Supplemental Material

CBE—Life Sciences Education

Knekta *et al.*

Supplemental material
Section 1. Checking properties of the data

Summary of the most essential analysis for checking the data for EFA/CFA purposes.

Missing values and outliers.

Data should always be checked for missing values. Some statistical software cannot handle data with missing values. Yet, simply deleting participants with missing values has the potential to bias the results of any subsequent analysis. Participants with large percentages of missing values (e.g., > 90%) should always be considered for possible deletion, as these individuals may not have taken the survey seriously and imputing that much missing data can be problematic. In most other instances, best practice dictates that a missing data technique, such as full information maximum likelihood or multiple imputation (Enders, 2010), should be used to estimate missing values. If the number of missing values across the data set is relatively low (< 5%) and the data are deemed to be missing at random (i.e. there is no systematic reason for the missing responses[1]), the methods used for estimation of the missing values will have only a minor impact on the results (Tabachnick and Fidell, 2013). In these cases individuals with a few missing values might even be deleted, but if so, this should be reported and justified in the methods section of the manuscript. If the missing values are non-randomly distributed or if there is a substantial amount of missing data across participants, a missing data technique must be used in order to retain as much of the sample as possible.

Outliers can also have a large effect on the factor analysis, therefore the presence of both univariate and multivariate outliers have to be considered before performing factor analysis. Univariate outliers can be found by looking at descriptive statistics such as the minimum and maximum values for a survey items, as well as frequency histograms of the responses. Multivariate outliers can be screened for using, for example, Mahalanobis distance, leverage, and influence (Raykov and Marcoulides, 2008; Tabachnick and Fidell, 2013). On surveys, multivariate outliers are respondents having an unusual reply pattern. An example of a multivariate outlier that could be subject to deletion would be a respondent replying 1 to all items on a Likert scale, although some items are reverse-scored. This would indicate that the respondent was string responding (choosing the same response for all items), and thus not providing valid responses to the survey items.

***Factorability*.**

To be able to find or confirm factors within a data there first has to be, at least, a few sizeable correlations between individual items. This so-called factorability can be tested with Kaiser's measure of sampling adequacy, values of 0.6 and above are suggest good factorabilty (Tabachnick and Fidell, 2013). Also, an inter-item-correlation matrix should be used to visually inspect the data and confirm that there are numerous moderate to strong correlations among the items. The expected size of the correlation depends to some extent on the sample size, but inter-item correlations should at least exceed 0.30 if they are expected to be on the same factor (Tabachnick and Fidell, 2013).

---

[1]Whether the data is missing at random or not can be tested with the Little's Missing Completely at Random (MCAR) test[1]. It should be noted that Little's MCAR only tests the null hypothesis that the observed missing responses are consistent with MCAR missing data but cannot definitively show that the missingness is truly MCAR (Enders, 2010).

Normality and linearity.

Univariate and multivariate normality within the data are preferable when performing factor analysis (Raykov and Marcoulides, 2008; Tabachnick and Fidell, 2013). Univariate normality can be assessed by measures of skewness and kurtosis. A graphical examination of a frequency histogram for each question can be helpful. A common guideline is that skewness and kurtosis should be less than |2.0| (Bandalos and Finney, 2010). Some researchers suggest a more liberal standard for kurtosis, < |7.0| (Bandalos and Finney, 2010). Mardia's multivariate normality test is one commonly available test for multivariate normality. Significant multivariate skewness or kurtosis values indicate multivariate non-normality. Factor analysis is a multivariate procedure and one can have multivariate non-normality even when all univariate statistics indicate that the data is normal.

Factor analysis is built on the analysis of the covariance matrix in the data and assumes linear relationships between items and between items and the factors (Raykov and Marcoulides, 2008; Tabachnick and Fidell, 2013). Linearity can be checked by inspecting scatter plots between two variables or residual plots. If severe nonlinearity and or non-normality are found, transformation of variables can be considered. If moderate non-normality exists, polyserial or polychoric correlation coefficients or estimators robust against non-normality (for example MLR or PAF) can be considered. Non-linear factor analysis methods exist (Yalcin and Amemiya, 2001), but are far less common.

Multicollinearity

For factor analysis, it is important that the variables not are too highly correlated (multicollinearity). Multicollinearity can cause statistically unstable and unreliable results. Multicollinearity can be detected with the help of the variance inflation factor (VIF) or tolerance. If the VIF is above 10 or the value of tolerance is less than 0.1, multicollinearity is problematic. Multicollinearity can depend on two variables being too highly correlated (correlation >0.90) or too many moderately high correlations over a number of items (too many items with correlations > 0.70). If multicollinearity is indicated, one or several of the variables that are too highly correlated should preferably be deleted.

Example of how to analyze whether the data meet the assumptions for a factor analysis using Diekman's et al. (2010) goal endorsement example.

Missing values and outliers

No items were missing more than 1.3% of their values and this missingness was at random (Little's MCAR test: Chi-Square = 677.719, df = 625, Sig. = 0.075 implemented with the BaylorEdPsych package; Beaujean, 2012). Thus, missing data should only have a minor impact on the results and missing data handling procedures would yield similar results. For the EFA, only cases with complete items were used in the analysis. For the CFA, we used full-information maximum likelihood in the estimation procedure to handle the missingness. Minimum and maximum values of the items were analyzed to ensure no univariate outliers. Thirty-eight (38) cases with high Mahalanobis distance ($p < 0\ .001$) were identified as potential multivariate outliers. Each of the 38 cases were inspected in detail and we found no justification for removing any of the responses.

Factorability

The inter-item correlation matrix showed that there were several correlations above 0.3 and the Kaiser's measure of sampling adequacy value was 0 .91 which indicated good factor ability. The Kaiser test was implemented using the psych package (Revelle, 2017) and correlations were visualized using corrplot package (Wei and Simko, 2017).

Normality and linearity

Most items had a skewness and kurtosis below |1.0| and all items had a skewness below |2.0| and kurtosis below |4.0|. In alignment with these tests, graphical examination of frequency histograms did indicate a slight negative skewness for some items. Thus, most of the items were univariate normal but some items showed slight non-normality. These same items showed some indications of non-linearity when regressed against other items and when examining standardized residual plots. Mardia's multivariate normality test (implemented with the psych package, Revelle 2017) showed significant multivariate skewness and kurtosis values which indicated multivariate non-normality. We employed robust estimation methods to handle non-normality and non-linearity, in subsequent factor analyses.

Multicollinearity

Multicollinearity was investigated by examining inter-item-correlations and tolerance values from multiple regressions implemented with the olsrr package (Hebbali, 2018). The highest inter-item correlation was 0.73 and the lowest tolerance was 0.36. Thus, the data did not show evidence of multicollinearity.

References

Bandalos, D. J. & Finney S. J. (2010). Factor analysis. Exploratory and confirmatory. In Hancock G. R., and Mueller R. O. (Eds), *The Reviewer's Guide to Quantitative Methods in the Social Science* (pp. 93 – 114). New York: Routledge.

Beaujean, A. A. (2012). BaylorEdPsych: R package for Baylor University Educational Psychology Quantitative Courses. Available from https://CRAN.R-project.org/package=BaylorEdPsych. Retrived February 24, 2019

Diekman A. B., Brown E. R., Johnston A. M., & Clark E. K. (2010). Seeking congruity between goals and roles: A new look at why women opt out of science, technology, engineering, and mathematics careers. *Psychological Science 21*(8), 1051-1057.

Enders C. K. (2010). Applied missing data analysis. Guilford Press, New York

Hebbali, A. (2018) olsrr: Tools for building OLS Regression Models, version 0.5.1. Available from https://cran.r-project.org/web/packages/olsrr. Retrived February 24, 2019

Raykov, T. & Marcoulides G. A. (2008). *An Introduction to Applied Multivariate Analysis.* New York: Routledge.

Revelle, W. (2017) psych: Procedures for Personality and Psychological Research (Version 1.7.8), Northwestern University, Evanston, Illinois, USA. Available from https://CRAN.R-project.org/package=psych. Retrived February 24, 2019

Tabachnick B.G. & Fidell L. S. (2013). *Using Multivariate Statistics*, 6th Ed. Boston: Pearson

Wei, T., & Simko, V. (2017). R package "corrplot": Visualization of a Correlation Matrix (Version 0.84). Available from https://github.com/taiyun/corrplot. Retrived February 24, 2019

Yalcin, I., & Amemiya, Y. (2001). Nonlinear factor analysis as a statistical method. *Statistical science, 16*(3), 275-294.

SupplementalMaterial

Section 2. Interpreting R Output from a CFA and EFA.

The resulting output from a CFA or EFA run in Ris immense and can feeloverwhelming in the beginning. In this annotated output section, we highlight the aspects of the output that we recommend using in Knekta et al. to assess how well the models fit the data. The output is based on the example data set provided in the Supplemental Materials for this paper titled “EFAsampledata.csv” as well as the example R syntax provided (“EFACFAexamplesyntax.R”). The data in the example dataset is modeled after the real dataset used in the paperbut is smaller, so values will not be identical to that in the paper.

Key to annotation:
Blue text in Courier New font: The syntax input used to return this output in R.
Black text in Courier New font: R output returned.
Red boxes:the pieces of the outputwe recommendfocusing on to interpret a CFA or EFA.
*Text beginning with * and in Times New Roman Font:* Notes on our interpretation of the output highlighted in the red boxes as well as explanations to help the reader interpret the output.

A. Confirmatory factor analysis output

```
CFA2<- 'Self =~ go1 + go2 + go3 + go4 + go5 + go6 + go7 + go8 + go9 +
go10 + go11 + go12 + go13 + go14
Other =~ go15 + go16 + go17 + go18 + go19 + go20 + go21 + go22 + go23'

C2f_fit <- cfa(CFA2, estimator = "mlr", missing = "fiml", data= data)

summary(C2f_fit, fit.measures=TRUE,  standardized=TRUE, rsquare=TRUE)
```

```
lavaan (0.5-23.1097) converged normally after  36 iterations

  Number of observations                           365

  Number of missing patterns                         1

  Estimator                                         ML      Robust
Minimum Function Test Statistic                879.302     712.881
  Degrees of freedom                               229         229
  P-value (Chi-square)                           0.000       0.000
  Scaling correction factor                                  1.233
    for the Yuan-Bentler correction
```

* This is the chi-squared test of model fit. Significance indicates the data do not fit the model.For this and the rest of the model fit statistics, we focus on the robust results column (labeled “Robust”) rather than the maximum likelihood results column (“ML”). The robust tests account for the non-normality and non-linearity that we observed when we examined our raw data (see Supplemental Materials Section 1 for what to look for in raw data before running a factor analysis).

```
Model test baseline model:

  Minimum Function Test Statistic             3651.000    2899.963
  Degrees of freedom                               253         253
  P-value                                        0.000       0.000

User model versus baseline model:

Comparative Fit Index (CFI)                     0.809       0.817
Tucker-Lewis Index (TLI)                        0.789       0.798

Robust Comparative Fit Index (CFI)                          0.821
```

*CFI measures whether the current model fits the data better than a model assuming no relationships between the observed variables. In our case, this incremental fit index is less than the .95 cutoff, indicating a lack of fit.

```
  Robust Tucker-Lewis Index (TLI)                            0.802

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)            -12800.248  -12800.248
  Scaling correction factor                                 1.276
    for the MLR correction
  Loglikelihood unrestricted model (H1)    -12360.598  -12360.598
  Scaling correction factor                                 1.244
    for the MLR correction

  Number of free parameters                        70          70
  Akaike (AIC)                              25740.497   25740.497
  Bayesian (BIC)                            26013.490   26013.490
  Sample-size adjusted Bayesian (BIC)       25791.409   25791.409

Root Mean Square Error of Approximation:

RMSEA                                           0.088       0.076
  90 Percent Confidence Interval          0.082  0.094       0.070  0.082
  P-value RMSEA <= 0.05                          0.000       0.000

Robust RMSEA                                                0.085
  90 Percent Confidence Interval                             0.078 0.092
```

*RMSEA is a parsimony adjusted fit index. It measures how closely the model reproduces the actual observed data patterns. More complex models (in this case, models with more factors) will, by their nature, more closely match the observed data, but that does not necessarily increase the model's predictive ability. To account for this issue, RMSEA introduces a penalty for model complexity. In this case,RMSEA exceeds the 0.06 cut off and the 90% confidence interval is also bounded away from 0.06.Together, these measuresindicate a lack of fit.

```
Standardized Root Mean Square Residual:
```

```
SRMR                                           0.080       0.080
```

*SRMR is an absolute fit index. SRMR is the average difference between the sample variances and covariances and the model estimated variances and covariances(i.e. the average correlation residuals which appears later in this output). In our case, it indicates adequate fit as it just meets the .08 cut off.

```
Parameter Estimates:

  Information                                  Observed
  Standard Errors                    Robust.huber.white

Latent Variables[1]:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  Self =~
    go1               1.000                               0.938    0.617
    go2               1.087    0.089   12.162    0.000    1.020    0.665
    go3               0.671    0.088    7.660    0.000    0.630    0.611
    go4               0.418    0.083    5.047    0.000    0.392    0.406
    go5               1.093    0.090   12.103    0.000    1.026    0.646
    go6               0.669    0.109    6.128    0.000    0.628    0.534
    go7               0.822    0.118    6.949    0.000    0.772    0.566
    go8               0.953    0.084   11.320    0.000    0.894    0.600
    go9               0.922    0.122    7.564    0.000    0.865    0.596
    go10              0.554    0.083    6.637    0.000    0.519    0.547
    go11              0.698    0.085    8.216    0.000    0.655    0.471
    go12              0.692    0.095    7.308    0.000    0.649    0.561
    go13              0.564    0.089    6.310    0.000    0.529    0.511
    go14              0.785    0.088    8.876    0.000    0.736    0.474
  Other =~
    go15              1.000                               0.978    0.869
    go16              1.091    0.059   18.623    0.000    1.067    0.795
    go17              1.110    0.060   18.628    0.000    1.085    0.800
    go18              0.809    0.079   10.280    0.000    0.791    0.577
    go19              0.687    0.079    8.732    0.000    0.672    0.540
    go20              1.037    0.068   15.192    0.000    1.014    0.754
    go21              0.998    0.052   19.256    0.000    0.976    0.840
    go22              0.650    0.091    7.135    0.000    0.636    0.417
    go23              0.928    0.088   10.532    0.000    0.908    0.466
```

*The above section focuses on the factors specified by the researcher.For each factor (*Self* and *Other*) we are given information about how each individual item (go1 – go23) is associated with the factor we have specified that it should represent (i.e. the factor loading). The first column (Estimate) is the factor loadingfor each item. This is either estimated by the model or fixed.Notice that under each factor the first item has an estimate of 1. These items are fixed to 1 by the researcher or by the CFA program. This is done to give the factor an interpretable scale. Thefactor loading can be interpreted similarlyto regression coefficients: for each unit increase in the appropriatefactor (*Self* or *Other*), the model predicts an estimated increase in the specific item. For example, a one-unit increase in *Self* predicts a 1.087

[1] In the main paper we refer to the latent variables as constructs

increase in go2.The values are in the original metric of the item used to set the scale, so values greater than 1.00 have stronger relationships to the latent variable than the reference item, and weaker otherwise. The next column (Std.err) is the standard error ofthis estimate for each parameter.

The third and fourth columnsare related to the Wald test (Z-value and P(z>|z|)), which tests whether the value of the factor loading is significantly different from zero and, thus,actually contributes to the factor. If an item does not contribute to the factor, this indicates the item is not working well in the specified model.

The final two columns are only printed because we set standardized=TRUE in the initial model. These columns are standardized factor loadings. In the first column (Std.lv) only the factors (*Self* and *Other*) were standardized. In the second column (Std.all) both the factors and the items themselves are standardized. This second column is the one we focus on and is probably the most easily interpretable column in this entire table. Since the factor loadings in the Std.all columnare standardized, the values are bounded between -1 and 1. On the *Self*-factor items, we see that the factor explains more of go5 than it does of go10.

```
Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  Self ~~
    Other             0.320    0.072    4.430    0.000    0.349    0.349
```

* The Covariance table measures the degrees to which two variables in the model relate to one another. In our case, we are testing to what degree the factors are correlated with one another. The positive covariance tells us that as one factor increases in value so does the other. It is the default in the CFA command for thefactors to be allowed to covary. Std.all tells us the strength of the correlation between the two factors. This correlation is at the latent level. This means that if a researcher used the mean values from each itemto calculate summed scores for the *Self* and the*Other* factor, and we correlated these two summed scores, then the resulting value would not be 0.349. Instead, 0.349 is an “error-free” correlation, as the measurement errors in the items for each factor is removed in the factor analysis process.

```
Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .go1               4.159    0.080   52.289    0.000    4.159    2.737
.go2                 4.825    0.080   60.092    0.000    4.825    3.145
   .go3               6.214    0.054  115.240    0.000    6.214    6.032
   .go4               6.274    0.051  124.083    0.000    6.274    6.495
   .go5               4.595    0.083   55.289    0.000    4.595    2.894
   .go6               5.795    0.061   94.248    0.000    5.795    4.933
   .go7               5.510    0.071   77.277    0.000    5.510    4.045
   .go8               4.455    0.078   57.114    0.000    4.455    2.989
   .go9               5.159    0.076   67.912    0.000    5.159    3.555
   .go10              6.367    0.050  128.164    0.000    6.367    6.708
   .go11              5.258    0.073   72.233    0.000    5.258    3.781
   .go12              5.830    0.061   96.185    0.000    5.830    5.035
   .go13              6.082    0.054  112.194    0.000    6.082    5.872
   .go14              4.584    0.081   56.374    0.000    4.584    2.951
   .go15              6.126    0.059  104.005    0.000    6.126    5.444
   .go16              5.932    0.070   84.446    0.000    5.932    4.420
```

```
   .go17              5.742    0.071   80.867    0.000    5.742    4.233
   .go18              5.460    0.072   76.056    0.000    5.460    3.981
   .go19              5.858    0.065   89.979    0.000    5.858    4.710
   .go20              5.523    0.070   78.410    0.000    5.523    4.104
   .go21              6.060    0.061   99.656    0.000    6.060    5.216
   .go22              4.866    0.080   61.045    0.000    4.866    3.195
   .go23              4.562    0.102   44.725    0.000    4.562    2.341
    Self              0.000                               0.000    0.000
    Other             0.000                               0.000    0.000
```

*The Intercepts tableof a CFA tell us the expected value for an item when the all the predictors are 0. In our model, the factor (*Other* or *Self*) is the only predictor for each item and it is standardized so that 0 is the mean value for the factor. Thus, the intercept estimate (column 1) for each item is simply the mean value for each item on the raw scale. We can, for example, see that most students seem to find success (item go10) to be important for themselves (Estimate = 6.367 were 7 = very important).

```
Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .go1               1.429    0.134   10.698    0.000    1.429    0.619
   .go2               1.313    0.145    9.056    0.000    1.313    0.558
   .go3               0.664    0.082    8.097    0.000    0.664    0.626
   .go4               0.779    0.091    8.588    0.000    0.779    0.835
   .go5               1.469    0.130   11.280    0.000    1.469    0.583
   .go6               0.986    0.090   10.971    0.000    0.986    0.715
   .go7               1.260    0.123   10.234    0.000    1.260    0.679
   .go8               1.422    0.126   11.273    0.000    1.422    0.640
   .go9               1.359    0.124   10.988    0.000    1.359    0.645
   .go10              0.631    0.068    9.218    0.000    0.631    0.701
   .go11              1.505    0.149   10.069    0.000    1.505    0.778
   .go12              0.920    0.096    9.570    0.000    0.920    0.686
   .go13              0.793    0.091    8.701    0.000    0.793    0.739
   .go14              1.871    0.166   11.280    0.000    1.871    0.775
   .go15              0.310    0.047    6.622    0.000    0.310    0.245
   .go16              0.662    0.091    7.312    0.000    0.662    0.368
   .go17              0.663    0.079    8.348    0.000    0.663    0.360
   .go18              1.256    0.118   10.647    0.000    1.256    0.667
   .go19              1.095    0.107   10.212    0.000    1.095    0.708
   .go20              0.782    0.103    7.592    0.000    0.782    0.432
   .go21              0.398    0.058    6.873    0.000    0.398    0.295
   .go22              1.915    0.164   11.707    0.000    1.915    0.826
   .go23              2.973    0.211   14.105    0.000    2.973    0.783
    Self              0.880    0.156    5.646    0.000    1.000    1.000
    Other             0.957    0.127    7.561    0.000    1.000    1.000
```

*The variance table represents the error variance or the amount of the actual scores on the items that are not predicted by the model we entered. From the Wald test columns (z-value; p>|z|) we see that the factors do not perfectly predict the observed values; the estimated residual varianceis significantly different from zero for all items and factors.This is expected and does not indicate whether a model is a good fitto the data.This variance table is related to the R^2 table below (1 – the standardized error variance = R^2).

```
R-Square:
                   Estimate
    go1               0.381
    go2               0.442
    go3               0.374
    go4               0.165
    go5               0.417
    go6               0.285
    go7               0.321
    go8               0.360
    go9               0.355
    go10              0.299
    go11              0.222
    go12              0.314
    go13              0.261
    go14              0.225
    go15              0.755
    go16              0.632
    go17              0.640
    go18              0.333
    go19              0.292
    go20              0.568
    go21              0.705
    go22              0.174
    go23              0.217
```

*R^2 are the standardized factor loadings squared. It demonstrates how much of the variance in the item can be explained by the factor.R^2 can also be calculated by subtracting the standardized error variance in the previous table from 1. Most of the R^2 values in the proposed model are below the .5 cutoff indicating that thefactors are not responsible for explaining a majority of the variance in many of the items. Instead, unmodeled factors or measurement error are contributing most the responses to these items. This hints at the potentially poor fit of the model to the data.

```
C2f_mods <- modificationIndices(C2f_fit, minimum.value = 2)
C2f_mods

      lhs op  rhs     mi mi.scaled    epc sepc.lv sepc.all sepc.nox
75   Self =~ go15  9.931     8.051 -0.133  -0.125   -0.111   -0.111
78   Self =~ go18 12.252     9.933  0.260   0.244    0.178    0.178
79   Self =~ go19 11.878     9.630  0.238   0.223    0.179    0.179
82   Self =~ go22 25.703    20.838  0.458   0.430    0.282    0.282
86  Other =~  go3 10.575     8.574  0.168   0.164    0.160    0.160
87  Other =~  go4 10.318     8.366  0.174   0.170    0.176    0.176
92  Other =~  go9  5.092     4.128 -0.166  -0.162   -0.112   -0.112
94  Other =~ go11  3.002     2.434 -0.131  -0.128   -0.092   -0.092
95  Other =~ go12  2.038     1.653  0.086   0.084    0.072    0.072
96  Other =~ go13  5.579     4.523  0.131   0.128    0.123    0.123
97  Other =~ go14  7.289     5.909  0.228   0.223    0.144    0.144
98    go1 ~~  go2  7.044     5.711  0.217   0.217    0.093    0.093
99    go1 ~~  go3  6.633     5.377 -0.146  -0.146   -0.093   -0.093
100   go1 ~~  go4  5.738     4.652 -0.141  -0.141   -0.096   -0.096
101   go1 ~~  go5  9.203     7.461  0.260   0.260    0.108    0.108
102   go1 ~~  go6 10.373     8.410 -0.218  -0.218   -0.122   -0.122
103   go1 ~~  go7  6.774     5.492 -0.201  -0.201   -0.097   -0.097
104   go1 ~~  go8 25.390    20.584  0.417   0.417    0.184    0.184
107   go1 ~~ go11  8.741     7.086  0.245   0.245    0.116    0.116
108   go1 ~~ go12  4.518     3.663 -0.140  -0.140   -0.080   -0.080
109   go1 ~~ go13 10.880     8.820 -0.200  -0.200   -0.127   -0.127
110   go1 ~~ go14  8.639     7.004  0.271   0.271    0.115    0.115
112   go1 ~~ go16  3.133     2.540  0.101   0.101    0.049    0.049
114   go1 ~~ go18  3.523     2.856  0.140   0.140    0.067    0.067
117   go1 ~~ go21  2.933     2.378 -0.078  -0.078   -0.044   -0.044
118   go1 ~~ go22  4.731     3.836  0.198   0.198    0.086    0.086
121   go2 ~~  go4  2.560     2.075 -0.092  -0.092   -0.062   -0.062
122   go2 ~~  go5  6.633     5.378  0.216   0.216    0.089    0.089
123   go2 ~~  go6  6.720     5.448 -0.171  -0.171   -0.095   -0.095
124   go2 ~~  go7  7.116     5.769 -0.201  -0.201   -0.096   -0.096
125   go2 ~~  go8 28.156    22.827  0.429   0.429    0.188    0.188
126   go2 ~~  go9  5.633     4.567 -0.187  -0.187   -0.084   -0.084
128   go2 ~~ go11  2.760     2.237  0.134   0.134    0.063    0.063
129   go2 ~~ go12 14.905    12.084 -0.248  -0.248   -0.140   -0.140
131   go2 ~~ go14  6.452     5.231  0.229   0.229    0.096    0.096
132   go2 ~~ go15  2.809     2.277  0.068   0.068    0.039    0.039
133   go2 ~~ go16  6.706     5.437 -0.143  -0.143   -0.070   -0.070
134   go2 ~~ go17  8.116     6.580 -0.158  -0.158   -0.076   -0.076
136   go2 ~~ go19  3.113     2.524  0.119   0.119    0.062    0.062
139   go2 ~~ go22  5.673     4.600  0.210   0.210    0.090    0.090
141   go3 ~~  go4  7.149     5.796  0.107   0.107    0.108    0.108
142   go3 ~~  go5 10.436     8.461 -0.188  -0.188   -0.115   -0.115
147   go3 ~~ go10 22.689    18.395  0.176   0.176    0.180    0.180
148   go3 ~~ go11  8.474     6.870 -0.164  -0.164   -0.114   -0.114
150   go3 ~~ go13  9.302     7.542  0.126   0.126    0.118    0.118
151   go3 ~~ go14  4.455     3.612 -0.133  -0.133   -0.083   -0.083
154   go3 ~~ go17  3.546     2.875  0.073   0.073    0.052    0.052
158   go3 ~~ go21  2.288     1.855  0.047   0.047    0.039    0.039
159   go3 ~~ go22  8.479     6.874 -0.180  -0.180   -0.115   -0.115
161   go4 ~~  go5  3.272     2.652 -0.109  -0.109   -0.071   -0.071
164   go4 ~~  go8  8.908     7.222 -0.175  -0.175   -0.122   -0.122
165   go4 ~~  go9 13.774    11.167 -0.212  -0.212   -0.152   -0.152
```

334	167	go4	~~	go11	7.773	6.302	-0.164	-0.164	-0.122	-0.122
335	168	go4	~~	go12	6.147	4.984	0.116	0.116	0.104	0.104
336	169	go4	~~	go13	54.431	44.129	0.317	0.317	0.317	0.317
337	170	go4	~~	go14	2.243	1.818	0.098	0.098	0.066	0.066
338	176	go4	~~	go20	5.893	4.778	-0.106	-0.106	-0.082	-0.082
339	177	go4	~~	go21	2.491	2.020	0.052	0.052	0.046	0.046
340	179	go4	~~	go23	5.924	4.802	0.199	0.199	0.105	0.105
341	180	go5	~~	go6	3.031	2.457	-0.121	-0.121	-0.065	-0.065
342	181	go5	~~	go7	6.877	5.576	-0.207	-0.207	-0.096	-0.096
343	182	go5	~~	go8	12.149	9.849	0.296	0.296	0.125	0.125
344	184	go5	~~	go10	3.951	3.204	-0.111	-0.111	-0.073	-0.073
345	185	go5	~~	go11	8.248	6.687	0.243	0.243	0.110	0.110
346	187	go5	~~	go13	4.928	3.995	-0.137	-0.137	-0.084	-0.084
347	188	go5	~~	go14	3.174	2.573	0.168	0.168	0.068	0.068
348	190	go5	~~	go16	3.122	2.531	0.103	0.103	0.048	0.048
349	192	go5	~~	go18	3.991	3.236	0.152	0.152	0.070	0.070
350	194	go5	~~	go20	11.001	8.919	-0.206	-0.206	-0.096	-0.096
351	196	go5	~~	go22	2.100	1.702	0.135	0.135	0.056	0.056
352	197	go5	~~	go23	6.798	5.512	0.302	0.302	0.098	0.098
353	198	go6	~~	go7	42.154	34.176	0.408	0.408	0.255	0.255
354	199	go6	~~	go8	11.473	9.302	-0.228	-0.228	-0.130	-0.130
355	200	go6	~~	go9	13.840	11.221	0.244	0.244	0.143	0.143
356	203	go6	~~	go12	18.879	15.306	0.233	0.233	0.171	0.171
357	205	go6	~~	go14	4.820	3.908	-0.165	-0.165	-0.090	-0.090
358	215	go7	~~	go8	9.807	7.951	-0.240	-0.240	-0.118	-0.118
359	216	go7	~~	go9	41.792	33.883	0.484	0.484	0.245	0.245
360	219	go7	~~	go12	11.982	9.714	0.211	0.211	0.134	0.134
361	221	go7	~~	go14	4.255	3.450	-0.177	-0.177	-0.083	-0.083
362	222	go7	~~	go15	3.497	2.835	-0.072	-0.072	-0.047	-0.047
363	224	go7	~~	go17	2.692	2.183	0.087	0.087	0.047	0.047
364	231	go8	~~	go9	2.949	2.390	-0.138	-0.138	-0.064	-0.064
365	232	go8	~~	go10	4.025	3.263	-0.108	-0.108	-0.077	-0.077
366	233	go8	~~	go11	6.937	5.624	0.216	0.216	0.104	0.104
367	234	go8	~~	go12	18.283	14.823	-0.279	-0.279	-0.162	-0.162
368	235	go8	~~	go13	5.027	4.076	-0.135	-0.135	-0.087	-0.087
369	244	go8	~~	go22	4.331	3.511	0.188	0.188	0.083	0.083
370	245	go8	~~	go23	2.083	1.689	0.163	0.163	0.056	0.056
371	248	go9	~~	go12	4.600	3.729	0.137	0.137	0.081	0.081
372	250	go9	~~	go14	2.752	2.231	-0.148	-0.148	-0.066	-0.066
373	260	go10	~~	go11	5.012	4.063	0.121	0.121	0.092	0.092
374	263	go10	~~	go14	13.809	11.195	-0.224	-0.224	-0.152	-0.152
375	264	go10	~~	go15	2.027	1.643	-0.039	-0.039	-0.036	-0.036
376	265	go10	~~	go16	3.131	2.538	-0.066	-0.066	-0.052	-0.052
377	266	go10	~~	go17	3.748	3.039	0.073	0.073	0.056	0.056
378	271	go10	~~	go22	3.398	2.755	-0.110	-0.110	-0.076	-0.076
379	272	go10	~~	go23	4.284	3.473	-0.154	-0.154	-0.083	-0.083
380	273	go11	~~	go12	7.847	6.362	-0.183	-0.183	-0.114	-0.114
381	274	go11	~~	go13	4.098	3.323	-0.122	-0.122	-0.085	-0.085
382	276	go11	~~	go15	2.301	1.866	-0.063	-0.063	-0.041	-0.041
383	279	go11	~~	go18	2.162	1.753	0.110	0.110	0.058	0.058
384	285	go12	~~	go13	3.819	3.096	0.093	0.093	0.078	0.078
385	299	go13	~~	go17	3.973	3.221	-0.083	-0.083	-0.059	-0.059
386	303	go13	~~	go21	6.188	5.017	0.083	0.083	0.069	0.069
387	312	go14	~~	go21	3.079	2.496	-0.090	-0.090	-0.050	-0.050
388	315	go15	~~	go16	7.860	6.372	0.093	0.093	0.062	0.062
389	317	go15	~~	go18	14.049	11.390	-0.150	-0.150	-0.097	-0.097
390	318	go15	~~	go19	11.616	9.417	-0.126	-0.126	-0.090	-0.090
391	320	go15	~~	go21	6.249	5.066	0.070	0.070	0.054	0.054

```
321  go15 ~~ go22  8.762     7.103 -0.142  -0.142   -0.083   -0.083
323  go16 ~~ go17 27.980    22.684  0.228   0.228    0.125    0.125
324  go16 ~~ go18  3.858     3.128 -0.105  -0.105   -0.057   -0.057
325  go16 ~~ go19  2.528     2.050 -0.079  -0.079   -0.047   -0.047
326  go16 ~~ go20 16.611    13.467 -0.183  -0.183   -0.102   -0.102
327  go16 ~~ go21  4.849     3.931 -0.078  -0.078   -0.050   -0.050
330  go17 ~~ go18  2.623     2.127 -0.087  -0.087   -0.047   -0.047
331  go17 ~~ go19  5.968     4.838 -0.122  -0.122   -0.072   -0.072
333  go17 ~~ go21 14.082    11.417 -0.133  -0.133   -0.085   -0.085
334  go17 ~~ go22 16.086    13.042 -0.260  -0.260   -0.126   -0.126
335  go17 ~~ go23  2.624     2.127  0.132   0.132    0.050    0.050
336  go18 ~~ go19 67.086    54.389  0.524   0.524    0.307    0.307
337  go18 ~~ go20  4.391     3.560  0.119   0.119    0.064    0.064
339  go18 ~~ go22  8.182     6.633  0.240   0.240    0.115    0.115
340  go18 ~~ go23 11.225     9.101 -0.351  -0.351   -0.131   -0.131
343  go19 ~~ go22 38.132    30.915  0.481   0.481    0.254    0.254
345  go20 ~~ go21  3.718     3.014  0.071   0.071    0.045    0.045
```

*This call requests modification indices. Modification indices help the user diagnose why the model may not fit the data well. The modification index is actually a measure of how much the chi-squared value of the model (this was the very first fit index examined in this output) would change if the additional element suggested were added to the model. This does not mean a researcher should actually blindly add these elements, but instead,the indices should be used as indicators of model misfit. In general, larger modification indices (mi column) should be examined first.

The modification indices above are of two types: either (1) specifying an additional factor loading (‘=~’) or (2) correlating the errors between two items (‘~~’). For the additional factor loadings, the name of the factor is on the left, with the item on the right. In our model specification, this would mean that the item on the right should correlate on *both* the *Self* and *Other* factors, indicating that the other factor explains some of the variances in the response to that item. The suggested correlated errors signal that these items share common variance apart from the factor they are onor across the two factors. There are several possible reasons that items may be related to one another: order of presentation, similar wording, nearly identical content, or just plain measurement error. While all of these reasons are important to understand, what can be most helpful is examining *sets* of items that covary together. For example in our model go18 (working with people) and go19 (connecting with others) covary beyond what is expected by the model.On average they have a higher correlation than other items on the same subscale have to each other. The modification indicestell us that correlating the errors of these two items would reduce the chi-squared value by 67.086.This can indicate either the presence of a separate factor or a more nuanced, fine-grained distinction present in the overall factor.

Overall, we have a lot of large modification indices for this CFA, suggesting our model did not fit the data. This might indicate that two subscales are not sufficient or that our data structure does not match the assumptions of a CFA.

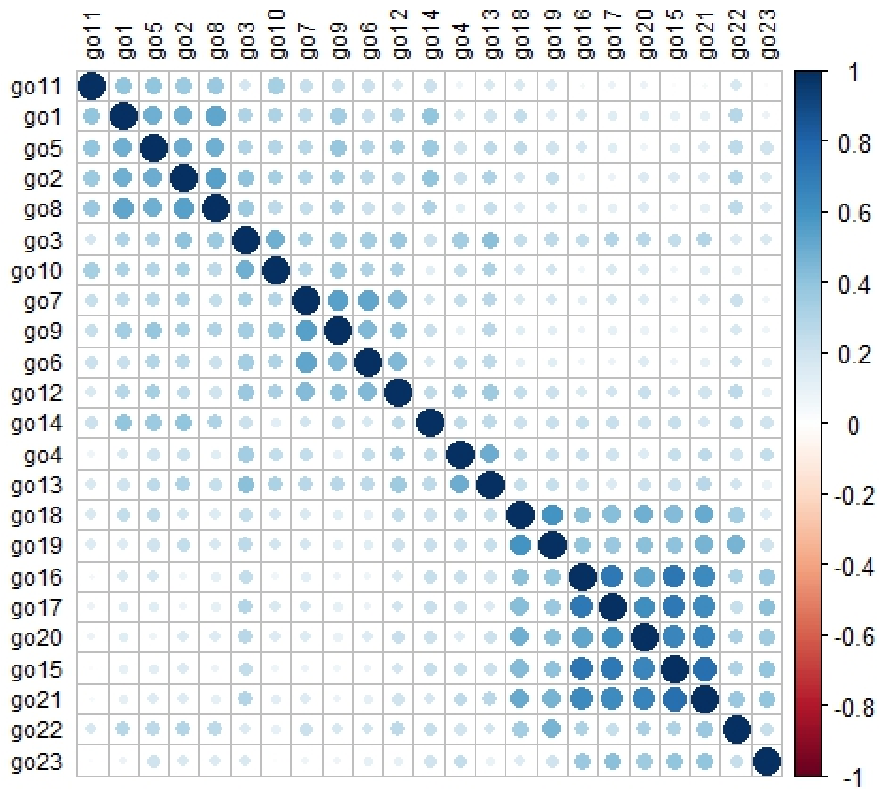
```r
residuals(C2f_fit, type = "cor")$cor
```

```
     go1    go2    go3    go4    go5    go6    go7    go8    go9    go10   go11   go12   go13   go14   go15   go16   go17   go18   go19   go20   go21   go22   go23
go1   0.000
go2   0.072  0.000
go3  -0.075 -0.003  0.000
go4  -0.084 -0.053  0.095  0.000
go5   0.084  0.067 -0.091 -0.061  0.000
go6  -0.103 -0.077  0.013  0.023 -0.053  0.000
go7  -0.080 -0.077 -0.012  0.012 -0.078  0.220  0.000
go8   0.150  0.147 -0.003 -0.107  0.099 -0.111 -0.099  0.000
go9  -0.026 -0.066 -0.022 -0.134  0.005  0.122  0.205 -0.052  0.000
go10 -0.021 -0.030  0.151  0.027 -0.060  0.014 -0.012 -0.065  0.035  0.000
go11  0.100  0.052 -0.099 -0.113  0.093 -0.040 -0.029  0.091 -0.043  0.082  0.000
go12 -0.066 -0.112  0.034  0.093 -0.030  0.148  0.114 -0.136  0.068  0.014 -0.101  0.000
go13 -0.108 -0.030  0.100  0.290 -0.070 -0.004 -0.016 -0.075 -0.025  0.038 -0.076  0.068  0.000
go14  0.099  0.080 -0.071  0.061  0.058 -0.081 -0.074  0.019 -0.057 -0.135 -0.009 -0.011  0.025  0.000
go15 -0.082 -0.049  0.058  0.110 -0.075 -0.074 -0.085 -0.099 -0.113 -0.091 -0.118  0.015  0.062  0.087  0.000
go16 -0.002 -0.105  0.071  0.111  0.001 -0.072 -0.036 -0.048 -0.082 -0.087 -0.077  0.008  0.063  0.099  0.031  0.000
go17 -0.034 -0.102  0.123  0.087 -0.044 -0.066  0.014 -0.057 -0.064  0.008 -0.050  0.034  0.017  0.095  0.015  0.081  0.000
go18  0.119  0.061  0.125  0.168  0.128  0.002  0.049  0.021  0.014  0.076  0.073  0.135  0.135  0.118 -0.065 -0.045 -0.037  0.000
go19  0.027  0.117  0.152  0.145  0.085  0.031  0.078  0.061  0.014  0.087  0.059  0.106  0.142  0.134 -0.061 -0.038 -0.058  0.283  0.000
go20 -0.024 -0.015  0.114  0.034 -0.091  0.005 -0.003 -0.014 -0.059  0.006 -0.036  0.077  0.080  0.094 -0.004 -0.071  0.011  0.054  0.007  0.000
go21 -0.070 -0.054  0.114  0.142 -0.042 -0.045 -0.019 -0.063 -0.093 -0.034 -0.090  0.037  0.122  0.043  0.023 -0.029 -0.048  0.024  0.009  0.029  0.000
go22  0.192  0.199  0.069  0.139  0.172  0.112  0.127  0.180  0.091  0.026  0.106  0.185  0.122  0.164 -0.059 -0.019 -0.104  0.108  0.240  0.007  0.029  0.000
go23 -0.015  0.056  0.067  0.181  0.104  0.016 -0.021  0.056 -0.025 -0.070 -0.055  0.011  0.012  0.125 -0.006  0.006  0.041 -0.122 -0.046  0.004  0.004  0.044  0.000
```

*This call requests the correlation residuals. These correlation residuals are presented in the form of a correlation matrix. They represent the difference between the matrix the model creates and the actual correlation matrix of the observed data. Large residual between two variables suggests that there is a relationship not being captured by the model. Specifically, we look for residuals greater than 0.1. These are plentiful in this matrix, implying that the model is not a good fit to the data. For example, go1 and go22 (first column of the matrix) have a correlation residual of 0.192.

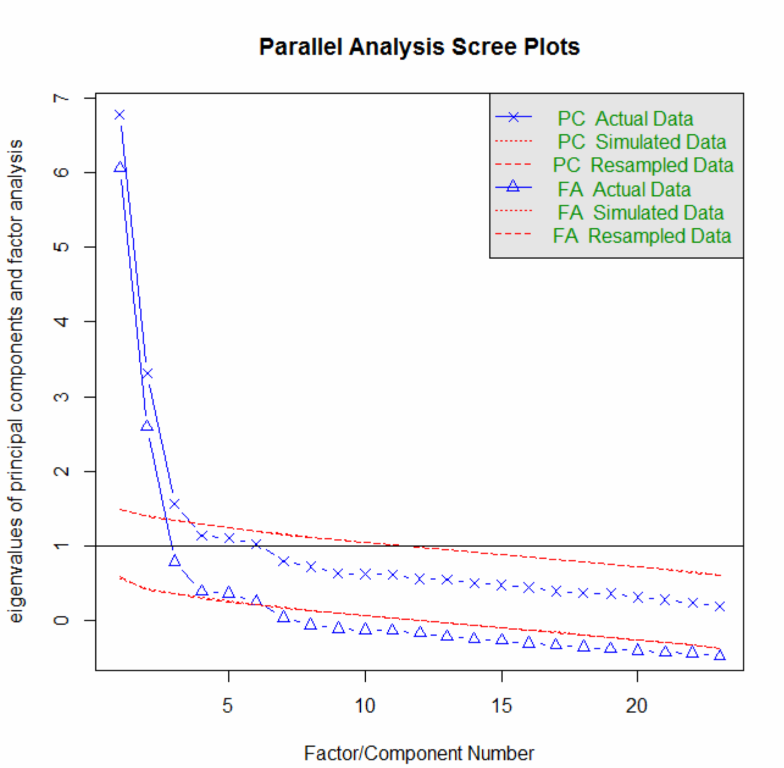
B. Exploratory Factor Analysis Output

```
corrplot((efa_cormat), order = "hclust", tl.col='black', tl.cex=.75)
```



* There are two methods for looking at the relationships between items in the data. The first is a traditional correlation matrix, which is not shown here for brevity. The second is a correlation plot shown above. This plot is easier to use than the correlation matrix because it clusters the items that show the strongest relationships to each other. This makes it easy to identify potential factors. The size and intensity of the color of the circle in each square indicated the strength of the relationship between two items. Looking at this plot there seem to be at least 3 sets of items that are highly correlated with each other. If a researcher has a small sample size this may be what you use to argue that a survey has or does not have the same factor structure with your population as it did in earlier published uses of the survey. With a larger sample size, researchers can continue on to the next steps of the EFA to more confidently identify the structure of the survey.

```
fa.parallel(data)
```




```
Parallel analysis suggests that the number of factors =  6  and the number of
components =  3
```

* This command produces a visual scree plot as well as two recommended number of factors to consider. The plot has a lot of information in it. On the x-axis is a range of possible factors to use in the model from 0 to over 20. On the y-axis are the eigenvalues for those factors. An eigenvalue is a measure of how much of the variance of the observed variables a factor explains. In the plot,there are two lines of connected symbols: triangles and Xs. Each triangle or X is a component/factor. The first represents the information gained (eigenvalue) by having the first factor. The second is the information gained by having a second factor and so on. The Xs are the principal components, and the triangles are the factors. Principal components analysis is usually an item reduction method so will usually yield a lower of components than the factor analysis. This gives the researcher a reasonable range of factors to test.

There are multiple strategies for identifying the number of potential factors from this plot. First, some researchers recommend only retaining factors with an eigenvalue greater than 1 (indicated by being above the horizontal black line at 1). Factors with an eigenvalue less than one account for a small proportion of the variability in the dataset and generally do not add as much information to the model. This is not a hard and fast rule, especially if there are theoretical reasons to consider additional factors. For instance, if a researcher designed a survey to have three factors and the third is below one but close to it, the researcher could argue to retain it.

A second way to evaluate the number of factors is the scree test. This involves looking at the shape of the line of triangles or Xs. There is generally a steep curve at the beginning and then a leveling out of the line. A flat line between two factors indicates there is no change information with the addition of the new factor, so there is no reason to add that factor.In our plots, there are two inflection points, one large one between 4 and 5 factors and then another smaller one between 7 and 8. So, adding a 8th factor does not add any additional information. This tells us to focus on EFA models with 7 or fewer factors. It is also evident from the shape of the plot that there is a lot of information added by the first three factors (notice how steep the slope is). This tells us we probably want to consider models with three or more factors. So, this suggests we test a series of EFA models with 3 to 7 factors.

A final approach is parallelanalysis. On the plot, you can see two dotted lines that each cross the actual data lines. These dotted lines are simulated data. They are based on the same sample size and number of variables as the actual data, but they are randomly generated. Eigenvalues are then repeatedly calculated for this random data set. These dotted lines represent the eigenvalues we would expect if there were no real relationships between the variables in the dataset. Thus, we only are interested in the actual factors with eigenvalues that exceed the eigenvalues you would see from data if relationships between variables are simply due to chance. Thus, looking at the plot, we should only keep the factors above the red line. In our case parallel analysis suggests three to six factors. A verbal summary of the parallel analysis is printed below the figure.

```
Efa5<-fa(r = efa_cormat,nfactors = 5,rotate = "oblimin",fm = "pa", max.iter = 500)
Efa5
Factor Analysis using method =  pa
Call: fa(r = efa_cormat, nfactors = 5, rotate = "oblimin", max.iter = 500,
    fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
       PA1   PA2   PA3   PA5   PA4   h2   u2 com
go1   0.01  0.70  0.04  0.03 -0.05 0.51 0.49 1.0
go2  -0.05  0.68  0.00  0.05  0.11 0.54 0.46 1.1
go30.14  0.18  0.27 -0.02  0.32 0.42 0.58 3.0
go4   0.03 -0.01 -0.02  0.00  0.74 0.55 0.45 1.0
go5   0.01  0.63  0.09  0.03  0.01 0.49 0.51 1.1
go6   0.00 -0.03  0.66 -0.03  0.08 0.46 0.54 1.0
go7   0.02 -0.03  0.74  0.03 -0.01 0.54 0.46 1.0
go8   0.03  0.76 -0.04 -0.03 -0.01 0.53 0.47 1.0
go9   0.01  0.15  0.67 -0.04 -0.09 0.52 0.48 1.1
go10-0.07  0.18  0.32  0.05  0.19 0.31 0.69 2.5
go11 -0.09  0.49  0.10  0.11 -0.08 0.29 0.71 1.3
go12  0.02 -0.01  0.50  0.09  0.20 0.42 0.58 1.4
go13 -0.02  0.06  0.11  0.06  0.59 0.47 0.53 1.1
go14  0.15  0.44 -0.05  0.00  0.13 0.28 0.72 1.5
go15  0.87 -0.02 -0.04  0.01  0.04 0.78 0.22 1.0
go16  0.80  0.04 -0.03  0.00  0.00 0.65 0.35 1.0
go17  0.87  0.00  0.08 -0.06 -0.04 0.70 0.30 1.0
go18  0.11  0.04  0.01  0.66  0.02 0.56 0.44 1.1
go19  0.00  0.02  0.00  0.78  0.02 0.63 0.37 1.0
go20  0.62 -0.02  0.08  0.20 -0.05 0.57 0.43 1.3
go21  0.68 -0.06  0.00  0.21  0.08 0.70 0.30 1.2
go22  0.09  0.21  0.03  0.41 -0.02 0.32 0.68 1.6
go23  0.52  0.15 -0.08 -0.14  0.09 0.27 0.73 1.5
```

*In this table the rows represent the individual items (go1 – go23) and the first five columns are the factors (PA1 – PA5). For each factor and item, a pattern coefficient is provided(similar to factor loadings in CFA). Ideally, an item has a high pattern coefficient (close to .8)for one factor and the pattern coefficients are close to zero for each of the other factors. In our case, we lookedfor items that had a pattern coefficient above 0.40[2] on one factor and not above 0.30 on any other factor. In our example, two items (go3 and go10) have low pattern coefficients across all the factors. These items are good candidates for removal from our final solution.

The next two columns are different ways of looking at how well each item is explained by the model. The first (h2) is a measure of communality. Communalityis the proportion of variance in the variable that is explained by all the factors in the model. The closer this value is to one, the more the variance in the item is explained by the model. The next column is the unique variance (u2) for the variable. This is the amount of variance not explained by the latent variables (1- h2). Ideally, items would have low uniqueness and high communality. In our example, go11, go14, go23 might be candidates for removal because they have a low communality.

The final column (com) is not a commonality measure;instead, it is Hoffman's index of complexity. The statistics describe the average number of factors necessary to explain the item. In an ideal case this number would be 1, meaning exactly one factor is necessary to explain the item. We can see go3, which we saw had similarloadings for several factors, has a high level of complexity indicating it needs multiple factors to explain it. This would make it a candidate for removal.

```
                       PA1  PA2  PA3  PA5  PA4
SS loadings           3.57 2.77 2.19 1.59 1.38
Proportion Var        0.16 0.12 0.10 0.07 0.06
Cumulative Var        0.16 0.28 0.37 0.44 0.50
Proportion Explained  0.31 0.24 0.19 0.14 0.12
Cumulative Proportion 0.31 0.55 0.74 0.88 1.00
```

*SS loadings are the sum of squared loadings (pattern coefficients for all items squared and summed for a factor). Generally, we consider factors worth saving if they have an SS loading greater than one. In our case, all our factors are greater than one.

Proportion Var is the proportion of variance in the data explained by a particular factor.The higher this number, the more of the variance in the data it explains. We can see, for example, that the first factor explains 16% of the variance in the data.The next row, Cumulative Var, is the amount of the variance explained by each factor summed. In the first column, it represents just the variance explained by PA1. In column two, it is the variance explained by PA1 + PA2 (.16 + .12) and in column three it is the variance explained by all three factors and so on. The closer the final column's value is toone, the better the fit of the model to the data.In total 50% of the variance in the data was explained by our model.

[2] This is a very generous guideline. We used it for the first deletion because we wanted to keep as many items as possible from the original scale. If an item continues to show pattern coefficients below 0.5 over repeated data collections, researchers should consider whether it should be kept in the scale or not.

The final two rows examine how the factors contribute to the amount of variance explained. From Proportion Explained we see the PA1 accounts for 31% of the explained variance, PA2 explains 24% and so on. Cumulative Proportion just sums those values. This will sum to 1 by the last column.

```
With factor correlations of
     PA1  PA2  PA3  PA5  PA4
PA1 1.00 0.18 0.16 0.58 0.32
PA2 0.18 1.00 0.52 0.29 0.33
PA3 0.16 0.52 1.00 0.25 0.39
PA5 0.58 0.29 0.25 1.00 0.34
PA4 0.32 0.33 0.39 0.34 1.00
```

*This table reveals the correlations between the factors.

```
Mean item complexity =  1.3
Test of the hypothesis that 5 factors are sufficient.

The degrees of freedom for the null model are  253  and the objective function was
10
The degrees of freedom for the model are 148  and the objective function was  0.86

The root mean square of the residuals (RMSR) is  0.03
The df corrected root mean square of the residuals is  0.04

Fit based upon off diagonal values = 0.99
Measures of factor score adequacy
                                                  PA1  PA2  PA3  PA5  PA4
Correlation of (regression) scores with factors   0.96 0.92 0.90 0.89 0.85
Multiple R square of scores with factors          0.92 0.84 0.81 0.80 0.73
Minimum correlation of possible factor scores     0.84 0.68 0.62 0.59 0.45
```