IRT workshop

Christopher David Desjardins

16 April 2014

Overview

Equating and Linking

- Equating is the process of converting scores on one form of a test to scores on another form of the same test, where forms are defined as a unique set of items.
- In CTT, equating is necessary since strictly parallel forms are very difficult to construct.
- Problem is to put the scores on one test on the same scale/metric as another test.
- The purpose is to transform the scores in such a way that it makes no difference which form of the test an examinee takes.

Conditions necessary for Equating

1. Two forms of a test measure exactly the same trait
2. The two forms yield scores that are equally reliable
3. Equating transformation is invertible

- Samples of 2000 seem adequate while 3000 is preferred for calibrating 3-PL models
- 3000 is more than needed typically for linear equating but not for most equipercentile methods

CTT equating

- Equipercentile equating
  - Non-linear transformation of one set of scores such that scores on the two tests will be equal if they correspond to the same percentile rank in another group of examinees
- Linear equating
  - Linear transformation of one set of scores such that scores on the two tests will be equal if they correspond to the same number of standard deviations from the mean in another group of examinees
- Methods may involve observed or estimated true scores
- Some use anchor items or anchor persons
- May be horizontal (groups with same ability) or vertical (groups with differing ability)
- Other forms of CTT equating exist
- For `R`, see the `equate` package and this vignette by Dr. Albano

Factors affecting validity of CTT equating

- Difficulty and length of tests
- Difficulty and length of anchor items
- Similarity of two groups and variability of ability levels
- Reliability of the two tests
- How the tests are scored
- Interactions of these factors

- These factors disappear in IRT because
  - Item parameters are invariant between subgroups within a population pending a linear transformation
  - θ and b are on same scale
  - a can be expressed on the same metric as θ and b
- Equating is done through linking
  - Transform of difficulty and discrimination parameters in one group to those of another group
  - Puts θ on the same scale and automatically does equating
  - c doesn't need to be transformed
  - Can use this to create items banks of items calibrated on the same scale regardless of the ability levels of those on the group that the items were calibrated on

Simultaneous calibration

- Useful when item response data are available on two different groups that might have different θ levels, have taken different items, and have different numbers of examinees
- Estimates item parameters for both groups in a single calibration run and treats non-overlapping items (persons) as missing data.
- Results in a single scale that spans the θ range of the combined group

Common items

- Common set of items administered to two or more groups, along with a unique set of items.
- Longer anchor tests better (at least 10 items)
- Distribution of item difficulties is important: A rectangular distribution of items seems to be best (based on simulations)
- The linking transformation is defined from the regression of the difficulty estimates of the anchor test items in the base group on those of the target group:

$$b_j^* = \beta b_j + \alpha$$

which implies that

$$\theta_j^* = \beta\theta_j + \alpha$$

- The same transformation is applied to the θ estimates
- The discrimination estimates: $a_j^* = a_j/\beta$

Common persons

- Requires that one group of persons take both sets of items that are otherwise administered to two different groups
- Less frequent than common items
- The linking transformation is defined from the linear regression of the θ estimates in one group on those of the other (target) group:

$$\theta_j^* = \beta\theta_j + \alpha$$

  which implies that

$$b_j^* = \beta b_j + \alpha$$

- Discrimination calculated in the same manner

Toy Example

```
> dif <- data.frame(base = c(1.5,.1,NA,.3,.5,1.2),
+  target = c(NA,NA, 2.1,-.2,1,2))
> dis <- data.frame(base = c(.8,.5,NA,1.1,1.5,2),
+ target = c(NA,NA,1.1,.8,1.5,1.2))
> ## Calculate B and A
> m0 <- lm(base ~ target, data = dif)
> A <- coef(m0)[1]
> B <- coef(m0)[2]
> ## Find scores on linked metric
> theta_unlink <- c(-1.4, .7, 2)
> theta_link <- B*theta_unlink + A
> theta_link

[1] -0.2692308  0.5730769  1.0945055
```