Basic Psychometrics: An R Tutorial

Bob Gore, Ph.D.
Moffitt Cancer Center

September 2021

1 Introduction

Measurement is the foundation of quantitative science. Without solid measures of the variables we study, the conclusions of our research are likely to be shaky. Psychologists have devoted monumental effort to the development of statistical methods to ensure that measures of human behavior are *reliable* and *valid*. This tutorial will go through some basic concepts that are essential to effective use of behavioral measures. After that, an exercise will be presented in which you can practice with a public dataset. The goal of this tutorial is to provide a ready reference for staff statistical analysts who may not have a background in psychology yet must analyze psychological test data.

2 Basic Conceptual Tools

2.1 Validity

Suppose you ask a group of people to record all of the food they eat for a week. The measure of interest might be the number of calories consumed during the week. If your measure is valid, then the responses of your participants should be accurate. For example, you should get the same number of calories regardless of whether targeted participants record their own food consumption or their behavior is observed by another person and recorded by that observer. Of course this assumes that the observer report is also valid. Two valid measures should give essentially the same results.

While the establishment of the validity of a measure will rarely be your task, the procedure is generally as follows:

- Obtain *gold standard* measures of the variable of interest
- Obtain the measure you want to validate, using the same observational units
- Calculate a correlation coefficient between the two sets of measures

This procedure for validation has an obvious flaw, one that is not well known to the psychological researchers who use it. If two measures should truly give exactly the same set of numbers, then they should not merely correlate, but the two sets of measures should give identical results. If a vial of blood was divided into two samples and sent to the same laboratory in two different test tubes requesting the same laboratory test, the numbers so obtained should be exactly the same. A correlation coefficient is not sensitive to changes in scale, however. If a laboratory begins to add a constant to every value or multiply every value by a constant (through some process problem), the correlation coefficient alone would not detect the problem. Another technique, such as Lin's Concordance coefficient, would be needed. In such cases, the intraclass correlation coefficient can also be used.

In most research, the PI will already have chosen measures with well-established validity. However, there are times when the PI will not have been able to do so. If there is no gold standard measure of validity to which things can be compared, the next best thing is to engage in what is called *construct validation*, in which an overall pattern of intercorrelations between the target measure of interest and other measures is examined to see if it makes sense. A correlation matrix, which might include a mix of Pearson's, Spearmans, Tetrachoric, and other correlation measures (such as phi) might be inspected. Ideally, you would use such a correlation matrix to show two things:

- Your measure of interest has positive correlations with other variables where positive correlations are expected (convergent validity)
- Your measure of interest has negative correlations where negative correlations are expected (convergent validity)
- Your measure of interest has near zero correlations which measures that are not supposed to be correlated with it (known as discriminant validity)

Where there is no gold standard measure of a construct but where the domain of possible test items is easy to sample from, a second approach is to use a content validation approach.

For example, if a researcher wanted to measure the ability to add a pair of two digit numbers, there are a limited number of possibilities (the numbers 10 to 99). By randomly sampling from the definable set of numbers, the researcher can develop a measure that has content validity.

A second example comes from medical research. If a medical illness has a well defined set of signs and symptoms, then a measure of that illness can be developed by writing a number of items that should logically assess each of the signs and symptoms. This is often how psychiatric questionnaires are developed to measure mental health related constructs (i.e., psychological variables that cannot be directly observed).

A final form of validity that is often misunderstood is so-called *face validity*. This concept arose out of settings where job applicants take psychological tests to measure such attributes as integrity, honesty, dependability, and the ability to get along with others. Some tests have been developed with tricky questions, such as "I would enjoy the work of a bodyguard." Such tricky questions were included on some early measures of psychopathology in order to try to find ways to assess mental health problems that people might be trying to conceal, as well as to detect malingered mental health problems that the test-taker does not really have. When these measures were taken into the workplace setting, concerns were raised about how the job applicant responds to such items, and particularly whether the job applicant finds them off putting. Face validity only has to do with whether the test-taker can see what a question item is trying to measure. It is not so much a form of validity as a form of maintaining good rapport with the test-taker. It is often confused with content validity, described above.

2.2 Reliability

While validity is hard to establish because in many domains, there is no gold standard measure or easy way to define the universe of possible test items that could be written, reliability is easy to establish. It is based on straightforward, practical statistics and simple, inexpensive research designs.

Reliability addresses the question as to whether a group of psychological test items shares a common source of variation, relatively free from random error. Consider the following purely hypothetical set of items that might assess attitudes toward smoking cigarettes:

> For each item below, answer on the following scale (write the appropriate number next to each test item):
>
> | Strongly Agree | Agree | Neither Agree nor Disagree | Disagree | Strongly Disagree |
> |---|---|---|---|---|
> | 1 | 2 | 3 | 4 | 5 |

1. Smoking would greatly increase my chances of getting cancer
2. If I were to quit smoking today, my risk of getting cancer would be lower than if I continued smoking
3. Smoking does not really raise people's risk of getting cancer

What researchers need to show in order to demonstrate the reliability of a scale such as the one above is that the three items are driven by a common source of variation, and that they are relatively free of nuisance sources of variation, such as the mood they are in at the moment they answer the items. What *nuisance* means in a particular study depends on the researcher's hypotheses. As a result, it is somewhat inaccurate to describe a measure as having well established reliabilty *in general*.

The most efficient way to show reliability is to calculate a coefficient, such as Cronbach's alpha (for items such as those above which are analyzed as if they are on an interval or ratio scale and as if there is an underlying continuous psychological variable being assessed by them). Be careful, when the items are dichotomous (true/false, correct/wrong) to use a technique such as KR-20 that is designed for dichotomous measures.

What these measures effectively do is to estimate the amount of variation in the *observed* test score (the sum across the items) that is attributable to the common source of variation, called *true score*. A high number (preferably above .80) demonstrates acceptable *internal consistency reliability*.

Consider the following hypothetical dataset (which has been generated to have known characteristics):

SubjectID	Q1	Q2	Q3
1	1	2	5
2	1	1	4
3	1	1	5
4	3	1	5
5	2	3	3
6	2	3	3
7	4	3	3
8	4	5	4
9	5	5	2
10	5	5	1

The dataset above is provided in the R file that is supplied for your use.

The `ltm` package in R has a function to compute Cronbach's alpha with a confidence interval. More information can be found at:

`https://www.statology.org/cronbachs-alpha-in-r/`

You may be puzzled about the fact that the three items provided above are on an ordinal scale. Why do we not use methods designed specifically for ordinal items? This can be done (using item response theory, if sample sizes are quite large) but typically is not done for reasons of how psychology developed as a field.

When we add up the item scores to form a total score, we are relying on a tacit assumption that the true score has the same influence on responses to each of the three items above. This assumption can only be supported through the use of a latent variable modeling technique such as confirmatory factor analysis. This is because the common source of variation in the items, true score, is not directly observable (and so, latent).

The function for the calculation of Cronbach's alpha is `cronbach.alpha()`, as follows:

```
> cronbach.alpha(dataset,CI=TRUE)

Cronbach's alpha for the 'dataset' data-set

Items: 4
Sample units: 10
alpha: 0.479

Bootstrap 95% CI based on 1000 samples
 2.5% 97.5%
0.187 0.712
```

Surprisingly, Cronbach's alpha is unacceptably low at .479. What went wrong? It's a good idea to start by inspecting a correlation matrix. Note that our variables (Q1, Q2, and Q3) are in columns 2-4 of the data frame.

```
> cor(dataset[,2:4])
           Q1         Q2         Q3
Q1  1.0000000  0.8167883 -0.7094635
Q2  0.8167883  1.0000000 -0.7646982
Q3 -0.7094635 -0.7646982  1.0000000
```

Now we can see the problem. If you have set up your data correctly, all of the correlation coefficients should be positive. However, Q3 has negative correlations with Q1 and Q2. If you review the wording of that item, you can see why. We need to recode Q3 so that it runs in the correct direction. To do this, we subtract all of the scores from 6 (because 5 is the highest possible value and 1 is the lowest). If M is the maximum possible value of an item and m is the minimum possible value (not the minimum in the dataset, but the theoretical minimum), then the general equation to reverse score an item such as Q3 is:

$$ReversedItemScore = M + m - ItemScore$$

For convenience, in the R code associated with this exercise, I have taken the dangerous step of recoding Q3 without saving it in a new variable. This is a bad idea. It would have been better to create a new variable, such as Q3r, and put the reverse coded information there.

```
> dataset$Q3 = 6-dataset$Q3
> cor(dataset[,c(2:4)])
          Q1        Q2        Q3
Q1 1.0000000 0.8167883 0.7094635
Q2 0.8167883 1.0000000 0.7646982
Q3 0.7094635 0.7646982 1.0000000
> cronbach.alpha(dataset,CI=TRUE)

Cronbach's alpha for the 'dataset' data-set

Items: 4
Sample units: 10
alpha: 0.915

Bootstrap 95% CI based on 1000 samples
 2.5% 97.5%
0.828 0.954
```

Although not provided here, it would be wise to review scatterplots to ensure that the relationships you see are basically linear and don't show evidence that missing values have been coded in such a way as to create high artifact correlations).

Now we see an appropriately high alpha value (.915) whose entire 95% CI is above .80. This is an ideal scale, according to the results so far, but in general if you at least get a Cronbach's alpha above .80 you can argue that a scale has good internal consistency reliability.

2.3 Confirmatory Factor Analysis

Researchers generally compute scale scores by simply adding or averaging the individual item responses. As we saw just now, two simple ways to get this wrong are to add up items that have been reverse coded and to add items where missing values are coded to a number (such as 999). There is no substitute for a common sense review of item content and basic descriptive statistics, correlations, and scatterplots.

The R package `lavaan` supports latent variable modeling. If you want to go Bayesian, you can use `blavaan`. There are some very good examples of how to use lavaan to do this kind of analysis here:

`https://github.com/janlammertyn/lavaan-material/blob/master/brown-cfa/allExamples.R`

For researchers who wish to simply average a set of items to create a summary scale score, one of two models needs to hold: the strictly parallel model, or the tau-equivalent model. In the strictly parallel model, every item is influenced to the same extent by true score, and to the same extent by error. In the tau equivalent model, every item is influenced to the same extent by true score, but items can vary in terms of how much influence errors have on observed item responses. There are three basic steps in a lavaan analysis:

1. Specify your model of how the latent variable influences the observed scores
2. Request a confirmatory factor analysis of the model you have specified
3. Review the fit statistics

In a particularly strong analysis, the analyst will partition the sample into a sample for developing a sound psychometric model and a second partition for cross-validation. All of the exploratory analysis which are not driven by a theoretical model will be conducted on the first partition. After an approach has been developed, it should be applied exactly to the cross-validation partition of the sample to see whether it fits reasonably well there. If such a procedure is not followed, and the researcher instead makes a number of ad hoc changes based on statistics, the resulting scale may only work well in the particular sample in which these analyses have been done. This is the classic problem of *overfitting*, the easiest solution to which is cross-validation.

Here is the `lavaan` code to use for the strictly parallel model:

```
# strictly parallel model
# specify the model
model.parallel <- '
 trueScore =~ v1*Q1 + v1*Q2 + v1*Q3
 Q1 ~~ v1*Q1
```

```
 Q2 ~~ v1*Q2
 Q3 ~~ v1*Q3
'
# fit the model
fit.parallel <- cfa(model.parallel,data=dataset)
summary(fit.parallel,fit.measures=TRUE)
# examine the residuals
resid(fit.parallel)
# examine the modification indices
modindices(fit.parallel)
```

Here is the code for the tau equivalent model:

```
# tau equivalent model
# specify the model
model.tauEquivalent <- '
 trueScore =~ v1*Q1 + v1*Q2 + v1*Q3
'

# fit the model
fit.tauEquivalent <- cfa(model.tauEquivalent,data=dataset)
summary(fit.tauEquivalent,fit.measures=TRUE)
# examine the residuals
resid(fit.tauEquivalent)
modindices(fit.tauEquivalent)
```

Notice the diference: in the parallel model (but not in the tau equivalent model) you will see the following lines:

```
 Q1 ~~ v1*Q1
 Q2 ~~ v1*Q2
 Q3 ~~ v1*Q3
```

These are the lines that ensure that `lavaan` knows to estimate a model with equal error variances across the items. Because that is not an assumption of the tau equivalent model, we don't include that code for testing tau equivalence.

Here is how to interpret the elements of the model statement for the parallel model.

```
model.parallel <- '    # create a container called model.parallel
                       # and fill it with the contents between single quotes
 trueScore =~ v1*Q1 + v1*Q2 + v1*Q3 # trueScore is the latent variable,
                                    # and it is indicated (=~) by Q1, Q2, and Q3
                                    # v1 is the weight to estimate
                                    # (weight to place on trueScore when predicting Q1, Q2, Q3)
 Q1 ~~ v1*Q1 # Q1 covaries with (~~) v1*Q1 # which is really a latent variable contribution.
 Q2 ~~ v1*Q2 # Q2 covaries with (~~) v1*Q2 # in practice these statements force equal errors
 Q3 ~~ v1*Q3 # Q3 covaries with (~~) v1*Q3 # across the three variables Q1, Q2, Q3
'    # end quotation to tell R you are finished with the model specification.
```

2.4 Evaluation of Model Fit

Ideally, you will find a model where the strictly parallel or tau equivalent model fits well. The main things to look at are RMSEA, CFI, and TLI. Here are the guidelines commonly used:

1. RMSEA ideally should be below .10, and most authors recommend RMSEA $< .05$. High is bad when it comes to RMSEA. The statistic - a modified non-central chi-square - is based on the divergence between the observed covariance matrix and the one that would be observed if your model were true.
2. TLI and CFI: ideally both of these should be above .95. Higher values are better. The possible range is 0 to 1. Values outside of that range are cause for concern about whether your model is being estimated appropriately.

2.5 Output for the Strictly Parallel Model

In the output below, the main features to examine are these:

1. Is the number of observations correct?
2. Is the number of model parameters correct?
3. Are the degrees of freedom correct?
4. Is the RMSEA .10 or less (better yet, .05 or less)?
5. Is the CFI .95 or more?
6. Is the TLI .95 or more?
7. Are there any residuals above .10 in absolute value terms?
8. The AIC and BIC are useful for comparing two or more models (lower is better)

We do have 10 observations. I am as puzzled as you are by the fact that lavaan only sees one model parameter to be estimated. I am looking into this. The total df available are K(K+1)/2 where K is the number of variables. For 3 variables, this is 3(4)/2=6. We lost one df when estimating the one parameter that lavaan is counting. The RMSEA of .027 is great, and the CFI/TLI are fantastic at very near 1. This is because I generated the data using a strictly parallel model (if you review the R code you'll see how). There are some residuals that are too high. I believe these are mainly due to very small sample size (10 cases).

```
> summary(fit.parallel,fit.measures=TRUE)
lavaan 0.6-9 ended normally after 10 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                         1

  Number of observations                            10

Model Test User Model:

  Test statistic                                 5.036
  Degrees of freedom                                 5
  P-value (Chi-square)                           0.411

Model Test Baseline Model:

  Test statistic                                20.325
  Degrees of freedom                                 3
  P-value                                        0.000

User Model versus Baseline Model:

  Comparative Fit Index (CFI)                    0.998
  Tucker-Lewis Index (TLI)                       0.999

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)                -46.283
  Loglikelihood unrestricted model (H1)        -43.764

  Akaike (AIC)                                  94.565
  Bayesian (BIC)                                94.868
  Sample-size adjusted Bayesian (BIC)           91.872

Root Mean Square Error of Approximation:

  RMSEA                                          0.027
  90 Percent confidence interval - lower         0.000
```

```
  90 Percent confidence interval - upper         0.441
  P-value RMSEA <= 0.05                          0.427

Standardized Root Mean Square Residual:

  SRMR                                           0.225

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  trueScore =~
    Q1        (v1)    1.000
    Q2        (v1)    1.000
    Q3        (v1)    1.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .Q1        (v1)    1.000
   .Q2        (v1)    1.000
   .Q3        (v1)    1.000
    trueScore         1.484    0.813    1.826    0.068

> # examine the residuals
> resid(fit.parallel)
$type
[1] "raw"

$cov
   Q1     Q2     Q3
Q1 -0.124
Q2  0.496  0.006
Q3 -0.084  0.066 -0.834
```

3 Things I have Not Covered

1. How to analyze dichotomous items. At this point, I tend to prefer mplus, which is costly. I am quite sure there is a way to handle logistic functions in lavaan. I would google it.
2. How to handle missing data. At this point, I would tend to use something like full information maximum likelihood analysis in mplus. I believe there is a way to do FIML in lavaan. I would use google.
3. How to do CFA in SAS. There are excellent resources online.
4. How to deal with situations where the tau equivalent and strictly parallel assumptions do not hold. At this point, it would be wise to get an expert in psychometrics involved to do more elaborate analysis. You might try dropping items one at a time to see if you can improve the model fit by excluding an item. However, this is the kind of ad hoc model revision that begs for cross validation samples.
5. How to handle a mix of data types (continuous, ordinal, categorical, dichotomous, etc.) That is a problem for mplus in my opinion, but there may be a lavaan solution
6. Heywood cases. You can obtain impossible parameter estimates, such as negative variance estimates. Bayesian analysis (blavaan) should prevent Heywood cases. Review your model carefully to make sure you did not do somethign wrong. However, Heywood cases are like ants at a picnic: they are a real problem and not easy to eliminate entirely from your life.

4 Exercises

1. Modify the R code provided so that the model represents tau equivalence but is not strictly parallel. Re-run the confirmatory factor analysis and observe the effect of your modification on the RMSEA, CFI, TLI when the parallel model is fit versus the tau equivalent model.
2. Modify the R code so that neither the tau equivalent nor the parallel model hold. See what effect that has on RMSEA, CFI, and TLI when you fit the parallel and tau equivalent models.
3. We had some unacceptably high residuals (greater than .10). Increase the size of the sample in the R code to 100 and 1000 and see whether the problem of high residuals still occurs when you fit the parallel model.
4. If you are really interested, download the following publicly available dataset. See if the parallel or tau equivalent model holds for the entire scale. The scale (the Nerdy Personality Attributes Scale) was developed based on empirical correlations with other measures. `osf.io/5njdx/`.
5. See if you can find a short subset of items from the NPAS (see previous item) for which the parallel assumption holds.