5-1-2017

An Unbiased Estimator Of The Greatest Lower Bound

Nol Bendermacher

Radboud University Nijmegen, Netherlands, Bendermacher@hotmail.com

Recommended Citation

An Unbiased Estimator Of The Greatest Lower Bound

Nol Bendermacher
Radboud University Nijmegen
Nijmegen, Netherlands

The greatest lower bound to the reliability of a test, based on a single administration, is the Greatest Lower Bound (GLB). However the estimate is seriously biased. An algorithm is described that corrects this bias.

Keywords: test reliability, greatest lower bound, GLB, unbiased estimate, capitalization on chance

Introduction

In classical test theory the concept of reliability refers to the precision of a measurement. In order to estimate the reliability of a test one needs two or more measurements applied to the same subjects. However, in many situations it is impossible to repeat a test administration under the same conditions. The next best thing is to estimate a lower bound to the reliability.

The current study is restricted to the reliability of tests that consist of a number of items and to the situation where the test is administered only once. The total score is the sum of scores on the individual items. According to classical test theory, the score x_{ij} of person i on item j consists of two parts: the true score τ_{ij} and an error component $\varepsilon_{ij} : x_{ij} = \tau_{ij} + \varepsilon_{ij}$. The error component includes not only real measurement errors but also the information that is unique to the item. It is assumed that these error components are uncorrelated with the true parts, as well as with each other. As a consequence the covariance matrix $\mathbf{\Gamma}$ of the items is the sum of two component matrices: the covariance matrix $\mathbf{\Gamma}_\tau$ of the true parts and the covariance matrix $\mathbf{\Gamma}_\varepsilon$ of the error components:

Nol Bendermacher is a retired member of the Research Technische OndersteuningsGroep (research technical support group). Email at bendermacher@hotmail.com.

$$\boldsymbol{\Gamma} = \boldsymbol{\Gamma}_\tau + \boldsymbol{\Gamma}_\varepsilon$$

The assumption of uncorrelated errors implies that $\boldsymbol{\Gamma}_\varepsilon$ is a diagonal matrix. Therefore the off-diagonal cells of $\boldsymbol{\Gamma}$ and $\boldsymbol{\Gamma}_\tau$ are identical.

The reliability of a test consisting of v items is defined as:

$$\rho_{tt} = 1 - \frac{\sigma_e^2}{\sigma_a^2} \tag{1}$$

where σ_e^2 is the sum of the error variances, and σ_a^2 is the variance of the test scores.

$$\sigma_e^2 = TR(\boldsymbol{\Gamma}_e) = \sum_{i=1}^{v} \boldsymbol{\Gamma}_{eii} \tag{2}$$

$$\sigma_a^2 = \sum_{i=1}^{v} \sum_{j=1}^{v} \boldsymbol{\Gamma}_{ij} \tag{3}$$

According to these definitions the formula of reliability can be rewritten as

$$\rho_{tt} = 1 - \frac{\sigma_e^2}{\sigma_a^2} = 1 - \frac{\sum_{i=1}^{v} \boldsymbol{\Gamma}_{eii}}{\sum_{i=1}^{v} \sum_{j=1}^{v} \boldsymbol{\Gamma}_{ij}} \tag{4}$$

The Greatest Lower Bound

From (4) it becomes clear, given the covariance matrix $\boldsymbol{\Gamma}$, that the reliability is maximal if the trace of the error covariance matrix $\boldsymbol{\Gamma}_e$ is minimal. As Jackson and Agunwamba (1977) remark, the only restrictions that the classical model imposes on the elements of $\boldsymbol{\Gamma}_\varepsilon$ are

(1) $0 \le \boldsymbol{\Gamma}_{eii} \le \boldsymbol{\Gamma}_{ii}$ (5)

(2) $\boldsymbol{\Gamma}_\tau = \boldsymbol{\Gamma} - \boldsymbol{\Gamma}_\varepsilon$ is non-negative definite

Therefore, if the set of values $\mathbf{\Gamma}_e$ can be found that maximizes its trace under these restrictions, the result is the smallest possible value for the reliability, given the covariance matrix $\mathbf{\Gamma}$. This value is the greatest possible lower bound to the reliability, called the GLB. Its possible values are restricted to the range [0,1]. A procedure to estimate it from a given covariance matrix is described in Ten Berge, Snijders and Zegers (1981).

A serious problem with the GLB is that it suffers from a phenomenon known as capitalization on chance: if it is estimated from a sample it tends to overestimate the population value. The bias increases with decreasing sample size and with lower values of the GLB; see Shapiro and ten Berge (2000). Moreover, the bias will be larger with a larger number of items.

To illustrate the seriousness of the problem: imagine a set of 40 items, completely uncorrelated and all with a unit normal distribution. Because the covariance matrix of these items is diagonal, the GLB for the test is zero. However, if samples of size 200 are drawn from the population, the average GLB-estimate from these samples is about 0.56.

Finding an unbiased estimator

Bendermacher (2010) describes an algorithm which reduces the bias in the estimated GLB by the use of a bootstrapping procedure. A large number of samples are drawn (with replacement) from the observed data with sample sizes equal to the size of the observed sample. For each sample the GLB is computed and the difference between the average of the sample-GLBs and the observed GLB is taken as an estimate of the bias. If this difference is subtracted from the observed GLB, the result is a less-biased estimate. The algorithm to be explained in this article starts in the same way, but it proceeds a few steps further and thereby manages to reduce the bias to a negligible quantity.

The algorithm tries to reconstruct the population covariance matrix $\mathbf{\Gamma}$ and then takes the GLB of this reconstructed matrix $\mathbf{G}_p$ as an unbiased estimator of the population GLB. The reconstruction is based on the following simple starting points:

1. The population-GLB β is smaller than the observed sample-GLB b_o. Theoretically this is incorrect (take for instance the case $\beta = 1$), but in almost all practical situations it will hold.

2. The population matrix Γ is similar to the sample covariance matrix $\mathbf{G}_o$.

3. If samples $\mathbf{G}_s$ are drawn from the reconstructed covariance matrix $\mathbf{G}_p$ (with the same size as the sample from which the observed matrix $\mathbf{G}_o$ was computed) their uncorrected GLB has as its expectation the observed value b_o.

The reconstruction of Γ will be called $\mathbf{G}_p$. It is built by adjustments to $\mathbf{G}_o$, which lower the value of its GLB. Because the three starting points still leave a considerable room in the exact way they are operationalized, several approaches were investigated, like adding error variances to the diagonal of $\mathbf{G}_o$, shrinking the off-diagonal cells, and reflecting some items to make their item-rest correlations negative. All these methods succeed in finding a covariance matrix that complies with the three starting points, but that does not mean by itself that the resulting GLB is an unbiased estimator. After some trial and error based on analyses of samples from two large real life data files, the following procedure appears to produce the best results by far:

1. Given the observed covariance matrix $\mathbf{G}_o$, compute the estimate $\mathbf{G}_t$ of Γ_τ with on its diagonal the minimal true variances and with its off-diagonals equal to those of $\mathbf{G}_o$. Example:

$\mathbf{G}_o=$				$\mathbf{G}_t=$			
6.4259	…	…	…	3.0717	…	…	…
3.0040	3.9210	…	…	3.0040	3.6019	…	…
1.5511	1.2191	5.0580	…	1.5511	1.2191	0.9501	…
1.2958	0.3373	1.0951	14.3406	1.2958	0.3373	1.0951	1.8588

    The GLB of $\mathbf{G}_o$ is $b_o = 0.5666$.

2. Multiply the diagonals of $\mathbf{G}_t$ by a factor $c \leq 1$. Call the resulting matrix $\mathbf{G}^*$.The rationale is that if Γ has a lower GLB than $\mathbf{G}_o$ its minimal true variances must be relatively smaller. How the factor c should be chosen will be explained later on.

    The example with $c = 0.69543$:

G*=

2.1358	…	…	…
3.0040	2.5045	…	…
1.5511	1.2191	0.6606	…
1.2958	0.3373	1.0951	1.2924

3. Due to its lowered diagonal elements, **G*** will have some negative eigenvalues.

   Compute the eigenvectors V and eigenvalues Λ of **G***, such that **G*** = $V\Lambda V^T$. Example:

   Λ=

6.4570	…	…	…
…	1.4324	…	…
…	…	-0.3546	…
…	…	…	-0.9415

4. Replace the negative eigenvalues of **G*** by zeros and add their (negative) values to the smallest non-negative eigenvalues without letting them become negative. Call the result Λ^*. Example:

   Λ^*=

6.4570	…	…	…
…	1.3630	…	…
…	…	0.0000	…
…	…	…	0.0000

5. Compute **G*** = $V\Lambda^* V^T$; its trace will be $c.TR(\mathbf{G}_t)$

6. Complete the reconstruction of the population matrix by replacing the diagonal of **G*** by that of **G**$_o$: $\mathbf{G}_p$ = **G*** − DIAG(**G***) + DIAG(**G**$_o$). Example:

   G$_p$=

6.4259	…	…	…
2.5760	3.9210	…	…
1.4686	1.4016	5.0580	…
1.1435	1.0547	0.6671	14.3406

7.   Compute the GLB b_p of $\mathbf{G}_p$. This is the corrected estimate of the population GLB. In the example, $b_p = 0.5005$.

There remains a crucial question: what is the correct value of the factor c in step 2 of the above procedure. The answer is based on the third starting point. The factor c must be chosen such that the expected GLB of samples from $\mathbf{G}_p$ is equal to the observed GLB b_o. This means that one can start from a well chosen guess c, compute $\mathbf{G}_p$ and perform a bootstrapping run in which a large number of samples matrices $\mathbf{G}_{si}$ are drawn from $\mathbf{G}_p$.

The average b_s of the sample GLB-values, as compared to the observed GLB b_o, is used to update the choice of c, and the process is repeated until the correct value has been found. More details are given in the section Algorithm. This procedure requires several bootstrapping runs, each generating a vast number of samples. Therefore it is important to have an efficient algorithm that keeps the number of bootstrap runs at a minimum.

Drawing samples from a covariance matrix

How a sample covariance matrix can be derived from a population matrix without knowing the underlying raw data will now be explicated. The algorithm requires covariance matrices based on samples from the data from which $\mathbf{G}_o$ is computed. If these data are available one might actually draw such samples and compute covariance matrices from them. However, because the algorithm implies a number of bootstrapping runs, with a large amount of samples for each run, such a procedure would be very time consuming. Moreover, the algorithm also requires sampling from modified covariance matrices for which no raw data are available. Fortunately it is possible to compute these sample covariance matrices directly from the observed or constructed covariance matrix and the given or assumed distributions of the items.

If a sample of raw data is given, estimates of the distributions of the items can be derived from that sample. If no information is available about the distributions of the items one may assume a multivariate normal distribution.

Sampling from a given $v \times v$ covariance matrix $\mathbf{G}$ with sample size k can be performed as follows:

1.   Compute, by Cholesky triangularization, a matrix $\mathbf{C}$ such that $\mathbf{C}\mathbf{C}^T = \mathbf{G}$.

2.   Generate k times a vector of v independently chosen random drawings using the distributions of the v items. Compute the covariance matrix $\mathbf{G}_z$ from these vectors, as if they were observed cases.

3.   Compute a matrix $\mathbf{G}^*$ by dividing each cell of $\mathbf{G}_r$ by the standard deviations of the two items involved: $\mathbf{G}^*_{ij} = \dfrac{\mathbf{G}_{zij}}{s_i s_j}$.

4.   Compute the sample matrix as $\mathbf{G}_s = C\mathbf{G}^* C^T$

The average of the GLB-values of the matrices $\mathbf{G}_z$ (see step 2) gives an estimate of the expected sample GLB b_z under the null hypothesis that $\mathbf{G}_p$ has GLB-value zero. If the observed GLB (b_o) is clearly less than b_z the corrected estimate b_p can immediately be set to zero.

If one assumes a multivariate normal distribution of the items, the v independently chosen drawings mentioned in step 2 can be drawings from a unit normal distribution. To speed up the program one may construct in advance a long list (say 4000 numbers) of drawings from a unit normal distribution by taking equally spaced values between 0 and 1 and computing the inverse of the cumulative normal distribution function for them. Sampling from a unit normal distribution then comes down to randomly choosing from this list, using a uniform random generator.

Algorithm

This description of the algorithm uses the following definitions:

Go   the observed covariance matrix
Gp   the current reconstruction of $\boldsymbol{\Gamma}$
bo   the GLB of Go
bp   the GLB of Gp, i.e. the provisional estimate of β
bs   the average GLB of the samples from the most recent bootstrap run
bz   the average of the GLB-values of samples simulated under the null hypothesis of uncorrelated items
bt   the intended GLB-value for an updated reconstruction Gp

The algorithm consists of the following steps:

Step 1: Choose a precision criterion `Precision`; 0.001 will do well.

Choose `MaxSteps` = the maximum number of steps in the main algorithm; suggested value: 100.

Set `CurrentPrecision` = `Precision` × 5; set `ShrinkFactor` = $0.2^{1/5}$

`ShrinkFactor` will be used to decrease `CurrentPrecision` in five steps towards `Precision`.

Step 2: Perform a bootstrap run in which samples are drawn from `Go` until the standard error of the mean of sample GLB-values is less than `CurrentPrecision` or a maximum number of samples is drawn.

The main results are: bz, bs and `Significance`. `Significance` gives the proportion of samples generated under the null hypothesis of uncorrelated items with a GLB-value greater than bo.

Step 3: If $bo < bz \times 0.9$ or `Significance` ≥ 0.5, then set `Bestbp` = 0 and go to step 16

Step 4: Initialize some variables: `BestDiff` = 9, `Bestbp` = bo, `BestCount` = 0, `Count` = 0

Step 5: Find successive new versions of the reconstructed population matrix `G`p by repeating steps 6-15

Step 6: Increase `Count`; If `Count` > `MaxSteps` go to step 16

Step 7: Find a new bt:

If $bs \leq bo$ then

set `LowLim` = $\text{MIN}(bs, bp)$

set `UppLim` = $\text{MAX}(\text{LowLim}, \text{UppLim})$

set bt = (`LowLim` + `UppLim`) / 2

else perform steps 7a - 7d

Step 7a: Set `UppLim` = bp

Step 7b: Set LowLim $=$ MIN(LowLim,UppLim)

Step 7c.   Find a second order polynomial $y = f(x)$ through the points $(x,y) = (bz,0)$, (bs,bp) and $(1,1)$

and find $bt = f(bo)$.

Compute the predictor matrix $\mathbf{P}$ and the criterion vector $\mathbf{Q}$:

$$\mathbf{P} = \begin{bmatrix} b_z^2 & b_z & 1 \\ b_s^2 & b_s & 1 \\ 1 & 1 & 1 \end{bmatrix}; \mathbf{Q} = \begin{bmatrix} 0 \\ b_p \\ 1 \end{bmatrix}$$

If $\mathbf{P}$ is singular set
$bt = $ MIN$(1,$MAX$(0,bp - (bs - bo) \times 1.2)$

else compute the weights $W = \mathbf{P}^{-1}\mathbf{Q}$ and set
$bt = W_1 b_o^2 + W_2 b_o + W_3$

Step 8.   IF Count $= 1$ set $bt=$ MIN$(bt,0.95)$

Step 9.   Find a new estimate Gp such that its GLB bp is close enough to bt, i.e. until ABS$(bp{-}bt) <$ CurrentPrecision) or a maximum of steps is taken.

Compute the GLB bp of Gp. The details of this step are described later.

Step 10.   Perform a bootstrap run and compute the average value bs of the sample GLB's.

Step 11.   Compute Diff $=$ ABS$(bs{-}bo)$

If Diff $<$ BestDiff then

set BestDiff $=$ Diff; set Bestbp $=$ bp; and set BestCount $=$ Count

Step 12.   If Diff $\leq$ CurrentPrecision then

If CurrentPrecision $=$ Precision go to step 16

else set CurrentPrecision $=$ Precision

Step 13.   If `BestCount` $\leq$ `Count` $- 5$ then

            If `CurrentPrecision` $=$ `Precision` go to step 16

            else set `CurrentPrecision` $=$ `Precision`

Step 14.   If `Count` < 4
            set `CurrentPrecision` $=$ `ShrinkFactor` $\times$ `CurrentPrecision`

            If `Count` $= 4$ set `CurrentPrecision` $=$ `Precision`

Step 15.   Go back to step 6

Step 16.   Set $bp = \text{MAX}(0, \text{MIN}(\text{Bestbp}, 1)$

Step 17.   Now bp is the final value of the corrected GLB

     Some explanations:

at Step 1:   The algorithm may be very time consuming. Therefore the required precision is varied from 5 times `Precision` in the first cycle to `Precision` in the fifth and following cycles.

at Step 9:   The factor c and the corresponding matrix `Gp` can be found by the following algorithm:

         Step 9a.   Set `Lowc` $= 0$; Set `Highc` $= 1$; set `Lowb` $= bz$; set `Highb` $= bo$

         Step 9b.   Repeat steps 9c through 9h

         Step 9c.   Set `Midc` $=$ (`Lowc` $+$ `Highc`)/2

                      If $\text{ABS}($`Highb` $-$ `Lowb`$) <$ `CurrentPrecision` go to step 9i

         Step 9d.   Copy `Go` to `Gp`

         Step 9e.   If `MidC` $\geq 1 -$ `Precision` set `Midb` $= bo$

                      else ... (steps 9f through 9h)

Step 9f. Replace the diagonal of Gp by $Midc$ times the vector of minimal true variances of Go

Compute the eigenvectors V and the diagonal matrix Λ with eigenvalues of Gp

Step 9g. Set $T_1 = TR(Gp)$; set T_2 = sum of the negative eigenvalues in Λ.

Replace the negative eigenvalues by zero.

Loop over the positive eigenvalues λ_i from smallest to greatest:

If $\Lambda_{i,i} \geq T_2$ then set $\Lambda_{i,i} = \Lambda_{i,i} - T_2$ and continue with step 9h

else set $T_2 = T_2 - \Lambda_{i,i}$ and set $T_2 = 0$; continue the loop over the eigenvalues

Step 9h. Recompute $Gp = V\Lambda V$ with the adjusted eigenvalues given by Λ

Replace the diagonal of Gp by that of Go and compute its GLB bp.

Set $Midb = bp$

Step 9i. If ABS($bt - Midb$) < CurrentPrecision go to step 9k

If a maximum (e.g. 30) number of cycles (9c through 9h) is taken go to step 9k

If $bt < MidB$ set HighC = MidC

else set LowC = MidC

Step 9j. Go back to step 9c

Step 9k. Now Gp is the wanted matrix with its GLB bp close to bt.

Border effects

The correction procedure as it was specified above may fail for extreme observed GLB-values b_o. For low values, there may be no population matrix possible with b_o as its expected sample value. This happens if the observed GLB is lower than the expected sample value b_z under the null hypothesis $b_p = 0$. In such cases the corrected estimate can immediately be set to 0. For high values of b_o, the problem is not that easy to be solved. If the observed GLB b_o is (almost) 1, the estimator $b_p = 0.99...$ complies with the three starting points, but samples from a population with a lower value might as well have a GLB equal to or close to 1. In such cases the algorithm may erroneously overestimate the population GLB.

Evaluating the estimation procedure

In order to test the quality of the above procedure several large datasets were downloaded (personality-testing.info, n.d.), not including the files used in the trial and error phase. From each of these datasets one or more tests were selected and from each test 100 or 50 samples were taken, consisting of randomly chosen cases. Cases with missing values were not allowed to enter the samples.

As a result several sets were available each consisting of a large population and 100 or 50 samples extracted from it. The mean of the corrected GLB-values over the samples renders an estimate of the expected value of the corrected GLB. If the correction algorithm works correctly, these expected corrected GLB's should be (almost) equal to their corresponding population values. The tests were taken from the following data collections:

1. 16PF, test 1, items A1-A10, ordinal scores (1-5), 49159 cases

2. 16PF, test 2, items B1-B13, ordinal scores (1-5), 49159 cases

3. 16PF, test 3, items C1-C10, ordinal scores (1-5), 49159 cases

4. ECR, items Q1-Q36, ordinal scores (1-5), 17386 cases

5. MSSCQ, items Q1-Q100, ordinal scores (1-5), 17685 cases

Table 1 summarizes the main results, with column definitions as follows:

test	name of the test
# files	number of sample files taken from the large population file
v	test length
n	sample size
β	the GLB-value computed from the large population file; the average b_p (in column 7) should be close to this value
b_o	the mean of the uncorrected observed GLB-values from the sample files
b_p	the mean of the corrected GLB-values from the sample files; it should be close to the population value β
b_z	the mean of the expected GLB-values under the null hypothesis of uncorrelated items
SE(b_p)	the standard error of the mean of the corrected GLB-values
duration	the average time (mm:ss) needed to analyze a single sample file on a basic desk top computer

Table 1. Results of the testing procedure.

test	# files	v	n	β	b_o	b_p	b_z	SE(b_p)	duration
16PF_1	100	10	100	0.6716	0.7559	0.6791	0.3389	0.0075	0:02
16PF_2	100	13	200	0.5581	0.6410	0.5571	0.3099	0.0084	0:06
16PF_3	100	10	500	0.4404	0.4722	0.4373	0.1671	0.0060	0:07
ECR	100	32	100	0.9016	0.9601	0.9052	0.6889	0.0023	1:11
ECR	100	32	200	0.9016	0.9410	0.9044	0.5184	0.0018	1:02
ECR	100	32	500	0.9016	0.9247	0.9072	0.3543	0.0010	1:05
ECR	100	32	1000	0.9016	0.9142	0.9011	0.2545	0.0006	1:09
MSSCQ	50	100	100	0.9675	0.9986	0.9834	0.9725	0.0012	53:36
MSSCQ	50	100	200	0.9675	0.9924	0.9782	0.8406	0.0008	28:20
MSSCQ	50	100	500	0.9675	0.9828	0.9712	0.6138	0.0006	24:44
MSSCQ	50	100	1000	0.9675	0.9772	0.9711	0.4651	0.0003	19:45

The result of these tests strongly suggest that the chosen algorithm reduces the bias in the GLB to a negligible quantity. However, the procedure becomes laborious when the observed GLB is close to unity. It should also be noticed that the expected GLB under the null hypothesis of uncorrelated items (b_z) may become extremely high when the ratio v/n is almost 1.

The assumption of multivariate normality of the items

Above, all scales consisted of ordinal items with a small set of possible scores and their distributions could be estimated from the observed data. If only a covariance matrix is available without information about the distribution of the item scores, one might fall back on the assumption of multivariate normality, but this assumption will frequently be incorrect. In order to get an impression of the seriousness of violations of this assumption, the tests described in the previous section were repeated, now replacing drawings from the actual distributions by drawings from normal distributions. The results are given in Table 2. These results suggest an analysis based on the assumption of multivariate normality will deliver a correct estimator of the GLB, even if the assumption is incorrect.

Table 2. Results using actual distributions and results assuming multinormality.

test	β	Actual Distributions		Normal Distributions	
		b_p	b_z	b_p	b_z
16PF_1	0.6716	0.6791	0.3389	0.6710	0.3397
16PF_2	0.5581	0.5571	0.3099	0.5539	0.3106
16PF_3	0.4404	0.4373	0.1671	0.4353	0.1645
ECR	0.9016	0.9052	0.6889	0.8923	0.6873
ECR	0.9016	0.9044	0.5184	0.9034	0.5206
ECR	0.9016	0.9072	0.3543	0.9066	0.3534
ECR	0.9016	0.9011	0.2545	0.9011	0.2549
MSSCQ	0.9675	0.9834	0.9725	0.9498	0.9576
MSSCQ	0.9675	0.9782	0.8406	0.9723	0.8403
MSSCQ	0.9675	0.9712	0.6138	0.9704	0.6142
MSSCQ	0.9675	0.9711	0.4651	0.9707	0.4659

Table 3. Distributions (proportions) of the 10 items in scale 16PF.

	Items									
	1	2	3	4	5	6	7	8	9	10
Score 1	0.0425	0.0238	0.0350	0.0316	0.0203	0.0185	0.0221	0.0729	0.2560	0.1771
Score 2	0.1163	0.0748	0.0893	0.1087	0.0678	0.0753	0.0699	0.3092	0.4727	0.4053
Score 3	0.1511	0.1787	0.1337	0.1664	0.1767	0.2588	0.1285	0.2486	0.1420	0.2340
Score 4	0.4764	0.4749	0.4771	0.5267	0.4964	0.4732	0.5469	0.2770	0.0997	0.1500
Score 5	0.2137	0.2479	0.2648	0.1666	0.2387	0.1742	0.2326	0.0922	0.0296	0.0336

As an illustration, Table 3 shows the distributions of the items of the population 16PF. The scores are clustered into only 5 categories and the

distribution over these categories is different for the individual items. Nevertheless the estimation of the GLB remains practically unbiased.

Conclusion

It is clear that under the assumptions of the classical test theory and without additional assumptions, the measure known as the Greatest Lower Bound (GLB) is the highest possible lower bound to the reliability of a test. Unfortunately the use of this measure is severely hindered by its bias for small or even moderate samples. It is possible to remove this bias by the given algorithm.

The ideas of this article are implemented in a program called GLBFind, which is available at http://www.ru.nl/socialewetenschappen/rtog/software/statistische/kunst/glbfind/.

References

Bendermacher, N. (2010). Beyond alpha: lower bounds for the reliability of tests. *Journal of Modern Applied Statistical Methods, 9*(1), 95-102.

Jackson, P. H., & Agunwamba, C. C. (1977). Lower bounds for the reliability of the total score on a test composed of non- homogeneous items I: Algebraic lower bounds. *Psychometrika, 42*(4), 567-578. doi: 10.1007/bf02295979

personality-testing.info. (n.d.). Raw data from online personality tests. Retrieved from http://personality-testing.info/_rawdata/

Ten Berge, J. M. F., Snijders, T. A. B., & Zegers, F. E. (1981). Computational aspects of the greatest lower bound to the reliability and constrained minimum trace factor analysis. *Psychometrika, 46*(2), 201-213. doi: 10.1007/bf02293900

Shapiro, A., & ten Berge, J. M. F. (2000). The asymptotic bias of minimum trace factor analysis with applications to the greatest lower bound to reliability. *Psychometrika, 65*(3), 413-425. doi: 10.1007/bf02296154