unidim: An index of scale homogeneity and unidimensionality

William Revelle[1] and David Condon[2]
Northwestern University University of Oregon

How to evaluate how well a psychological scale measures just one construct is a recurring problem in assessment. We introduce an index, u, of the unidimensionality and homogeneity of a scale. u is just the product of two other indices: τ (a measure of τ equivalence) and ρ_c (a measure of congeneric fit). By combining these two indices into one, we provide a simple index of the unidimensionality and homogeneity of a scale. We evaluate u through simulations and with real data sets. Simulations of u across one factor scales ranging from 3 to 24 items with various levels of factor homogeneity show that τ and therefore u are sensitive to the degree of factor homogeneity. Additional tests with multifactorial scales representing 9, 18, 27, 36 items with a hierarchical factor structure varying in a general factor loading show that ρ_c and therefore u are sensitive to the general factor saturation of a test. We also demonstrate the performance of u on 45 different publicly available personality and ability measures. Comparisons with traditional measures (i.e., $\omega_h, \alpha, \omega_t$, CFI, ECV) show that u has greater sensitivity to unidimensional structure and less sensitivity to the number of items in a scale. u is easily calculated with open source statistical packages and is relatively robust to sample sizes ranging from 100 to 5,000.

Public Significance

How to evaluate how well a psychological scale measures just one construct is a recurring problem in assessment. We present an intuitively easy to understand and easily calculated new index of scale quality, u, which combines measures of scale unidimensionality and item homogeneity. Using open source software on publicly available data sets we compare this index to conventional measures. In multiple simulations we assess the utility of u. Across 12 levels of general factor saturation, four item levels, and samples sizes ranging from 100 to 5,000, we show the utility of u as an index of scale quality. We include the R code for our simulations and analysis of existing data sets.

Evaluating the dimensionality of a measure has been an ongoing challenge ever since Spearman (1904) introduced his factor model of intelligence. Today, it is widely recognized that the majority of psychological tests — even those intended to measure only one construct — produce a range of response pattern (that is, a degree of "higgledypiggledyness", Walker, 1931, p 75) that is quite large. This has prompted the development of several techniques for assessing the unidimensionality of scales. But, as pointed out by McDonald (1981) because unidimensionality holds or does not hold, the proper question is not "is a test unidimensional", but rather how well does a unidimensional model fit the data. That is, we should ask if the fit of a one factor model is "satisfactory". Hattie (1985) reviewed and then evaluated (Hattie, 1984) 87 ways to assess unidimensionality and stated that "Unidimensionality can be rigorously defined as the existence of one latent trait underlying the set of items" (Hattie, 1985, p 152). Although a test with more than 3 items can never be truly unidimensional, the question becomes how to assess how *close* a test is to unidimensionality (Ten Berge and Socan, 2004).

An important characteristic of unidimensionality — one that motivates our aim of assessing it better — is that it represents a necessary condition for the meaningful application of Item Response Theory (IRT) models. In the context of IRT, Stout (1990) introduced the concept of *essential dimensionality* which he defined as follows:.

> "Essential dimensionality, much in the spirit of counting the number of dimensions in a factor analytic model, is the number of major latent dimensions with minor dimensions ignored. Essential unidimensionality, the

We thank Richard Zinbarg for helpful comments on an earlier draft.
contact: William Revelle revelle@northwestern.edu

This a preprint version of October 4, 2024 as submitted to *Psychological Methods*. It is currently in press. The final version will differ from this due to copyediting.

> existence of exactly one major dimension, then provides a justification for carrying out IRT based statistical analyses that require unidimensionality." (Stout, 1990, 293).

While the procedure for estimating essential dimensionality suggested by Stout (1990) and implemented with his colleagues (Nandakumar and Stout, 1993; Stout et al., 2001) requires very large samples, their motivation is worth emphasizing, especially given recent increases in the use IRT in psychological assessment (Thomas, 2019). Assessing the degree of unidimensionality is (or *should* be) a necessary step in claiming that scores for any given scale reflect one underlying construct. If a multi-item scale is not adequately unidimensional, then scores reflect the underlying construct of interest as well as other, extraneous sources of variance. In these cases, the use of IRT is not justified.

However, in the context of ability and personality measurement, even essential unidimensionality seems not to be desired, as scales need to reflect the complexity of the constructs being assessed (Tellegen, 1991; Tellegen and Waller, 2008) and scales need to "unidimensional enough" (Embretson and Reise, 2025). Although resulting in single scale values, measures as complex as the Stanford-Binet or the Wechsler Intelligence Scale for Children are clearly multidimensional, but dominated by a single factor:

> A test needs to measure a well defined construct, but the construct need not be at the first order. Many useful test scores in education and psychology are at a higher order. The composite score on the ACT Test reflects student performance in English, mathematics, reading, and science. The general intelligence score of the Wechsler Intelligence Scale for Children reflects performance in verbal comprehension, working memory, perceptual reasoning, and processing speed. In the NEO-PI personality measure, each domain (for example, extraversion) is composed of multiple facets (warmth, gregariousness, assertiveness, excitement seeking, and positive emotions). (Davenport et al., 2015, p 6)

In this article, we suggest a combined index of unidimensionality and homogeneity, u, which compares favorably to some of the current alternatives. u is both easy to understand and calculate and it addresses the weaknesses of several popular alternatives (i.e., α, ω_t, CFI, or Explained Common Variance).[3] Before explaining the derivation of u however, we first discuss the history and shortcomings of these alternatives.

Extant Alternatives

To understand the challenge of evaluating unidimensionality, consider a multi-item scale designed to measure a single construct (T) with inevitable random error (E). That is, each item x_i contributes construct-specific variance but is "befuddled by error" (McNemar, 1946, p 246). In terms from classical test theory (Spearman, 1904; Lord and Novick, 1968; McDonald, 1999), this may be represented as

$$x_i = \lambda_i \tau + \epsilon_i \tag{1}$$

with variances

$$\sigma_i^2 = \lambda_i^2 \sigma_\tau^2 + \sigma_\epsilon^2 \tag{2}$$

where the λ_i reflect factor loadings of items on the construct and τ the construct "true score", and ϵ_i random error.

The question of unidimensionality thus becomes one of assessing how well Equation 1 fits the data. In the unlikely case that the λ_i are equal for all items and the variances of the ϵ_i are all equal, the items are said to be *parallel*. That is, all items contribute equally with respect to the construct measured by the scale. Similarly, subsets of the items from the scale would represent *parallel forms*. If the λ_i are equal for all items but the variances of the ϵ_i are unequal, the items are said to be τ *equivalent*. That is, each item has the same *True* score but different error variances. If the λ_i are unequal (as is the typical case for most measures used in psychology), the items are said to be *congeneric*. For parallel items, all the correlations and covariances among items will be identical. For τ equivalent items, the covariances will be identical but the correlations will not. For congeneric items, neither the covariances nor the correlations need to be identical. In all three cases, the values of x_i fit a one factor latent variable model (Lucke, 2005; McDonald, 1999).

The relationships among these values are often used to evaluate the internal consistency of scales as an estimate of reliability as with, for example, Cronbach's α (aka Guttman's λ_3) (Cronbach, 1951; Guttman, 1945). For a scale (X) that is scored as the unit weighed sum of its items ($X = \Sigma x_i$), internal consistency reliability is defined as the proportion of construct-specific variance to total observed variance. That is, with the assumptions of Equation 2

$$\rho_{xx} = \frac{\sigma_\tau^2}{\sigma_X^2}.$$

Although most frequently used as an index of internal consistency reliability, α is sometimes used — mistakenly, in our opinion — as a means of assessing unidimensionality as well. α is a particularly poor measure

[3]We include α and ω_t not because they are useful indices of uinidimensionality nor of homogeneity, but because they are popularly misused as such.

because it assumes (without testing) the case of τ equivalence (Cronbach, 1951). In other words, all of the items are presumed to have the same (true score) relationship to the measured construct. This means α is just a function of the number of items (k) and the average covariance of the items ($\bar{\sigma}_{ij}$):

$$\alpha = \frac{\sigma_X^2 - \Sigma\sigma_i^2 + k\bar{\sigma}_{ij}}{\sigma_X^2} = \frac{k\bar{\sigma}_{ij}}{(\bar{\sigma_i^2} + (k-1)\bar{\sigma}_{ij})} = \frac{k}{k-1}\frac{\sigma_X^2 - \Sigma\sigma_i^2}{\sigma_X^2} \tag{3}$$

Before the introduction of modern computers, the advantage of α was that it could be calculated from just the test variances (σ_X^2) and the sum of the item variances ($\Sigma\sigma_i^2$), and it did not require finding the covariances and examining the internal structure of the test. This simplicity of calculation and subsequent introduction into popular proprietary software packages probably accounts for the widespread use of α (Cho and Kim, 2015),

As an alternative to α, McDonald (1999) expanded equation 1 to account for the relationship of any item to (multiple) group factors in addition to the general and specific/error terms[4]. Thus, the observed score for a particular subject on an item is the sum of the products of factor scores ($\boldsymbol{g}, \boldsymbol{f}, \boldsymbol{s}, \boldsymbol{e}$) and loadings ($\boldsymbol{c}, \boldsymbol{A}, \boldsymbol{D}$) on these factors:

$$\boldsymbol{x} = \boldsymbol{c}\boldsymbol{g} + \boldsymbol{A}\boldsymbol{f} + \boldsymbol{D}\boldsymbol{s} + \boldsymbol{e}. \tag{4}$$

Assuming standardized general (g) and group factor (f) variances, then the total variance of a test may be decomposed into that due to a general factor, to the sum of the group factors, to specific (D) and error factors and will be

$$\sigma_t^2 = \mathbf{1'cc'1} + \mathbf{1'AA'1} + \mathbf{1'DD'1} + \mathbf{1'}\boldsymbol{\sigma_\epsilon^2}1. \tag{5}$$

Taking advantage of this extension of equation 1 to that of 4 McDonald (1999) introduced two coefficients, both called ω to determine the amount of reliable variance in a test. Using notation suggested by Zinbarg et al. (2005) the first coefficient, ω_t, reflects the total amount of common variance among the items while the second. ω_h, reflects the amount due to the general factor

$$\omega_t = \frac{\mathbf{1'cc'1} + \mathbf{1'AA'1}}{\sigma_X^2}. \tag{6}$$

In that the communality of an item (h_i^2) is the sum of the squared factor loadings for that item, ω_t may be found by replacing the item variances (σ_i^2) with the amount of common variance (h_i^2) for each item:

$$\omega_t = \frac{\sigma_X^2 - \Sigma\sigma_i^2 + \Sigma h_i^2}{\sigma_X^2}. \tag{7}$$

ω_h is found by summing the loadings on the general factor and comparing the square of their sum to the test variance:

$$\omega_h = \frac{(\Sigma c_i)^2}{\sigma_X^2} = \frac{\mathbf{1'cc'1}}{\sigma_X^2}. \tag{8}$$

Although ω_t may be found directly from the communalities of a factor analysis, ω_h requires either a hierarchical factoring of the original data followed by a subsequent Schmid-Leiman Transformation (Schmid and Leiman, 1957) or by a bifactor solution (Holzinger and Swineford, 1937; Reise, 2012) with a general factor and a number of group factors.

As ω_h represents the proportion of general factor variance to total test variance, it directly address the question of how well a one factor model fits the data. In addition, ω_h avoids the difficulty introduced by using ω_t (the proportion of all common variance to total test variance) as an estimate of unidimensionality. In cases where one or more subset(s) of items share group variance that is not fully explained by variance on the general factor among all items, ω_h will be a smaller proportion than ω_t and a much better estimate of unidimensionality. In addition, because the relative contribution of the variance of a single item to the total variance decreases as the number of items increases, the asymptotic value of ω_h is

$$\omega_{h_\infty} = \frac{\mathbf{1'cc'1}}{\mathbf{1'cc'1} + \mathbf{1'AA'1}} = \frac{\omega_h}{\omega_t}. \tag{9}$$

Importantly, ω_t reflects the total amount of common variance among the items rather than the amount due to any single factor or dimension, and this limits its usefulness as a measure of unidimensionality. The difficulty is that ω_t does not indicate the extent to which items co-vary on underlying dimensions beyond the primary or "general" factor.

In fact, α (Equation 3) and ω_t (Equation 7) are similar in that both are the ratio of a reduced variance $\sigma_X^2 - \delta$ to the total variance σ_X^2 (where $\delta = \Sigma\sigma_i^2 - k\bar{\sigma}_{ij}$ for α or $\delta = \Sigma\sigma_i^2 - \Sigma h_i^2$ for ω_t). Thus, in both cases, the numerator increases linearly by the number of items, but the denominator by the square of the number of items. Importantly, both coefficients will increase asymptotically to 1 as the number of items increases.

Although Revelle and Condon (2019) have previously recommended reporting ω_h, α, and ω_t for all scales in order to estimate reliability and ω_h for the amount of first factor saturation, unfortunately ω_h is not appropriate for very short scales. This is because ω_h requires at least 2 (and preferably $\geq$ 3) factors in order to find a *hierarchical* solution or even a bifactor structure. Since the degrees of freedom for a factor model with k factors and n variables is:

[4] Specific and error are confounded unless using repeated measures to assess specific variance.

$$\frac{n(n-1)}{2} - nk + \frac{k(k-1)}{2}$$

the minimum number of variables needed for a 2 factor model to be defined is 5. Note that this case does not allow for a proper hierarchical solution using a Schmid-Leiman transformation for ω_h, as this requires 3 lower level factors (Zinbarg et al., 2006). The minimum number of variables needed to properly estimate a hierarchical solution is 6. Thus, in the case of $n \leq 4$ $\omega_h = \omega_t$.

Another index of the percent of general factor variance is *Explained Common Variance* (ECV) which compares the amount of variance extracted by the first factor to the amount explained by all factors (Rodriguez et al., 2016; Sijtsma, 2009; Ten Berge and Socan, 2004). ECV as discussed by Ten Berge and Socan (2004) compares the size of the eigenvalues extracted using *minimium rank factor analysis* (Shapiro and ten Berge, 2002; Ten Berge and Kiers, 1991). Sijtsma (2009) compares ECV and α for multiple examples and shows where ECV does not changing while α ranges from .2 to .9, as well as examples of α not varying and ECV varying from .33 to 1.0. It is important to recognize that ECV just measures how large the first factor is compared to the remaining factors. ECV is not an index of *general* factor variance, ω_h will detect the absence of a general factor.

Yet one more way to test for unidimensionality is to ask how many factors best fit a scale. If the best estimate is greater than one, clearly the scale is not unidimensional. However, the number of factors problem is notoriously difficult (Horn and Engstrom, 1979) and variations on the method of parallel analysis Horn (1965) have spawned a small industry (Lorenzo-Seva et al., 2011; Revelle and Rocklin, 1979; Timmerman and Lorenzo-Seva, 2011; Velicer, 1976; Zwick and Velicer, 1986). Even if these methods suggest one factor, the results may be misleading in terms of the extent of unidimensionality. For example, a completely homogeneous scale of six items with all correlations of .16 and and ECV of 1.00 (table 5 from Sijtsma, 2009) is seen as no different than the six political items from Ten Berge and Socan (2004) with an ECV of .82. Both appear to be one factor using parallel analysis with minimum rank factor analysis, but the latter is intuitively much less unidimensional than the first.

In recent discussions of the relationship between α, internal consistency, and unidimensionality, Davenport et al. (2015); Green and Yang (2015) consider how a test measuring a single construct does not necessarily have to be unidimensional, if the construct itself is complex (e.g., the ACT composite score, or the Wechsler Intelligence Scale for Children). Similar points have been raised by Embretson and Reise (2025) and Tellegen (1991).

Here we introduce a very simple alternative index for unidimensionality — u, which may be found using the `unidim` function in the *psych* package (Revelle, 2024a) in R (R Core Team, 2024) — and examine its properties using both simulated and real data.

Unidim: an index of unidimensionality

The logic is deceptively simple: Unidimensionality implies that a one factor model (λ) of the data fits the covariances or the correlations of the data. If this is the case, then the factor model implies that $R = \lambda\lambda' + U^2$ will have residuals of 0[5]. That is, that the observed correlations will equal those of the factor model. To evaluate this, find the residuals of the observed covariances minus the modeled covariances, sum the off-diagonal elements of these squared residuals ($F_m = \sum_{i \neq j}(R_{ij} - \lambda_i\lambda'_j)^2$), and compare this to the sum of off-diagonal elements of the squared original correlations ($F_o = \sum_{i \neq j} R_{ij}^2$). This is the measure of congeneric fit:

$$\rho_c = 1 - \frac{F_m}{F_o} = \frac{F_o - F_m}{F_o}.$$

When fitting a one factor model, ρ_c is a direct measure of the fit of a congeneric model to the observed correlations/covariances.

A statistically more elegant estimate is that of the Compararative Fit Index (Bentler, 1990), which compares a fit statistic (i.e., χ^2) of the model less the expected value ($F_m = \chi^2_m - df_m$) to that of the original data ($F_o = \chi^2_o - df_o$). Constraining both of the fits to be positive leads to the CFI:

$$CFI = \frac{max(\chi^2_0 - df_0, 0) - max(\chi^2_m - df_m, 0)}{max(\chi^2_0 - df_0, 0)}. \quad (10)$$

Clearly ρ_c is a direct index of the fit of a model, whereas the CFI is a comparison of the fit statistics. Although very similar in their values for data that are almost unidimensional we prefer ρ_c for its performance across less unidimensional data sets and the lack of a need to specify sample size. In the following tables and figures, we compare these two indices as well as a larger set of fit statistics.

Although both the CFI and ρ_c can be large when some of the loadings are very small or differ drastically in their magnitude, it is probably not a good idea to think of the items as forming a useful scale. Thus, an alternative measure (the τ statistic) compares the observed correlations (r_{ij}) to the mean correlation ($\bar{r_{ij}}$) and considers 1 - the ratio of the sum of the squared residuals to the sum of the squared correlations:

$$\tau = 1 - \frac{\sum_{i \neq j}(r_{ij} - \bar{r_{ij}})^2}{\sum_{i \neq j} r_{ij}^2} \quad (11)$$

[5]Although we show this for correlation matrices this also works for the more general covariance matrix.

τ will achieve a maximum if the item correlations are all identical (a parallel test model, McDonald, 1999). Indeed, for dichotomous data, given the equivalence of factor analytic models using tetrachoric correlations and weighted least squares, with 2PL IRT models (Kamata and Bauer, 2008; Takane and de Leeuw, 1987), the Rasch (1966) model — a one parameter IRT model with the assumption that all items have equally good discrimination — is functionally a τ equivalent model.

We define the product of ρ_c and τ as our measure of unidimensionality, u.[6] That is, congeneric fit x tau equivalent fit as a measure of unidimensionality. In the following tables, we show how u behaves in various simulations as well as with real data. We also show the behavior of the u statistic as a function of sample size and compare it with CFI and ECV as a function of sample size (Figure 1). These demonstrations use the functions `unidim` and `omega` as implemented in the *psych* package (Revelle, 2024a) for the open source statistical system R (R Core Team, 2024). Output from both functions is also shown in the `reliability` function.

Tests with simulations

In the following two sections, we examine 16 one factor models, four two factor models as well as 48 hierarchical models with varying levels of a general factor. To show the robustness of our results, we simulate each of these models 100 times with four levels of simulated sample size.

To demonstrate the `unidim` function on unidimensional data, we simulated four one-factor models for 3, 6, 12 and 24 items. Loadings for each model were specified for a three-item model as large (.7, .6, .5), medium (.6, .5, .4), small (.5, .4, .3), or mixed (.7, .5, .3) loadings. To form 6, 12, or 24 item structure, the basic loadings were repeated 2, 4, and 8 times. We used the `sim` and `sim.minor` functions to generate the data. Both functions generate a latent variable model by multiplying the factor loading matrix by a matrix of random normal deviates and then adding normally distributed error. `sim.minor` follows the advice of MacCallum and Tucker (1991) who distinguished between the factor model we want (pure factors) and a generating model of pure factors with a number of smaller, nuisance factors. (For further examples of large and small factors, see Lorenzo-Seva et al., 2011; Timmerman and Lorenzo-Seva, 2011). In order to be able to analyze the actual raw scores, simulations were not done using the procedure of Tucker et al. (1969) which generates sample correlation matrices but rather by generating factor scores plus error following techniques similar to those of Timmerman and Lorenzo-Seva (2011).

Varying factor loadings of a one dimensional model

For ease of simulation, we formed data based upon the first four columns (pure) or all eight columns (minor) of Table 2. By partitioning the resulting correlation matrix appropriately, we are thus able to generate 16 different one-factor models. We show loadings for each of the models. The first set (column 1) contain large loadings (.7, .6, .5), the second medium (.6, .5, .4) loadings, and the third small (.5, .4, .3) loadings. The fourth column shows mixed loadings of .7, .5, and .3. The final four columns were used when generating minor factors, with loadings of ±.2 randomly assigned to variables. The partitioning of the overall 96 x 96 correlation matrix resulted in e.g., a 3 x 3 correlation matrix with large loadings (R[1:3,1:3], medium loadings (R[4:6;4:6], or a 6 x 6 with mixed loadings (R[c(10:12,22:24), c(10:12,22:24)]), etc.

Table 1

Four simulated loadings matrices. For each model, one of the first 4 columns was combined with columns 5-8. The loadings represent large, medium and small loadings, as well as a mixed set. The minor factors had loadings of ±.2 for four nuisance factors. To simulate an (e.g.) six item problem with medium loadings, the first six rows of the second and 5-8th columns were used.

Item	large	middle	small	mixed	m1	m2	m3	m4
1	0.7	0.6	0.5	0.7	0.0	0.0	0.2	-0.2
2	0.6	0.5	0.4	0.5	0.0	0.2	0.0	0.0
3	0.5	0.4	0.3	0.3	-0.2	0.0	0.0	0.2
4	0.7	0.6	0.5	0.7	0.2	0.2	0.2	0.2
5	0.6	0.5	0.4	0.5	0.0	-0.2	0.0	0.0
6	0.5	0.4	0.3	0.3	0.0	0.2	0.0	0.2
	...	...	...	..				
22	0.7	0.6	0.5	0.7	-0.2	0.0	-0.2	0.2
23	0.6	0.5	0.4	0.5	0.0	0.0	0.2	-0.2
24	0.5	0.4	0.3	0.3	-0.2	0.0	0.0	0.0

Data were generated for 500 simulated subjects using both the "pure" (just one of the first four columns) and the "noisy" (one + four minor factors) model. Solutions for 3, 6, 12, and 24 items per scale for 500 simulated participants are shown for pure factors in Table 3 and for noisy data in Table 4. The last four rows reflect scales formed from the first and second factors for 6, 10, 12, and 24 items[7]. These are clearly not unidimensional. Simulations were done with continuous data as well as six alternative categorical data (simulating items– Table 5). Several things to note in these tables: Following the Spearman-Brown equation (Brown, 1910; Spear-

[6] Thus, the square root of u is just the geometric mean of τ and ρ_c.

[7] Because we also are finding split half estimates, we limited our examples to 24 items to allow for finding all split half values from the 2,704,156 possible splits.

Table 2

The simulated loadings matrix. Rows 1-12 were repeated 8 times to generate the 96 item loadings. The loadings represent large, medium and small loadings, as well as a mixed set. The minor factors had loadings of ±.2 for four nuisance factors. The resulting 96 96 correlation matrix is then partitioned into unifactorial subsets. (E.g., R[1:3,1:3] represents the correlation matrix of 3 items with large correlations, R[4:6,4:6] the medium sized correlation matrix.*

Variable	Large	Medium	Small	Mixed	m1	m2	m3	m4
V1	0.7	0.0	0.0	0.0	0.0	0.0	0.2	-0.2
V2	0.6	0.0	0.0	0.0	0.0	0.2	0.0	0.0
V3	0.5	0.0	0.0	0.0	-0.2	0.0	0.0	0.2
V4	0.0	0.6	0.0	0.0	0.2	0.2	0.2	0.2
V5	0.0	0.5	0.0	0.0	0.0	-0.2	0.0	0.0
V6	0.0	0.4	0.0	0.0	0.0	0.2	0.0	0.2
V7	0.0	0.0	0.5	0.0	0.2	0.2	0.0	0.0
V8	0.0	0.0	0.4	0.0	-0.2	0.0	-0.2	0.2
V9	0.0	0.0	0.3	0.0	0.0	0.2	0.0	0.2
V10	0.0	0.0	0.0	0.7	-0.2	0.0	0.2	0.0
V11	0.0	0.0	0.0	0.5	0.0	0.0	0.2	-0.2
V12	0.0	0.0	0.0	0.3	-0.2	0.0	0.0	0.0
	...	...	...	...	...	...	...	...
V85	0.7	0.0	0.0	0.0	0.0	0.0	0.2	0.0
V86	0.6	0.0	0.0	0.0	0.0	0.0	0.0	-0.2
V87	0.5	0.0	0.0	0.0	0.2	0.2	0.0	0.0
V88	0.0	0.6	0.0	0.0	-0.2	0.2	0.0	0.0
V89	0.0	0.5	0.0	0.0	-0.2	0.0	0.0	0.2
V90	0.0	0.4	0.0	0.0	0.2	0.0	-0.2	-0.2
V91	0.0	0.0	0.5	0.0	0.0	0.0	0.2	-0.2
V92	0.0	0.0	0.4	0.0	-0.2	0.0	0.0	0.0
V93	0.0	0.0	0.3	0.0	0.2	-0.2	0.2	0.0
V94	0.0	0.0	0.0	0.7	0.0	0.2	0.2	0.0
V95	0.0	0.0	0.0	0.5	0.0	0.2	0.0	0.0
V96	0.0	0.0	0.0	0.3	0.0	0.0	-0.2	0.2

man, 1910), α and ω_t increase with the number of items in the scale. Neither statistic flags the scales formed from two orthogonal factors as poor fits. Because a hierarchical model is not identified for three item scales, ω_h was forced to a one factor solution for those scales but properly identifies the last four scales as having low values for general factor saturation. The behavior of the u statistic is very gratifying, in that it does not increase with the number of items per scale, varies as a function of the the range of loadings, and correctly identifies the last four scales as non-unidimensional. For these examples, the performance of the CFI and ECV statistics showed similar patterns: high for unidimensional scales, lower for multidimensional scales. This is in striking contrast to ω_t or α which show very "reasonable" values for these non-unidimensional scales.

Sensitivity to sample size

For practical purposes, we addressed the question of the effect of sample size on the u statistic. We simulated 200, 500, 1,000 and 5,000 participants using the factor structure shown in Table 2. For each simulation we performed 100 replications. We also examined the effect of sample size on ω_t, CFI and ECV. It is quite clear (Figure 1) that even for samples as small as 100, the u statistic could distinguish between unidimensional scales versus multidimensional scales. The pattern of results show that u is, in contrast to ω_t, not sensitive to the number of items in the scale, but is sensitive to unidimensionality. This is evident from comparisons of the first 16 to the last four columns of each panel. The CFI statistic was not sensitive to sample size nor range of factor loadings for one factor data but did drop when given two factor data. The ECV was sensitive to sample size, increasing with larger samples, and to factor structure (decreasing from factors with large to medium to low loadings). In contrast to u, with the exception of 3 item scales, ECV was sensitive to the number of items per factor, increasing from 6 to 12 to 24.

But what if the data are not unidimensional?

Our prior demonstrations examined various fit statistics for one factor models with various number of items and size and range of factor loadings. But what if the data are intentionally multidimensional? Ability measures typically are thought to represent a factor hierarchy of correlated lower order (group) factors with a higher level 'g' factor (Jensen and Weng, 1994). Indeed, measures of general ability are not meant to be purely unidimensional (Humphreys, 1994; Nandakumar, 1991). Similarly, measures of higher order personality scales (e.g., extraversion or neuroticism) are typically not meant to be truly unidimensional, but to include related lower level constructs (Tellegen and Waller, 2008). Variously known

Table 3

Various estimates of unidimensionality and reliability for 500 simulated participants for scales formed from 3, 6, 12, and 24 items with factor loadings as specified in Table 2. The last four rows report results for scales formed from two orthogonal subscales. u *is the unidimensionality statistic, τ and ρ_C are the τ and congeneric fits, ω_h and ω_t are the two omega statistics, α is the traditional estimate. Max split and min split represent the maximum and minimum split half reliabilities found by complete sampling of all $C^k_{k/2}$ possible split half coefficients. $\bar{r}$ reports the mean correlation in the scale. Median r is just the median correlation among the items. CFI is the comparative fit index, ECV is the Explained Common Variance.*

Model	u	τ	ρ_C	ω_h	α	ω_t	Max Split	Min Split	$\bar{r}$	Median r	CFI	ECV	N Items
high.3	0.99	0.99	1.00	0.61	0.61	0.61	0.58	0.52	0.34	0.36	1.00	0.98	3
med.3	0.99	0.99	1.00	0.54	0.54	0.54	0.50	0.45	0.28	0.28	1.00	0.98	3
low.3	0.97	0.97	1.00	0.34	0.33	0.34	0.32	0.26	0.14	0.14	1.00	0.97	3
mixed.3	0.92	0.92	1.00	0.45	0.44	0.45	0.44	0.31	0.21	0.18	1.00	0.97	3
high.6	0.96	0.96	0.99	0.69	0.76	0.78	0.78	0.73	0.35	0.33	0.99	0.74	6
med.6	0.91	0.92	0.99	0.41	0.67	0.74	0.70	0.65	0.26	0.26	1.00	0.73	6
low.6	0.85	0.88	0.97	0.21	0.52	0.62	0.56	0.47	0.15	0.15	1.00	0.69	6
mixed.6	0.73	0.74	0.98	0.26	0.61	0.71	0.66	0.55	0.21	0.18	0.99	0.55	6
high.12	0.94	0.95	0.99	0.76	0.87	0.87	0.89	0.84	0.35	0.35	0.99	0.90	12
med.12	0.88	0.90	0.98	0.38	0.78	0.81	0.82	0.75	0.23	0.23	0.97	0.74	12
low.12	0.78	0.83	0.94	0.48	0.66	0.68	0.71	0.57	0.14	0.14	0.99	0.76	12
mixed.12	0.77	0.78	0.98	0.26	0.77	0.81	0.82	0.71	0.22	0.20	0.99	0.73	12
high.24	0.95	0.95	1.00	0.11	0.93	0.93	0.95	0.92	0.36	0.35	1.00	0.93	24
med.24	0.90	0.92	0.98	0.64	0.88	0.88	0.91	0.85	0.23	0.23	0.99	0.88	24
low.24	0.81	0.85	0.95	0.66	0.82	0.82	0.86	0.77	0.16	0.16	0.99	0.84	24
mixed.24	0.78	0.79	0.98	0.36	0.88	0.89	0.91	0.82	0.24	0.21	0.99	0.90	24
F2.6	0.32	0.50	0.63	0.11	0.50	0.61	0.59	0.12	0.14	0.07	0.61	0.54	6
F2.10	0.30	0.49	0.62	0.07	0.66	0.73	0.75	0.09	0.16	0.06	0.61	0.46	10
F2.12	0.12	0.19	0.63	0.05	0.62	0.73	0.73	0.09	0.12	0.05	0.62	0.50	12
F2.24	0.23	0.34	0.68	0.08	0.79	0.84	0.87	0.32	0.14	0.07	0.65	0.58	24

Table 4

Various estimates of unidimensionality and internal consistency for 500 simulated participants for scales formed from 3, 6, 12, and 24 items with major and minor factor loadings as specified in Table 2. The last four rows report results for scales formed from two orthogonal subscales. Column headings are the same as in Table 3. The addition of a small amount of noise makes very little difference (compare with Table 3).

Model	u	τ	ρ_C	ω_h	α	ω_t	Max Split	Min Split	$\bar{r}$	Median r	CFI	ECV	N Items
high.3	0.98	0.98	1.00	0.59	0.58	0.59	0.56	0.47	0.32	0.30	1.00	0.98	3
med.3	0.99	0.99	1.00	0.53	0.53	0.53	0.51	0.44	0.27	0.29	1.00	0.97	3
low.3	0.90	0.90	1.00	0.43	0.39	0.43	0.43	0.28	0.18	0.19	1.00	0.91	3
mixed.3	0.91	0.91	1.00	0.54	0.52	0.54	0.53	0.36	0.27	0.22	1.00	0.98	3
high.6	0.93	0.93	1.00	0.43	0.76	0.82	0.78	0.74	0.35	0.34	1.00	0.74	6
med.6	0.92	0.94	0.97	0.55	0.67	0.70	0.71	0.61	0.26	0.24	0.96	0.76	6
low.6	0.82	0.84	0.97	0.39	0.54	0.62	0.59	0.47	0.16	0.15	0.99	0.61	6
mixed.6	0.80	0.81	0.99	0.56	0.68	0.72	0.71	0.61	0.26	0.22	0.99	0.79	6
high.12	0.93	0.94	0.99	0.65	0.86	0.87	0.89	0.83	0.34	0.34	0.97	0.88	12
med.12	0.91	0.94	0.97	0.66	0.80	0.81	0.84	0.74	0.25	0.25	0.94	0.81	12
low.12	0.79	0.84	0.95	0.49	0.69	0.70	0.75	0.61	0.16	0.16	0.96	0.77	12
mixed.12	0.81	0.83	0.99	0.30	0.81	0.83	0.85	0.72	0.26	0.23	0.99	0.75	12
high.24	0.93	0.94	0.99	0.60	0.92	0.93	0.95	0.89	0.34	0.34	0.95	0.89	24
med.24	0.89	0.91	0.97	0.64	0.89	0.89	0.92	0.83	0.25	0.25	0.95	0.86	24
low.24	0.77	0.83	0.93	0.58	0.82	0.83	0.87	0.74	0.16	0.16	0.90	0.79	24
mixed.24	0.80	0.82	0.98	0.67	0.89	0.90	0.92	0.82	0.26	0.23	0.98	0.86	24
F2.6	0.06	0.11	0.57	0.02	0.46	0.56	0.56	0.03	0.12	0.05	0.57	0.52	6
F2.10	0.30	0.48	0.63	0.07	0.65	0.73	0.76	0.09	0.16	0.09	0.62	0.52	10
F2.12	0.31	0.47	0.64	0.07	0.68	0.74	0.77	0.10	0.15	0.10	0.63	0.53	12
F2.24	0.30	0.47	0.62	0.06	0.81	0.84	0.88	0.29	0.15	0.11	0.58	0.53	24

Table 5

Comparing unidim (u) estimates for continuous and item (categorical) data for pure and minor simulated factors. The categorical data were generated with six categories using the `sim` and `sim.minor` functions in psych.

Model	pure continuous	pure categorical	minor continuous	minor categorical
high.3	0.99	0.98	0.98	0.97
med.3	0.99	0.97	0.99	0.96
low.3	0.97	0.83	0.90	0.94
mixed.3	0.92	0.87	0.91	0.86
high.6	0.96	0.96	0.93	0.94
med.6	0.91	0.95	0.92	0.95
low.6	0.85	0.84	0.82	0.81
mixed.6	0.73	0.84	0.80	0.76
high.12	0.94	0.94	0.93	0.93
med.12	0.88	0.92	0.91	0.93
low.12	0.78	0.79	0.79	0.83
mixed.12	0.77	0.81	0.81	0.73
high.24	0.95	0.95	0.93	0.94
med.24	0.90	0.90	0.89	0.91
low.24	0.81	0.79	0.77	0.82
mixed.24	0.78	0.80	0.80	0.74
F2.6	0.32	0.06	0.06	0.32
F2.10	0.30	0.09	0.30	0.27
F2.12	0.12	0.10	0.31	0.28
F2.24	0.23	0.22	0.30	0.31

as Factorially Homogeneous Item Dimensions or FHIDs (Comrey, 1984) or Homogeneous Item Composites or HICs (Hogan and Hogan, 2007), these lower level constructs may be formed into higher level scales. Structural analysis includes bi-factor solutions (Reise et al., 2007), (Reise et al., 2018) taking advantage of the bifactor model of Holzinger and Swineford (1937) (see also, Holzinger and Swineford, 1939) as well as the Schmid and Leiman (1957) higher level model. All of these approaches are intentionally multidimensional but the resulting scores tend to be interpreted as one higher order construct.

Jensen and Weng (1994) give a nice example of a higher level (g) factor arising from the composite of three lower level factors. Given a three factor model for nine items with loadings as shown in Table 6 (part A), and factor loadings on 'g' of .9, .8 and .7 result in factor correlations of .72 , .63 and .56. The 9 x 9 correlation matrix may be solved using a higher order solution or rotated using the Schmid and Leiman (1957) oblique factor solution to produce the factor solution shown in part B of Table 6. The u statistic for this set of 9 items is .84, with ω_h of .69 and CFI of .93.

This model was extended to the case of 18, 27, and 36 items by duplicating the loadings shown in Table 6 two, three or four times. To examine the effect of factor structure, factor intercorrelations, and sample size, the higher order loadings were set by specifying identical higher order loadings varying from 0 to 1. Data were generated for 100, 200, 500, and 1000 simulated cases with minor noise factors (MacCallum et al., 2007; MacCallum and Tucker, 1991) with random loadings of -.2 , 0, or .2. The data for 9, 18, 27, and 36 items were generated, with either continuous or categorical items (with five categories). In the categorical case, polychoric correlations were found although we did not expect much effect of the categorical versus continuous distinction (Rhemtulla et al., 2012). Results for the u, ω_h and CFI statistics are shown as a function of the number of items (9, 18, 27, 36) and the general factor loading for both the continuous and categorical cases (Figure 2). Figure 3 show the effects of sample size (100, 200, 500, 1000) on these estimates. To examine the extent these these results generalize across the size of group factors we redid these analyses with smaller group factors (loadings of .6, .5 and .4) for the case of categorical variables (Figure 4) and sample size (Figure 5).

Inspecting these four figures, it is clear that all three measures are sensitive to the g loading, with the u statistic being less sensitive to the number of items and whether they are continuous or categorical. Although both u and ω_h varied across the entire range of g loadings, the CFI estimate was not sensitive to variations in g loadings less than .5. That is, for rejecting unidiminsionality (g < .9 all three statistics did a good job, but CFI was not good index of poor fit.

A reviewer asked why we bother to introduce a new estimate of unidimensionality when we already have perfectly good ways of detecting whether a single factor fits the data? In a comparison of five procedures for estimating the optimal number of factors to extract Zwick and Velicer (1986) show that two are far better than the others. Emphasizing Principal Components, the Minimum Average Partial correlation test (MAP Velicer, 1976; Zwick and Velicer, 1986) finds the number of components that minimize the average squared partial

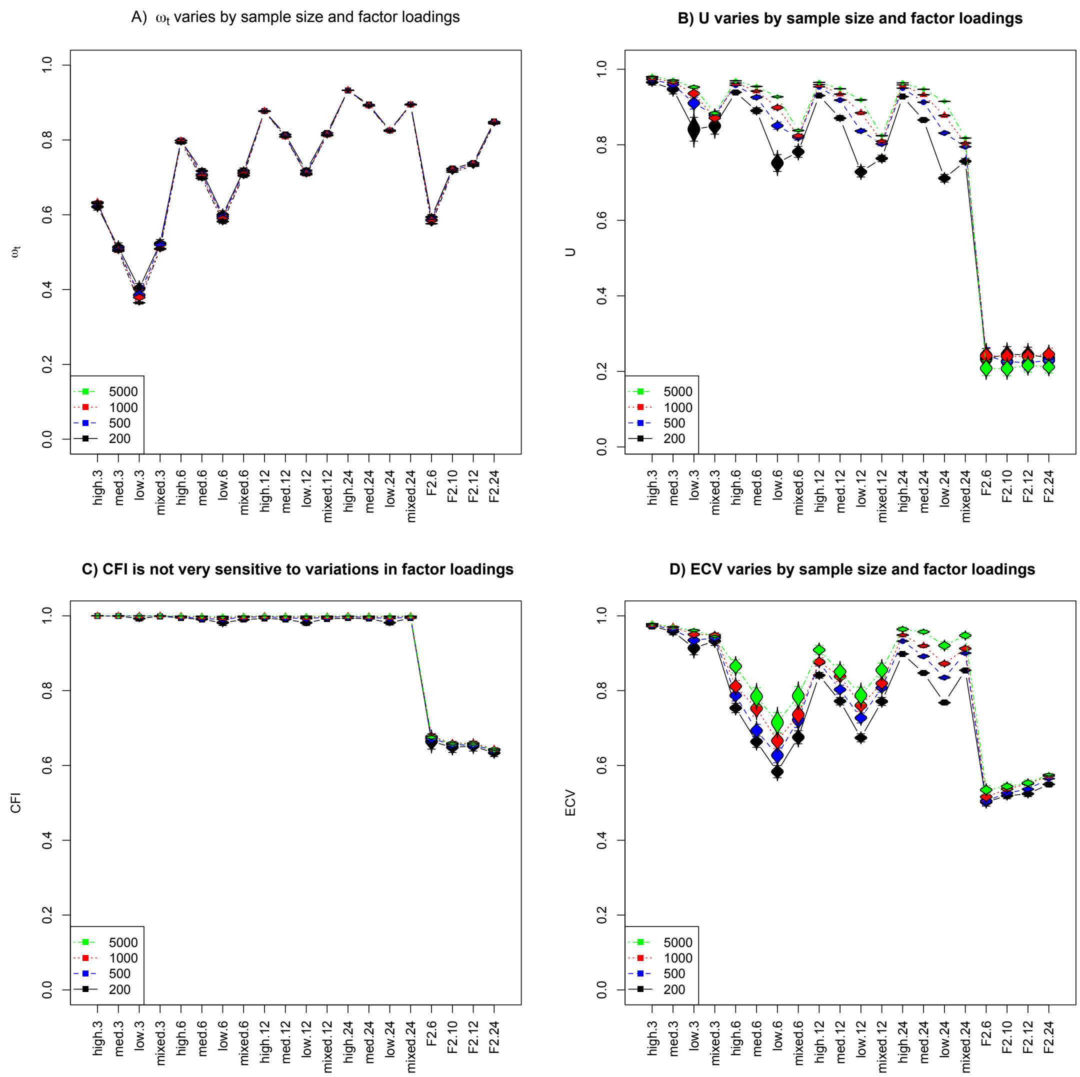

Figure 1

The ω_t, u, CFI *and* ECV *statistics behave very differently across the number of items per factor and sample size. Panel A shows how ω_t increases with the number of items/factor and varies by the size of the factor loadings. It is relatively insensitive to the non-unidimensionality of scales formed from multiple factors and to sample size. Panel B shows how the* u *statistic does not increase with the number of items per scale, is sensitive to the the size of the factor loadings, and is very sensitive to the degree of non-unidimensionality (the four right hand observations in both panels). Panel C shows that the* CFI *varies only slightly by sample or range of factor loadings. Panel D shows that ECV varies by sample size and factor loadings. Sample sizes in all panels range from 200 (black) to 500, 1000, and 5000 (green) simulated participants. The cats-eyes shapes display standard deviations of 100 replications for each sample size.*

correlation of the residual matrix when partialling out the first n components. Parallel Analysis (PA, Horn, 1965; Humphreys and Montanelli, 1975) compares factors/components of a sample to those of random samples of the same number of variables and subjects and identifies the number of factors/components to those with eigen values of the observed sample greater than those of a random samples. In Figure 6 we compare MAP, PA for components, and PA for factors for 100 replications of our hierarchical data problem with 12 levels of general factor saturation, four levels of the number of items, and four sample sizes. All three procedures suggest one factor with g values of 1.0, and three factors for g values of $<$ .5 but are not as clear for g values between .6 and

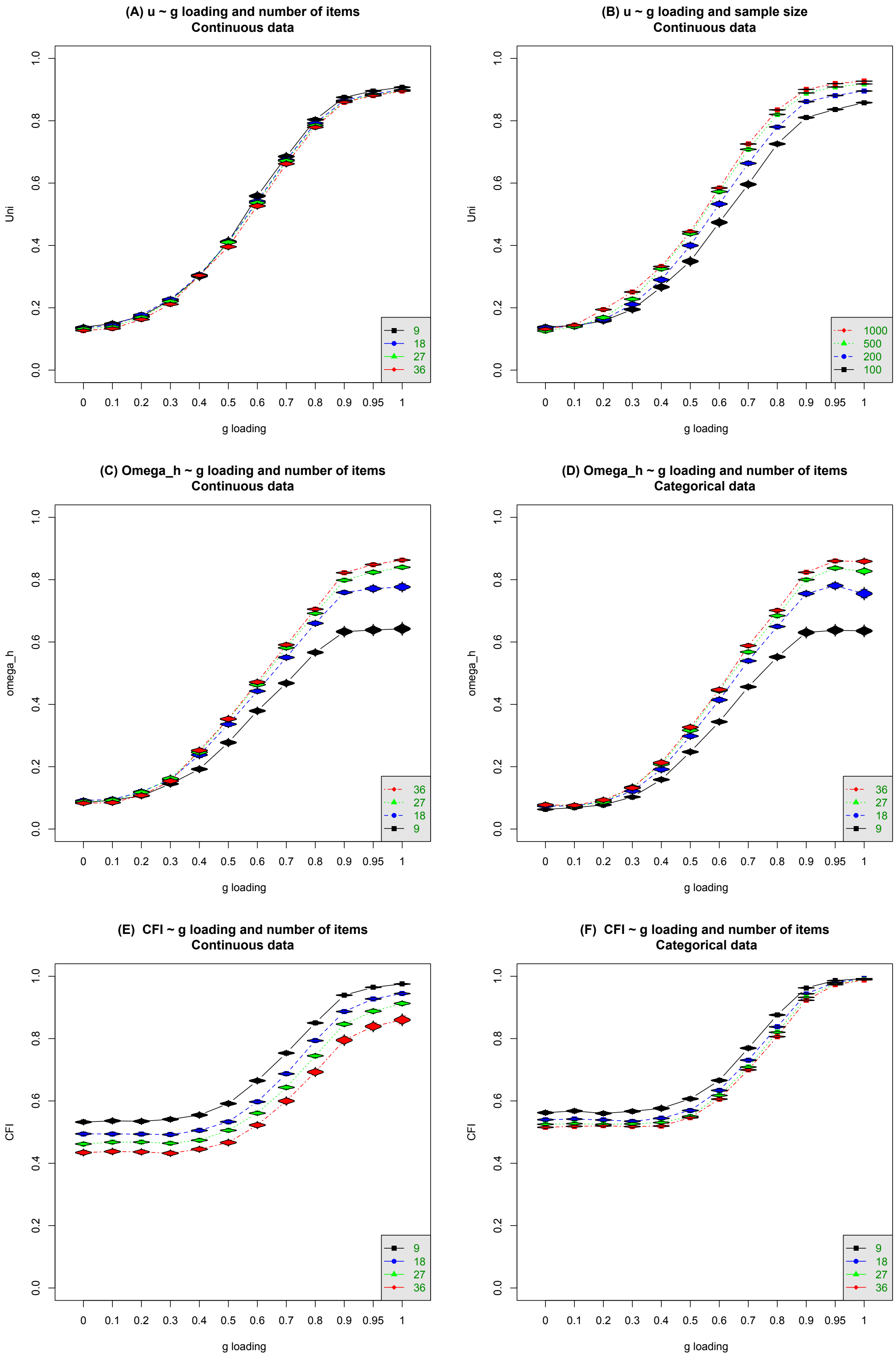

Figure 2

Unidim, ω_h, and CFI as functions of general factor loading (0 ... 1), number of items (9, 18, 27, 36) and type of data (continuous versus 5 level categorical). Left hand panel shows fits as function of number of items and g loadings for continuous data, right hand panel shows the effect of categorical data.. All models are include minor noise factors. Simulated sample sizes range for 100 to 1000 cases (see Figure 3). "Cats eyes" show the 95% confidence of the 100 simulations for each condition.

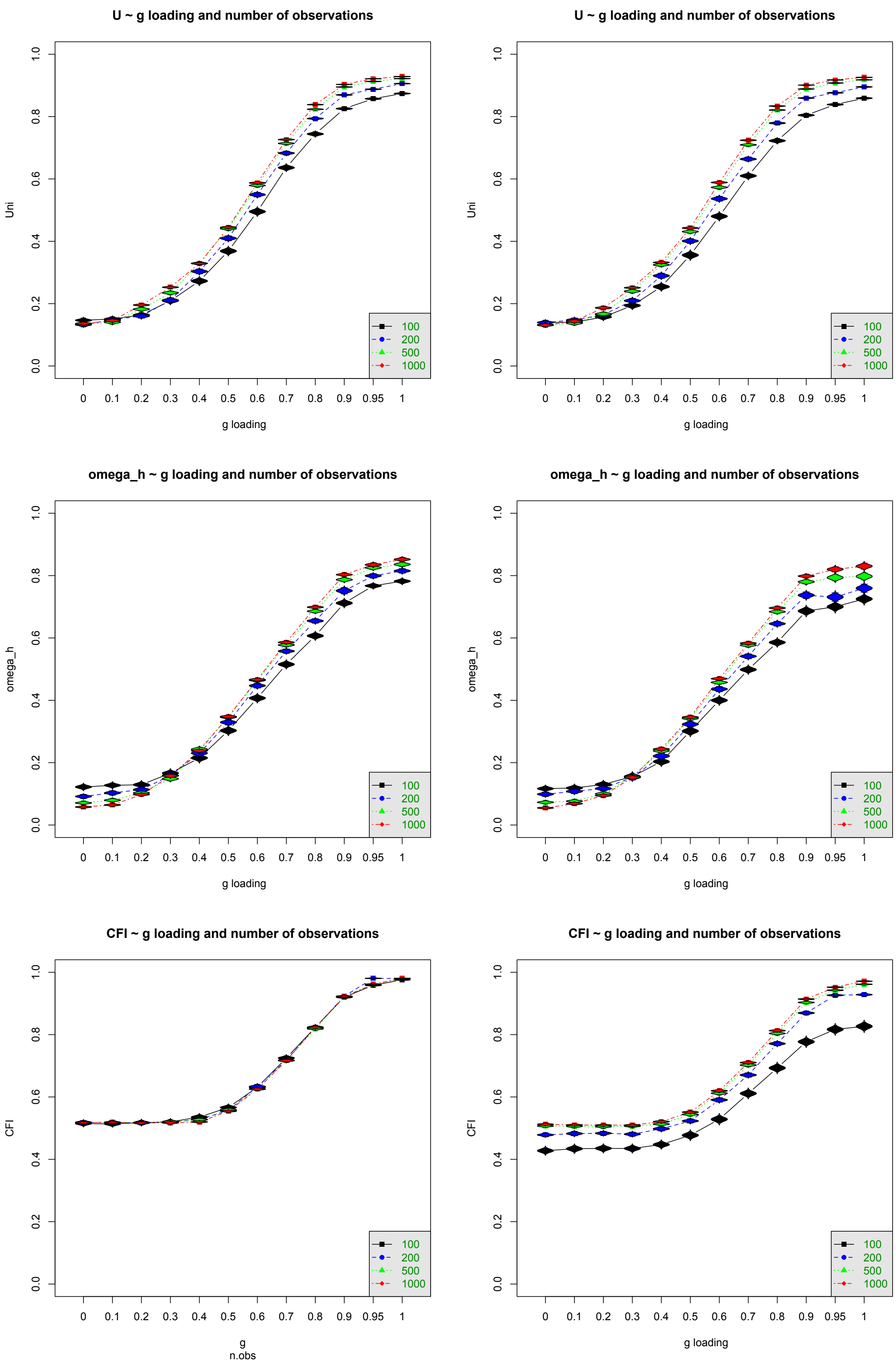

Figure 3

Unidim, ω_h, and CFI as functions of general factor loading and simulated sample size. Left hand panel shows continuous data, right hand panel shows categorical data. All models include minor noise factors. "Cats eyes" show the 95% confidence of the 100 simulations for each condition.

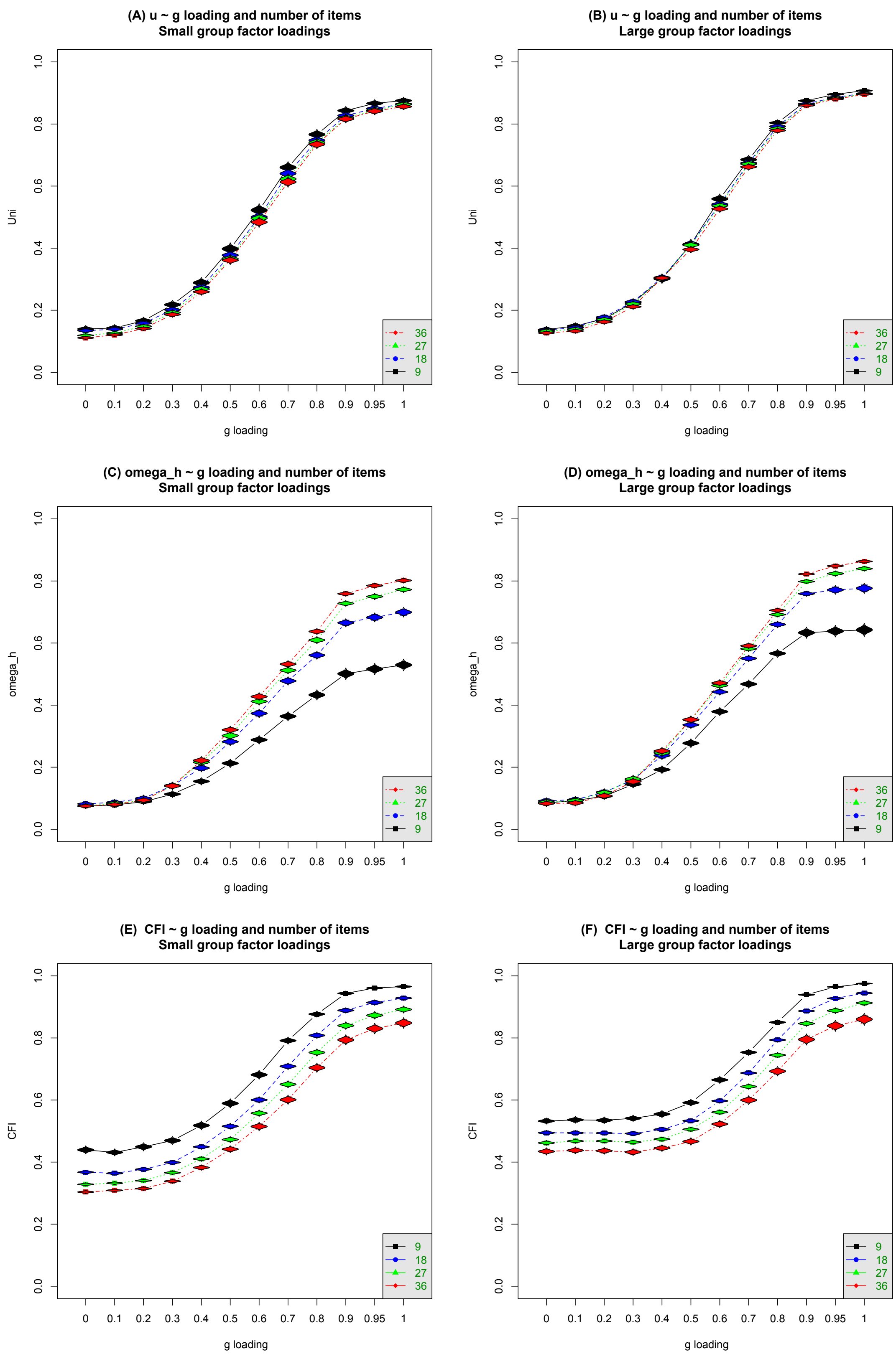

Figure 4

Unidim, ω_h, and CFI as functions of general factor loading and the total number of items (9, 18, 27, 36). Left hand panel shows low group factor loadings, right hand panel shows larger group factors loadings. All data are categorical and include minor noise factors. "Cats eyes" show the 95% confidence of the 100 simulations for each condition.

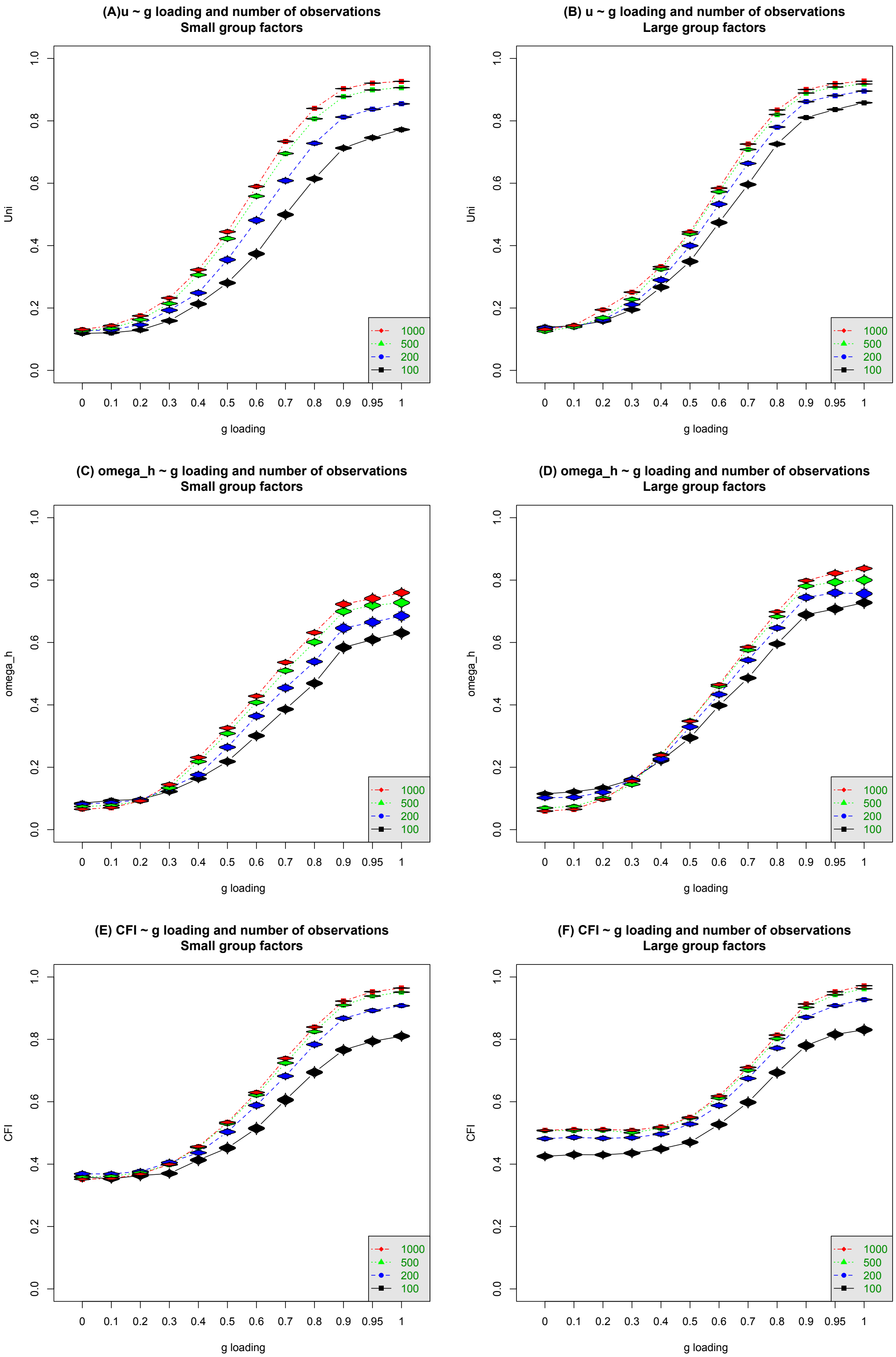

Figure 5

Unidim, ω_h, and CFI as functions of general factor loading and sample size (100, 200, 500, 1000). Left hand panel shows low group factor loadings, right hand panel shows larger group factors loadings. All data are categorical and include small noise factors. "Cats eyes" show the 95% confidence of the 100 simulations for each condition.

Table 6

Three factors with loadings for 9 items have g loadings as specified in the last row (part A). These may be transformed to a bifactor solution (part B). $U = .84, \omega_h = .69, CFI = .93$. A similar set of factors (F1, F2, F3) are then duplicated two, three or four times and the g loadings are set to be all 0, all .1, ... all 1.0, to create the results shown in Figure 3)

	Part A			Part B							
Variable	F1	F2	F3	g	F1*	F2*	F3*	h2	u2	p2	com
V1	0.80	0.00	0.00	0.72	0.35	0.00	0.00	0.64	0.36	0.81	1.44
V2	0.70	0.00	0.00	0.63	0.31	0.00	0.00	0.49	0.51	0.81	1.44
V3	0.60	0.00	0.00	0.54	0.26	0.00	0.00	0.36	0.64	0.81	1.44
V4	0.00	0.70	0.00	0.56	0.00	0.42	0.00	0.49	0.51	0.64	1.85
V5	0.00	0.60	0.00	0.48	0.00	0.36	0.00	0.36	0.64	0.64	1.85
V6	0.00	0.50	0.00	0.40	0.00	0.30	0.00	0.25	0.75	0.64	1.85
V7	0.00	0.00	0.60	0.42	0.00	0.00	0.43	0.36	0.64	0.49	2.00
V8	0.00	0.00	0.50	0.35	0.00	0.00	0.36	0.25	0.75	0.49	2.00
V9	0.00	0.00	0.40	0.28	0.00	0.00	0.29	0.16	0.84	0.49	2.00
g	0.90	0.80	0.70								

.9. Compare these curves to those showing u and CFI with values for the same range of g values in figures 2-5. Unlike u or the CFI, MAP and PA are very sensitive to the number of items in the scale. MAP and PA give different answers for 9 or 18 items as a function of the g loadings in the range from .7 to .9. In fact for 9 items MAP suggests one or slightly more the one factor across the entire g range.

Applying `unidim` to real data

The simulations were done with continuous and categorical item scores with known (simulated) structure. However, most real ability and personality items are categorical and the structures are assumed rather than known. `unidim` can be applied to such data using either tetrachoric or more generally polychoric correlations. In addition to showing results with simulated continuous and categorical data (Table 5, Figures 2, 3) we show the utility of the u statistic with real categorical data from three data sets available in the *psychTools* package (Revelle, 2024b) for the R statistical system. For reasons discussed in Chalmers (2017) and Revelle and Condon (2019), conventional reliability indices were found using Pearson correlations. However, the unidimensional estimates were found from factoring the tetrachoric or polychoric correlation matrices.

Of these data sets, the first, the `ability` data set includes 16 items for 1,525 participants from the International Cognitive Ability Resource (Condon and Revelle, 2014), which represents 4 lower level factors and one higher level factor (Table 7 part 1). The items are dichotomous. The second dataset, `bfi`, contains data from 2,800 participants who responded to 25 Likert-like items with six response choices ranging from "very inaccurate" to "very accurate." Although normally scored using only five separate constructs (the familiar Big Five traits shown in Table 7 part 2), composite scales were also formed here, for demonstration purposes, from two (E+O), three (A+C+N) or five (all) of these constructs. This is a particularly nice example of the advantage of the u statistic as contrasted with the more conventional α and ω_t statistics, for the latter show quite reasonable values (.71 - .84) for scales that are not unidimensional. u and ω_h on the other hand, show clear evidence (.30 - .41) for multidimensionality (Table 7 part 3). What is interesting is that for these nominally unifactorial scales, the CFI and ECV statistics were quite low. The ECVs for the single construct `bfi` scales ranged from .79 to .87, values, while the CFIs ranged from .90 to .97 which, although larger than for the multi-construct scales (.52 - .61 for ECV, .40 - .64 for CFI), do not suggest strong unidimensionality.

The third dataset uses the 135 items of the SAPA Personality Inventory (Condon, 2018) for 4,000 participants (`spi`) to form 27 lower level scales; these also had six response options ("very inaccurate" to "very accurate") (Table 8). In this case, the u values for the SPI-27 reflect the higher degree of unidimensionality expected from brief 5-item scales with a median of .94 and ranging from .83 to .99. By contrast, α values are relatively lower (again, as expected, given the short scale lengths), and the ω_h values are difficult to interpret due to inadequate degrees of freedom. In contrast to the findings with the `bfi` scales, these 27 scales show higher levels of ECV with a median of .91 and ranging from .77 to .96.

Comparison of fit statistics

A reasonable question to ask is what is the relationship between these various estimates of unidimensionality? As one reviewer questioned, is our ρ_c measure any different from ECV? Conceptually our proposed measure, u, based upon the product of τ and ρ_c would seem to be similar to the goodness of fit of a one factor model (CFI) or the Explained Common Variance. To address

Table 7

Unidim coefficients and multiple reliability measures for the `ability` *data set with 16 items and 1,525 participants and the* `bfi` *data set with 2,800 participants for 25 items. The first three columns report* `unidim` *statistics* u, τ, ρ_c, *the next three columns the conventional* ω_h, α, *and* ω_t *internal consistency estimates, and the next two report the maximum and minimum split half reliabilities based upon a split half decomposition of the scales. The next two columns (*$\bar{r}$ *and Median* r*) are the mean and median within scale correlations, respectively. The last three columns are the CFI and ECV, followed by the number of items per scale. The ICAR16 represents a higher order factor composed of four lower level factors. Although the* `bfi` *is scored as five unidimensional constructs, the last three lines represent scales formed from two (E+O), three (A+C+N) or five (all) constructs. For the latter three scales, note how the high value of* α *and* ω_t *conflict with low values of* u *and* ω_h *which are better indicators of unidimensionality.*

Data set Scale	u	τ	ρ_C	ω_h	α	ω_t	Max Split	Min Split	Mean r	Median r	CFI	ECV	N Items
ICAR dataset													
ICAR16	0.90	0.93	0.96	0.56	0.83	0.85	0.87	0.73	0.23	0.21	0.76	0.78	16
reasoning	0.98	0.98	1.00	0.05	0.65	0.65	0.65	0.64	0.31	0.31	1.00	0.99	4
letters	0.99	0.99	1.00	0.65	0.67	0.68	0.68	0.66	0.34	0.33	1.00	0.98	4
matrix	0.93	0.95	0.98	0.37	0.54	0.59	0.58	0.49	0.23	0.22	0.95	0.83	4
rotate	1.00	1.00	1.00	0.71	0.77	0.78	0.78	0.74	0.45	0.44	0.99	0.96	4
bfi dataset: single construct scales													
extraversion	0.96	0.97	0.99	0.55	0.76	0.82	0.78	0.66	0.39	0.38	0.97	0.86	5
openness	0.89	0.91	0.98	0.39	0.61	0.70	0.66	0.52	0.24	0.23	0.94	0.79	5
agreeableness	0.90	0.91	0.99	0.64	0.71	0.75	0.75	0.62	0.33	0.34	0.96	0.86	5
conscientious	0.95	0.97	0.98	0.53	0.73	0.77	0.74	0.64	0.35	0.34	0.93	0.84	5
neuroticism	0.94	0.95	0.98	0.71	0.81	0.85	0.83	0.69	0.47	0.41	0.90	0.87	5
bfi multi construct scales													
EO	0.49	0.62	0.79	0.36	0.71	0.77	0.79	0.38	0.20	0.21	0.64	0.61	10
ACN	0.43	0.63	0.68	0.33	0.78	0.81	0.87	0.47	0.19	0.14	0.44	0.52	15
All	0.38	0.56	0.68	0.30	0.82	0.84	0.89	0.58	0.15	0.13	0.40	0.56	25

Table 8

Unidim coeffficients and multiple reliability measures of the `spi-135` *(Condon, 2018). The SAPA Personality Inventory (Condon, 2018) has five higher order scales assessing the "Big Five" and 27 lower level scales assessing other aspects of personality. The 27 lower level measures have 5 items each. The* `spi` *data set in the psychTools has 4,000 observations on these 135 items plus 10 criteria/demographic variables. The columns are the same as in Table 7.*

Model	u	τ	ρ_C	ω_h	α	ω_t	Max Split	Min Split	Mean r	Median r	CFI	ECV	N Items
Compassion	0.99	0.99	1.00	0.80	0.88	0.89	0.87	0.82	0.59	0.58	0.99	0.94	5
Trust	0.99	0.99	1.00	0.80	0.87	0.89	0.87	0.81	0.58	0.58	0.98	0.94	5
Honesty	0.96	0.97	0.99	0.71	0.81	0.84	0.83	0.70	0.46	0.46	0.94	0.88	5
Conservatism	0.84	0.91	0.92	0.56	0.78	0.85	0.84	0.61	0.41	0.35	0.66	0.76	5
Authoritarianism	0.89	0.93	0.96	0.63	0.81	0.86	0.85	0.63	0.46	0.46	0.83	0.81	5
EasyGoingness	0.90	0.92	0.98	0.45	0.68	0.76	0.73	0.58	0.29	0.29	0.94	0.78	5
Perfectionism	0.83	0.84	0.99	0.34	0.70	0.74	0.72	0.53	0.31	0.33	0.97	0.87	5
Order	0.93	0.94	0.99	0.62	0.81	0.85	0.83	0.66	0.46	0.42	0.93	0.86	5
Industry	0.99	0.99	1.00	0.72	0.84	0.86	0.84	0.76	0.52	0.50	0.98	0.94	5
Impulsivity	0.98	0.98	1.00	0.72	0.87	0.90	0.87	0.80	0.58	0.58	0.99	0.96	5
SelfControl	0.91	0.94	0.96	0.49	0.76	0.83	0.80	0.60	0.39	0.36	0.87	0.79	5
EmotionalStability	0.99	0.99	1.00	0.65	0.85	0.89	0.84	0.76	0.52	0.50	0.99	0.93	5
Anxiety	0.99	0.99	1.00	0.83	0.90	0.91	0.89	0.83	0.64	0.62	0.97	0.95	5
Irritability	0.98	0.99	0.99	0.78	0.89	0.91	0.89	0.79	0.61	0.60	0.94	0.91	5
WellBeing	0.99	0.99	1.00	0.80	0.90	0.92	0.90	0.81	0.63	0.63	0.95	0.92	5
EmotionalExpressiveness	0.93	0.94	0.99	0.73	0.80	0.83	0.83	0.68	0.45	0.43	0.95	0.90	5
Sociability	0.97	0.98	0.99	0.66	0.85	0.89	0.85	0.75	0.53	0.50	0.95	0.88	5
Adaptability	0.93	0.94	0.99	0.62	0.80	0.84	0.82	0.68	0.44	0.42	0.95	0.88	5
Charisma	0.94	0.96	0.98	0.67	0.82	0.86	0.84	0.72	0.47	0.43	0.88	0.85	5
Humor	0.94	0.94	0.99	0.68	0.78	0.82	0.81	0.64	0.42	0.40	0.97	0.91	5
AttentionSeeking	0.93	0.93	0.99	0.80	0.88	0.90	0.89	0.77	0.58	0.67	0.94	0.92	5
SensationSeeking	0.97	0.98	0.99	0.77	0.86	0.89	0.87	0.77	0.55	0.54	0.92	0.91	5
Conformity	0.90	0.94	0.96	0.67	0.82	0.87	0.85	0.67	0.47	0.47	0.82	0.83	5
Introspection	0.94	0.95	0.99	0.56	0.78	0.84	0.81	0.68	0.41	0.41	0.94	0.87	5
ArtAppreciation	0.90	0.90	1.00	0.68	0.80	0.83	0.81	0.65	0.44	0.46	0.98	0.92	5
Creativity	0.97	0.98	1.00	0.70	0.85	0.86	0.85	0.77	0.52	0.53	0.98	0.94	5
Intellect	0.99	0.99	1.00	0.81	0.86	0.87	0.84	0.78	0.54	0.52	0.99	0.96	5
median	0.94	0.96	0.99	0.68	0.82	0.86	0.84	0.72	0.47	0.47	0.96	0 .90	5

this question, we report the correlations of these and additional measures (Table 9) for the continuous and categorical datas sets shown in Figure 2. Table 10 compares the effect of larger versus smaller group factor loadings. In all of these case, we show the correlation with the g loading as well. For both continuous and categorical data and for small versus large group factor loadings the *u* statistic is more closely related to the g loading (r=.96 for both cases with larger group factor loadings, .9 for smaller group factor loadings) than are the other estimates.

Table 9

Correlation of fit statistics across 12 levels of g loadings. Lower off diagonal for continuous variables, upper off diagonal for categorical variables, N observations range from 100 to 1000, number of items fit range from 9 to 36. See Figure 2 for the data.

Variable	g	Uni	τ	ρ_c	ω_h	CFI	ECV
g		0.96	0.95	0.94	0.92	0.84	0.92
Uni	0.96		0.99	0.99	0.96	0.89	0.96
τ	0.95	0.98		0.96	0.94	0.83	0.92
ρ_c	0.95	0.99	0.96		0.95	0.91	0.96
ω_h	0.94	0.96	0.94	0.96		0.83	0.96
CFI	0.89	0.92	0.85	0.93	0.88		0.91
ECV	0.94	0.96	0.92	0.96	0.96	0.96	

Table 10

Correlation of fit statistics across 12 levels of g loadings. Lower off diagonal values represent lower group factor loadings, upper off diagonal larger group factor loadings.

Variable	g	Uni	τ	ρ_c	ω_h	CFI	ECV
g		0.96	0.95	0.95	0.93	0.83	0.93
Uni	0.90		0.98	0.99	0.96	0.88	0.96
τ	0.89	0.98		0.96	0.93	0.82	0.92
ρ_c	0.89	0.98	0.96		0.95	0.90	0.96
ω_h	0.85	0.90	0.88	0.88		0.82	0.96
CFI	0.81	0.91	0.86	0.94	0.75		0.90
ECV	0.88	0.93	0.90	0.92	0.95	0.83	

How should we interpret these various fit statistics and what should we do if the fits are not perfect? Even with data meant to be represented by one factor, (i.e, g loadings of 1 in the hierarchical factor model– Figure 2) the fits are not perfect. Unlike ω_h or CFI, *u* is not sensitive to the number of items in the scale, but all of these measures are sensitive to the number of observations (Figures 3, 5). The magnitude of ω_h is positively correlated with the number of items, while CFI is negatively correlated. All measures are larger for more observations, with the exception of CFI for continuous data.

Summary and Conclusion

The problem of assessing the fit of a unidimensional model of a scale has been a challenge for many years, and many solutions have been offered. Here we have suggested a simple index, *u*, which is the the product of indices of τ equivalance and ρ_c or congeneric equivalence. *u* is simple to calculate (e.g., using the `unidim` or `reliability` functions in the *psych* package for R). We compare the *u* index to five popular indices of scale quality, α, ω_h ω_t, CFI, and ECV and show that unlike α, ω_t, or ECV, *u* is insensitive to the number of items in a scale and unlike the CFI is sensitive to factor structure. *u* is robust across sample sizes from 100 to 5000 in assessing how well a unidimensional model fits the data.

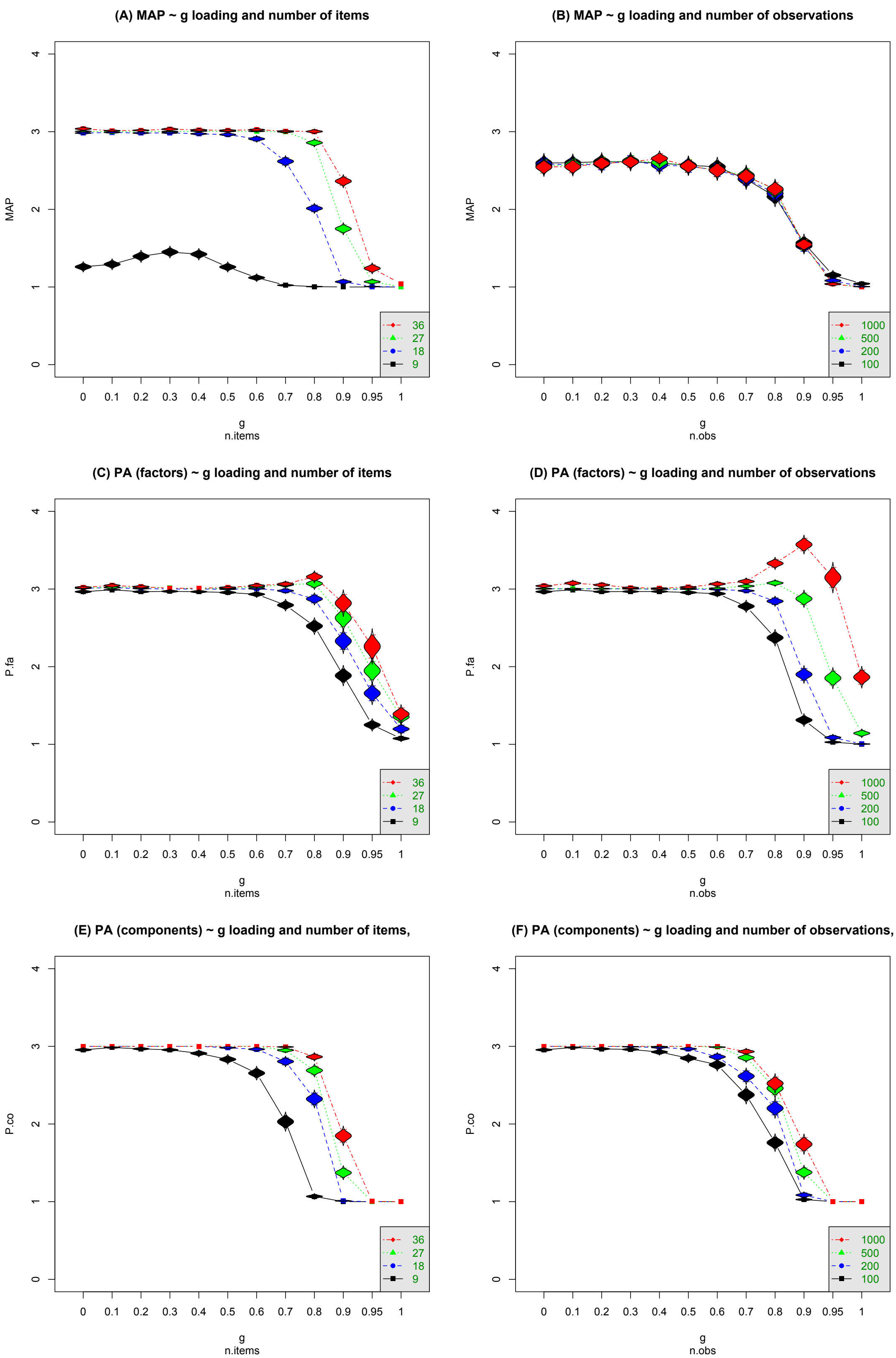

Figure 6

Alternative measures of the number of factors in a hierarchical model include the Minimum Average Partial (MAP), parallel analysis of the number of factors (P.fa) , or parallel analysis of the number of components (P.co). For all 3 estimates, we show the statistic as a function of the g loadings of the generating model, as well as the number of items (panels A, C, E) and the number of observations (Panels B, D, F). Catseyes show the confidence intervals for 100 replications.

References

Bentler, Peter M (1990). Comparative fit indexes in structural models. *Psychological bulletin*, 107(2):238–246.

Brown, William (1910). Some experimental results in the correlation of mental abilities. *British Journal of Psychology*, 3(3):296–322.

Chalmers, R. Philip (2017). On misconceptions and the limited usefulness of ordinal alpha. *Educational and Psychological Measurement*, 78(6):1056–1071.

Cho, Eunseong and Kim, Seonghoon (2015). Cronbach's coefficient alpha: Well known but poorly understood. *Organizational research methods*, 18(2):207–230.

Comrey, Andrew L. (1984). Comparison of two methods to identify major personality factors. *Applied Psychological Measurement*, 8(4):397–408.

Condon, David M. (2018). The SAPA Personality Inventory: An empirically-derived, hierarchically-organized self-report personality assessment model. *PsyArXiv*.

Condon, David M. and Revelle, William (2014). The International Cognitive Ability Resource: Development and initial validation of a public-domain measure. *Intelligence*, 43:52–64.

Cronbach, Lee J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika*, 16:297–334.

Davenport, Ernest C, Davison, Mark L, Liou, Pey-Yan, and Love, Quintin U (2015). Reliability, dimensionality, and internal consistency as defined by cronbach: Distinct albeit related concepts. *Educational measurement : issues and practice.*, 34(4).

Embretson, Susan E. and Reise, Steven P. (2025). Testing assumptions and assessing fit. In *Item Response Theory: Foundations for Psychologists and Social Scientists.*, chapter 8. Routledge/Taylor & Francis.

Green, Samuel B and Yang, Yanyun (2015). Evaluation of dimensionality in the assessment of internal consistency reliability: Coefficient alpha and omega coefficients. *Educational measurement: issues and practice.*, 34(4):14–20.

Guttman, Louis (1945). A basis for analyzing test-retest reliability. *Psychometrika*, 10(4):255–282.

Hattie, John (1984). An empirical study of various indices for determining unidimensionality. *Multivariate Behavioral Research*, 19(1):49–78.

Hattie, John (1985). Methodology review: Assessing unidimensionality of tests and ltems. *Applied Psychological Measurement*, 9:139–164.

Hogan, Robert and Hogan, Joyce (2007). *The Hogan Personality Inventory (3rd Editionr)*. Hogan Assessment Systems, Tulsa, OK.

Holzinger, Karl and Swineford, Frances (1937). The bifactor method. *Psychometrika*, 2(1):41–54.

Holzinger, Karl and Swineford, Frances (1939). A study in factor analysis: the stability of a bi-factor solution. Department of Education, University of Chicago, Chicago, Il.

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

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

Humphreys, Lloyd G. (1994). Intelligence from the standpoint of a (pragmatic) behaviorist. *Psychological Inquiry*, 5(3):179–192.

Humphreys, Lloyd G. and Montanelli, Richard G. (1975). An investigation of the parallel analysis criterion for determining the number of common factors. *Multivariate Behavioral Research*, 10(2):193–205.

Jensen, Arthur R. and Weng, Li-Jen (1994). What is a good g? *Intelligence*, 18(3):231–258.

Kamata, Akihito and Bauer, Daniel J. (2008). A note on the relation between factor analytic and item response theory models. *Structural Equation Modeling: A Multidisciplinary Journal*, 15(1):136–153.

Lord, Frederic M. and Novick, Melvin R. (1968). *Statistical theories of mental test scores*. The Addison-Wesley series in behavioral science: quantitative methods. Addison-Wesley Pub. Co, Reading, Mass.

Lorenzo-Seva, Urbano, Timmerman, Marieke E., and Kiers, Henk A. L. (2011). The hull method for selecting the number of common factors. *Multivariate Behavioral Research*, 46(2):340–364.

Lucke, Joseph F (2005). "rassling the hog": The influence of correlated item error on internal consistency, classical reliability, and congeneric reliability. *Applied psychological measurement*, 29(2):106–125.

MacCallum, Robert C., Browne, Michael W., and Cai, Li (2007). Factor analysis models as approximations. In Cudeck, Robert and MacCallum, Robert C., editors, *Factor analysis at 100: Historical developments and future directions*, pages 153–175. Lawrence Erlbaum Associates Publishers, Mahwah, NJ.

MacCallum, Robert C and Tucker, Ledyard R (1991). Representing sources of error in the common-factor model: Implications for theory and practice. *Psychological Bulletin*, 109(3):502–511.

McDonald, Roderick P (1981). The dimensionality of tests and items. *British Journal of mathematical and statistical Psychology*, 34(1):100–117.

McDonald, Roderick P. (1999). *Test theory: A unified treatment*. L. Erlbaum Associates, Mahwah, N.J.

McNemar, Quinn (1946). Opinion-attitude methodology. *Psychological Bulletin*, 43(4):289–374.

Nandakumar, Ratna (1991). Traditional dimensionality versus essential dimensionality. *Journal of Educational Measurement*, 28(2):99–117.

Nandakumar, Ratna and Stout, William (1993). Refinements of Stout's procedure for assessing latent trait unidimensionality. *Journal of educational statistics*, 18(1):41–68.

R Core Team (2024). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rasch, G. (1966). An item analysis which takes individual differneces in to account. *British Journal of Mathematical and Statistical Psychology*, 19(1):49–57.

Reise, Steven P. (2012). The rediscovery of bifactor measurement models. *Multivariate Behavioral Research*, 47(5):667–696.

Reise, Steven P., Bonifay, Wes, and Haviland, Mark G. (2018). *Bifactor Modelling and the Evaluation of Scale Scores*, chapter 22, pages 675–707. John Wiley & Sons, Ltd.

Reise, Steven P., Morizot, Julien, and Hays, Ron (2007). The role of the bifactor model in resolving dimensionality issues in health outcomes measures. *Quality of Life Research*, 16:19–31.

Revelle, W. and Rocklin, T. (1979). Very Simple Structure - alternative procedure for estimating the optimal number of interpretable factors. *Multivariate Behavioral Research*, 14(4):403–414.

Revelle, William (2024a). *psych:Procedures for Psychological, Psychometric, and Personality Research*. Northwestern University, Evanston, https://CRAN.r-project.org/package=psych, 2.4.7 edition. R package version 2.4.7.

Revelle, William (2024b). *psychTools Tools to Accompany the Psych Package for Psychological Research*. Northwestern University, Evanston, psychTools. R package version 2.4.3.

Revelle, William and Condon, David M. (2019). Reliability from α to ω: A tutorial. *Psychological Assessment.*, 31(12):1395–1411.

Rhemtulla, Mijke, Brosseau-Liard, Patricia É, and Savalei, Victoria (2012). When can categorical variables be treated as continuous? a comparison of robust continuous and categorical sem estimation methods under suboptimal conditions. *Psychological methods*, 17(3):354.

Rodriguez, Anthony, Reise, Steven P, and Haviland, Mark G (2016). Evaluating bifactor models: Calculating and interpreting statistical indices. *Psychological methods*, 21(2):137–150.

Schmid, John Jr. and Leiman, John. M. (1957). The development of hierarchical factor solutions. *Psychometrika*, 22(1):83–90.

Shapiro, Alexander and ten Berge, Jos M.F. (2002). Statistical inference of minimum rank factor analysis. *Psychometrika*, 67(1):79–94.

Sijtsma, Klaas (2009). On the use, the misuse, and the very limited usefulness of Cronbach's alpha. *Psychometrika*, 74(1):107–120.

Spearman, Charles (1904). "General Intelligence," objectively determined and measured. *American Journal of Psychology*, 15(2):201–292.

Spearman, Charles (1910). Correlation calculated from faulty data. *British Journal of Psychology*, 3(3):271–295.

Stout, William, Froelich, Amy Goodwin, and Gao, Furong (2001). Using resampling methods to produce an improved dimtest procedure. In Boomsma, Anne, van Duijn, Marijtje A. J., and Snijders, Tom A. B., editors, *Essays on Item Response Theory*, pages 357–375. Springer New York, New York, NY.

Stout, William F. (1990). A new item response theory modeling approach with applications to unidimensionality assessment and ability estimation. *Psychometrika*, 55(2):293–325.

Takane, Yoshio and de Leeuw, Jan (1987). On the relationship between item response theory and factor analysis of discretized variables. *Psychometrika*, 52:393–408. 10.1007/BF02294363.

Tellegen, Auke (1991). Personality traits: Issues of definition, evidence, and assessment. In *Thinking clearly about psychology: Essays in honor of Paul E. Meehl*, volume 2, pages 10–35. University of Minnesota Press.

Tellegen, Auke and Waller, Niels G (2008). Exploring personality through test construction: Development of the multidimensional personality questionnaire. *The Sage handbook of personality theory and assessment*, 2:261–292.

Ten Berge, Jos M. F. and Socan, Gregor (2004). The greatest lower bound to the reliability of a test and the hypothesis of unidimensionality. *Psychometrika*, 69(4):613–625.

Ten Berge, Jos MF and Kiers, Henk AL (1991). A numerical approach to the approximate and the exact minimum rank of a covariance matrix. *Psychometrika*, 56:309–315.

Thomas, Michael L (2019). Advances in applications of item response theory to clinical assessment. *Psychological assessment*, 31(12):1442–.

Timmerman, Marieke E and Lorenzo-Seva, Urbano (2011). Dimensionality assessment of ordered polytomous items with parallel analysis. *Psychological methods*, 16(2):209.

Tucker, Ledyard, Koopman, Raymond, and Linn, Robert (1969). Evaluation of factor analytic research procedures by means of simulated correlation matrices. *Psychometrika*, 34(4):421–459.

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

Walker, David A. (1931). Answer-pattern and score-scatter in tests and examinations. *British Journal of Psychology. General Section*, 22(1):73–86.

Zinbarg, R. E., Revelle, W., Yovel, I., and Li, W. (2005). Cronbach's α, Revelle's β, and McDonald's ω_H: Their relations with each other and two alternative conceptualizations of reliability. *Psychometrika*, 70(1):123–133.

Zinbarg, R. E., Yovel, I., Revelle, W., and McDonald, R. P. (2006). Estimating generalizability to a latent variable common to all of a scale's indicators: A comparison of estimators for ω_h. *Applied Psychological Measurement*, 30(2):121–144.

Zwick, William R. and Velicer, Wayne F. (1986). Comparison of five rules for determining the number of components to retain. *Psychological Bulletin*, 99(3):432 – 442.

Appendix: R code

Here we show the R code used in the simulations.

The first section is for Tables 1-5 and Figure 6 showing various effects on the one factor model.

The next section is for the variation in the g factor loadings.

One factor models

R code

```
library(psych)
library(psychTools)
library(parallel)  #to allow multiple cores (for speed)
#requires psych version 2.4.3 or above

#First simulate data for tables 1-5

#The basic data structure
#we create a 96 x 96 matrix, from which we take partitions for the various examples

fx4 <- matrix(c(.7,.6,.5,rep(0,12),.6,.5,.4,rep(0,12) ,.5,.4,.3,rep(0,12),.7,.5,.3),ncol=4)
fx96 <- rbind(fx4,fx4,fx4,fx4,fx4,fx4,fx4)
rownames(fx96) <- paste0("V",1:96)
colnames(fx96) <- paste0("F",1:4)
 keys96 <- factor2cluster(fx96)
 rownames(keys96) <- paste0("V",1:96)
 keys.96 <- keys2list(keys96)
 keys.12 <-keys2list(keys96[1:12,])
keys.24 <- keys2list(keys96[1:24,])
 keys.48 <- keys2list(keys96[1:48,])
# names(keys.12) <- paste0("F",1:4,".12")
names(keys.12) <- paste0(cs(high, med, low, mixed),".",3)
names(keys.24) <- paste0(cs(high, med, low, mixed),".",6)
 names(keys.48) <- paste0(cs(high, med, low, mixed),".",12)
names(keys.96) <- paste0(cs(high, med, low, mixed),".",24)
keys.6.2.factors <- list(c(keys.12$high,keys.12$med))
keys.10.2.factors <- list(c(keys.24$high[1:5],keys.24$med[1:5]))
keys.12.2.factors <- list(c(keys.48$high[1:6],keys.48$med[1:6]))
keys.24.2.factors <- list(c(keys.96$high[1:12],keys.96$med[1:12]))
names(keys.6.2.factors) <- "F2.6"
names(keys.10.2.factors) <- "F2.10"
names(keys.12.2.factors) <- "F2.12"
names(keys.24.2.factors) <- "F2.24"
#names(keys.48.2.factors) <- "F2.48"
 keys <- c(keys.12,keys.24,keys.48,keys.96,keys.6.2.factors,keys.10.2.factors,
       keys.12.2.factors,keys.24.2.factors)

#one run to create Tables 3 and 4
n.obs=500
set.seed(42)  #for reproducible results
sim.96 <- sim(fx96,n=n.obs)  #generate n.obs participants for 96 variables
minor.96  <- sim.minor(fbig=fx96,n=n.obs, n.small=4) #include nuisance factors
#create item like data
sim.96.item <- sim(fx96,n=n.obs, items=TRUE)  #generate n.obs participants for 96 variables
minor.96.item  <- sim.minor(fbig=fx96,n=n.obs, n.small=4,items=TRUE) #include nuisance factors

reliability.pure.500 <- reliability(keys, sim.96$observed) #
reliability.minor.500 <- reliability(keys, minor.96$observed,n.sample=50000)


uni.pure.cont <- unidim(keys,sim.96$observed)
uni.minor.cont <-  unidim(keys,minor.96$observed)
#do it for categoriical data
uni.pure.item <- unidim(keys,sim.96.item$observed,cor="poly")
uni.minor.item  <- unidim(keys,minor.96.item$observed,cor="poly")

uni.sum <- data.frame(pure.cont=uni.pure.cont$uni[,1:3],
                      pure.item=uni.pure.item$uni[,1:3],
            minor.cont = uni.minor.cont$uni[,1:3],
             minor.item = uni.minor.item$uni[,1:3])

 #now, a cleaner table, just the u values
 uni.sum.1<- data.frame(pure.cont=uni.pure.cont$uni[,1],
             pure.item=uni.pure.item$uni[,1],
            minor.cont = uni.minor.cont$uni[,1],
            minor.item = uni.minor.item$uni[,1])

matPlot(uni.sum.1, xlas=3 ,legend=3,
       main="Unidim as a function of number of items, factor loadings and item type")
```

```
#now try to get confidence intervals on the basic stats
# we do 100 runs for each condition

# We create a short function to simulate the data
#It uses the sim functions from psych
#also the reliability and unidim functions
#use 10 replications for testing purposes

simulate.unidim <- function(fx,n.obs=1000, n.replications = 10, keys=NULL,
           minor=FALSE, items=FALSE) {
        results <- list()
        uni.results <- list()
        vnames <- cs(omega.h ,alpha, omega.tot,  ECV, Uni, r.fit ,fa.fit ,n.items)
        if(is.null(keys)) {nvar <- 1} else {nvar  <-  length(keys)}

for (i in (1: n.replications)) {  # short loop
        if(minor) {sim.data <- sim.minor(fbig=fx, n=n.obs, n.small=4,items=items)
         } else {sim.data <- sim(fx,n=n.obs, items=items)}
        if(items) {R <- lowerCor(sim.data$observed, cor="poly", show=FALSE)$rho
         } else {R <- lowerCor(sim.data$observed,show=FALSE)}

        temp  <- reliability(keys,R,split=FALSE,n.obs=n.obs)$result.df
        uni.temp <- unidim(keys,R, n.obs=n.obs)$fa.stats
        results[[i]] <- temp
uni.results[[i]] <- uni.temp
        }  #end of loop


#organize the results
result.df <- matrix(unlist(results),byrow=TRUE,ncol=13*nvar)
uni.df <- matrix(unlist(uni.results), byrow=TRUE, ncol=9*nvar)
return(result.df)
}  # the end of our short simulation function


#now use this function to generate the replications of the data
set.seed(42)         #allows for reproducible results


 sim100 <- simulate.unidim(fx96,n.obs=100,n.replications=100,keys=keys)   # n = 100
 sim200 <- simulate.unidim(fx96,n.obs=200,n.replications=100,keys=keys)
  sim500 <- simulate.unidim(fx96,n.obs=500,n.replications=100, keys=keys)
  sim1k <- simulate.unidim(fx96,n.obs=1000,n.replications=100, keys=keys)
   sim5k <- simulate.unidim(fx96,n.obs=5000,n.replications=100, keys=keys)


 #now combine the u and omega_t estimates across data sets to graph them
#this is particularly clunky
 total.sims <- data.frame(N=c(rep(200,100), rep(500,100),
         rep(1000,100),rep(5000,100)),
      rbind(sim200[,41:80], sim500[,41:80], sim1k[,41:80], sim5k[,41:80]),
     rbind(sim200[,221:260], sim500[,221:260] , sim1k[,221:260], sim5k[,221:260]))

 #clean up the names for the graphics
temp <- gsub("omega.tot","",colnames(total.sims))
 temp <- gsub("Uni","u",temp)
  temp[18:21] <- c("F1+F2 6","F1+F2 10","F1+F2 12","F1+F2 24")
temp[38:41] <- c("F1+F2 6","F1+F2 10","F1+F2 12","F1+F2 24")
colnames(total.sims) <- temp

#n = 100
#figure not included for compactness
 error.bars.by(total.sims[,22:41],total.sims[,1],by.var=FALSE,
     v.labels=colnames(total.sims[2:21]),
      ylab="U",xlab="", las=3,main="B) U varies by sample size and factor loadings",
      legend=3,
      labels =c(200,500,1000,5000), lty=1:4,col =cs(black, blue,red, green))


error.bars.by(total.sims[,2:21],total.sims[,1],by.var=FALSE,v.labels=colnames(total.sims[2:21]),
    ylab=expression(omega[t]), las=3,xlab="",ylim=c(0,1),main=expression(paste("A)  ",
       omega[t]," varies by sample size and factor loadings")),
       legend=3,     labels =c(100,500,1000,5000), lty=1:4,col =cs(black, blue,red, green))

error.bars.by(total.sims2[,42:61],total.sims2[,1],by.var=FALSE, v.labels=names(keys),
    ylab="CFI", las=3,xlab="", ylim=c(0,1),
    main=" C) CFI is not very sensitive to variations in factor loadings",
    labels =c(200,500,1000,5000), lty=1:4,col =cs(black, blue,red, green))

 error.bars.by(total.sims2[,62:81],total.sims2[,1],by.var=FALSE,v.labels=names(keys),
    ylab="ECV",  las=3,xlab="",ylim=c(0,1),
```

```
    main=" D) ECV varies by sample size and factor loadings",
      labels =c(200,500,1000,5000), lty=1:4,col =cs(black, blue,red, green))


   #min n =200

     total.sims <- data.frame(N=c(rep(100,100), rep(500,100), rep(1000,100),rep(5000,100)),
      rbind(sim200[,41:80], sim500[,41:80], sim1k[,41:80], sim5k[,41:80]),
     rbind(sim200[,221:260], sim500[,221:260] , sim1k[,221:260], sim5k[,221:260]))

 error.bars.by(total.sims[,2:21],total.sims2[,1],by.var=FALSE,v.labels=names(keys),
    ylab=expression(omega[t]), las=3,xlab="",ylim=c(0,1),main=expression(paste("A)  ",
       omega[t]," varies by sample size and factor loadings"))

 #      legend=3,     labels =c(200,500,1000,5000), lty=1:4,col =cs(black, blue,red, green))

      error.bars.by(total.sims[,22:41],total.sims2[,1],by.var=FALSE,v.labels=names(keys),
      ylab="U",xlab="", las=3,main="B) U varies by sample size and factor loadings",
      labels =c(5000,1000,500,200), lty=1:4,col =cs(green, red, blue, black))


  #do legends separately to have more control

 error.bars.by(total.sims[,22:41],total.sims2[,1],by.var=FALSE,v.labels=names(keys),
      ylab="U",xlab="", las=3,main="B) U varies by sample size and factor loadings",
      labels =c(5000,1000,500,200), lty=1:4,col =cs(black, blue,red, green))
legend("bottomleft",lty=4:1,legend=c(5000,1000,500,200),col=cs(green, red, blue, black) ,pch=15)

#########
########'
```

Score the ability, bfi, and spi data sets

These data sets are in the *psychTools* package.

R code

```
library(psychTools)
#score the data from the real data sets
#
 results are in table 7
 #five single construct scales
bfi.keys.plus <-
  list(extraversion=c("-E1","-E2","E3","E4","E5"),
  openness = c("O1","-O2","O3","O4","-O5"),
  agree=c("-A1","A2","A3","A4","A5"),
  conscientious=c("C1","C2","C3","-C4","-C5"),
neuroticism=c("N1","N2","N3","N4","N5"),
#3 multiconstruct scales
EO = c("-E1","-E2","E3","E4","E5","O1","-O2","O3","O4","-O5"),
ACN =c("-A1","A2","A3","A4","A5","C1","C2","C3","-C4","-C5","N1","N2","N3","N4","N5"),
All = c("-E1","-E2","E3","E4","E5","O1","-O2","O3","O4","-O5","-A1","A2","A3","A4","A5",
      "C1","C2","C3","-C4","-C5","N1","N2","N3","N4","N5"))
bfi.rel <- reliability(bfi.keys.plus,bfi)




#add the ability results for table 7
rel.ab <- reliability(ability.keys,ability)
uni.ab <- unidim(ability.keys,ability,cor="tet", n.obs=1525)
ab.uni.rel <- data.frame(uni.ab$uni[,c(1:3)],rel.ab$result.df[,c(1:3,7:10)],
    uni.ab$uni[,c(7:8)],rel.ab$result.df[,11])
df2latex(ab.uni.rel)

rel.bfi<- reliability(bfi.keys,bfi)
uni.bfi <- unidim(bfi.keys, bfi, cor="poly")
bfi.uni.rel <- data.frame(uni.bfi$uni[,c(1:3)],rel.bfi$result.df[,c(1:3,7:10)],
   uni.bfi$uni[,c(7:8)],rel.bfi$result.df[,11])
df2latex(bfi.uni.rel)



EO <- c(bfi.keys$extraversion,bfi.keys$openness)
ACN <- c(bfi.keys$agree,bfi.keys$conscientious,bfi.keys$neuroticism)
bfi.plus <- list(extraversion = bfi.keys$extraversion,openness=bfi.keys$openness,
              agreeableness=bfi.keys$agree, conscientious=bfi.keys$conscientious,
              neuroticism = bfi.keys$neuroticism,
              EO = EO, ACN = ACN,
```

```
            All = c(EO, ACN))

rel.bfi<- reliability(bfi.plus,bfi)
uni.bfi <- unidim(bfi.plus, bfi, cor="poly")
bfi.uni.rel <- data.frame(uni.bfi$uni[,c(1:3)],rel.bfi$result.df[,c(1:3,7:10)],
          uni.bfi$uni[,c(7:8)],rel.bfi$result.df[,11])


#now, do the analyses for Table 8

spi.rel <- reliability(spi.keys,spi)
spi.uni <- unidim(spi.keys,spi,cor="poly")
spi.uni.rel <- data.frame(spi.uni$uni[,c(1:3)],spi.rel$result.df[,c(1:3,7:10)],
         spi.uni $uni[,c(7:8)], spi.rel $result.df[,11])
```

Simulation of multidimensional models

This next section simulates the data for figures 1-4. It takes considerable time and so using multi-cores will speed it up.

First we define two functions which then we call repeatedly.

R code

```
# a little function to take the output of the VSS function and grab just some of the output.
psych.summary.vss<- function(x) {

wh.max <- which.max(x$cfit.1)
wh.map <- which.min(x$map)
results <- list(VSS1=wh.max, MAP=wh.map)
return(results)
}

#the next code was
#developed June/July/September 2024 help test the uni and omega statistics

#requires psych_2.4.9 or newer

#first create a "small" function to do one case (for four levels of number of items)

sim.gen<- function(n.trials=10,n.obs=500, gload=NULL, fload=NULL, nfactors=3,
        rotate="oblimin",categorical=FALSE,minor=FALSE,VSS=FALSE) {
 if (is.null(gload))
        gload = matrix(c(0.9, 0.8, 0.7), nrow = 3)
if (is.null(fload)) fload <- matrix(c(0.8, 0.7, 0.6, rep(0, 9), 0.7, 0.6, 0.5, rep(0, 9),
      0.6, 0.5, 0.4), ncol = 3)

 fload.18 <- rbind(fload,fload)
 fload.27 <- rbind(fload,fload.18)
 fload.36 <- rbind(fload.18,fload.18)
#do this once
jen <- sim.hierarchical(gload=gload,fload=fload,n=n.obs)
R <- cor(jen$observed)
rel <-  suppressMessages(suppressWarnings(reliability(R,nfactors, n.obs=n.obs)))
cn <- colnames(rel$result.df)

results <- list()   #   We will store the results in a list

#define a local function to do the simulation
#this will be called repeateldy inside the sim.gen function

bigFunction <- function(ntrials,gload,fload, n.obs ,nfactors, categorical, rotate, minor, VSS) {
jen9 <- sim.hierarchical(gload=gload,fload=fload,n=n.obs,categorical=categorical,low=-2,
     high=2,minor=minor)
jen18 <- sim.hierarchical(gload=gload,fload=fload.18,n=n.obs,categorical=categorical,low=-2,
     high=2,minor=minor)
jen27 <- sim.hierarchical(gload=gload,fload=fload.27,n=n.obs,categorical=categorical,low=-2,
      high=2,minor=minor)
jen36 <- sim.hierarchical(gload=gload,fload=fload.36,n=n.obs,categorical=categorical,low=-2,
       high=2,minor=minor)

if(!categorical) {R9 <- cor(jen9$observed)
                  R18 <-  cor(jen18$observed)
                  R27 <-  cor(jen27$observed)
                  R36 <- cor(jen36$observed) } else {
                  R9 <- polychoric(jen9$observed,correct=0)$rho
                 R18 <- polychoric(jen18$observed,correct=0)$rho
                   R27 <- polychoric(jen27$observed,correct=0)$rho
```

```
                R36  <- polychoric(jen36$observed,correct=0)$rho

                  }
if(VSS) {  #do we want to do all the extra hassle of finding MAP, PA, etc.
   vss9 <- suppressWarnings(VSS(R9,n.obs=n.obs))
    vss18 <- suppressWarnings(VSS(R18,n.obs=n.obs))
     vss27 <- suppressWarnings(VSS(R27,n.obs=n.obs))
   vss36 <- suppressWarnings(VSS(R36,n.obs=n.obs))
   p9 <-  suppressWarnings(fa.parallel(R9,n.obs=n.obs))
    p18 <- suppressWarnings(fa.parallel(R18,n.obs=n.obs))
    p27 <- suppressWarnings(fa.parallel(R27,n.obs=n.obs) )
    p36 <- suppressWarnings(fa.parallel(R36,n.obs=n.obs))


   vss.result <-c(psych.summary.vss(vss9)$VSS1,psych.summary.vss(vss18)$VSS1,
          psych.summary.vss(vss27)$VSS1,psych.summary.vss(vss36)$VSS1)
    map.result <-c(psych.summary.vss(vss9)$MAP,psych.summary.vss(vss18)$MAP,
            psych.summary.vss(vss27)$MAP,psych.summary.vss(vss36)$MAP)
    pa.result <- c(p9$nfact, p18$nfact,p27$nfact,p36$nfact)
    pa.resultc <-  c(p9$ncomp, p18$ncomp,p27$ncomp,p36$ncomp)
   }
#combine the results from four passes
rel <-  suppressMessages(suppressWarnings(reliability(R9,nfactors,rotate=rotate,n.obs=n.obs)))
rel18 <-  suppressMessages(suppressWarnings(reliability(R18,nfactors,rotate=rotate, n.obs=n.obs)))
rel27 <-  suppressMessages(suppressWarnings(reliability(R27,nfactors,rotate=rotate,n.obs=n.obs)))
rel36 <-  suppressMessages(suppressWarnings(reliability(R36,nfactors,rotate=rotate,n.obs=n.obs)))
results <- t(matrix(unlist(rbind(rel$result.df,rel18$result.df,rel27$result.df,
      rel36$result.df)),ncol=16))
if(VSS)  results <-rbind(results,vss.result,map.result, pa.result,pa.resultc)
return(results)
} #end bigFunction
results  <- mcmapply(bigFunction,c(1:n.trials),MoreArgs=list(gload=gload,fload=fload,
        n.obs=n.obs,nfactors=nfactors, categorical=categorical, rotate=rotate,
          minor=minor, VSS=VSS))


#results is now a list of n.trials matrices of 16 or 20 x 4
if(VSS) {mat <- matrix(unlist(results),ncol=(length(cn)+4),byrow=TRUE)
    colnames(mat) <- c(cn, "VSS","MAT","P.fa","P.co")
       } else {mat <- matrix(unlist(results),ncol=length(cn),byrow=TRUE)
colnames(mat)  <- cn}
mat.df<- data.frame(n.obs=n.obs,mat)
return(mat.df)
}


#options(mc.cores=4)    #use multi-cores if available.


#
sim.gen(gload=gload,n.trials=n.trials,categorical=categorical, n.obs=1000,
    minor=minor, VSS=VSS))

#do it for 0 .. 1
set.seed(42)


  #use this function multiple times and the aggregate the results

       #compare smaller factor loadings

categorical <- TRUE
minor=TRUE
n.trials=100
set.seed(42)
gload<- NULL
VSS <- TRUE
if (minor)        fload <- matrix(c( 0.6, 0.5, 0.4, rep(0, 9), .5,.4,.3, rep(0, 9),
      0.5,.4,.3), ncol = 3) } else {minor <- NULL}


#do it for 0 .. 1
#this could probably be done in a loop, but easier to for one simulation run
set.seed(42)
gload <- rep(0,3)
v0.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
             n.obs=100, minor=minor, VSS=VSS)
v0.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
          n.obs=200, minor=minor, VSS=VSS)
v0.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
             n.obs=500, minor=minor, VSS=VSS)
```

```
system.time(v0.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
              n.obs=1000, minor=minor, VSS=VSS))

set.seed(42)
gload <- rep(.1,3)
v.1.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=100,
       minor=minor, VSS=VSS)
v.1.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=200,
      minor=minor, VSS=VSS)
v.1.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=500,
     minor=minor, VSS=VSS)
v.1.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,n.obs=1000,
          minor=minor, VSS=VSS)


set.seed(42)
gload <- rep(.2,3)
v.2.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=100,
      minor=minor, VSS=VSS)
v.2.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=200,
    minor=minor, VSS=VSS)
v.2.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=500,
    minor=minor, VSS=VSS)
system.time(v.2.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
         n.obs=1000, minor=minor, VSS=VSS))

set.seed(42)
gload <- rep(.3,3)
v.3.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=100,
     minor=minor, VSS=VSS)
v.3.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=200,
    minor=minor, VSS=VSS)
v.3.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=500,
      minor=minor, VSS=VSS)
system.time(v.3.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
        n.obs=1000, minor=minor, VSS=VSS))

set.seed(42)
gload <- rep(.4,3)
v.4.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=100,
     minor=minor, VSS=VSS)
v.4.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=200,
       minor=minor, VSS=VSS)
v.4.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical, n.obs=500,
      minor=minor, VSS=VSS)
system.time(v.4.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
        n.obs=1000, minor=minor, VSS=VSS))

set.seed(42)
gload <- rep(.5,3)
system.time(v.5.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
          n.obs=100, minor=minor, VSS=VSS))
system.time(v.5.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
           n.obs=200, minor=minor, VSS=VSS))
v.5.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
          n.obs=500, minor=minor, VSS=VSS)
system.time(v.5.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
        n.obs=1000, minor=minor, VSS=VSS))



set.seed(42)
gload <- rep(.6,3)
system.time(v.6.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
     n.obs=100, minor=minor, VSS=VSS))
system.time(v.6.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
      n.obs=200, minor=minor, VSS=VSS))
v.6.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
      n.obs=500, minor=minor, VSS=VSS)
system.time(v.6.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
        n.obs=1000, minor=minor, VSS=VSS))


set.seed(42)
gload <- rep(.7,3)
system.time(v.7.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
      n.obs=100, minor=minor, VSS=VSS))
system.time(v.7.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
         n.obs=200, minor=minor, VSS=VSS))
v.7.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
       n.obs=500, minor=minor,VSS=VSS)
```

```
system.time(v.7.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
        n.obs=1000, minor=minor, VSS=VSS))



set.seed(42)
gload <- rep(.8,3)
system.time(v.8.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
           n.obs=100, minor=minor, VSS=VSS))
system.time(v.8.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
  n.obs=200, minor=minor, VSS=VSS))
v.8.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
     n.obs=500, minor=minor, VSS=VSS)
system.time(v.8.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,categorical=categorical,
         n.obs=1000, minor=minor, VSS=VSS))

set.seed(42)
gload <- rep(.9,3)
system.time(v.9.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,
      n.obs=100,minor=minor, VSS=VSS,categorical=categorical,rotate="cluster"))
system.time(v.9.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,
         n.obs=200,minor=minor, VSS=VSS,categorical=categorical,rotate="cluster"))
v.9.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,
       n.obs=500,minor=minor, VSS=VSS,categorical=categorical,rotate="cluster")
system.time(v.9.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,
        n.obs=1000,minor=minor, VSS=VSS,categorical=categorical,rotate="cluster"))


set.seed(42)
gload <- rep(.95,3)
system.time(v.95.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,n.obs=100,
   minor=minor, VSS=VSS,categorical=categorical,rotate="cluster"))
system.time(v.95.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,n.obs=200,
        minor=minor,VSS=VSS,categorical=categorical,rotate="cluster"))
v.95.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,n.obs=500,
       minor=minor, VSS=VSS,categorical=categorical,rotate="cluster")
system.time(v.95.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,n.obs=1000,
         minor=minor, VSS=VSS,categorical=categorical,rotate="cluster"))

set.seed(42)
gload <- rep(1,3)    #100 percent g loading  which blows up omega_h unless we use rotate="cluster"
system.time(v1.100 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,n.obs=100,
          minor=minor, VSS=VSS,categorical=categorical,rotate="cluster"))
system.time(v1.200 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,n.obs=200,
          minor=minor, VSS=VSS,categorical=categorical,rotate="cluster"))
v1.500 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,n.obs=500,
minor=minor, VSS=VSS,categorical=categorical,rotate="cluster")
system.time(v1.1000 <- sim.gen(gload=gload, fload=fload,n.trials=n.trials,
           n.obs=1000,minor=minor, VSS=VSS,categorical=categorical,rotate="cluster"))


}

nr <- nrow(v.9.1000)

g <- c(rep(0,nr *4),rep(.1,nr *4),
       rep(.2,nr *4),rep(.3,nr *4),rep(.4,nr *4),
rep(.5,nr *4),rep(.6,nr*4),rep(.7,nr*4),rep(.8,nr*4), rep(.9,nr*4),
  rep(.95,nr*4),rep(1,nr*4))


#now, look at the lower value of fload
new.vss.sum.df <- rbind(v0.100,v0.200,v0.500,v0.1000,
           v.1.100,v.1.200,v.1.500,v.1.1000,
          v.2.100,v.2.200,v.2.500,v.2.1000,
          v.3.100,v.3.200,v.3.500,v.3.1000,
           v.4.100,v.4.200,v.4.500,v.4.1000,
          v.5.100,v.5.200,v.5.500,v.5.1000,
          v.6.100,v.6.200,v.6.500,v.6.1000,
          v.7.100,v.7.200,v.7.500,v.7.1000,
          v.8.100,v.8.200,v.8.500,v.8.1000,
             v.9.100,v.9.200,v.9.500,v.9.1000,
              v.95.100,v.95.200,v.95.500,v.95.1000,
          v1.100, v1.200, v1.500, v1.1000)



new.vss.sum.df <- data.frame(g,new.vss.sum.df)


#now draw the graphs for low fload
```

```
#these graphical functions are repeated multiple times with different parameters
#not shown for compactness


 error.bars.by(omega_h  ~ g +n.items , data=low.cat.sum.df,
    main="omega_h ~ g loading and number of items",
    ylim=c(0,1),
       v.labels =cs(0,.1,.2,.3,.4,.5,.6,.7,.8,.9,.95,1),
       xlab=" g loading",
       legend=1, lab=c(9,18,27,36),
       colors=cs(black,blue,green,red))
```