Statistical analyses and computer programming in personality

A chapter for the
Cambridge University Press Handbook of Personality Psychology

William Revelle, Lorien Elleman and Andrew Hall
Department of Psychology, Northwestern University

Abstract
The use of open source software for statistical analysis provides personality researchers new tools for exploring large scale data sets and allows developing and testing new psychological theories. We review the development of these techniques and consider some of the major data analytic strategies. We provide example code for doing these analyses in R.
Prologue: A brief history of Open Source Statistical Software

It is hard to imagine early in the 21st century that many of the statistical techniques we think of as modern statistics were developed in the late 19th and early 20th centuries. What we now call regression was the degree of “reversion to mediocrity” as introduced by Francis Galton (1886). A refinement of regression was the measure of “co-relations” (Galton, 1888) which was specified as the relationship of deviations as expressed in units of the probable error. Galton’s insight of standardization of units became known as ‘Galton’s coefficient’ which, when elaborated by Pearson (1895,8,9) became what we now know as the Pearson Product Moment Coefficient. Because the Pearson derivations were seen as too complicated for psychologists, Charles Spearman (1904b) explained the developments of Pearson and his associates to psychologists where “the meaning and working of the various formulae have been explained sufficiently, it is hoped, to render them readily usable even by those whose knowledge of mathematics is elementary” (p 73 Spearman, 1904b). Later in that same paper, he then developed reliability theory and the correction for attenuation.

In his second amazing publication that year he developed the basic principles of factor analysis and laid out his theory of general intelligence (Spearman, 1904a).

Fundamental theorems of factor analysis (Thurstone, 1933; Eckart and Young, 1936) and estimates of reliability (Brown, 1910; Spearman, 1910; Kuder and Richardson, 1937; Guttman, 1945) came soon after and were all developed when computation was done by “computers” who were humans operating desk calculators. If not difficult, computation was tedious in that it required repeatedly finding sums of squares and sums of cross products. When examining the correlation structure of multiple items, the number of correlations went up by the square of the number of items, and thus so did the computational load.

Test theory as developed in the 1930s and 1940s led to a number of convenient shortcuts by which the sums of test and item variances could be used to estimate what the correlations of composites of items would be if certain assumptions were met. Thus, the coefficients of Cronbach (1951); Guttman (1945); Kuder and Richardson (1937) (α, λ3 and KR20) were meant to be estimates based upon the structure of covariances without actually finding the covariances.

Another shortcut was to dichotomize continuous data in order to find correlations. For example, in an ambitious factor analysis of individual differences in behavior among psychiatric patients (Eysenck, 1944) made use of this shortcut by dichotomizing continuous variables and then finding the Yule (1912) coefficient. Unfortunately, such a procedure produced a non-positive definite matrix which makes reanalysis somewhat complicated.

With the need to calculate flight paths for large artillery shells and rockets, the ideas developed by Babbage for his “Analytical Engine” in 1838 (Bromley, 1982) and algorithms for programming (Lovelace, 1842) were converted from punched cards and automatic looms into the electrical tubes and circuit boards of von Neuman (Isaacson, 2014). The age of computation had arrived. It was now possible to properly analyze covariance structures.

For personality psychologists, these were exciting times, for it allowed for the cal-
calculation of larger correlation matrices and the use of factor analysis with more elegant algorithms than the centroid approach. Indeed, Raymond Cattell moved from Harvard to the University of Illinois because he was promised access to the new ‘ILLIAC’ computer (and would not have to teach undergraduates).

Main frame computers and proprietary software

At first, software for these new devices was tailor made for the particular machine, and there was a plethora of programming languages and operating systems. In the late 1950s the programming language FORTRAN (later renamed to be Fortran when computers learned about lower case) was developed at IBM for the numeric computation necessary for scientific computing and then translated for other operating systems. While some programs would run on IBM 709 and 7090s, others would only work on the super computers of the time, the Control Data Corporation’s 1604 and 6400. An early package of statistical programs, developed for the IBM 7090 in 1961 at UCLA was the BioMedical Package (BMDP). At first BMDP was distributed for free to other universities but BMDP (Dixon and Brown, 1979) eventually became a commercial package which has since disappeared. Two other statistical systems (SAS and SPSS), originally developed at other universities (North Carolina State and Stanford) and shared with others, were also developed.

These three major software systems for doing statistics came from three somewhat different traditions. BMDP was developed for biomedical research, SAS for agriculture research, and SPSS for statistics in the social sciences (SPSS, 2008). Although all three systems were originally developed at universities and were freely distributed to colleagues, all three soon became incorporated as for profit corporations. All were developed for main frame computers where instructions were originally given in stacks of Hollerith Cards (known to many as IBM cards), containing 80 characters per card. All of the programs made use of the FORTRAN programming language and still, many years later, have some of their main frame genealogy embedded in their systems. Older researchers still shudder at the memories of needing to wait for 24 hours after turning in a box of cards only to discover one typo negated the entire exercise.

S and R: interactive statistics

In contrast to the statistical analyses done on mainframes, S and subsequently R were developed for interactive statistics. The S computing ‘environment’ was developed for Unix in the 1970’s at Bell labs by John Chambers and his associates (Becker, Chambers, and Wilks, 1988). It was meant to take advantage of interactive computing where the user could work with his/her data to better display it and understand it. After several iterations it became the defacto statistical package for those using the Unix operating system. In 1992, two statisticians, Ross Ihaka and Robert Gentleman, at the University of Otago, in New Zealand started adapting S to run on their Mac computers. It incorporated the list oriented language Scheme and emphasized object-oriented programming. Most importantly, they
shared the design specifications with other interested developers around the world and intentionally did not copyright the code. Rather, they, with the help of John Chambers and the rest of the R Development Core Team, deliberately licensed R under the GNU General Public License of the Free Software Foundation which allows users to copy, use, and modify the code as long as the product remains under the GPL (R Core Team, 2018).

Perhaps the real power of R is that because it is open source, it is extensible. That means that anyone can contribute packages to the overall system. That, and the power of the GPL and open source software movement has led to an amazing effect. From the original functions in R and the ones written by the R Core Team, more than 12,600 packages have been contributed to the CRAN, the Comprehensive R Archive Network, and at least (as of this writing) more than 34,000 packages are available on GitHub. R has become the lingua franca of statistics and many new developments in psychological statistics are released as R functions or packages. When writing methodology chapters or articles, the associated R code to do the operations may also be given (as we do in this chapter).

With the growing recognition of the importance of replicable research, the publication of the R scripts and functions to do the analysis, as well as the release of R readable data sets is an essential step in allowing others to understand and repeat what we have done. Because the source code of all of the R packages is available, users can check the accuracy of the code and report bugs to the developers of the particular package. This is the ultimate peer review system, in that users use, review, and add to the entire code.

Given its open source nature and growing importance in scientific computing, much of the rest of this chapter will be devoted to discussing how particular analyses can be done in R. This is not to deny that commercial packages exist, but to encourage the readers of this handbook to adopt modern statistics. The actual code used for the tables and figures is included in the Appendix.

Finally, little appreciated by many users of R is that it is not just a statistical environment, it is also a very high level programing language. Although some of the packages in R are written in Fortran or C++, many packages are written in R itself. R allows operations at the higher matrix level and allows for object-oriented programming. Each function operates on ‘objects’ and returns another ‘object’. That is, functions can be chained together to add value to previous operations. This allows users to include standard functions in their own functions with the output available for even more functions. Actual programming in R is beyond the scope of this chapter, but is worth learning for the serious quantitative researcher. Without developing packages, a willingness to write more and more complicated scripts is a positive benefit.

Getting and using R

R may be found at https://www.r-project.org and the current release is distributed through the Comprehensive R Archive Network https://cran.r-project.org. Popular interfaces to R include Rstudio (https://www.rstudio.com) which is particularly useful for PCs (the Mac version comes with its own quite adequate interface). Once R is
downloaded and installed, it is useful to install some of the powerful packages that have been added to it. We will make use of some of these packages, particularly the \texttt{psych} package which has been specifically developed for personality and psychological research (Revelle, 2018). See the appendix for detailed instructions.

Data, Models, and Residuals

The basic equation in statistics is that:

\[ \text{Data} = \text{Model} + \text{Residual} \iff \text{Residual} = \text{Data} - \text{Model}. \quad (1) \]

That is, the study of statistics is the process of modeling our data. Our models are approximations and simplifications of the data (Rodgers, 2010). Our challenge as researchers is to find models that are good approximations of the data but that are not overly complicated. There is a tradeoff between the two goals of providing simple descriptions of our data and providing accurate descriptions of the data. Consider the model that the sun rises in the East. This is a good model on average and as a first approximation, but is actually correct only twice a year (the equinoxes). A more precise model will consider seasonal variation and recognize that in the northern hemisphere, the sun rises progressively further north of east from the spring equinox until the summer solstice and then becomes more easterly until the fall equinox. An even more precise model will consider the latitude of the observer.

If we think of degrees of freedom in models as money, we want to be frugal but not stingy. Typically we evaluate the quality of our models in terms of some measure of goodness of fit. Conceptually, fit is some function of the size of the residuals as contrasted to the data. Because almost all models will produce mean residuals of zero, we typically invoke a cost function such as ordinary least squares to try to find a model that minimizes our squared residual. As an example, the algebraic mean is that model of the data that minimizes the squared deviations around it (the variance).

The following pages will consider a number of statistical models, how to estimate them, and how to evaluate their fit. All of what follows can be derived from a serious consideration of Equation 1.

In what follows we discuss two types of variables and three kinds of relationships. In a distinction reminiscent of the prisoners in the cave discussed by Plato in the \textit{The Republic} (Plato, 1892), we consider two classes of variables: those which we can observe, and those which we can not observe but are the latent causes of the observed variables. Our observations are of the movement of the shadows on the cave’s wall; we need to infer the latent causes of these shadows. Many of our tools of data analysis (e.g., factor analysis, reliability theory, structural equation modeling, and item response theory) are just methods for estimating latent variables and their inter-relationships from the pattern of relationships between observed variables. Traditionally we make this distinction by using Greek letters for unobserved (latent) population values and Roman letters for observed
values. When we portray patterns of relationships graphically (e.g., Figure 1) we show observed variables as boxes and latent variables as circles. As may be seen in Figure 1, there are three kinds of relationships between our variables: relations between observed variables, relations between latent and observed variables, and relations between latent variables. Theories are organizations of latent variables as they represent the relationships between observed variables.

Basic Descriptive Statistics

Before any data analysis may be done, the data must be collected. This is more complicated than it seems, for it involves consideration of the latent variables of interest; presumed observed markers of these latent variables; the choice of subjects (are they selected randomly, systematically, are they volunteers, are they WEIRD (Henrich, Heine, and Norenzayan, 2010)), the means of data collection (self report, observer ratings, life narratives, computerized measurement, web based measures, etc.); the number of times measures are taken (e.g., once, twice for test-retest measures or measures of change, multiple times in studies of growth changes or of emotions over time); the lags between repeated measures (minutes, hours, days, or years) and whether there are experimental manipulations to distinguish particular conditions (Revelle, 2007).

Once the data are collected, it is of course necessary to prepare them for data analysis. That is to say, to transfer the original data into a form suitable for computer analysis. If hand coding must be done (e.g., scoring life narratives, Guo, Klevan, and McAdams, 2016) the separate ratings must be entered in a way that allows for computer based analysis (e.g., a reliability calculation) to be made.

Typically the data are organized as a two dimensional table (e.g., a spreadsheet in EXCEL or OpenOffice) with subjects as rows and variables as columns. If there are repeated measures per subject, the data might have a separate row for each occasion, but with one column identifying the subject and another the occasion. Consider the data in Table 1 which are taken from an example data set msqR in the psych package (Revelle, 2018) from R\(^1\). The msqR data set was collected over about 10 years as part of a long term series of studies of the interactive effect of personality and situational stressors on cognitive performance.

An under-appreciated part of data analysis is the basic data cleaning necessary to work with real data. Mistakes are made at data entry, participants fall asleep, other participants drop out, some do not answer every question, some participants are intentionally deceptive in their responses. It is important before doing any analysis to find basic descriptive statistics to look for impossible responses, to examine the distribution of responses, and to attempt to detect outliers. However, as discussed by Wilcox (2001), merely examining the shape of the distribution is not enough to detect outliers and it is useful to apply robust estimators of central tendency and relationships. The WRS2 package (Mair, Schoenbrodt,
Figure 1. The basic set of statistical relationships may be seen as relations among and between observed variables (boxes) and latent variables (circles).
Table 1: A representative sample of eight subjects with time 1 and time 2 observations on 10 emotion terms. The data are from the msqR data set (N=3,032) which has repeated measures for 2,086 participants. The full data set is used for many of following examples. It is included in the psych package. The data are shown in ‘long’ format with repeated measures ‘stacked’ on top of each other to represent multiple time points. ‘Wide’ format would represent the different time points as separate columns for each subject.

<table>
<thead>
<tr>
<th>Line #</th>
<th>id</th>
<th>time</th>
<th>anxis</th>
<th>at-es</th>
<th>calm</th>
<th>cnfdn</th>
<th>cntst</th>
<th>jttry</th>
<th>nervs</th>
<th>rlxed</th>
<th>tense</th>
<th>upset</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>69</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>70</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>71</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>72</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>5</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>74</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>75</td>
<td>7</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>76</td>
<td>8</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

and Wilcoxon, 2017) implements many of the robust statistics discussed by Wilcoxon and his colleagues (Wilcoxon and Keselman, 2003; Wilcoxon, 2005). For example, the algebraic mean is just the sum of the observations divided by the number of observations. The trimmed mean is the same after a percentage (e.g., 10%) are removed from the top and bottom of the distribution. The trimmed mean is more robust to outliers than is the algebraic mean. The median is the middle observation (the 50th percentile) and is an extreme example of a trimmed mean (with trim=.5). The minimum and maximum observations, and the resulting range are most useful for detecting improper observations. Skew and Kurtosis are functions of the third and fourth power of the deviations from the mean (Mardia, 1970). The describe function will also report various percentiles of the distribution including the Inter Quartile Range (25th to 75th percentiles) (Table 2).

Tests of statistical significance: Normal theory and the bootstrap

Those brought up in the Fisherian tradition of Null Hypothesis Significance Testing (NHST) traditionally compare fit statistics to their expected value given normal theory. Fits are converted into standardized scores (z scores) and then probabilities are found from the normal distribution. This works well with large samples where errors are in fact random. For smaller samples, the variation of estimates of mean differences compared to the sample based standard error are larger than expected given the normal. This problem led to the
Table 2: Descriptive statistics for the data in Table 1. An important step before doing any more advanced analysis is to search for outliers by examining the data for impossible values and comparing the values of the algebraic versus trimmed mean versus the median.

<table>
<thead>
<tr>
<th>Variable</th>
<th>vars</th>
<th>n</th>
<th>mean</th>
<th>sd</th>
<th>med</th>
<th>trmmed</th>
<th>mad</th>
<th>min</th>
<th>max</th>
<th>range</th>
<th>skew</th>
<th>kurtosis</th>
<th>se</th>
<th>IQR</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>1</td>
<td>16</td>
<td>4.50</td>
<td>2.37</td>
<td>4.5</td>
<td>4.50</td>
<td>2.97</td>
<td>1</td>
<td>8</td>
<td>7</td>
<td>0.00</td>
<td>-1.45</td>
<td>0.59</td>
<td>3.50</td>
</tr>
<tr>
<td>time</td>
<td>2</td>
<td>16</td>
<td>1.50</td>
<td>0.52</td>
<td>1.5</td>
<td>1.50</td>
<td>0.74</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0.00</td>
<td>-2.12</td>
<td>0.13</td>
<td>1.00</td>
</tr>
<tr>
<td>anxious</td>
<td>3</td>
<td>16</td>
<td>0.62</td>
<td>0.62</td>
<td>1.0</td>
<td>0.57</td>
<td>0.74</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0.35</td>
<td>-0.96</td>
<td>0.15</td>
<td>1.00</td>
</tr>
<tr>
<td>at.ease</td>
<td>4</td>
<td>16</td>
<td>1.94</td>
<td>0.57</td>
<td>2.0</td>
<td>1.93</td>
<td>0.00</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2.02</td>
<td>-0.19</td>
<td>0.14</td>
<td>0.25</td>
</tr>
<tr>
<td>calm</td>
<td>5</td>
<td>16</td>
<td>1.98</td>
<td>0.81</td>
<td>2.0</td>
<td>1.93</td>
<td>0.00</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>1.51</td>
<td>-0.20</td>
<td>0.20</td>
<td>0.25</td>
</tr>
<tr>
<td>confident</td>
<td>6</td>
<td>16</td>
<td>1.75</td>
<td>0.68</td>
<td>2.0</td>
<td>1.71</td>
<td>0.74</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>0.29</td>
<td>-1.04</td>
<td>0.17</td>
<td>1.00</td>
</tr>
<tr>
<td>content</td>
<td>7</td>
<td>16</td>
<td>2.06</td>
<td>0.68</td>
<td>2.0</td>
<td>2.07</td>
<td>0.00</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>0.06</td>
<td>-0.98</td>
<td>0.17</td>
<td>0.25</td>
</tr>
<tr>
<td>jittery</td>
<td>8</td>
<td>16</td>
<td>0.44</td>
<td>0.51</td>
<td>0.0</td>
<td>0.43</td>
<td>0.00</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.23</td>
<td>-2.07</td>
<td>0.13</td>
<td>1.00</td>
</tr>
<tr>
<td>nervous</td>
<td>9</td>
<td>16</td>
<td>0.31</td>
<td>0.48</td>
<td>0.0</td>
<td>0.29</td>
<td>0.00</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.73</td>
<td>-1.55</td>
<td>0.12</td>
<td>1.00</td>
</tr>
<tr>
<td>relaxed</td>
<td>10</td>
<td>16</td>
<td>1.94</td>
<td>0.57</td>
<td>2.0</td>
<td>1.93</td>
<td>0.00</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>-0.02</td>
<td>-0.19</td>
<td>0.14</td>
<td>0.00</td>
</tr>
<tr>
<td>tense</td>
<td>11</td>
<td>16</td>
<td>0.31</td>
<td>0.48</td>
<td>0.0</td>
<td>0.29</td>
<td>0.00</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.73</td>
<td>-1.55</td>
<td>0.12</td>
<td>1.00</td>
</tr>
<tr>
<td>upset</td>
<td>12</td>
<td>16</td>
<td>0.12</td>
<td>0.34</td>
<td>0.0</td>
<td>0.07</td>
<td>0.00</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2.06</td>
<td>2.40</td>
<td>0.09</td>
<td>0.00</td>
</tr>
</tbody>
</table>

introduction of the $t$ statistic for comparing the means of smaller groups (Student, 1908) (t.test) and the $r$ to $z$ transformation (r2z) for tests of correlations (Fisher, 1921). (Use cor.test for one correlation, corr.test for many correlations). Most functions in R will return both the statistic and the probability of that statistic. Many will also return a confidence interval for the statistic. But the probabilities (and therefore the confidence intervals) are a function of the effect size, the sample size, and the distribution of the parameter being estimated. Unfortunately, not all tests can be assumed to be normally distributed and it is unclear how to find the distribution of arbitrary parameters of a distribution (e.g., the median). Extending ideas such as the ‘Jackknife’ proposed by Tukey (1958), Efron (1979) proposed the ‘bootstrap’ (having considered such names as the ‘shotgun’), a powerful use of random sampling (Efron and Gong, 1983).

The basic concept of the bootstrap is to treat the observed sample as the entire population, and then to sample repeatedly from this ‘population’ with replacement; find the desired estimate (e.g., the mean, the median, the regression weight) and then do this again, and again, and many times again. Each sample, although the same size as the original sample, will contain (on average) 63.2% of the subjects in the original sample, with 36.8% being repeated at least once\footnote{This perhaps unintuitive amount is $1 - \frac{1}{e}$ and is the limit of the probability of an item not being repeated as the number of cases increases ($p = 1 - (1 - \frac{1}{n})^n$).} The resulting distribution of the estimated value can be used to find confidence intervals without any appeal to normal theory.

For those who use NHST, it is important to understand the probability that a real effect is detected, the power, is not the same as the probability of rejecting the ‘nil’ hypothesis that an effect is 0 (Cohen, 1988,9,9; Streiner, 2003). A number of R packages (e.g.
One of the great advances of modern statistics is the use of the bootstrap and other randomization tests. In the space of seconds, 1,000 to 100,000 bootstrap resamples can be calculated for almost any statistic. We will use this procedure when we find the confidence intervals for correlation coefficients (Table 5) and in particular for the effect of mediation in a regression model.

Correlation and Regression

Originally proposed by Galton (1888) and refined by Pearson (1895) and Spearman (1904b), the linear regression coefficient and its standardized version, the correlation coefficient are the fundamental statistics of research. In terms of deviation scores \((x = X - \bar{X} \text{ and } y = Y - \bar{Y})\)

\[
r_{xy} = \frac{\Sigma xy}{\sqrt{\Sigma x^2 \Sigma y^2}}
\]  

Depending upon the characteristics of the data, the correlation as defined in Equation 2 has many different names. If both X and Y are continuous, then the resulting correlation is known as the Pearson Product Moment Correlation Coefficient (or just Pearson r). If converted to ranks, as Spearman’s \(\rho\), and if both X and Y are dichotomous as the \(\phi\) coefficient (Table 3). Three of the correlations shown are estimates of what the latent continuous correlation would be if two continuous latent variables (\(\chi\) and \(\psi\)) were artificially dichotomized (the tetrachoric), or split into multiple levels (the polychoric) correlation.

Because the first four of these correlations are just different forms of the Pearson r (albeit it in different forms), the same estimation function can be used. In core R this is just the \texttt{cor} function or to find covariances the \texttt{cov} function. The last three require specialized functions written for polytomous (or dichotomous) data (i.e., the \texttt{psych} package functions \texttt{polyserial}, \texttt{tetrachoric} and \texttt{polychoric}. All of these functions are combined in \texttt{mixed.cor}.

The tetrachoric correlations of ability data (answers are right or wrong) and the polychoric correlations of self report temperament scales (typically on a 1-4, 1-5, or 1-6 scale), being the modeled correlation of continuous latent scores, will be larger in absolute value than the Pearson correlations of the same data. In addition, these estimates of the latent correlations are not affected by differences in distributions the way the Pearson r on the observed variables is. An example of the difference between a Tetracoric and a Pearson \(\phi\) is seen in Table 7 where \(\phi = .32\) but the inferred relationship between two continuous variables was .54.

The ubiquitous correlation coefficient

The correlation is also a convenient measure of the size of an effect (Ozer, 2007). It has long been been known that the difference in means compared to the within group
Table 3: A number of correlations are Pearson $r$ in different forms, or with particular assumptions. The first four use $r = \frac{\sum x_i y_i}{\sqrt{\sum x_i^2 \sum y_i^2}}$. The last three are based upon assumptions of normality of a latent $X$ and $Y$, with an artificial dichotomization or categorization into discrete (but ordered) groups.

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>symbol</th>
<th>X</th>
<th>Y</th>
<th>Assumptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pearson</td>
<td>$r$</td>
<td>continuous</td>
<td>continuous</td>
<td></td>
</tr>
<tr>
<td>Spearman</td>
<td>$\rho$</td>
<td>ranks</td>
<td>ranks</td>
<td></td>
</tr>
<tr>
<td>Point bi-serial</td>
<td>$r_{pb}$</td>
<td>dichotomous</td>
<td>continuous</td>
<td></td>
</tr>
<tr>
<td>Phi</td>
<td>$\phi$</td>
<td>dichotomous</td>
<td>dichotomous</td>
<td></td>
</tr>
<tr>
<td>Bi-serial</td>
<td>$r_{bis}$</td>
<td>dichotomous</td>
<td>continuous</td>
<td>normality</td>
</tr>
<tr>
<td>Tetrachoric</td>
<td>$r_{tet}$</td>
<td>dichotomous</td>
<td>dichotomous</td>
<td>bivariate normality</td>
</tr>
<tr>
<td>Polychoric</td>
<td>$r_{pc}$</td>
<td>categorical</td>
<td>categorical</td>
<td>bivariate normality</td>
</tr>
</tbody>
</table>

standard deviation: the $d$ statistic of Cohen (1962) is a better way to compare the difference between two groups than Student’s $t$ statistic. For it is the size of the difference that is important, not the significance. An undue reliance of “statistical significance” has ignored the basic observation that the test of significance $= \text{size of effect} \times \text{size of study}$ (Rosenthal, 1994) and that the resulting $p$ value is a non-linear function of the size of the effect. To remedy this problem, Cohen (1962) developed the $d$ statistic for the comparison of two groups (use $\text{cohen.d}$). Generalizations for multiple groups or continuous variables allow the translation of many alternative indices of effect size into units of the correlation coefficient (see Table 4). Robust alternatives to $d$ (found by $\text{d.robust}$) express differences in terms of trimmed means and Winsorized variances (Algina, Keselman, and Penfield, 2005; Erceg-Hurn and Mirosevich, 2008). Basic principles in reporting effect sizes are available in a recent tutorial (Pek and Flora, 2018).

**Multiple Regression and the general linear model**

Just as the $t$-test and the $F$-test may be translated into correlations units, so they can be thought of in terms of the general linear model (Judd and McClelland, 1989):

$$\hat{Y} = \mu + \beta X + \epsilon.$$  (3)

$X$ can be an experimental design matrix with one or more independent grouping variables but it can also include a set of person variables. In the case of just one dichotomous grouping variable, then Equation 3 is just the regression of the two levels of $X$ with the dependent variable and is similar to the comparison of the means of Student’s $t$ (Student, 1908). $t$ is typically expressed as difference of means compared to the standard error of that difference but is better expressed as an effect size multiplied by one half the square root of the degrees of freedom (df) or the ratio of the correlation times the square root of
Table 4: Alternative Estimates of effect size. Using the correlation as a scale free estimate of effect size allows for combining experimental and correlational data in a metric that is directly interpretable as the effect of a standardized unit change in x leads to r change in standardized y.

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Estimate</th>
<th>r equivalent as a function of r</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pearson correlation</td>
<td>$r_{xy} = \frac{C_{xy}}{\sigma_x \sigma_y}$</td>
<td>$r_{xy}$</td>
</tr>
<tr>
<td>Regression</td>
<td>$b_{y,x} = \frac{C_{xy}}{\sigma_x^2}$</td>
<td>$r = b_{y,x} \frac{\sigma_y}{\sigma_x}$</td>
</tr>
<tr>
<td>Cohen’s d</td>
<td>$d = \frac{X_1 - X_2}{\sigma_x}$</td>
<td>$r = \frac{d}{\sqrt{d^2 + 4}}$</td>
</tr>
<tr>
<td>Hedge’s g</td>
<td>$g = \frac{X_1 - X_2}{s_y}$</td>
<td>$r = \frac{g}{\sqrt{g^2 + 4(df/N)}}$</td>
</tr>
<tr>
<td>t - test</td>
<td>$t = \frac{d \sqrt{df}}{2}$</td>
<td>$r = \sqrt{t^2/(t^2 + df)}$</td>
</tr>
<tr>
<td>F-test</td>
<td>$F = \frac{d^2 df}{4}$</td>
<td>$r = \sqrt{F/(F + df)}$</td>
</tr>
<tr>
<td>Chi Square</td>
<td>$\chi^2 = \frac{df}{n}$</td>
<td>$r = \sqrt{\chi^2 / n}$</td>
</tr>
<tr>
<td>Odds ratio</td>
<td>$d = \frac{\ln(OR)}{1.81}$</td>
<td>$r = \sqrt{\frac{\ln(OR)}{1.81}}$</td>
</tr>
<tr>
<td>$r_{equivalent}$</td>
<td>$r$ with probability p</td>
<td>$r = r_{equivalent}$</td>
</tr>
</tbody>
</table>

the degrees of freedom to the coefficient of alienation ($\sqrt{1 - r^2}$, Brogden, 1946).

$$t = \frac{X_1 - X_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} = \frac{d \sqrt{df}}{2} = \frac{r}{\sqrt{1 - r^2}} \sqrt{df}$$  \(4\)

where the degrees of freedom are $n_1 + n_2 - 2$ (Rosnow and Rosenthal, 2003). The slope of the regression is the effect size, dividing this by the coefficient of alienation and multiplying by the square root of df converts the regression to a t. It is found in R with the t.test function.

If X has two categorical grouping variables (e.g., $x_1$ and $x_2$), then we have

$$\hat{y} = \mu + \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_1 x_2 + \epsilon$$  \(5\)

which for categorical values of X is just the traditional analysis of variance of two main effects and an interaction (Fisher, 1925). This may be found using the aov function which acts on categorical variables and returns the traditional ANOVA output. With unbalanced repeated measures designs, the lme function included in the lme4 package (Bates, Maechler, Bolker, and Walker, 2015) allows a specification of random and fixed effects.

The advantage of the general linear model for psychologists interested in individual differences is that continuous person variables can be included in the same model as experimental variables. This is a great improvement from prior approaches which would artificially dichotomize the person variable into high and low groups in order to use an
ANOVA approach. By retaining the continuous nature of the predictor, we improve the power over the ANOVA test.

As an example of using the general linear model, we use a data set from Tal-Or, Cohen, Tsfati, and Gunther (2010) that is discussed by Hayes (2013). Tal-Or et al. (2010) measured the effect of an experimental manipulation of salience of a news article (cond) on presumed media influence (PMI), perceived importance of the issue (import), and reported willingness to change one’s behavior (reaction)\(^3\). The observed correlations are found by using the `lowerCor` function and are given in Table 5.

Table 5: Correlations of the conditions with Perceived Media Influence, Importance of the message, and Reaction to the message (Tal-Or et al., 2010). As is traditional in NHST, correlations that are larger than would be expected by chance are marked with ‘magic astericks’. Confidence intervals for these correlations are shown given normal theory (upper and lower normal) as well as estimated by 1,000 bootstrap resamplings of the data (lower and upper empirical).

<table>
<thead>
<tr>
<th>Variable</th>
<th>cond</th>
<th>pmi</th>
<th>imprt</th>
<th>rectn</th>
</tr>
</thead>
<tbody>
<tr>
<td>cond</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pmi</td>
<td>0.18*</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>import</td>
<td>0.18*</td>
<td>0.28**</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>reaction</td>
<td>0.16</td>
<td>0.45***</td>
<td>0.46***</td>
<td>1.00</td>
</tr>
</tbody>
</table>

*Note:* ***\(p < .001\); **\(p < .01\); *\(p < .05\).

Empirical and normal theory based confidence intervals from `cor.ci`.

<table>
<thead>
<tr>
<th>Variable</th>
<th>lwr.m</th>
<th>lwr.n</th>
<th>estmt</th>
<th>uppr.n</th>
<th>uppr.m</th>
</tr>
</thead>
<tbody>
<tr>
<td>cond-pmi</td>
<td>0.01</td>
<td>0.01</td>
<td>0.18</td>
<td>0.36</td>
<td>0.36</td>
</tr>
<tr>
<td>cond-import</td>
<td>0.02</td>
<td>0.00</td>
<td>0.18</td>
<td>0.35</td>
<td>0.35</td>
</tr>
<tr>
<td>cond-rectn</td>
<td>-0.01</td>
<td>-0.01</td>
<td>0.16</td>
<td>0.33</td>
<td>0.33</td>
</tr>
<tr>
<td>pmi-import</td>
<td>0.09</td>
<td>0.09</td>
<td>0.28</td>
<td>0.45</td>
<td>0.45</td>
</tr>
<tr>
<td>pmi-rectn</td>
<td>0.31</td>
<td>0.31</td>
<td>0.45</td>
<td>0.57</td>
<td>0.56</td>
</tr>
<tr>
<td>imprt-rectn</td>
<td>0.32</td>
<td>0.32</td>
<td>0.46</td>
<td>0.60</td>
<td>0.61</td>
</tr>
</tbody>
</table>

There are multiple ways to analyze these data. We could naively do three t-tests of the experimental manipulation, find all of the intercorrelations, or do a regression predicting reaction from the condition, perception of media influence (PMI) and perceived importance of the message (Importance). All of these alternatives are shown in the appendix. The `setCor` and `mediate` functions will also draw the regressions as path diagrams (Figure 3).

\(^3\)With the kind permission of Nurit Tal-Or, Jonathan Cohen, Yariv Tsfati, and Albert C. Gunther, these data were added to the `psych` package as the Tal-Or data set.
Mediation and Moderation

In the Tal-Or et al. (2010) data set, the experimental manipulation affected the dependent variable of interest (reaction) but also two other variables (perception of media influence and perceived importance of the message). There is a direct effect of condition on reaction, as well as indirect effects through PMI and Importance. Conventional regression shows the direct effect of reaction controlling for the indirect effects that go through PMI and import. The total effect of condition on reaction is their covariance divided by the condition variance and is known as the $c$ effect ($c = \frac{\sigma_{xy}}{\sigma_{x}^{2}} = \frac{125}{25} = .5$). If we label the paths from cond to PMI ($a_1 = .48$) and from condition to import ($a_2 = .63$), then the indirect effects are the sum of the products through the two mediators ($b_1 = .40, b_2 = .32 \rightarrow a_1b_1 + a_2b_2 = .48 \times .40 + .63 \times .32 = .4$) then the direct effect is the total less the indirect effect ($c’ = c - ab = .1$). We say that the effect of the experimental manipulation is mediated through its effect on perceived importance and perceived media influence. The error associated with the mediating term (ab) or the sum of product terms ($a_1b_1 + a_2b_2$) needs to be found by bootstrapping the model multiple times (Preacher, 2015; Hayes, 2013; Preacher, Rucker, and Hayes, 2007; MacKinnon, 2008). By default, the mediate function in psych does 5,000 bootstrap iterations. See the Appendix for sample output. Other packages in R that are specifically designed to test mediation hypotheses include the mediation (Tingley, Yamamoto, Hirose, Keele, and Imai, 2014) and MBESS (Kelley, 2017) packages.

When doing regressions, we sometimes are interested in the interactions of two of the predictor variables. For instance, when examining how women react to discriminatory treatment of a hypothetical other Garcia, Schmitt, Branscombe, and Ellemers (2010)\(^4\) considered the interactive effects of beliefs about inequality and type of protest (individual vs. collective vs. none) as they affected the appraisal of the other person. This example of a moderated regression is discussed by Hayes (2013).

Interactions (also known as moderation, or moderated regression) are found by entering the product term of the two interacting variables. There are several questions to ask in this analysis that will change the interpretability of the results. For example, should the data be mean-centered before finding the product term, and should the path models be done using standardized or unstandardized regressions? The recommendation from Aiken and West (1991) and Cohen, Cohen, West, and Aiken (2003) is to mean center. However, Hayes (2013) rejects this advice. In both cases, the interaction terms will be identical, but the main effects will differ depending upon centering or not centering. The argument for mean centering is to remove the artificial correlation between the main effects and the interaction term. For with positive numbers X and Y, their product XY will be highly correlated with X and Y. This means that the linear effects of X and Y will be underestimated. The setCor and mediate functions will by default mean center the data before

\(^4\)With the kind permission of Donna M. Garcia, Michael T. Schmitt, Nyla R. Branscombe, and Naomi Ellemers, the data are included as the Garcia data set in the psych package.
**Figure 2.** Regression and mediation approaches to the Tal-Or et al. (2010) data set. The curved lines represent covariances, the straight lines, regressions. Panel A shows the full regression model, Panel B shows the total effect (c=.5) and the direct effect (c’ = .1) removing the indirect effect (ab) through PMI (.19) and through Import (.20).

Finding the product term, however this option can be modified. The `lm` does not and so we need to take an extra-step to do so. The function `scale` will mean center (and by default standardize). The second question, whether to standardize or not, is one of interpretability. Unstandardized coefficients are in the units of the predictors and the criteria and show how much the DV changes per unit change in each IV. The standardized coefficients, on the other hand are unit free and show how much change occurs per standard deviation change in the predictors. Standardization allows for easier comparison across studies but at the cost of losing the direct meaning of the regression slope. In the Appendix we show the code for mean centering using `scale` and then using the `lm` function to do the regression with the interaction term. We also show how the `setCor` function combines both operations.

**Correlation, regression and decision making**

When reporting standardized regression weights ($\beta_i$) the amount of variance in the dependent variable accounted for by the regression model is $R^2 = \Sigma \beta_i r_i$. However it is important to recognize that the slopes ($\beta_i$) are the optimal fit for the observed data and that the fit will probably not be as good in another sample. This problem of overfitting is particularly problematic in machine learning (see below) when the number of variables used in the regression is very large. Thus, regression functions will report the $R^2$ as well as shrunken or adjusted $R^2$ which estimate what the fit would be in another sample. For
A: Moderated regression (std. and mean centered)

B: Response to sexism varies as type of protest

Figure 3. Two ways of showing moderation effects: Panel A, as a path diagram with the product term or Panel B: as a plot of the continuous variable (sexism) showing the individual regression slopes for the three protest conditions. Data from Garcia et al. (2010).

n subjects and k variables, the adjusted $\hat{R}^2 = 1 - \frac{(1 - R^2) \frac{n-1}{n-k-1}}{n}$ (Cohen et al., 2003), that is, there will be more shrinkage for small sample sizes and a large number of predictors.

The $R^2$ for a particular model is maximized by using the regression weights, but because of what is known as the “Robust beauty of improper linear models” (Dawes, 1979) or the principal that “it don’t make no nevermind” (Wainer, 1976), as long as the predictors are moderately correlated with the criterion, using unit weights (1, 0, -1) works almost as well. Weights are said to be ‘fungible’ (Waller, 2008; Waller and Jones, 2010) in that an infinite set of weights will do almost as good a job as the optimal weights.

Although the variance in the criterion accounted for by the predictors is $R^2$, it is better to report the actual R, which reflects the amount of change in the criterion for unit changes in the predictors (Ozer, 2007). Change is linear with R, not $R^2$. This is particularly important when discussing the correlation of a dichotomous predictor with a dichotomous outcome (e.g., applicants are selected or not selected for a job, they succeed or they fail). Consider the four outcomes shown in Table 6 applied to a decision study by Danielson and Clark (1954) and elaborated on by Wiggins (1973). Of 504 military inductees, 89 were later diagnosed as having psychiatric problems requiring their discharge. How well could this future diagnosis be predicted? Using a screening test given to all of the inductees, 55% of the future psychiatric diagnoses could be predicted, with a false alarm (false positive) rate of 19%. This leads to an accuracy of classification (Valid Positives + Valid Negatives) of .76 and a sensitivity of .55 and a specificity of .81 (Table 7). In this kind of binary
decision, the \( \phi \) coefficient is a linear function of the difference between the percent of Valid Positives and the number expected due to the base rates (BR) times the selection ratio (SR):

\[
\phi = \frac{VP - BR \times SR}{\sqrt{(BR)(1-BR)(SR)(1-SR)}}
\]  

(6)

In the case of \( BR = SR = .5 \), 50% accuracy means a 0 correlation, 60% a correlation of .2, 70% a correlation of .4, etc. That is, the number of correct predictions is a linear function of the correlation (Ozer, 2007; Rosenthal and Rubin, 1982; Wiggins, 1973).

An alternative approach when considering accuracy in decision making is known as ‘signal detection theory’ which was developed to model the detection of a signal in a background of noise (Green and Swets, 1966). \( d' \) (d-prime) reflects the sensitivity of the observer and \( \beta \) the criterion the observer was using to make the decision. Similar ideas are seen in the NHST approach to significance testing, where effect size is equivalent to \( d' \) and the criterion used (.05, .01) is the decision criterion. The relationship predicted accuracy as a function of the selection ratio, the base rates, and the size of the correlation was discussed by Taylor and Russell (1939) who present tables for different values. The equivalence of these various procedures in seen in (Figure 4) which presents graphically the cell entries in Table 7. The AUC (area under the curve) function will take the two by two table of decision theory and report \( d', \phi, r_{tetrachoric} \) as well as total accuracy, sensitivity, and specificity.

Table 6: The four outcomes of a decision. Subjects above a particular score on the decision axes are accepted, those below are rejected. Similarly, the criterion of success is such that those above a particular value are deemed to have succeed, those below that value to have failed. All numbers are converted into percentages of the total.

<table>
<thead>
<tr>
<th>Decision = Predicted Outcome</th>
<th>Accept</th>
<th>Reject</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outcome</td>
<td>Valid Positive (VP)</td>
<td>False Negative (FN)</td>
</tr>
<tr>
<td>Success</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Failure</td>
<td>False Positive (FP)</td>
<td>Valid Negative (VN)</td>
</tr>
<tr>
<td>Selection Rate (SR)</td>
<td>1-Selection Rate (1-SR)</td>
<td></td>
</tr>
</tbody>
</table>

Accuracy = \( \frac{Valid\ Positive + Valid\ Negative}{Total\ Subjects} \)

Sensitivity = \( \frac{Valid\ Positive}{(Valid\ Positive + False\ Negative)} \)

Specificity = \( \frac{Valid\ Negative}{(Valid\ Positive + False\ Negative)} \)

\( \Phi = \frac{VP-\ BR \times SR}{\sqrt{BR(1-BR)(SR)(1-SR)}} \)
Table 7: Applying decision theory to a prediction problem: the case of predicting future psychiatric diagnoses from military inductees. (Data from Danielson and Clark (1954) as discussed by Wiggins (1973).

<table>
<thead>
<tr>
<th>Predicted Positive</th>
<th>Predicted Negative</th>
<th>Row Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td>True Positive</td>
<td>49</td>
<td>40</td>
</tr>
<tr>
<td>True Negative</td>
<td>79</td>
<td>336</td>
</tr>
<tr>
<td>Column Totals</td>
<td>118</td>
<td>376</td>
</tr>
</tbody>
</table>

Fraction of Total

<table>
<thead>
<tr>
<th>Predicted Positive</th>
<th>Predicted Negative</th>
<th>Row Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td>True Positive</td>
<td>.097</td>
<td>.079</td>
</tr>
<tr>
<td>True Negative</td>
<td>.157</td>
<td>.667</td>
</tr>
<tr>
<td>Column Totals</td>
<td>.234</td>
<td>.746</td>
</tr>
</tbody>
</table>

Accuracy = .097 + .667 = .76
Sensitivity = .097/(.097 + .079) = .55
Specificity = .667 / (.667+.157) = .81
Phi = \( \frac{.097 - .196 \times .234}{\sqrt{.196 \times .804 \times .234 \times .747}} \) = .32

Latent Variable Modeling: EFA, CFA and SEM

There has long been tension in psychological research between understanding and causal explanation versus empirical prediction and control. The concept of the underlying but unobservable cause that accounts for the patterns of observed correlations was implicit in the measurement models of Spearman (1904a) and considered explicitly by Borsboom, Mellenbergh, and van Heerden (2003). This is the logic of the reflective latent variable indicate in the paths of Figure 1 from the latent variables \( \chi_1 \) or \( \chi_2 \) to the observed variables \( X_1 \ldots X_6 \). Figure 1 represents multiple causal paths: from the latent \( \chi_{1..3} \) to the observed \( X_{1..9} \) and from the latent \( \eta_{1..2} \) to the observed \( Y_{1..6} \) as well as from the latent predictors ( \( \chi_{1..3} \)) to the latent criteria ( \( \eta_{1..2} \)). In this perspective, items are reflective measures of the latent trait (Loevinger, 1957; Bollen, 2002) and can be thought to be caused by the latent trait. The contrasting approach of prediction and control was traditionally the domain of behaviorists emphasizing the power of environmental stimuli upon particular response patterns. Stimulus-response theory had no need for latent variables; outcomes were perfectly predictable from the stimulus conditions. In personality research this was the appeal of empirical keys for the MMPI (Hathaway and McKinley, 1943; Butcher, Dahlstrom, Graham, Tellegen, and Kaemmer, 1989), or Strong’s Vocational Interest Test (Strong, 1927), and continues now with the statistical learning procedures we will discuss.
Figure 4. Signal detection theory converts the frequencies of a 2 x 2 table into normal equivalents and shows the relative risks of false positives and false negatives. The number of valid positives will increase at a cost of increasing false positives. Figure from the AUC function with input from Table 7.

later. In this empirical approach, scales are composites formed of not necessarily related items. The items are said to be formative indicators that “cause” the latent variable.

Exploratory Factor Analysis

The original concept for factor analysis was Spearman’s recognition that the correlations between a number of cognitive ability tests was attenuated due to poor measurement. When correcting for measurement error (see the reliability section where we discuss such corrections for attenuation) all of the cognitive domains were correlated almost perfectly. The underlying latent factor of these tests was thought to be a measure of general intelligence.

Although the initial calculations were done on tables of correlations, when a kindly mathematician told Thurstone in 1931 that his generalization of Spearman’s procedure was just the taking the square root of a matrix (Bock, 2007), Thurstone immediately applied
this new matrix algebra to his ability measures and produced his *Vectors of the Mind* (Thurstone, 1933). Correlations were no longer arranged in tables, they were now elements of “correlation matrices”. Factor analysis was seen as the approximation of a matrix with one of lesser rank. In modern terminology, factor analysis is just an *eigen decomposition* problem and is a very straightforward procedure.

For any symmetric matrix, \( R \) of rank \( n \) there is a set of *eigen vectors* that solve the equation \( x_i R = \lambda_i x_i \) and the set of \( n \) eigenvectors are solutions to the equation

\[
XR = \lambda X
\]

where \( X \) is a matrix of orthogonal eigenvectors and \( \lambda \) is a diagonal matrix of the *eigenvalues*, \( \lambda_i \). Finding the eigenvectors and eigenvalues is computationally tedious, but may be done using the *eigen* function. That the vectors making up \( X \) are orthogonal means that \( XX' = I \) and they form the *basis space* for \( R \) that is: \( R = XX' \). In plain terms, it is possible to recreate the correlation matrix \( R \) in terms of an orthogonal set of vectors (the *eigenvectors*) scaled by their associated *eigenvalues*.

We can find the principal components of \( R \) by letting

\[
C = X\sqrt{\lambda}
\]

and therefore

\[
R = CC'.
\] (7)

But such a decomposition is not very useful, because the size (rank) of the \( X \) matrix is the same as the original \( R \) matrix. However, if the components are in rank order of their eigenvalues, the first \( k \) \((k < n)\) components will provide the best fit to the \( R \) matrix when compared to any other set of vectors. Such a principal components analysis (PCA) is useful for optimally describing the observed variables. The components are merely weighted sums of the variables and may be used in applied prediction settings. The components are the \( k \) orthogonal sums that best summarize the total variability of the correlation matrix. The *pca* function will do this analysis.

An alternative model, the *common factor* model attempts to fit the variance that the \( n \) variables have in common and ignores that variance which is unique to each variable.

\[
R \approx FF' + U^2.
\] (8)

where \( F \) is of rank \( k \), and \( U^2 \) is a diagonal matrix of rank \( n \). The \( U^2 \) matrix may be thought of as the residual variance when we subtract the model (\( FF' \)) from the data (\( R \)) \( U^2 = R - FF' \). Although it would seem that these two equations (7, 8) are quite similar, they are not. For in the first case, the components are formed from linear sums of the variables, while in the second, the variables reflect the linear sums of the factors.

Equation 7 can be solved directly for \( C \), but equation 8 has different solutions for \( F \) depending upon the values in the \( U^2 \) matrix which in turn depend upon the value of \( k \).
If we know the amount of variance each variable shares in common with all of the other variables (this is known as the communalinity and is \( h_i^2 = 1 - U_i^2 \)) then we can solve for the factors. But, unfortunately, we do not know \( U^2 \) unless we know \( F \). The solution to this conundrum takes advantage of the power of computers to do iterative solutions. Make an initial guess of \( U^2 \), solve equation 8 for \( F \) and take the resulting \( U^2 \) as the input for the next iteration. Repeat these steps until the change in \( U^2 \), from one step to the next is very small and then quit (Spearman, 1927; Thurstone, 1933,9,9).

Consider the correlation matrix in Table 8. A conventional initial estimate for the communalities (\( \text{diag}(I-U^2) \)) might be the Squared Multiple Correlation (SMC) of each variable with all the others (the last line of Table 8 shows these values). Enter these in the diagonal of the matrix and solve for \( F \). Unfortunately exploratory factor analysis is not quite as simple as this for there are at least four decisions that need to be made: what kind of correlation to use, which factor extraction algorithm to use, how many factors to extract, and what rotation or transformation should be applied?

Table 8: The Thurstone correlation matrix is a classic data set discussed in detail by R. P. McDonald (McDonald, 1985,9) and and is used as example in the sem package as well as in the PROC CALIS manual for SAS. These nine tests were grouped by Thurstone and Thurstone (1941) (based on other data) into three factors: Verbal Comprehension, Word Fluency, and Reasoning). The original data came from Thurstone and Thurstone (1941) but were reanalyzed by Bechtoldt (1961) who broke the data set into two. McDonald, in turn, selected these nine variables from the larger set of 17 found in Bechtoldt.2. The sample size is 213.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vocabulary</td>
<td>0.83</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>0.78</td>
<td>0.78</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First.Letters</td>
<td>0.44</td>
<td>0.49</td>
<td>0.46</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Four.Letter.Words</td>
<td>0.43</td>
<td>0.46</td>
<td>0.42</td>
<td>0.67</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Suffixes</td>
<td>0.45</td>
<td>0.49</td>
<td>0.44</td>
<td>0.59</td>
<td>0.54</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Letter.Series</td>
<td>0.45</td>
<td>0.43</td>
<td>0.40</td>
<td>0.38</td>
<td>0.40</td>
<td>0.29</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pedigrees</td>
<td>0.54</td>
<td>0.54</td>
<td>0.53</td>
<td>0.35</td>
<td>0.37</td>
<td>0.32</td>
<td>0.56</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>Letter.Group</td>
<td>0.38</td>
<td>0.36</td>
<td>0.36</td>
<td>0.42</td>
<td>0.45</td>
<td>0.32</td>
<td>0.60</td>
<td>0.45</td>
<td>1.00</td>
</tr>
<tr>
<td>SMC</td>
<td>0.74</td>
<td>0.75</td>
<td>0.67</td>
<td>0.55</td>
<td>0.52</td>
<td>0.43</td>
<td>0.48</td>
<td>0.45</td>
<td>0.43</td>
</tr>
</tbody>
</table>

**Which correlation?**

If the data are continuous (or have at least 8-10 response levels), then the normal Pearson r is the appropriate measure of relationship. But if the data are dichotomous (as would be the case for items if scoring correct/correct on an ability test) or polytomous (as is normally the case when scoring personality questionnaires with a 1-5 or 1-6 rating scale), then it is better to use the tetrachoric correlation (for dichotomous items) or its generalization to polytomous items, the polychoric correlation. The principal reason for
doing so is that the Pearson correlation for items that differ in their mean endorsement rate can not have a high correlation value and is attenuated. As discussed earlier, the tetrachoric is the modeled correlation of the latent traits affecting the scores on the items not the observed scores of the items themselves.

Unfortunately, using tetrachoric correlations will frequently produce correlation matrices which are said to be non-positive-definite, which means some of the eigen values of the matrix are negative. With appropriate assumptions, such matrices can be corrected (smoothed) by adding a small number to any negative eigen value, adjusting the positive ones to keep the same total, and then recreating the matrix from the original eigen vectors and the adjusted eigen values (Wothke, 1993). This is done in the cor_smooth function.

**Factor extraction.** Factors are approximate solutions to Equation 8 and have a degree of misfit. Each factoring method attempts to minimize this misfit. The basic fitting equation is

\[
E = \frac{1}{2} tr[(R - FF')W]^2
\]  

where \(tr\) means the trace (sum of the diagonals) of a matrix. If \(W\) is the identity matrix, minimizing \(E\) is equivalent to ordinary least squares (OLS); if \(W = R^{-1}\), it is generalized least squares (GLS), and if \(W = FF'^{-1}\) it is maximum likelihood (ML) (Loehlin, 2004). Maximum likelihood (Lawley and Maxwell, 1962,9) has the advantage that under normal theory it finds the model that maximizes the likelihood of the data given the model, but with the disadvantage that it requires taking the inverse of the model. GLS is a close approximation of ML, but requires that the original correlation matrix be invertible. OLS does not require taking inverses but will not produce ‘optimal’ solutions (in the ML sense). OLS (and the variant known as minimum residual (Harman and Jones, 1966) has the advantage that it is more robust to violations of the model and will produce meaningful solutions even in the presence of many, minor, ‘nuisance’ factors (MacCallum, Browne, and Cai, 2007). Empirically, although not minimizing the ML criterion, minres solutions are very close to it. All of these factor extraction techniques are available in the fa function in the psych package as are alpha factoring (Kaiser and Caffrey, 1965) and minimum rank factoring (Shapiro and ten Berge, 2002).

**Number of factors.** An unsolved problem in EFA is how many factors to extract. Henry Kaiser is said to have solved the problem every day before breakfast, but the challenge is to find the solution (Horn and Engstrom, 1979). Perhaps the best known solution (Kaiser, 1970) is also the worst: extract as many factors as the number of principal components with eigen values larger than 1. This procedure, although the default for many commercial packages, routinely will extract too many factors (Revelle and Rocklin, 1979). Statistical criteria (e.g., extract factors as long as the \(\chi^2\) of the residual matrix is significant) suffer from the problem of being dependent upon sample size: the larger the sample, the more factors are extracted. An appealing technique is to plot the successive eigen values and look for a sharp break. Where the scree of trivial factors suddenly jumps to
larger values, stop factoring (Cattell, 1966b). Another useful technique involving the plot of the eigen values is to compare observed values versus those from random data (Horn, 1965). When the observed eigen values are less than those from random data, too many factors have been extracted. This is a useful rule of thumb, but seems to break down with more than about 500-1000 subjects, at which point the random eigen values are all essentially 1.0. Yet another approach is to plot the size of the average minimum partial correlation of the residual matrix. Where this achieves a minimum is an appropriate place to stop (Velicer, 1976). For factoring items, a comparison of the goodness of fit of models which zero out all except the largest loading for each item seems to produce a reasonable estimate (Revelle and Rocklin, 1979). Finally, continuing the the extraction of factors as long as they are interpretable is not unreasonable advice, although those of us unable to interpret many factors will tend to be biased towards extracting fewer. The \texttt{nfactors} function applies all of these tests, but unfortunately the typical result is that none of them agree.

\textit{Rotations and Transformation}. Given a factor solution $F$ with elements (loadings) of $f_{ij}$ what is the best way to interpret it? The loadings reflect the correlation of the factors with the items and differ by item and by factor. The sum of the squared loadings for each item (row wise) is the amount of variance in that item accounted for by all of the factors. This is known as the communality ($h_i^2 = \sum f_{ij}^2$). Items with high communality are well explained by all of the factors, those with low communality are badly explained. For the same value of communality, a variable is said to be more complex if several variables are needed to explain its variance (have high loadings) and less complex if just one variable has a high loading. An index of item complexity is $c_i = \frac{(\sum f_{ij}^2)^2}{\sum f_{ij}^4}$ which will achieve a minimum of 1 if all of the explained variance in an item is due to one factor (Hofmann, 1978). A similar measure of factor complexity is to do the operation column wise. Multiplying the $F$ matrix by a orthogonal transformation matrix($T$) will not change the communalities but can change the item and factor complexities. In the orthogonal case, this is known as rotation, if the resulting solution has correlated factors, we should refer to this as an oblique transformation. We want to chose a transformation that provides a more ‘simple structure’ (Thurstone, 1947) than the original $F$ matrix. A number of different solutions to this problem take advantage of the \textit{GPArotation} package (Bernaards and Jennrich, 2005) and are included in the \texttt{fa} function. Browne (2001) discusses how many of these are part of the (Crawford and Ferguson, 1970) family of rotations. Some of the most frequently used include \texttt{Varimax} (Kaiser, 1958,9) and \texttt{Quartimax} (Neuhaus and Wrigley, 1954) for orthogonal rotations and \texttt{oblimin} (Harman, 1976; Jennrich, 1979), \texttt{Promax} (Hendrickson and White, 1964) \texttt{Bifactor} (Holzinger and Swineford, 1937; Reise, 2012) and \texttt{Geomin} (Yates, 1988) for oblique solutions. Unfortunately, some of these rotation procedures achieve local minima in their fitting functions and it is recommended to do multiple random restarts to confirm solutions. The net result of an oblique tranformation is the factor \textit{pattern} matrix.
(F) and the factor structure matrix (S = Fφ) where φ is the correlation between the factors. When reporting an oblique transformation, it is important to show both the pattern (F) and the correlation (φ).

Factor score indeterminancy. A problem with factor analysis is that although the model is well defined at the structure level (modeling the covariances of the variables) it is indeterminate at the individual score level (Grice, 2001). Factor scores are best estimates of an individual’s score but should not be equated with the factor. Factor score estimates correlate with the latent factors, but this correlation may be far from unity. The fa function returns the correlation of the factor scores with the factor. If the correlation is less than .707 (an $R^2$ of .5), then two estimates of the factor score vector may actually be negatively correlated with each other. Correlations of the factors with the factor score estimates are a function of the number of variables marking the factor as well as the communality of the variables.

Table 9: The minres solution using the fa function of the Thurstone data set. The factor solution was then transformed to simple structure using the oblimin transformation. $h^2$ is the communality estimate, $u^2$ is the unique variance associated with the variable, com is the degree of item complexity. The pattern coefficients are shown as well as the correlation (φ) between the factors. Because this is an oblique solution, the correlation matrix is reproduced by $FφF' + U^2$. The sums of squares for an oblique solution are the diagonal elements of $φF'F$.

<table>
<thead>
<tr>
<th>Variable</th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
<th>$h^2$</th>
<th>$u^2$</th>
<th>com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>0.90</td>
<td>-0.03</td>
<td>0.04</td>
<td>0.82</td>
<td>0.18</td>
<td>1.01</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>0.89</td>
<td>0.06</td>
<td>-0.03</td>
<td>0.84</td>
<td>0.16</td>
<td>1.01</td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>0.84</td>
<td>0.03</td>
<td>0.00</td>
<td>0.74</td>
<td>0.26</td>
<td>1.00</td>
</tr>
<tr>
<td>First.Letters</td>
<td>0.00</td>
<td>0.85</td>
<td>0.00</td>
<td>0.73</td>
<td>0.27</td>
<td>1.00</td>
</tr>
<tr>
<td>Four.Letter.Words</td>
<td>-0.02</td>
<td>0.75</td>
<td>0.10</td>
<td>0.63</td>
<td>0.37</td>
<td>1.04</td>
</tr>
<tr>
<td>Suffixes</td>
<td>0.18</td>
<td>0.63</td>
<td>-0.08</td>
<td>0.50</td>
<td>0.50</td>
<td>1.20</td>
</tr>
<tr>
<td>Letter.Series</td>
<td>0.03</td>
<td>-0.01</td>
<td>0.84</td>
<td>0.73</td>
<td>0.27</td>
<td>1.00</td>
</tr>
<tr>
<td>Pedigrees</td>
<td>0.38</td>
<td>-0.05</td>
<td>0.46</td>
<td>0.51</td>
<td>0.49</td>
<td>1.96</td>
</tr>
<tr>
<td>Letter.Group</td>
<td>-0.06</td>
<td>0.21</td>
<td>0.63</td>
<td>0.52</td>
<td>0.48</td>
<td>1.25</td>
</tr>
</tbody>
</table>

SS loadings 2.65 1.87 1.49

Confirmatory Factor Analysis

With the introduction of statistical measures of fit, it is now possible to fit and then test particular factor models (Jöreskog, 1978; Rindskopf and Rose, 1988). Because most models do not fit in an absolute sense, model comparison is recommended. Fit
Table 10: Comparing the Varimax orthogonally rotated PCA (RC\textsubscript{1-3}), Minimum Residual (MR\textsubscript{1-3}), obliquely transformed PCA (TC\textsubscript{1-3}) and oblique Minimum Residual (MR\textsubscript{1-3}) solutions. In order to show the structure more clearly, loadings > .30 are boldfaced.

<table>
<thead>
<tr>
<th>Variable</th>
<th>RC1</th>
<th>RC2</th>
<th>RC3</th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
<th>TC1</th>
<th>TC2</th>
<th>TC3</th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>0.86</td>
<td>0.24</td>
<td>0.23</td>
<td>0.90</td>
<td>0.01</td>
<td>0.03</td>
<td>0.83</td>
<td>0.25</td>
<td>0.26</td>
<td>0.90</td>
<td>-0.03</td>
<td>0.04</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>0.85</td>
<td>0.31</td>
<td>0.19</td>
<td>0.88</td>
<td>0.10</td>
<td>-0.02</td>
<td>0.83</td>
<td>0.32</td>
<td>0.22</td>
<td>0.89</td>
<td>0.06</td>
<td>-0.03</td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>0.85</td>
<td>0.26</td>
<td>0.19</td>
<td>0.89</td>
<td>0.04</td>
<td>-0.01</td>
<td>0.78</td>
<td>0.28</td>
<td>0.23</td>
<td>0.84</td>
<td>0.03</td>
<td>0.00</td>
</tr>
<tr>
<td>First.Letters</td>
<td>0.23</td>
<td>0.82</td>
<td>0.23</td>
<td>0.03</td>
<td>0.84</td>
<td>0.07</td>
<td>0.79</td>
<td>0.23</td>
<td>0.00</td>
<td>0.85</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Four.Letter.Words</td>
<td>0.18</td>
<td>0.79</td>
<td>0.30</td>
<td>-0.03</td>
<td>0.81</td>
<td>0.16</td>
<td>0.21</td>
<td>0.71</td>
<td>0.29</td>
<td>-0.02</td>
<td>0.75</td>
<td>0.10</td>
</tr>
<tr>
<td>Suffixes</td>
<td>0.31</td>
<td>0.77</td>
<td>0.06</td>
<td>0.17</td>
<td>0.79</td>
<td>-0.14</td>
<td>0.31</td>
<td>0.62</td>
<td>0.13</td>
<td>0.18</td>
<td>0.63</td>
<td>-0.08</td>
</tr>
<tr>
<td>Letter.Series</td>
<td>0.25</td>
<td>0.16</td>
<td>0.83</td>
<td>0.10</td>
<td>-0.01</td>
<td>0.84</td>
<td>0.23</td>
<td>0.18</td>
<td>0.80</td>
<td>0.03</td>
<td>-0.01</td>
<td>0.84</td>
</tr>
<tr>
<td>Pedigrees</td>
<td>0.53</td>
<td>0.08</td>
<td>0.61</td>
<td>0.49</td>
<td>-0.14</td>
<td>0.55</td>
<td>0.45</td>
<td>0.17</td>
<td>0.52</td>
<td>0.38</td>
<td>-0.05</td>
<td>0.46</td>
</tr>
<tr>
<td>Letter.Group</td>
<td>0.10</td>
<td>0.31</td>
<td>0.80</td>
<td>-0.11</td>
<td>0.21</td>
<td>0.82</td>
<td>0.16</td>
<td>0.31</td>
<td>0.63</td>
<td>-0.06</td>
<td>0.21</td>
<td>0.63</td>
</tr>
</tbody>
</table>

Statistics include $\chi^2$, the Root Mean Square Error of Approximation (RMSEA) which adjusts the $\chi^2$ for the degrees of freedom and sample size (N) ($\text{RMSEA} = \sqrt{\frac{\chi^2 - df}{N-1}}$), the standard deviation of the residuals (RMSR), the Akaike Information Criterion ($AIC = \chi^2 + k(k - 1) - 2df$) which also considers the number of variables in the model (k), and the Bayesain Information Criterion ($BIC = \chi^2 + \ln(N)(k(k + 1)/2 + df)$). These fits are actually estimates of misfit. That is, the larger the $\chi^2$, the less well the model fits the data. The question then becomes how bad is bad? Barrett (2007) gives very strict interpretation of what makes a good model; Marsh, Hau, and Wen (2004) suggests there is no golden rule of fit, and Loehlin and Beaujean (2017) give a very useful discussion of how to report fit statistics. The most important thing to remember is that this is a model comparison procedure where we compare multiple models to see which is better, not which is correct.

EFA is seen as a hypothesis generation procedure and CFA as a hypothesis confirmation procedure: the initial model might be derived from an EFA and then tested using CFA on a different data set. A powerful but easy to use package to do this is lavaan (Rosseel, 2012). Other CFA packages include sem (Fox, Nie, and Byrnes, 2016) and OpenMX (Neale, Hunter, Pritikin, Zahery, Brick, Kickpatrick, Estabrook, Bates, Maes, and Boker, 2016). lavaan syntax is very straightforward, and allows one to specify and test any particular model.

An important use of CFA is evaluating whether the factor structure of a set of variables is the same across time, or across groups. These are important questions when comparing people across groups or people over time, for it is not possible to make comparisons if the measures are different. Three tests of factor invariance are typically considered: configural, metric, and scaler (also known as weak, strong, and strict). Configural asks whether the structure is the same across groups, metric asks whether the loadings are the same (or do not differ very much), and scaler ask whether the means and intercepts of
the factors are the same. Testing for invariance is thus a set of comparisons of structures across groups. The `measurementInvariance` function in the `semTools` package (semTools Contributors, 2016) has been developed to do this in `lavaan`.

**Structural Equation Modeling**

Combining observations, latent variables and regression into one structural model (see Figure 1) seems to be an obvious step. How to combine the path tracing rules of Wright (1920), with factor analysis and reliability theory by using modern estimation algorithms was, however, an important insight. Developed independently by Keesling (1972), Jöreskog (1977) and Wiley (1973) the use of fitting regression models with latent variables was soon identified with a computer algorithm for Linear Structural Relations (LISREL) (Jöreskog, 1978; Joreskog and Sorbom, 1993) (see Tarka (2018) for a thorough history). Very influential texts on SEM include those of Bollen (1989, 2002), Loehlin and Beaujean (2017) and Mulaik (2009). How to report SEM is discussed by McDonald and Ho (2002) and others.

In addition to LISREL, the development of the proprietary programs EQS (Bentler, 1995) and MPLUS (Muthén and Muthén, 2007) made SEM available to many. Now, with the introduction into R of the `sem` (Fox, Nie, and Byrnes, 2013), `lavaan` (Rosseel, 2012) and OpenMx (Neale et al., 2016) packages, SEM and CFA are part of the open source armamentarium for all. Bayesian approaches are available in the `blavaan` (Merkle and Rosseel, 2016) which takes advantage of the `lavaan` package.

Combining formative causal variables with reflective indicator variables is done in MIMIC models (multiple indicators, multiple causes) where a latent variable is seen as formed from a set of causal variables but in turn causes another latent variable which is indicated by a number of reflective measured variables. An early example of a MIMIC model is the causative effect of education, occupation, and income on the latent variable of social status which in turn effects the latent variable of social participation which is measured by church attendance, memberships and the number of friends seen (Jöreskog and Goldberger, 1975).

Perhaps one of the most powerful uses of SEM techniques is in examining growth curves over time. Given a set of participants at time 1, what happens to them over weeks, months, or years? Growth curve models allow for the separation of trait and state effects (Cole, Martin, and Steiger, 2005) as well as an examination of change (McArdle and Bell, 2000; McArdle, 2009). Tutorials for using `lavaan` include growth curve analysis and are included in the help pages for `lavaan`.

In an important generalization of the problem of fungible regression weights (Waller, 2008; Waller and Jones, 2010) Lee, MacCallum, and Browne (2018) showed how with a very small decrease in fit (increase in RMSEA), path coefficients in equivalent models that fit equally well can actually differ in sign. This is just one of the many cautions in how to interpret SEM results. For SEM fit statistics are merely fits of a model to the data. What is needed is comparisons of the fit of alternative/equivalent models. It is important to realize...
that reversing the direction of causal arrows in many SEM models does not change the fit, but drastically changes the interpretation (MacCallum, Wegener, Uchino, and Fabrigar, 1993).

Reliability: correlating a test with a test just like it

A powerful use of correlation is assessing reliability. All measures are contaminated with an unknown amount of error. Reliability is just the fraction of the measures that is not error and is the correlation of a measure with another measure that is just like the first measure (Revelle and Condon, 2018a,0). Using $V_x$ to represent observed total test variance and $\sigma^2_e$ to represent unobserved error variance then the reliability ($r_{xx}$) of a measure is

$$r_{xx} = 1 - \frac{\sigma^2_e}{V_x}. \quad (10)$$

In terms of the observed and latent variables in Figure 5, $x = \chi + \epsilon_1$, and a test just like it is $x' = \chi + \epsilon_2$ with correlation $r_{xx}$. To infer the latent correlation between $\chi$ and $\eta$, $r_{\chi\eta}$, we can correct the observed correlation $r_{xy}$ for the reliabilities $r_{xx}$ and $r_{yy}$

$$r_{\chi\eta} = \frac{r_{xy}}{\sqrt{r_{xx}r_{yy}}}, \quad (11)$$

Equation 11 was proposed by Spearman (1904b) and the problem since then has been how to estimate the reliability. This is important because if we underestimate the reliability, we will overestimate the disattenuated correlation (Equation 11). There are several different ways to estimate reliability. All agree on the basic principle that items represent some unknown amount of latent score and another unknown amount of error score; the problem is how to measure the relative contributions. Before the era of modern computers, several shortcuts were proposed that – with some very strong assumptions – would allow reliability to be found from the total test variance and the sum of the item variances. Equivalent forms of this procedure are known as (KR20, Kuder and Richardson, 1937) ($\lambda_3$, Guttman, 1945) and ($\alpha$, Cronbach, 1951). Essentially these coefficients are a function of the average inter-item correlation and do not depend upon the structure of the test items. They are all available in the alpha function in psych.

Model based reliability measures

Procedures that take into account the internal structure of the test using factor analysis (so called model based procedures) include $\omega_t$ and $\omega_h$ of McDonald (1999), and various estimates of the greatest lower bound of the test reliability (Bentler, 2017). By applying factor analytic techniques, it is possible to estimate the amount of variance in a test that is attributable to a general factor ($\omega_h$) (Revelle and Zinbarg, 2009; Zinbarg, Revelle, Yovel, and Li, 2005), as well as the general plus all group factors (total reliability or $\omega_t$). With modern computing techniques, these model based estimates are easy to find.
Figure 5. The basic concept of reliability and correcting for attenuation. All four observed variables \((x, x', y, y')\) reflect the latent variables \(\chi\) and \(\eta\) but are contaminated by error \((\delta_1, \delta_2, \epsilon_1, \epsilon_2)\). Adjusting observed correlations \(r_{xy}\) by reliabilities \((r_{xx'}, r_{yy'})\) estimates underlying latent correlations \(\rho_{\chi\eta}\). (See Equation 11). Observed variables and correlations are shown in conventional Roman fonts, latent variables and latent paths in Greek fonts.

(e.g., \textit{omega} will find \(\omega_h\) and \(\omega_l\) as well as \(\alpha\)). Estimates of \(\omega_h\) and \(\omega_l\) are preferred over \(\alpha\) because they are sensitive to the actual structure of the test. \(\alpha\) is an estimate based upon the average correlation and the very strong assumption that all the items are equally good measures of the latent trait. (More formally, the items are assumed to be \(\tau\) equivalent: they all have equal loadings on the latent trait). This was a reasonable assumption to make as a short cut before we had computers. Now, there is no reason to settle for the shortcut. It is not uncommon to find tests with moderate levels of \(\alpha\) that actually measure two unrelated or only partially related constructs. It is only by applying model based techniques that we can identify the problem (e.g., Rocklin and Revelle, 1981).
Reliability of raters

Rather than evaluating the reliability of a test, sometimes we want to know how much raters agree with each other when making judgements. If the ratings are numeric, this is done by finding one of several Intraclass Correlations (ICC). Depending upon whether we view raters as random or fixed, and whether the raters rate all subjects or just one each, and whether we pool the judgements of different raters, we end up with six different ICCs (Shrout and Fleiss, 1979; Revelle and Condon, 2018b), all of which are found by the ICC function. ICC uses the power of R to chain functions together: it performs a one or two way analysis of variance, extracts the mean squares or estimates of variance components, and finds the appropriate ratio of variance components.

If the ratings are categorical rather than numeric, it is possible to compare the agreement of two raters in terms of Cohen’s Kappa statistic (Cohen, 1960) which corrects the observed proportions of agreement for the expectations given the marginal base rates. This is done by the cohen.kappa function. An example of such ratings is given by Guo et al. (2016) who had raters evaluate the presence or absence of particular themes in a set of life narratives.

Structure vs. Process

Factor analysis and estimates of reliability typically examine the structure of personality items and tests. In terms of Cattell’s data box of people by measures by occasions (Cattell, 1946) this an example of what Cattell called “R” analysis (people over measures). They do not tell us about how people differ over time. Repeated measures allow us to examine the process of change. With the use of multi-level modeling techniques, it is possible to examine individual differences in within person structure over time (Cattell called this “S” analysis).

Statistical analysis of within subject variability

Although initially done using daily diaries (Bolger, Davis, and Rafaeli, 2003), with the use of personal digital assistants and now cell phone apps, it is possible to collect intensive within subject data once to many times per day for several weeks or even months (Fisher, 2015; Wilt, Funkhouser, and Revelle, 2011; Wilt, Bleidorn, and Revelle, 2017; Wilt and Revelle, 2017). Excellent reviews of how to analyze these intensive longitudinal data include Hamaker, Ceulemans, Grasman, and Tuerlinckx (2015) and Hamaker and Wichers (2017). Shrout and Lane (2012) and Revelle and Wilt (2019) provide useful tutorials for examining multilevel reliability, and the multilevel.reliability function will do all these calculations in R. A basic question is whether the data are ergodic (each individual subject can be seen as representing the entire group) or whether the patterning of each individual is a meaningful signal in its own right. Different approaches to ergodicity include those of Nesselroade and Molenaar (2016) and Revelle and Wilt (2016).
Functions to do these within subject analyses include multilevel regression using \texttt{lme4} and \texttt{nlme}, correlational structures within and between groups using \texttt{statsBy} and examining factor structures for invariance across subjects using \texttt{measurementinvariance} in the \texttt{semTools} package. The \texttt{multilevel} package (Bliese, 2016) comes with very useful documentation on doing some of the more complicated forms of multilevel analyses.

\textit{Computational modeling of process}

Another very powerful approach to studying within person processes is computational modeling. This is not a statistical approach so much as a way to develop and test the plausibility of theories. It is however, an important use of computer analysis for it can compare and test the fit of alternative dynamic models. Read and Miller and their colleagues (Read, Vanman, and Miller, 1997; Read, Monroe, Brownstein, Yang, Chopra, and Miller, 2010; Yang, Read, Denson, Xu, Zhang, and Pedersen, 2014; Read, Brown, Wang, and Miller, 2018) have implemented a neural network model of the structure and dynamics of individual differences. (Read et al., 2010). Pickering (2008) has implemented several different representations of Gray’s Reinforcement Sensitivity Theory (Gray, 1991; Gray and McNaughton, 2000). Although written in MatLab, it is straight forward to translate them into R. A model of the dynamics of action (DOA, Atkinson and Birch, 1970) was implemented as program for main frame computers (Atkinson, Bongort, and Price, 1977) and then reparameterized as the Cues-Tendency Action model and implemented as the \texttt{cta} function (Revelle, 1986; Revelle and Condon, 2015). The CTA model has been combined with the RST model into the \texttt{cta.rst} function by Brown (2017) and provided good fits to empirical data.

\textit{Other statistical techniques}

\textit{Aggregating data by geographic location}

Most personality analysis focuses on individuals, but many interesting questions may be asked concerning differences in the aggregated personality of groups. Whether aggregating the scores of subjects by college major, socioeconomic status, or geographic location, the typical first step is the same: find the mean score for each group. The \texttt{statsBy} function provides mean values for any number of variables by a selected grouping variable, as well as a suite of statistics and output for analysis of aggregated variables. For example, the \texttt{statsBy} function outputs a correlation matrix for both between groups (the \texttt{rbg} object) and within groups (the \texttt{rwg} object). The correlation between aggregated groups is known to sociologists as the ecological correlation (Robinson, 1950). The \texttt{rbg} object weights correlations by the number of subjects in each group due to the fact that estimates of a mean are more accurate with more subjects. A non-weighted between groups correlation matrix could be obtained by applying the \texttt{cor} function to the \texttt{mean} object of \texttt{statsBy} output.
When analyzing aggregated data, it is critical to keep in mind that a correlation between two variables may not be consistent between groups and within groups (Yule, 1903; Simpson, 1951). Kievit, Frankenhuis, Waldorp, and Borsboom (2013) provide some excellent illustrations of this phenomenon, known as the Yule-Simpson paradox, where the within group correlations are of the opposite sign of the between groups correlations; for example, although a higher dosage of a medication is positively related to the likelihood of patient recovery across genders, dosage is negatively correlated with patient recovery within each gender. Another striking example of the effect is the finding that although at the aggregate level, the University of California seemed to discriminate against women in their admission policy, the individual departments actually discriminate in their favor (Bickel, Hammel, and O’Connell, 1975). The simpsons package (Kievit and Epskamp, 2012) allows for detailed examination of data that can produce this effect. The important point, as made by Kievit et al. (2013) and Robinson (1950) is not to dismiss aggregate level relationships but to realize that that the level of generalization depends upon the level of analysis.

How large are the effects of aggregation? Two coefficients (intraclass correlations or ICCs) are reported when examining aggregated data. ICCs describe variance ratios for each aggregated variable in terms of within and between group variance components. ICC1 is an effect size that indicates the percentage of variance in subjects’ scores that is explained by group membership (Shrout and Fleiss, 1979). Although sometimes expressed in terms of the between group and within group means squares from the traditional analysis of variance approach, a clearer definition may be expressed as the variance ratio of between group variance (σbg²) to the total variance. The total is the sum of the between (σbg²) and within group (σwg²) variance. ICC2 (also known as ICC1k) takes into account the average number of observations within group (n̄) and is a measure of the reliability of the group mean differences. It is the Spearman-Brown reliability formula applied to ICC1:

\[
\text{ICC1} = \frac{\sigma_{bg}^2}{\sigma_{bg}^2 + \sigma_{wg}^2} \quad \text{ICC2} = \frac{\sigma_{bg}^2}{\sigma_{bg}^2 + \frac{\sigma_{wg}^2}{n̄}}.
\]  

(12)

In plain English, ICC2 indicates the extent to which the aggregated scores of a variable are reliably different from one another. Assuming one’s current data are a random sample, if a new random sample is collected with the same average number of participants per group, ICC2 estimates the correlation between the group scores of the first sample and the second sample (James, 1982). ICC2 indicates that aggregated scores are still reliable (i.e., a high ICC2) even if there is a miniscule amount of variance explained in aggregation (i.e., a low ICC1), provided one has enough subjects per group (i.e., a large n̄). The statsBy function outputs ICC1 (the ICC1 object), ICC2 (the ICC2 object) and the number of subjects in each group who responded to each variable (the n object).

In the last two decades, international collaborations and online personality assessments have collected enormous data sets with samples an order of magnitude larger than
what was possible a few decades ago (e.g., Gosling, Vazire, Srivastava, and John, 2004; Revelle, Condon, Wilt, French, Brown, and Elleman, 2016). Geographical psychology is a subfield that has taken advantage of these large data sets, investigating how and why psychological phenomena are aggregated by residence of geographic locations (Rentfrow and Jokela, 2016), both large (e.g., countries; McCrae and Terracciano, 2008) and small (e.g., postal codes; Jokela, Bleidorn, Lamb, Gosling, and Rentfrow, 2015). Relatively straightforward correlational analyses at an aggregated geographic level (e.g., Rentfrow, Gosling, and Potter, 2008) are replicable (e.g., Elleman, Condon, Russin, and Revelle, 2018). Geographical psychology researchers have started to explore novel approaches, such as determining the extent to which a psychological phenomenon is spatially clustered between locations; for a psychological variable, is a given location more similar to neighbor locations than more distant locations (e.g., Jokela et al., 2015). Bleidorn, Schönbrodt, Gebauer, Rentfrow, Potter, and Gosling (2016) explored both individual and aggregated levels of their data to calculate “person-city personality fit” and found that this fit was related to the life satisfaction of individuals. The complexities of spatial analysis are beyond the scope of this chapter but see Rentfrow and Jokela (2016) for an overview and Rentfrow (2014) for details. The spdep package (Bivand and Piras, 2015) supplies functions pertaining to spatial autocorrelation, weights, statistics, and models.

Statistical Learning Theory

The study of individual differences has expanded beyond academic personality research to computer scientists who are taking advantage of the “big data” possible to collect through web based techniques. Algorithms popular among computer scientists to analyze individual differences data in a prediction context are known generically as “machine learning” or “statistical learning theory.” Although some of these techniques are new, some of them repackaged traditional methods with new labels (e.g., the $\phi$ coefficient of Pearson and Heron, 1913, has been ‘rediscovered’ as a measure of fit but with a new name: the Matthews Correlation Coefficient). We include a brief discussion of these techniques as many personality researchers will likely interact with computer scientists and it is worth learning the “new” vocabulary.

Machine learning is a term without a universally agreed-upon definition. In general, it concerns the prediction of outcome variables from models trained on other datasets and it can be used to refer to a broad range of techniques from logistic regression to neural networks (Hastie, Tibshirani, and Friedman, 2001). The core emphasis in machine learning is on generalization of algorithmic performance to new data, meaning that researchers must shift their focus away from explanation of underlying constructs and toward prediction when using these techniques (Yarkoni and Westfall, 2017). For the purposes of this chapter, it is useful to illustrate some of the methods in the domain of machine learning that are distinct from the traditional regression-based modeling that is more common in psychology.

One such method falls underneath the umbrella term of classification and regression tree (CART) methods. CART methods involve the recursive separation of observations
into distinct subgroups with a goal of enhancing subgroup homogeneity. The algorithm will first segment the observations based on the value of the variable that it has found to lead to the largest reduction in impurity, the exact measure of which depends on the type of problem at hand (e.g., Gini impurity index, entropy). Each of the subgroups created in this split will then be considered separately for further partitioning, until a desired fit to the training data has been reached. This process can be computed automatically through the rpart package (Therneau and Atkinson, 2018). CART methods are readily amenable to the creation of attractive output in the form of decision trees, which graphically display a series of sequential steps constituting the final partitions determined by the algorithm. These figures are interpretable by non-statisticians with little training, enhancing the applicability of these methods to a variety of applied contexts. Unfortunately, there is a downside to these methods: classification and regression trees tend to overfit the training sample data, meaning that they tend to extend beyond interpretable signal in a dataset to incorporate noise. Decision tree methods tend to produce models with a high amount of variability. As such, while predictive accuracy may be acceptable on the training dataset, these methods tend to not perform as well as simpler models when applied to out-of-sample data. Fortunately, their predictive capabilities can be dramatically enhanced by incorporating a class of techniques called ensemble methods, which aggregate many different trees to harness their power.

In general, ensemble methods take advantage of the power of averages to create estimates that have lower variance than any of the constituent observations. The logic behind the utility of averages can be understood through an extension of the Central Limit Theorem, in which the variance of the mean of a group of n independent observations, each with a variance of $\sigma^2$, is $\sigma^2/n$. As such, if we were able to create trees from many different training datasets and aggregate the results, the resultant model would have lower variance than any individual model alone (James, Witten, Hastie, and Tibshirani, 2013). Unfortunately, this is not typically a feasible process, as it would be rare to have many different training datasets at the ready. However, we are able to approximate this process through a method of bootstrap random sampling in which we repeatedly take random samples with replacement from our training dataset, creating many different bootstrapped datasets. This method of aggregation can be applied to a variety of statistical techniques, but applying this method to decision trees, we are able to fit a tree to each of these bootstrapped training datasets, aggregate the results, and get an ultimate prediction that does not suffer as much from the high variance concerns of the individual trees. Thus, ensemble methods form a relatively simple means by which the predictive power of classification and regression trees can be enhanced, making them compelling options for personality researchers with the goal of enhancing prediction.

Three of the most popular ensemble techniques used in the context of tree-based machine learning methods are bagging, random forests, and boosting. In bagging, observations are randomly sampled from the training dataset through bootstrapping methods to create many distinct datasets. A different tree is then grown based on each of these
bootstrapped datasets, resulting in many trees, each of which have a potentially different set of decision rules on which they based their predictions. The results of these trees are then aggregated to create a final prediction for each observation. Predictions are evaluated on the ‘out-of-bag’ observations that were left out of the bootstrap samples, providing a validation set of all observations not used in the creation of a subset of trees (James et al., 2013). Random forest methods can be viewed as special forms of bagging in which variables are randomly selected alongside the observations in the training dataset. More specifically, during the process of growing the tree from the training dataset, variables are randomly selected at each node to be used in the creation of the tree. Why add this additional step? While the general process of bagging reduces variance by bootstrapping multiple trees, each of the trees will be correlated due to the inclusion of identical variables at every split. In certain cases (i.e., when one variable is much more important in the prediction of the outcome than others) this will lead to many trees that are nearly identical. Averaging trees that are very similar will not lead to as large a reduction in variance as averaging uncorrelated trees. Random forest methods address this by changing the variables made available at each potential node split in the trees, effectively decorrelating the trees and theoretically decreasing the variance of a given predictive model. In general, if $p$ is the number of predictors, $\sqrt{p}$ is the number of variables selected at each split for classification problems and $p/3$ variables are selected at each split for regression problems, although this number should be determined based on hyperparameter tuning given the dataset at hand (Hastie et al., 2001). Both bagging and random forest techniques can be computed in R using the randomForest package (Liaw and et al., 2002). By setting the ‘mtry’ argument to equal the number of variables in the model, a bagged model will be run, otherwise it will be random forest.

Boosting methods are conceptually similar to bagging and random forests, but alter the way that the trees update their information. Specifically, boosting involves sequentially updating each tree by fitting the residuals of the tree before it. In this manner, boosting attempts to target the areas of weakness of the previous trees and update them accordingly. The boosting algorithm completes this procedure slowly to avoid overfitting, each time fitting the residuals of the previous model before adding this fitted tree back to the original tree after applying a shrinkage parameter, thus updating the residuals. The power in the predictive ability of boosting comes from its slow progression and sequential growth based on residuals of previous trees. However, this practice of fitting residuals can lead to overfitting if the tree grows too fast and thus we want the algorithm to proceed slowly. Boosting tends to outperform bagging and random forest on prediction metrics, but may not be as conceptually clear as those methods due to the fitting of residuals. The preference is up to the researcher. Popular packages in R to fit boosted models include xgboost (Chen, He, Benesty, Khotilovich, Tang, Cho, Chen, Mitchell, Cano, Zhou, Li, Xie, Lin, Geng, and Li, 2018) and gbm (Ridgeway et al., 2017). Python is another language commonly used in machine learning as it allows somewhat faster processing of very large data sets.

Taken together, these ensemble methods are referred to as “black box” methods,
which means that their exact inner workings remain unknown to the practitioner. In other words, while a researcher may be able to tell that a random forest model provides excellent predictive accuracy, they would not be able to view each decision tree used in the crafting of the prediction. As such, these ensemble methods are best used when the major aim of a project is prediction and they may not be appropriate for situations in which a precise theoretical model is desired. This may be disconcerting to personality psychologists who have largely been trained to prioritize theoretical modeling, but it is simply another possible way to analyze data with a predictive focus. Researchers are not left completely in the dark about the inner workings of the models, however: they are able to influence the way that the model constructs and aggregates the individual trees in these ensemble methods through the selection of various hyperparameters. These hyperparameters include the number of trees grown in all three methods, number of variables selected at each node in random forests, and how slowly the trees grow in boosting. Adjusting these hyperparameters to influence the performance of the ensemble methods is critical to predictive performance and can be done with through trial and error or cross validation.

The aforementioned methods are intriguing in that they provide the tools for creating a more prediction-focused study of individual differences. While promising in their potential impact on the field in the future, machine learning results should generally be viewed through a lens of caution, as many complicated methods may be outperformed by comparatively simpler methods of linear and logistic regression. A shift toward a more predictive science would be welcomed, but we must be sure to select methods to suit the problem at hand and not just apply these methods with abandon.

Conclusion

Personality research has come a long way from the simple correlation of Galton (1888), Pearson (1895), and Spearman (1904b). Advances in the past few years have brought powerful computation to the desk or lap of the individual researcher. Open source software has made complex research questions answerable by people anywhere in the world. Computational model techniques that used to take days on multi-million dollar computers can now be done in seconds on very affordable laptops. Data can be shared across the web, analyses can be duplicated using published and open source computer code. Statistical testing and modeling of psychological data and theory has never been easier for those willing to learn the modern methods.
References


Hathaway, S., and McKinley, J. (1943). Manual for administering and scoring the MMPI.


Lovelace, A. A. (1842). Sketch of the analytical engine invented by Charles Babbage, by LF Menabrea, officer of the military engineers, with notes upon the memoir by the translator. Taylor’s Scientific Memoirs, 3, 666–731.


Appendix

The R code for the various examples is shown here.

Table 1 was a subset of the msqR data set which is included in the psych package. Here we show the size of the entire data set (6411 rows by 79 columns), the number of subjects with repeated measures (2086), and how to form a subset of the first eight cases for both time 1 and time 2.

R code

```r
msq.items <- c(“anxious”, “at.ease”, “calm”, “confident”, “content”, “jittery”, “nervous”, “relaxed”, “tense”, “upset”) #these overlap with the sai
dim(msqR) #show the dimensions of the data set
colnames(msqR) #what are the variables
table(msqR$time) #show the number of observations with various repeated values
example <- msqR[c(1:8,69:76),c(cs(id,time),msq.items)]
df2latex(example) #make a \LaTeX table of the example data
```

**Descriptive statistics**

Table 2 shows descriptive statistics.

R code

```r
describe(msqR[c(1:8,69:76),c(cs(id,time),msq.items)],IQR=TRUE) #for the data in table 1
describe(msqR[c(cs(id,time),msq.items)]) #describe the entire data set
```

**Correlation and regression**

Table 5 shows the correlation matrix from the Tal-Or et al. (2010) data set. Here we show several different ways to show those correlations and to test for their significance. In this and the subsequent examples, we use the standard notation for \( y \sim x \). Unfortunately, the \( \sim \) symbol renders poorly and for those who want to directly copy from the pdf, the \( \sim \) symbol should be written in by hand.

R code

```r
describe(Tal_Or) #the descriptive statistics for the data.
t.test(reaction ~ cond, data=Tal_Or) # The t.test of interest
t.test(pmi ~ cond, data=Tal_Or) # Also test the effects on pmi
t.test(import ~ cond, data=Tal_Or) #and import
cor(Tal_Or) #the core-R command displays to 9 decimals
#or just show the lower diagonal of the correlations,
lowerCor(Tal_Or) #round the results to two decimals and abbreviate the names
corr.test(Tal_Or) # find the correlations, the raw p values and the adjusted p values
cor.ci(Tal_Or[1:4], n.iter=1000)
cor2latex(Tal_Or[1:4],stars=TRUE,adjust=“none”) #create the Table
```
STATISTICAL ANALYSIS

Produces this output:

```r
> describe(Tal_Or)  # the descriptive statistics for the data.

<table>
<thead>
<tr>
<th>vars</th>
<th>n</th>
<th>mean</th>
<th>sd</th>
<th>median</th>
<th>trimmed</th>
<th>mad</th>
<th>min</th>
<th>max</th>
<th>range</th>
<th>skew</th>
<th>kurtosis</th>
<th>se</th>
</tr>
</thead>
<tbody>
<tr>
<td>cond</td>
<td>123</td>
<td>0.47</td>
<td>0.50</td>
<td>0.00</td>
<td>0.46</td>
<td>0.00</td>
<td>1</td>
<td>1</td>
<td>0.11</td>
<td>-2.00</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>pmi</td>
<td>123</td>
<td>5.60</td>
<td>1.32</td>
<td>6.00</td>
<td>5.78</td>
<td>1.48</td>
<td>1</td>
<td>7</td>
<td>-1.17</td>
<td>1.30</td>
<td>0.12</td>
<td></td>
</tr>
<tr>
<td>import</td>
<td>123</td>
<td>4.20</td>
<td>1.74</td>
<td>4.00</td>
<td>4.26</td>
<td>1.48</td>
<td>1</td>
<td>7</td>
<td>-0.26</td>
<td>-0.89</td>
<td>0.16</td>
<td></td>
</tr>
<tr>
<td>reaction</td>
<td>123</td>
<td>3.48</td>
<td>1.55</td>
<td>3.25</td>
<td>3.44</td>
<td>1.85</td>
<td>1</td>
<td>7</td>
<td>0.21</td>
<td>-0.90</td>
<td>0.14</td>
<td></td>
</tr>
<tr>
<td>gender</td>
<td>123</td>
<td>1.65</td>
<td>0.48</td>
<td>2.00</td>
<td>1.69</td>
<td>0.00</td>
<td>1</td>
<td>2</td>
<td>0.62</td>
<td>-1.62</td>
<td>0.04</td>
<td></td>
</tr>
<tr>
<td>age</td>
<td>123</td>
<td>24.63</td>
<td>5.80</td>
<td>20.00</td>
<td>23.76</td>
<td>1.48</td>
<td>18</td>
<td>61</td>
<td>43</td>
<td>4.71</td>
<td>0.52</td>
<td></td>
</tr>
</tbody>
</table>

> t.test(reaction ~ cond, data=Tal_Or)  # The t.test of interest

Welch Two Sample t-test
data: reaction by cond
t = -1.7964, df = 120.98, p-value = 0.07492alternative hypothesis: true difference in means is not equal to 095 percent confidence interval:
-1.04196792 0.05058861sample estimates:
mean in group 0 mean in group 1
3.250000 3.745690

> cor(Tal_Or)  # the core-r command

<table>
<thead>
<tr>
<th>cond</th>
<th>pmi</th>
<th>import</th>
<th>reaction</th>
<th>gender</th>
<th>age</th>
</tr>
</thead>
<tbody>
<tr>
<td>cond</td>
<td>1.00</td>
<td>0.18</td>
<td>0.18</td>
<td>-0.13</td>
<td>0.03</td>
</tr>
<tr>
<td>pmi</td>
<td>0.18</td>
<td>1.00</td>
<td>0.28</td>
<td>0.16</td>
<td>0.45</td>
</tr>
<tr>
<td>import</td>
<td>0.18</td>
<td>0.28</td>
<td>1.00</td>
<td>0.46</td>
<td>0.07</td>
</tr>
<tr>
<td>reaction</td>
<td>0.18</td>
<td>0.45</td>
<td>0.46</td>
<td>1.00</td>
<td>0.01</td>
</tr>
<tr>
<td>gender</td>
<td>-0.13</td>
<td>-0.02</td>
<td>0.03</td>
<td>0.01</td>
<td>1.00</td>
</tr>
<tr>
<td>age</td>
<td>0.03</td>
<td>0.00</td>
<td>0.07</td>
<td>-0.08</td>
<td>-0.32</td>
</tr>
</tbody>
</table>

lowerCor(Tal_Or)  # round the results to two decimals and abbreviate the names

> corr.test(Tal_Or)  # find the correlations, the raw p values and the adjusted p values
Call:corr.test(x = Tal_Or)
Correlation matrix

<table>
<thead>
<tr>
<th>cond</th>
<th>pmi</th>
<th>import</th>
<th>reaction</th>
<th>gender</th>
<th>age</th>
</tr>
</thead>
<tbody>
<tr>
<td>cond</td>
<td>1.00</td>
<td>0.18</td>
<td>0.18</td>
<td>-0.13</td>
<td>0.03</td>
</tr>
<tr>
<td>pmi</td>
<td>0.18</td>
<td>1.00</td>
<td>0.28</td>
<td>-0.02</td>
<td>0.00</td>
</tr>
<tr>
<td>import</td>
<td>0.18</td>
<td>0.28</td>
<td>1.00</td>
<td>0.46</td>
<td>0.07</td>
</tr>
<tr>
<td>reaction</td>
<td>0.18</td>
<td>0.45</td>
<td>0.46</td>
<td>1.00</td>
<td>0.01</td>
</tr>
<tr>
<td>gender</td>
<td>-0.13</td>
<td>-0.02</td>
<td>0.03</td>
<td>0.01</td>
<td>1.00</td>
</tr>
<tr>
<td>age</td>
<td>0.03</td>
<td>0.00</td>
<td>0.07</td>
<td>-0.08</td>
<td>-0.32</td>
</tr>
</tbody>
</table>

Sample Size
[1] 123
Probability values (Entries above the diagonal are adjusted for multiple tests.)
To see confidence intervals of the correlations, print with the short=FALSE option

> cor.ci(Tal_Or[1:4], n.iter=1000)

Coefficients and bootstrapped confidence intervals

<table>
<thead>
<tr>
<th></th>
<th>cond</th>
<th>pmi</th>
<th>import</th>
<th>reaction</th>
<th>gender</th>
<th>age</th>
</tr>
</thead>
<tbody>
<tr>
<td>cond</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pmi</td>
<td>0.18</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>import</td>
<td>0.18</td>
<td>0.28</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>reaction</td>
<td>0.16</td>
<td>0.45</td>
<td>0.46</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

scale correlations and bootstrapped confidence intervals

<table>
<thead>
<tr>
<th></th>
<th>lower.emp</th>
<th>lower.norm</th>
<th>estimate</th>
<th>upper.norm</th>
<th>upper.emp</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>cond-pmi</td>
<td>0.02</td>
<td>0.02</td>
<td>0.18</td>
<td>0.35</td>
<td>0.34</td>
<td>0.03</td>
</tr>
<tr>
<td>cond-import</td>
<td>0.00</td>
<td>0.00</td>
<td>0.18</td>
<td>0.35</td>
<td>0.34</td>
<td>0.05</td>
</tr>
<tr>
<td>cond-react</td>
<td>-0.02</td>
<td>-0.02</td>
<td>0.16</td>
<td>0.33</td>
<td>0.32</td>
<td>0.09</td>
</tr>
<tr>
<td>pmi-import</td>
<td>0.10</td>
<td>0.11</td>
<td>0.28</td>
<td>0.44</td>
<td>0.43</td>
<td>0.00</td>
</tr>
<tr>
<td>pmi-react</td>
<td>0.30</td>
<td>0.30</td>
<td>0.45</td>
<td>0.57</td>
<td>0.58</td>
<td>0.00</td>
</tr>
<tr>
<td>import-react</td>
<td>0.32</td>
<td>0.31</td>
<td>0.46</td>
<td>0.60</td>
<td>0.59</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Mediation and Moderation

Mediation is just a different way of thinking of regression. It can be done using the `mediate` function. The first example just shows the regression analysis and draws the figure. The second example adds pmi and import as mediators. Compare the two outputs. See Figure 3.

```R
reg <- mediate(reaction ~ pmi + cond + import, data=Tal_Or)
moderate.diagram(reg, main="Regression")
reg
med <- mediate(reaction ~ cond + (pmi) + (import), data=Tal_Or)
print(med, short=FALSE)
```

> reg <- mediate(reaction ~ pmi + cond + import, data=Tal_Or)
> moderate.diagram(reg, main="Regression")
> reg

Mediation/Moderation Analysis
Call: mediate(y = reaction ~ pmi + cond + import, data = Tal_Or)

The DV (Y) was reaction. The IV (X) was pmi cond import. The mediating variable(s) = .

DV = reaction

slope se t p
pmi  0.40  0.09  4.26  4.0e-05
cond  0.10  0.24  0.43  6.7e-01
import  0.32  0.07  4.59  1.1e-05

With R² = 0.33
R = 0.57 R² = 0.33 F = 19.11 on 3 and 119 DF p-value: 3.5e-10

> med <- mediate(reaction ~ cond + (pmi)+ (import), data=Tal_Or)
> print(med, short=FALSE)

Mediation/Moderation Analysis
Call: mediate(y = reaction ~ cond + (pmi) + (import), data = Tal_Or)

The DV (Y) was reaction. The IV (X) was cond. The mediating variable(s) = pmi import.

Total effect (c) of cond on reaction = 0.5 S.E. = 0.28 t = 1.79 df= 119 with p = 0.077

Direct effect (c') of cond on reaction removing pmi import = 0.1
S.E. = 0.24 t = 0.43 df= 119 with p = 0.67

Indirect effect (ab) of cond on reaction through pmi import = 0.39
Mean bootstrapped indirect effect = 0.4 with standard error = 0.17
Lower CI = 0.09 Upper CI = 0.73
R = 0.57 R² = 0.33 F = 19.11 on 3 and 119 DF p-value: 3.5e-10

Full output

Total effect estimates (c)

reaction se t df Prob
cond  0.5 0.28 1.79 119 0.0766

Direct effect estimates (c')

reaction se t df Prob
cond  0.10 0.24 0.43 119 6.66e-01
pmi  0.40 0.09 4.26 119 4.04e-05
import  0.32 0.07 4.59 119 1.13e-05

'a' effect estimates

cond se t df Prob
pmi  0.48 0.24 2.02 121 0.0454
import  0.63 0.31 2.02 121 0.0452

'b' effect estimates

reaction se t df Prob
pmi  0.40 0.09 4.26 119 4.04e-05
import  0.32 0.07 4.59 119 1.13e-05

'ab' effect estimates

reaction boot sd lower upper
cond  0.39 0.4 0.17 0.09 0.73

'ab' effects estimates for each mediator
pmi boot sd lower upper
To show moderation, we use the Garcia et al. (2010) data set. We use the `scale` and `lm` functions from Core-R to do the regressions. We compare the mean centered versus non-mean centered results. Then we use `setCor` to combine these two steps. We include a demonstration of how to create the interaction plot of Figure 3.

```R
# First do the regular linear model
mod1 <- lm(respappr ~ prot2 * sexism, data=Garcia)  # do not mean center
centered <- scale(Garcia, scale=FALSE)  # mean center, do not standardize
centered.df <- data.frame(centered)  # convert to a data frame
mod.centered <- lm(respappr ~ prot2 * sexism, data=centered.df)
summary(mod1)  # the uncentered model
summary(mod.centered)  # the centered model

par(mfrow=c(1,2))
# compare two models (bootstrapping n.iter set to 5000 by default
# 1) mean center the variables prior to taking product terms
mod <- setCor(respappr ~ prot2 * sexism, data=Garcia,
               main="A: Moderated regression (std. and mean centered)"
)
mod
# demonstrate interaction plots
plot(respappr ~ sexism, pch = 23-protest, bg = c("black","red", "blue") [protest],
data=Garcia, main = "B: Response to sexism varies as type of protest")
by(Garcia,Garcia$protest, function(x) abline(lm(respappr ~ sexism,
data =x),lty=c("solid","dashed","dotted") [x$protest+1]))
text(6.5,3.5,"No protest")
text(3.1,3.9,"Individual")
text(3.1,5.2,"Collective")

> summary(mod1)  # the uncentered model
Call:
  lm(formula = respappr ~ prot2 * sexism, data = Garcia)

Residuals:
     Min      1Q  Median      3Q     Max
-3.4984 -0.7540  0.0801  0.8301  3.1853

Coefficients:
                     Estimate Std. Error t value     Pr(>|t|)
(Intercept)  6.5667     1.2095   5.429  2.83e-07 ***
prot2       -2.6866     1.4515  -1.851    0.06654 .
sexism      -0.5290     0.2359  -2.243   0.02668 *
prot2:sexism  0.8100     0.2819   2.873   0.00478 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.144 on 125 degrees of freedom
Multiple R-squared:  0.2962,   Adjusted R-squared:  0.2793
```
F-statistic: 17.53 on 3 and 125 DF, p-value: 1.456e-09

> summary(mod.centered) #the centered model

Call:
  lm(formula = respappr ~ prot2 * sexism, data = centered.df)

Residuals:
  Min       1Q   Median       3Q      Max
-3.4984 -0.7540  0.0801  0.8301  3.1853

Coefficients:
                         Estimate Std. Error t value    Pr(>|t|)
(Intercept)               -0.01184   0.10085  -0.117   0.90671
prot2                     1.45803   0.21670   6.728 5.52e-10 ***
sexism                    0.02354   0.12927   0.182   0.85579
prot2:sexism             0.80998   0.28191   2.873  0.00478 **

---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 1.144 on 125 degrees of freedom
Multiple R-squared: 0.2962, Adjusted R-squared: 0.2793
F-statistic: 17.53 on 3 and 125 DF, p-value: 1.456e-09

> #compare two models (bootstrapping n.iter set to 5000 by defalt
> # 1) mean center the variables prior to taking product terms
> mod <- setCor(respappr ~ prot2 * sexism, data=Garcia,
>    + main="A: Moderated regression (std. and mean centered)")
> mod

Multiple Regression from raw data
DV = respappr
  slope     se      t      p   VIF
prot2       0.51    0.08    6.73 5.5e-10 1
sexism      0.01    0.08    0.18  8.6e-01 1
prot2*sexism 0.22    0.08    2.87 4.8e-03 1

Multiple Regression
R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2   p
respappr 0.54  0.3  0.42  0.18  0.28    0.06  17.53   3 125  1.46e-09
> #demonstrate interaction plots
> plot(respappr ~ sexism, pch = 23- protest, bg = c("black","red", "blue") [protest],
+ data=Garcia, main = "B: Response to sexism varies as type of protest")
> by(Garcia,Garcia$protest, function(x) abline(lm(respappr ~ sexism,
+      data =x),lty=c("solid","dashed","dotted") [x$protest+1]))
Garcia$protest: 0
NULL
Garcia$protest: 1
NULL
Garcia$protest: 2
NULL
> text(6.5,3.5,"No protest")
Decision theory and Area under the curve

Table 6 and Figure 4 are examples of signal detection theory. This is done by giving the four cells to the \texttt{AUC} function.

\begin{verbatim}
AUC(c(49,40,79,336))
\end{verbatim}

\textbf{Decision Theory and Area under the Curve}

The original data implied the following 2 x 2 table

\begin{tabular}{|c|c|}
\hline
Predicted.Pos & Predicted.Neg \\
\hline
True.Pos & 0.097 & 0.079 \\
True.Neg & 0.157 & 0.667 \\
\hline
\end{tabular}

Conditional probabilities of

\begin{tabular}{|c|c|}
\hline
Predicted.Pos & Predicted.Neg \\
\hline
True.Pos & 0.55 & 0.45 \\
True.Neg & 0.19 & 0.81 \\
\hline
\end{tabular}

Accuracy = 0.76  Sensitivity = 0.55  Specificity = 0.81
with Area Under the Curve = 0.76
d.prime = 1  Criterion = 0.88  Beta = 0.15
Observed Phi correlation = 0.32
Inferred latent (tetrachoric) correlation = 0.53

\textbf{EFA}

The factor analysis of the \texttt{Thurstone} data set was done using the \texttt{fa} function. We specify that the number of subjects was 213. By default, we find a \texttt{minres} solution and use the \texttt{oblminin} rotation. We also show how to specify other factor extraction techniques, and other rotations. We just show the first solution.

\begin{verbatim}
fa(Thurstone,nfactors=3,n.obs=213)
fa(Thurstone,nfactors=3,n.obs=213,fm="mle") #use the maximum likelihood algorithm
fa(Thurstone,nfactors=3,n.obs=213, rotate="Varimax") #use an orthogonal rotation.
\end{verbatim}

\begin{verbatim}
> fa(Thurstone,nfactors=3,n.obs=213)
Factor Analysis using method = minres
Call: fa(x = Thurstone, nfactors = 3, n.obs = 213)
Standardized loadings (pattern matrix) based upon correlation matrix
   MR1 MR2 MR3 h2  u2 com
Sentences  0.90 -0.03 0.04 0.82 0.18 1.0
Vocabulary  0.89 0.06 -0.03 0.84 0.16 1.0
\end{verbatim}
STATISTICAL ANALYSIS

Here we find the reliability of the msqR items found in the first example. We select just the time 1 data. We show several different approaches. Because we have just 8 items and they represent two subfactors, we find \( \omega_h \) using a two factor solution.

```r
msq.items <- c("anxious", "at.ease", "calm", "confident", "content", "jittery", "nervous", "relaxed", "tense", "upset") #these overlap with the sai
msq1 <- subset(msqR, msqR$time==1)
alpha(msq1[msq.items], check.keys=TRUE)
```
omega(msq1[msq.items], nfactors=2)

alpha(msq1[msq.items], check.keys=TRUE)

Reliability analysis
Call: alpha(x = msq1[msq.items], check.keys = TRUE)

raw_alpha std.alpha G6(smc) average_r S/N ase mean sd median_r
0.83 0.83 0.86 0.33 5 0.0046 2 0.54 0.32

lower alpha upper 95% confidence boundaries
0.82 0.83 0.84

Reliability if an item is dropped:

raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
anxious- 0.83 0.83 0.85 0.34 4.7 0.0047 0.026 0.34
at.ease 0.80 0.80 0.83 0.31 4.1 0.0055 0.028 0.32
confident 0.83 0.81 0.84 0.32 4.2 0.0054 0.030 0.32
content 0.82 0.82 0.84 0.34 4.6 0.0049 0.025 0.32
jittery- 0.83 0.83 0.85 0.35 4.8 0.0047 0.027 0.33
nervous- 0.82 0.82 0.84 0.33 4.4 0.0049 0.030 0.32
relaxed 0.80 0.81 0.84 0.31 4.1 0.0055 0.030 0.31
tense- 0.81 0.81 0.83 0.32 4.2 0.0051 0.029 0.32
upset- 0.82 0.82 0.85 0.34 4.7 0.0049 0.033 0.35

Item statistics
n raw.r std.r r.cor r.drop mean sd
anxious- 1871 0.54 0.56 0.51 0.42 2.3 0.86
at.ease 3018 0.77 0.74 0.72 0.67 1.6 0.94
calm 3020 0.74 0.71 0.68 0.63 1.6 0.92
confident 3021 0.54 0.50 0.43 0.38 1.5 0.93
content 3010 0.64 0.59 0.55 0.50 1.4 0.92
jittery- 3026 0.52 0.55 0.48 0.41 2.3 0.83
nervous- 3017 0.59 0.64 0.60 0.52 2.6 0.68
relaxed 3023 0.76 0.73 0.70 0.66 1.6 0.91
tense- 3017 0.67 0.71 0.69 0.60 2.4 0.78
upset- 3019 0.54 0.58 0.50 0.45 2.6 0.68

Non missing response frequency for each item
1 2 3 miss
anxious 0.53 0.29 0.13 0.04 0.38
at.ease 0.14 0.33 0.35 0.18 0.00
calm 0.14 0.34 0.36 0.17 0.00
confident 0.16 0.33 0.37 0.14 0.00
content 0.17 0.35 0.35 0.13 0.01
jittery 0.54 0.31 0.12 0.04 0.00
nervous 0.70 0.22 0.06 0.02 0.00
relaxed 0.12 0.30 0.40 0.18 0.00
tense 0.59 0.28 0.10 0.03 0.00
upset 0.74 0.18 0.05 0.02 0.00

Warning message:
In alpha(msq1[msq.items], check.keys = TRUE) :
Some items were negatively correlated with total scale and were automatically reversed.
This is indicated by a negative sign for the variable name.
> omega(msq1[msq.items], nfactors=2)
Three factors are required for identification -- general factor loadings set to be equal. Proceed with caution.
Think about redoing the analysis with alternative values of the 'option' setting.

Omega
Call: omega(m = msq[msq.items], nfactors = 2)
Alpha: 0.83
G.6: 0.86
Omega Hierarchical: 0.45
Omega H asymptotic: 0.51
Omega Total 0.87

Schmid Leiman Factor loadings greater than 0.2

<table>
<thead>
<tr>
<th></th>
<th>g</th>
<th>F1*</th>
<th>F2*</th>
<th>h2</th>
<th>u2</th>
<th>p2</th>
</tr>
</thead>
<tbody>
<tr>
<td>anxious</td>
<td>0.36</td>
<td>-0.57</td>
<td>0.46</td>
<td>0.54</td>
<td>0.28</td>
<td></td>
</tr>
<tr>
<td>at.ease</td>
<td>0.52</td>
<td>0.59</td>
<td>0.64</td>
<td>0.36</td>
<td>0.43</td>
<td></td>
</tr>
<tr>
<td>calm</td>
<td>0.49</td>
<td>0.47</td>
<td>-0.21</td>
<td>0.51</td>
<td>0.49</td>
<td>0.48</td>
</tr>
<tr>
<td>confident</td>
<td>0.31</td>
<td>0.58</td>
<td>0.46</td>
<td>0.54</td>
<td>0.21</td>
<td></td>
</tr>
<tr>
<td>content</td>
<td>0.40</td>
<td>0.65</td>
<td>0.59</td>
<td>0.41</td>
<td>0.26</td>
<td></td>
</tr>
<tr>
<td>jittery</td>
<td>0.35</td>
<td>-0.52</td>
<td>0.40</td>
<td>0.60</td>
<td>0.31</td>
<td></td>
</tr>
<tr>
<td>nervous</td>
<td>0.43</td>
<td>-0.57</td>
<td>0.51</td>
<td>0.49</td>
<td>0.36</td>
<td></td>
</tr>
<tr>
<td>relaxed</td>
<td>0.51</td>
<td>0.48</td>
<td>-0.22</td>
<td>0.53</td>
<td>0.47</td>
<td>0.48</td>
</tr>
<tr>
<td>tense</td>
<td>0.50</td>
<td>-0.62</td>
<td>0.63</td>
<td>0.37</td>
<td>0.39</td>
<td></td>
</tr>
<tr>
<td>upset</td>
<td>0.35</td>
<td>-0.29</td>
<td>0.25</td>
<td>0.75</td>
<td>0.50</td>
<td></td>
</tr>
</tbody>
</table>

With eigenvalues of:

<table>
<thead>
<tr>
<th>g</th>
<th>F1*</th>
<th>F2*</th>
<th>h2</th>
<th>u2</th>
<th>p2</th>
</tr>
</thead>
<tbody>
<tr>
<td>general/max</td>
<td>1.13</td>
<td>max/min</td>
<td>1.05</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
mean percent general = 0.37 with sd = 0.1 and cv of 0.28

Explained Common Variance of the general factor = 0.37

The degrees of freedom are 26 and the fit is 0.24
The number of observations was 3032 with Chi Square = 721.36 with prob < 2.4e-135
The root mean square of the residuals is 0.04
The df corrected root mean square of the residuals is 0.05
RMSEA index = 0.094 and the 10 % confidence intervals are 0.088 0.1
BIC = 512.92

Compare this with the adequacy of just a general factor and no group factors
The degrees of freedom for just the general factor are 35 and the fit is 1.67
The number of observations was 3032 with Chi Square = 5055.64 with prob < 0
The root mean square of the residuals is 0.21
The df corrected root mean square of the residuals is 0.24
RMSEA index = 0.218 and the 10 % confidence intervals are 0.213 0.223
BIC = 4775.04

Measures of factor score adequacy

<table>
<thead>
<tr>
<th>g</th>
<th>F1*</th>
<th>F2*</th>
<th>h2</th>
<th>u2</th>
<th>p2</th>
</tr>
</thead>
</table>
Correlation of scores with factors | 0.67 | 0.77 | 0.76|
Multiple R square of scores with factors | 0.45 | 0.60 | 0.59|
Minimum correlation of factor score estimates | -0.09 | 0.19 | 0.17|

Total, General and Subset omega for each subset

<table>
<thead>
<tr>
<th>g</th>
<th>F1*</th>
<th>F2*</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Value 1</td>
<td>Value 2</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>Omega total for total scores and subscales</td>
<td>0.87</td>
<td>0.84</td>
</tr>
<tr>
<td>Omega general for total scores and subscales</td>
<td>0.45</td>
<td>0.33</td>
</tr>
<tr>
<td>Omega group for total scores and subscales</td>
<td>0.36</td>
<td>0.51</td>
</tr>
</tbody>
</table>