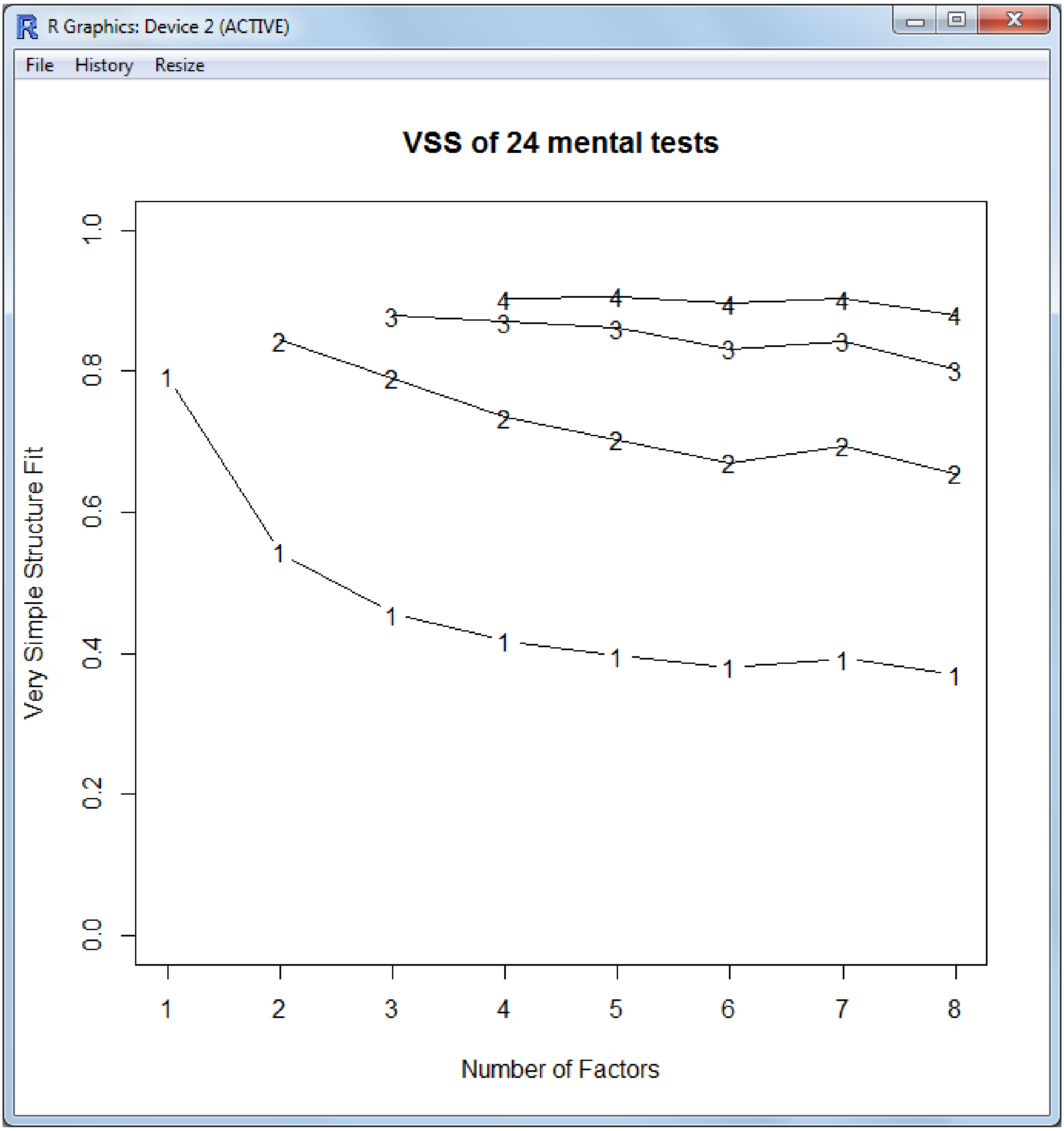
```
Apply the Very Simple Structure, MAP, and other criteria to determine
the appropriate number of factors.

Description:

     There are multiple ways to determine the appropriate number of
     factors in exploratory factor analysis. Routines for the Very
     Simple Structure (VSS) criterion allow one to compare solutions of
     varying complexity and for different number of factors. Graphic
     output indicates the "optimal" number of factors for different
     levels of complexity.  The Velicer MAP criterion is another good
     choice. 'nfactors' finds and plots several of these alternative
     estimates.

Usage:

     vss(x, n = 8, rotate = "varimax", diagonal = FALSE, fm = "minres",
     n.obs=NULL,plot=TRUE,title="Very Simple Structure",...)
     VSS(x, n = 8, rotate = "varimax", diagonal = FALSE, fm = "minres",
     n.obs=NULL,plot=TRUE,title="Very Simple Structure",...)
     nfactors(x,n=20,rotate="varimax",diagonal=FALSE,fm="minres",n.obs=NULL,
                  title="Number of Factors",pch=16,...)

Arguments:

       x: a correlation matrix or a data matrix

       n: Number of factors to extract - should be more than
          hypothesized!

  rotate: what rotation to use c("none", "varimax", "oblimin","promax")

diagonal: Should we fit the diagonal as well
```

Next, scroll to the bottom of the help file and copy / paste the relevant lines of script into the R console.



```
> help(vss)
> test.data <- Harman74.cor$cov
> VSS(test.data,title="VSS of 24 mental tests")
n.obs was not specified and was arbitrarily set to 1000.  This only affects the chi square values.

Very Simple Structure of  VSS of 24 mental tests
Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,
    n.obs = n.obs, plot = plot, title = title)
VSS complexity 1 achieves a maximimum of 0.79  with  1  factors
VSS complexity 2 achieves a maximimum of 0.84  with  2  factors

The Velicer MAP achieves a minimum of 0.02  with  4  factors
BIC achieves a minimum of  69.72  with  8  factors
Sample Size adjusted BIC achieves a minimum of  425.44  with  8  factors

Statistics by number of factors
  vss1 vss2   map dof chisq     prob sqresid  fit RMSEA  BIC SABIC complex eChisq  SRMR eCRMS eBIC
1 0.79 0.00 0.025 252  4583  0.0e+00    17.1 0.79 0.132 2842  3642     1.0   5310 0.098 0.103 3569
2 0.54 0.84 0.022 229  3105  0.0e+00    12.9 0.84 0.113 1523  2251     1.5   3001 0.074 0.081 1419
3 0.46 0.79 0.017 207  2195  0.0e+00    10.0 0.88 0.099  765  1422     1.8   1689 0.055 0.064  259
4 0.42 0.73 0.017 186  1689 2.3e-240     8.0 0.90 0.091  405   995     1.9    936 0.041 0.050 -349
5 0.40 0.70 0.021 166  1398 9.3e-194     7.3 0.91 0.087  252   779     2.0    743 0.037 0.047 -403
6 0.38 0.67 0.024 147  1221 1.2e-168     6.6 0.92 0.086  206   673     2.1    604 0.033 0.045 -411
7 0.39 0.69 0.028 129  1004 2.6e-135     5.7 0.93 0.083  112   522     2.1    450 0.029 0.042 -441
8 0.37 0.65 0.030 112   843 1.6e-112     5.4 0.93 0.082   70   425     2.4    369 0.026 0.041 -405
>
```

As mentioned previously, the tables of statistics do not provide a clear answer to the question of which factor model is best (i.e. how many factors should be extracted). However, if we review the associated plot, we can clearly see the 4–factor model is the best (i.e. highest; even when embedded within models with more than 4 factors, with good separation from previous models).



2 Conclusions

The intent of this article was to raise awareness of the dangers of using only one criteria or method for deciding upon the number of factors to extract when conducting factor analysis. This article also demonstrated the ease with which an analyst can compute and evaluate several such criteria to reach a

more informed decision. More extensive examples of the data analysis solutions are available at the RSS *Do-it-yourself Introduction to R* course page:
`http://www.unt.edu/rss/class/Jon/R_SC/`
Lastly, a copy of the script file used for the above examples is available here:
`http://www.unt.edu/rss/class/Jon/Benchmarks/VerySimpleStructure.R`

Until next time; remember what George Carlin said: *"just 'cause you got the monkey off your back doesn't mean the circus left town."*

3 References & Resources

Bernaards, C., & Jennrich, R. (2014). The 'GPArotation' package. Documentation, including the package manual and the package vignette, available at CRAN:
`http://cran.r-project.org/web/packages/GPArotation/index.html`

Carlin, G. (1937 - 2008). *Just One-Liners*.
`http://www.just-one-liners.com/ppl/george-carlin`

Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research, 1*(2), 245 – 276.

Horn, J. (1965). A rationale and test for the number of factors in factor analysis. *Psychometrika, 30*(2), 179 – 185.

Horn, J. L., & Engstrom, R. (1979). Cattell's scree test in relation to bartlett's chi-square test and other observations on the number of factors problem. *Multivariate Behavioral Research, 14*(3), 283 – 300.

McDonald, R. P. (1999). *Test Theory: A Unified Treatment*. Mahwah, NJ: Erlbaum.

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine, 2*, 559 – 572.

Revelle, W. (2014). The 'psych' package. Documentation, including the package manual and the package vignette, available at CRAN:
`http://cran.r-project.org/web/packages/psych/index.html`

Revelle, W., & Rocklin, T. (1979). Very simple structure: An alternative procedure for estimating the optimal number of interpretable factors. *Multivariate Behavioral Research, 14*, 403 - 414. Available at:
`http://personality-project.org/revelle/publications/vss.pdf`

Spearman, C. (1904). General Intelligence: Objectively Determined and Measured. *American Journal of Psychology, 15*, 201 – 292.

Statistics Canada. (2010). *Survey Methods and Practices*. Ottawa, Canada: Minister of Industry.

http://www.statcan.gc.ca/bsolc/olc-cel/olc-cel?lang=eng&catno=12-587-X

Thompson, B. (2004). *Exploratory and confirmatory factor analysis: Understanding concepts and applications*. Washington, DC: American Psychological Association.

Velicer, W. (1976). Determining the number of components from the matrix of partial correlations. *Psychometrika, 41*(3), 321 – 327.

This article was last updated on December 1, 2014.

This document was created using LaTeX