Exploratory Factor Analysis in R Using Some Advanced Support Functions

James H. Steiger
Psychology 312
Spring, 2013

As we discussed in lecture on “Practical Factor Analysis,” there are several steps involved in completing a factor analytic study.

1. Design the Study
2. Gather the Data
3. Choose the Model
4. Select *m*, the Number of Factors
5. Rotate the Factors
6. Interpret the Factors and Name Them
7. Obtain Scale Scores (if needed)

All of these steps require a careful analytical approach. In this handout, we’ll assume that a common factor analysis is being performed on the data that have already been gathered or made available. In that case, the next step is to select *m*, the number of common factors.

Selecting *m* is partly science, partly art and intuition. Once *m* is determined, the factor analysis is performed using maximum likelihood, and if the resulting rotated factor pattern is sufficiently interpretable, the analysis can proceed. If, however, the rotated pattern does not exhibit either simple structure or bifactor structure, then the experimenter may be moved to try a different *m*, depending on how clearcut the statistical criteria were.

This step seems straightforward, but it is hampered somewhat by the necessity to perform several alternative analyses. For example, there are several major approaches to deciding on a number of factors:

1. The Scree Test
2. The Likelihood Ratio Test
3. The RMSEA Fit Criterion

The ability to perform these analyses quickly and efficiently is built into our support routines. In a similar vein, there are several major approaches to rotation:

1. Orthogonal Rotation
2. Oblique Transformation
3. Orthogonal Bifactor Rotation
4. Oblique Bifactor Rotation

All of the above rotational techniques are performed, and the output formatted, with one of our advanced support functions. Let’s see how this works.

We'll begin with a famous data set — the *24 Psychological Variables* of Holzinger and Swineford. This data set, based on a sample of $n = 145$ independent observations, has been discussed in many places in the literature.

The choice of the correct number of factors with these data is not as straightforward as in some analyses.

We begin by loading the data. Make sure the `psych` and `plotrix` libraries are loaded. We store the correlation matrix in a variable called R.

```
> library(psych)
> source(
> 'http://www.statpower.net/Content/312/R Stuff/AdvancedFactorFunctions.txt'
>       )
> R <- as.matrix(Harman74.cor$cov)
```

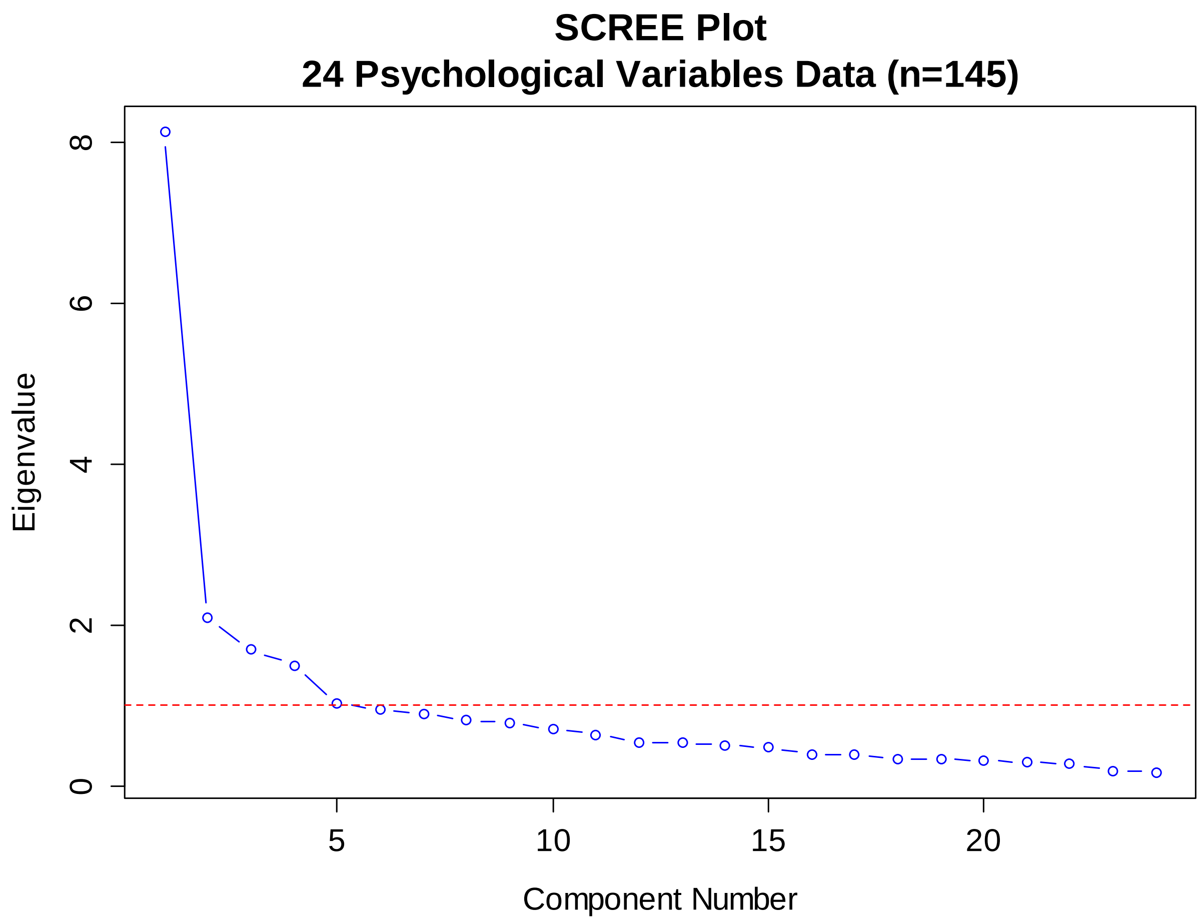
Next, we generate a scree.plot of the eigenvalues.

```
> Scree.Plot(R,main="SCREE Plot\n24 Psychological Variables Data
(n=145)")
```

The plot shows the eigenvalues of the correlation matrix in decreasing order. For ease of interpretation, a horizontal line is drawn at a height of 1.0.

As mentioned in class, one examines the plot for a noticeable "scree," a flattened area around 1.0. Eigenvalues to the left of the scree represent components that are worth retaining, and often correspond to the number of useful factors in a common factor analysis as well

The scree plot is somewhat ambiguous. The scree seems to start at the 5th factor, leading to a 4 factor solution. However, the eigenvalue of the 5th principal component is slightly larger than 1.



We move on to more advanced statistical criteria.

```
> FA.Stats(R,n.factors=1:5,n.obs=145,
> main="RMSEA Plot\n24 Psychological Variables Data  (n=145)",
> RMSEA.cutoff=0.05)
```

Here is the output. First we get a statistical table.

```
     Factors Cum.Eigen Chi-Square  Df             p.value RMSEA.Pt RMSEA.Lo RMSEA.Hi
[1,]       1    8.1354     622.91 252 0.00000000000000000 0.101100 0.091135 0.111126
[2,]       2   10.2315     420.24 229 0.00000000000020062 0.076153 0.064595 0.087541
[3,]       3   11.9241     295.59 207 0.00005121867331981 0.054517 0.039694 0.068100
[4,]       4   13.4259     226.68 186 0.02239559067132135 0.038974 0.015820 0.055621
[5,]       5   14.4511     186.82 166 0.12832636580567369 0.029513 0.000000 0.049486
```

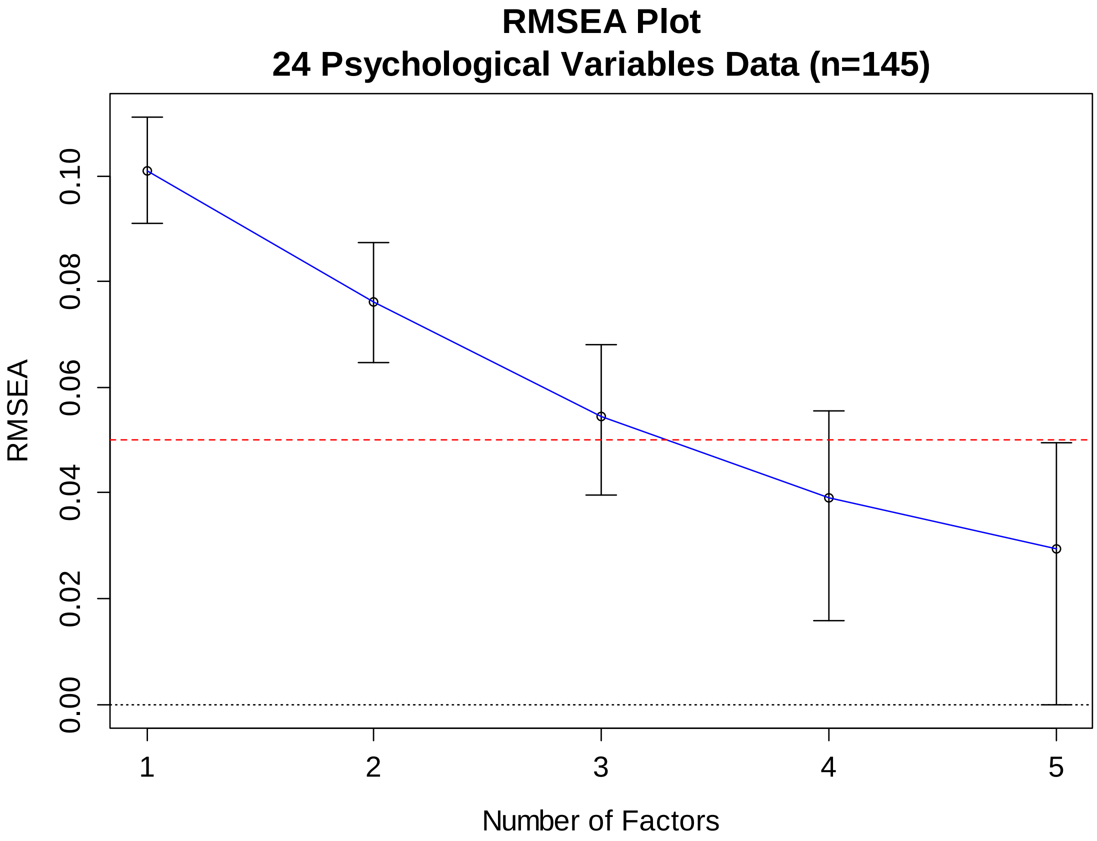
The chi-square statistic and its associated *p*-value test the hypothesis that the factor model fits perfectly. This hypothesis is rejected easily at the 0.05 level for 4 factors, but is not rejected for 5 factors. According to the traditional χ^2 “Accept-Support” logic, this might lead to a choice of a 5 factor solution.

Using fit critieria such as the RMSEA, the decision is more in doubt. With the RMSEA approach, one plots confidence intervals for the complexity-corrected index of model fit. This is discussed in great detail in the handout *Measures of Fit in Structural Equation Modeling*, available on the website in the Statistics Handouts section.

The RMSEA-based approach asks, "how well does the model fit, and how precisely have we determined it." The RMSEA index is crude, and can be criticized on a number of grounds, but the general idea seems to be superior to the chi-square approach. We no longer require fit to be perfect, but rather that it be determined with a reasonable degree of precision to be good.

A population RMSEA of 0.05 is considered by a number of experts to represent good fit. Some writers have misconstrued this to mean that a *point estimate* of 0.05 signifies good fit. It might, but under conditions of low precision (and a wide confidence interval), one might want to suspend judgment.

Surveying the confidence intervals in the RMSEA plot on the next page, we see that the intervals march steadily downward, with only a slight inflection. By the 4th factor, the point estimate is 0.039, and the confidence interval has an upper limit of 0.056. This is generally considered to be very good fit. Adding a 5th factor lowers the point estimate to 0.030, and the confidence interval includes zero (indicating a failure to reject the null hypothesis of perfect fit, i.e., RMSEA = 0, at the 0.05 level).



Classic sources and textbooks favored a 4-factor solution for the Holzinger-Swineford 24 variables data, and Jöreskög decided on a 4-factor solution in his December 1978 Psychometric Society Presidential Address article in *Psychometrika*.

Suppose we settle on 4 factors for now. The next step is to compute maximum likelihood factor loadings and rotate them to an interpretable structure. To make this relatively convenient for you, I've created some service functions.

A key service function is **`MLFA`**, which simultaneously produces several of the most commonly used rotation methods, reverses the sign of the loading if appropriate, orders the factors, blanks out loadings less than 0.25 in absolute value, and sorts the observed variables to help make simple structure more obvious. By default the **`MLFA`** routine returns all the loadings to 3 decimal places. However, the **`Loadings`** function allows you to reformat the loadings.

Included in the output are:

Unrotated (Orthogonal)
Varimax (Orthogonal)
Promax (Oblique)
Quartimin (Oblique)
Bifactor (Jennrich-Bentler 2011Orthogonal)
Bifactor (Jennrich-Bentler 2012 Oblique)

Additional rotational methods will be added as requested.

Here is the output from **`MLFA`**. I use the **`Loadings`** function to truncate loadings below 0.30.

```
out <- MLFA(Correlation.Matrix=R,n.factors=4,n.obs=145,promax.m=3)
Loadings(out,cutoff=.3,num.digits=2)
```

I'll skip over some of the output for efficiency.

Let's look at the varimax rotation first.

```
Varimax Loadings
------------------
                       Factor1 Factor2 Factor3 Factor4
GeneralInformation      0.74
PargraphComprehension   0.77
SentenceCompletion      0.81
WordClassification      0.57    0.34
WordMeaning             0.81
VisualPerception                0.69
PaperFormBoard                  0.57
Flags                           0.53
SeriesCompletion        0.37    0.50
Addition                                0.83
Code                                    0.51    0.37
CountingDots                            0.72
StraightCurvedCapitals          0.44    0.53
WordRecognition                                 0.55
NumberRecognition                               0.52
FigureRecognition               0.41            0.53
ObjectNumber                                    0.57
Cubes                           0.44
Deduction               0.38    0.40            0.30
NumericalPuzzles                0.38    0.44
ArithmeticProblems      0.37            0.50    0.30
NumberFigure                            0.34    0.46
FigureWord                                      0.37
ProblemReasoning        0.37    0.40            0.30

               Factor1 Factor2 Factor3 Factor4
SS loadings       3.65    2.87    2.66    2.29
Proportion Var    0.15    0.12    0.11    0.10
Cumulative Var    0.15    0.27    0.38    0.48
```

Ideally, each row would have only one nontrivial loading. But we see 7 variables with 2 and 3 variables with 3 nontrivial loadings.

Next, let’s check the Promax solution. Promax is perhaps the most frequently used oblique rotation method, although a number of experts consider quartimin to be superior.

```
Promax Loadings
-----------------
                       Factor1 Factor2 Factor3 Factor4
GeneralInformation      0.76
PargraphComprehension   0.79
SentenceCompletion      0.86
WordClassification      0.51
WordMeaning             0.84
VisualPerception                0.78
PaperFormBoard                  0.66
Flags                           0.58
Addition                                0.92
CountingDots                            0.73
WordRecognition                                 0.62
NumberRecognition                               0.58
FigureRecognition               0.37            0.54
ObjectNumber                                    0.63
Cubes                           0.49
StraightCurvedCapitals          0.46    0.45
SeriesCompletion                0.47
Code                                    0.46    0.31
ArithmeticProblems                      0.43
NumberFigure                                    0.44
FigureWord                                      0.35
Deduction                       0.34
NumericalPuzzles                0.36    0.34
ProblemReasoning                0.34

               Factor1 Factor2 Factor3 Factor4
SS loadings       3.26    2.82    2.28    2.01
Proportion Var    0.14    0.12    0.09    0.08
Cumulative Var    0.14    0.25    0.35    0.43

Factor Intercorrelations
------------------------
        Factor1 Factor2 Factor3 Factor4
Factor1    1.00    0.53    0.37    0.47
Factor2    0.53    1.00    0.43    0.52
Factor3    0.37    0.43    1.00    0.45
Factor4    0.47    0.52    0.45    1.00
```

Here we see 4 variables with 2 nontrivial loadings, but none with 3 nontrivial loadings. Notice, however, that the 4 factors are correlated round 0.40 – 0.50, making ultimate interpretability somewhat more problematic.

Next, let’s look at an orthogonal bifactor solution (it is very similar to the oblique bifactor solution that follows, because the factors are almost uncorrelated in the oblique solution.

```
Orthogonal Bifactor Loadings
----------------------------
                       Factor1 Factor2 Factor3 Factor4
VisualPerception        0.68
Flags                   0.51
WordClassification      0.59    0.39
Code                    0.56            0.32
CountingDots            0.56            0.43
StraightCurvedCapitals  0.67
FigureRecognition       0.54                    0.31
NumberFigure            0.57
Deduction               0.58
NumericalPuzzles        0.63
ProblemReasoning        0.57
SeriesCompletion        0.68
ArithmeticProblems      0.61
GeneralInformation      0.54    0.59
PargraphComprehension   0.52    0.64
SentenceCompletion      0.50    0.67
WordMeaning             0.51    0.69
Addition                0.48            0.72
Cubes                   0.41
PaperFormBoard          0.46           -0.37
FigureWord              0.44
WordRecognition         0.37                    0.45
NumberRecognition       0.37                    0.41
ObjectNumber            0.44                    0.44

               Factor1 Factor2 Factor3 Factor4
SS loadings       7.02    2.09    1.38    0.98
Proportion Var    0.29    0.09    0.06    0.04
Cumulative Var    0.29    0.38    0.44    0.48
```

Once the general factor is extracted, The remaining factors show a simple structure. Besides the general factor, no variable loads above 0.30 on more than one variable, and the clusters have a rather clear interpretation.

Now you try it. The **Thurstone** data set, is a correlation matrix based on $n = 213$ observations. Load it as follows:

```
> data(Thurstone)
```

Then factor analyze it, deciding on a number of factors and a simple structure. Justify all your decisions.