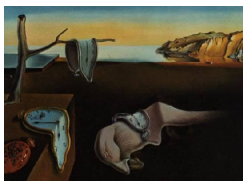

Journal of Statistical Software

December 2025, Volume 115, Issue 11. *doi: 10.18637/jss.v115.i11*

equateMultiple: An R Package to Equate Multiple Forms

Michela Battauz
Università degli Studi di Udine

Abstract

Item response theory (IRT) provides a framework for modeling the responses given to a test or questionnaire, which are assumed to depend on an underlying latent variable and on some item parameters. Due to identifiability issues, when the parameters are estimated separately on different datasets, the estimates of the item parameters and the predicted values of the latent variable are not directly comparable. Equating is a statistical procedure that can be used to convert these values to a common metric and to obtain comparable test scores. The R package **equateMultiple** implements methods to link the parameters estimated on many different datasets. After briefly reviewing the IRT models and the equating methods, this article illustrates the use of the package.

Keywords: equating, IRT, linking, testing.

1. Introduction

In IRT, the parameters of the model are usually identified by fixing the mean and the variance of the latent variable to zero and one. As a consequence, when the parameters are estimated separately on different groups of examinees, the item parameter estimates are not directly comparable. Separate estimation can be performed for various reasons. First, in educational testing security reasons require to prepare various forms of a test, which can be administered at the same time or in subsequent periods. Using all the data from previous administrations to estimate the item parameters could be cumbersome or even unfeasible, since the number of examinees and items grows with the number of administrations. Another reason to perform separate estimation is testing for differential item functioning (DIF), which occurs when the probability of giving a certain response to an item depends on characteristics of the subjects other than the latent variable of interest (Magis, Béland, Tuerlinckx, and De Boeck 2010). Hence, estimating the parameters separately on different groups of individuals can reveal

differences in the item parameter estimates (Battauz 2019). Similarly, separate estimation can be used to detect drifts of item parameters over time (Donoghue and Isham 1998; Battauz 2023). Whenever separate estimation is performed, to obtain values expressed on a common metric, it is necessary to covert the item parameters through a linear transformation that involves two unknown constants, called equating coefficients (Ogasawara 2001). The literature provides various methods to estimate these coefficients and to obtain comparable test scores (Kolen and Brennan 2014). While most of the literature focuses on the case of two separate groups, some works considered the case of many groups. The first proposal was given in Haberman (2009), who employed a regression model to estimate the equating coefficients. This approach represents an extension of the mean-geometric mean method for two forms to the case of multiple forms. Subsequently, Battauz (2017) extended other methods that were used for two forms to handle many forms. More recently, Battauz and Leôncio (2023) proposed a novel likelihood-based approach. The package **equateMultiple** (Battauz 2026) was originally developed to implement the methods proposed in Haberman (2009) and Battauz (2017). The latest versions of the package have included the likelihood-based method (Battauz and Leôncio 2023), improving the computational time with respect to the performance declared in the paper by employing C++ language. With the only exception of the R (R Core Team 2025) package **sirt** (Robitzsch 2025), which implements the method proposed by Haberman (2009) in the `linking.haberman` function, the other R packages deal only with the case of two forms to be equated. More specifically, the package **plink** (Weeks 2010) implements IRT-based methods, **kequate** (Andersson, Bränberg, and Wiberg 2013) applies the kernel method of test equating, **equateIRT** (Battauz 2015), besides computing the equating coefficients for pairs of forms, combines them to obtain the conversion through chains, while **equate** (Albano 2022) implements non-IRT methods. To the best of our knowledge, the non-R software that implements test equating methods is available in the applications **Equating Recipes** (Brennan, Wang, Kim, and Seol 2009), which provides a set of open-source functions to perform all types of equating described by Kolen and Brennan (2014) and other equating methods for pairs of forms, and **IRTEQ** (Han 2009), which employs IRT-based methods for two forms.

This article is organized as follows. Section 2 introduces the main IRT models for binary data and shows how to fit them in R. To this end, a real dataset available in the **equateMultiple** package is used. Section 3 briefly illustrates the IRT equating methods for multiple forms, which are applied to the real dataset using the **equateMultiple** package. Further insights on the identifiability issues and on the effect of composing forms of different items are provided in Section 5 by means of simulated data. Finally, Section 6 concludes the paper.

2. IRT models for binary data

Let Y_{ij} be the binary response of person i to item j. The **equateMultiple** package includes a real dataset regarding five forms of a math test administered to different groups of students. The following code loads the package and the data:

```
R> library("equateMultiple")
R> data("mathTest", package = "equateMultiple")
```

The dataset consists of a list of five data frames, each containing rows of responses from individuals to various items, with the item labels as the column headers.

```
R> head(mathTest[[1]])
```

```
  M0110 M0111 M0101 M0102 M0103 M0104 M0105 M0106 M0107 M0108 M0109
1     1     1     1     1     0     1     1     0     1     1     1
2     1     1     1     1     1     1     1     0     1     1     0
3     1     1     1     1     0     0     1     1     1     0     1
4     1     1     1     1     1     1     1     1     1     1     0
5     0     1     1     1     1     1     1     0     0     0     0
6     1     0     1     1     1     1     1     1     1     1     0
```

These data are frequently modeled using the three-parameter logistic (3PL) model, which expresses the probability of a correct response as follows

$$P(Y_{ij} = 1; \theta, a_j, b_j, c_j) = P(\theta; a_j, b_j, c_j) = c_j + (1 - c_j)\frac{\exp\{Da_j(\theta - b_j)\}}{1 + \exp\{Da_j(\theta - b_j)\}}, \tag{1}$$

where θ is the latent variable level, while a_j, b_j and c_j are parameters related to the item. More specifically, c_j is commonly called the guessing parameter since it corresponds to the lower asymptote of the function. When the guessing parameters are equal to zero, the model reduces to the two-parameter logistic (2PL) model. The parameter a_j, usually referred to as the discrimination parameter, determines the steepness of the function. If set to 1, the model results in the one-parameter logistic (1PL) model, also known as the Rasch model. To identify the 2-3PL models it is necessary to impose two constrains, usually by setting the mean of the latent variable to 0 and its variance to 1. For the Rasch model, one constrain is sufficient, e.g. the mean of the latent variable equal to 0. If, in addition, the variance is fixed at 1, it is possible to include a discrimination parameter constant across items, which corresponds to the standard deviation of the latent variable. Various R packages can be used to estimate these models, such as the packages **TAM** (Robitzsch, Kiefer, and Wu 2025), **ltm** (Rizopoulos 2006) or **mirt** (Chalmers 2012). Using **mirt**, the parameters of a 2PL model can be estimated separately for each data frame as follows:

```
R> library("mirt")
R> mods_mirt <- list()
R> for (i in 1:5)
+    mods_mirt[[i]] <- mirt(mathTest[[i]], 1, itemtype = "2PL", SE = TRUE)
```

Note that `SE` is set to `TRUE` to obtain the estimated covariance matrix of the item parameters, which is necessary later to obtain the standard errors of the quantities estimated in the equating process. The 2PL model was chosen to fit these data because the guessing parameters don't need to be converted (Kolen and Brennan 2014, p. 178), so it fully shows the equating methods and the conversion of the item parameters. Furthermore, fitting a 3PL model yields convergence errors for some of these datasets.

In the **mirt** package, the parameters of the model are actually estimated using the following parameterization

$$P(Y_{ij} = 1; \theta, a_j, b_j, c_j) = c_j + (1 - c_j)\frac{\exp(\beta_{1j} + \beta_{2j}\theta)}{1 + \exp(\beta_{1j} + \beta_{2j}\theta)}. \tag{2}$$

Furthermore, the guessing parameters are parameterized as follows:

$$c_j = \frac{\exp(\beta_{3j})}{1 + \exp(\beta_{3j})}. \tag{3}$$

So, the covariance matrix of the estimates obtained with **mirt** refers to the parameters β_{1j}, β_{2j} and β_{3j}. The function `modIRT` of the `equateIRT` package can be used to extract the item parameter estimates and their covariance matrix and to convert them to the parameterization adopted by Equation 1

```
R> mods_extract <- modIRT(mods_mirt, display = FALSE)
```

The argument `display` is set to `FALSE` in this case to avoid long prints, though it is advisable to leave it `TRUE` (the default) to observe the estimates and the standard errors. The output in the object `mods_extract` is of class `modIRT` and it consists of a list with length equal to the number of forms. Each element of this list contains a list with the item parameter estimates, their covariance matrix and the number of parameters of the adopted model.

If the item parameters are estimated using external software, it is still possible to use the `modIRT` function to build an object of class `modIRT`. In this case the inputs of the function should be a list of item parameter estimates (argument `coef`) and a list of covariance matrices (argument `var`). Attention should be paid to the parameterization of the item parameter estimates, which should be specified using argument `ltparam`, which is `TRUE` if the difficulty parameters are expressed as in Equation 2, and argument `lparam`, which is `TRUE` if the guessing parameters are in the form of Equation 3. If the parameters are expressed as in Equation 1, they should be both set to `FALSE`.

The estimation method commonly employed to estimate the parameters is the marginal maximum likelihood method (Bock and Aitkin 1981), which maximizes the marginal log-likelihood function obtained by assuming a standard normal distribution for the latent variable and integrating it out. Hence, the latent variable is assumed to have zero mean and variance equal to one in all the groups taking different forms, though the mean and the variance can vary across groups. As a consequence, the item parameter estimates are not on the same scale, so it is necessary to convert them to a common metric to obtain comparable values. This is the first step of the equating process, also known as linking.

3. Equating multiple forms

3.1. The equating design

To link two forms administered to different groups of examinees, the forms must have some items in common. This case is commonly referred to as the common-item nonequivalent groups design (Kolen and Brennan 2014). In case of multiple forms, it is sufficient that all the forms can be linked to the others through a path that connects them. This can be inspected using the function `linkp` of the **equateIRT** package, which shows the number of items in common between pairs of forms

```
R> linkage_plan <- linkp(mods_extract)
R> linkage_plan
```

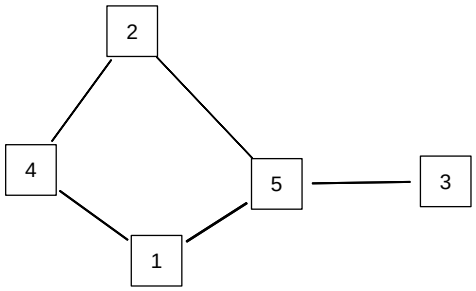

Figure 1: Linkage plan.

```
     [,1] [,2] [,3] [,4] [,5]
[1,]   11    0    0    3    5
[2,]    0   11    0    2    1
[3,]    0    0   11    0    3
[4,]    3    2    0   11    0
[5,]    5    1    3    0   11
```

For example, Forms 1 and 4 have 3 items in common, while Form 1 and 3 do not share any items. Forms 2 and 5 have one only item in common; however, they both present common items with other forms. To estimate the equating coefficients, it is sufficient to have one item in common between one form and all the others. However, it is advisable to have more common items to reduce the random error in the estimation of the equating coefficients and improve the stability of the results. The linkage plan can be represented using, for example, the **sna** package (Butts 2008, 2024) for social network analysis

```
R> library("sna")
R> set.seed(6)
R> gplot(linkage_plan, displaylabels = TRUE, vertex.sides = 4,
+    vertex.cex = 3, vertex.rot = 45, usearrows = FALSE,
+    label.pos = 5, label.cex = 1, vertex.col = 0, edge.lwd = 0.2)
```

Figure 1 shows that all the forms can be connected to the others, though not all of them are directly linked.

3.2. Equating methods for multiple forms

To introduce the methods proposed in the literature to estimate simultaneously the equating coefficients that link all the forms, we denote the item parameters expressed on the scale of the form chosen as base as a_j^* and b_j^*. Every parameter can be converted to the scale of the base form using the following equations

$$a_j^* = \frac{a_{jt}}{A_t} \tag{4}$$

and

$$b_j^* = b_{jt}A_t + B_t, \tag{5}$$

where A_t and B_t are the equating coefficients of form t.

The first proposal is given in Haberman (2009), who employed Equations 4 and 5 to specify the regression models

$$\log \hat{a}_{jt} = \log A_t + \log a_j^* + \varepsilon_{jt}^a \tag{6}$$

and

$$\hat{b}_{jt}\hat{A}_t = -B_t + b_j^* + \varepsilon_{jt}^b, \tag{7}$$

where ε_{jt}^a and ε_{jt}^b are error terms in the models that involve the discrimination and the difficulty parameters, respectively. The estimation of the parameters by least squares provides estimates of both the equating coefficients and the item parameters on a common metric, exploiting all the items at the same time. Since this method, when applied to only two forms, gives the same results as the mean-geometric mean method (Mislevy and Bock 1990), it was referred to as the *multiple mean-geometric mean* (MM-GM) method in Battauz (2017). Such paper provided extensions to the case of multiple forms of other methods already used for two forms. In particular, the *multiple mean-mean* (MM-M) method generalized the mean-mean method (Loyd and Hoover 1980) to the case of many forms and it requires to solve the following equations

$$\hat{A}_t = \frac{\sum_{j \in J_t} \hat{a}_{jt}}{\sum_{j \in J_t} \frac{\sum_{s \in U_j} \hat{a}_{js}}{\sum_{s \in U_j} \hat{A}_s}}, \quad t = 1, \ldots, T. \tag{8}$$

where J_t is the set of items administered in form t and U_j is the set of forms that include item j. After the coefficients A_t, $t = 1, \ldots, T$, are estimated, the coefficients B_t, $t = 1, \ldots, T$, are computed as in the MM-GM method. The coefficients of the base form are fixed at 1 and 0.

The methods based on the item response function (given in Equation 1) for two forms are the Haebara (Haebara 1980) and the Stocking-Lord (Stocking and Lord 1983) methods. The *multiple item response function* (MIRF) and the the *multiple test response function* (MTRF) methods generalize the Haebara and the Stocking-Lord methods to the case of multiple forms. They are obtained by defining an estimator for the item parameters on a common metric

$$\hat{a}_j^* = \frac{1}{u_j} \sum_{s \in U_j} \frac{\hat{a}_{js}}{\hat{A}_s} \quad \text{and} \quad \hat{b}_j^* = \frac{1}{u_j} \sum_{s \in U_j} (\hat{b}_{js}\hat{A}_s + \hat{B}_s), \tag{9}$$

converting them to the scale of form t

$$\hat{a}_{jt}^* = \hat{a}_j^* \hat{A}_t \quad \text{and} \quad \hat{b}_{jt}^* = \frac{\hat{b}_j^* - \hat{B}_t}{\hat{A}_t}, \tag{10}$$

and using these values in the item response function to define

$$P_{jt}^* = P(\theta; \hat{a}_{jt}^*, \hat{b}_{jt}^*, \hat{c}_{jt}), \tag{11}$$

and

$$P_{jt} = P(\theta; \hat{a}_{jt}, \hat{b}_{jt}, \hat{c}_{jt}). \tag{12}$$

The MIRF method requires to minimize the following function with respect to the equating coefficients

$$\sum_{t=1}^{T} \int_{-\infty}^{\infty} \sum_{j \in J_t} \left(P_{jt} - P_{jt}^* \right)^2 h(\theta) d\theta, \tag{13}$$

where $h(\cdot)$ is the density of a standard normal distribution. Instead, the MTRF method minimizes

$$\sum_{t=1}^{T} \int_{-\infty}^{\infty} \left(\sum_{j \in J_t} P_{jt} - P_{jt}^* \right)^2 h(\theta) d\theta. \tag{14}$$

A more detailed explanation of these methods and the derivation of the standard errors of the estimates of the equating coefficients can be found in Battauz (2017).

In a subsequent paper, Battauz and Leôncio (2023) proposed a novel likelihood-based method to estimate the equating coefficients in case of multiple forms. Differently from the previous proposals, this approach accounts for the correlation of the item parameter estimates of the same form and for their heteroskedasticity. Since the item parameters are estimated by marginal maximum likelihood, they are consistent and asymptotically normal. So, assuming a normal distribution for them

$$\begin{pmatrix} \hat{\boldsymbol{a}}_t \\ \hat{\boldsymbol{b}}_t \end{pmatrix} \sim N\left(\begin{pmatrix} \boldsymbol{a}_t \\ \boldsymbol{b}_t \end{pmatrix}, \boldsymbol{\Sigma}_t \right), \quad t = 1, \ldots, T, \tag{15}$$

where

$$\boldsymbol{a}_t = A_t \boldsymbol{a}_t^* \quad \text{and} \quad \boldsymbol{b}_t = \frac{1}{A_t}(\boldsymbol{b}_t^* - B_t),$$

it is possible to define a profile likelihood function for the equating coefficients, treating $\boldsymbol{a}_t^*$ and $\boldsymbol{b}_t^*$ as nuisance parameters. Here, $\boldsymbol{a}_t$ and $\boldsymbol{b}_t$ denote the vectors of true item parameters in form t, $\hat{\boldsymbol{a}}_t$ and $\hat{\boldsymbol{b}}_t$ are their estimates and $\boldsymbol{a}_t^*$ and $\boldsymbol{b}_t^*$ are the true parameters on the common scale. The covariance matrix $\boldsymbol{\Sigma}_t$ is assumed to be known and equal to the one estimated along with the item parameters. The maximization of the profile log-likelihood function yields estimates of the equating coefficients. All these methods are implemented in the R package **equateMultiple**.

3.3. An application of the multiple equating methods to real data

The following example applies the MM-M method to the data introduced in Section 2:

```
R> eq_mm <- multiec(mods_extract)
R> summary(eq_mm)

Equating coefficients:
 EQ Form Estimate  StdErr
  A   T1  1.00000 0.00000
  A   T2  0.94704 0.19577
  A   T3  0.49900 0.14970
  A   T4  1.50535 0.21478
  A   T5  1.04509 0.14703
  B   T1  0.00000 0.00000
  B   T2 -0.74833 0.21233
  B   T3 -0.36654 0.28395
  B   T4 -0.22767 0.17555
  B   T5 -0.85328 0.12708
```

These coefficients can be used to convert the item parameters to the scale of the base form from the scale of the other forms, as described in Equations 4 and 5. Similarly, the latent variable, which has zero mean and variance equal to 1 in all groups before conversion, can be converted to the scale of the base form using the equation $\theta_t A_t + B_t$. Hence, after the conversion, the standard deviation of the latent variable will be equal to A_t and the mean will be equal to B_t. For instance, in this application, the variability of the latent variable in the group of examinees that took Form 3 is smaller than the group that was administered Form 1 (since $A_3 < 1$), and the average value is lower (since $B_3 < 0$). The labels `T1`, ..., `T5` were assigned by default when running the `modIRT` function, and correspond to Forms 1 to 5.

To give another example, the following code employs the MIRF method and converts the item parameters to the scale of Form 5:

```
R> eq_irf <- multiec(mods_extract, method = "irf", base = 5)
R> summary(eq_irf)

Equating coefficients:
 EQ Form Estimate  StdErr
  A   T1  1.07414 0.13311
  A   T2  0.95216 0.18741
  A   T3  0.44643 0.10133
  A   T4  1.70976 0.27645
  A   T5  1.00000 0.00000
  B   T1  0.88384 0.14643
  B   T2  0.11032 0.17429
  B   T3  0.68281 0.14861
  B   T4  0.63805 0.17507
  B   T5  0.00000 0.00000
```

For the methods that require the optimization of an objective function, it is possible to specify the initial values using argument `start`, either as a vector of values or as the output of the `multiec` function previously called. Here is an example using the likelihood-based method and the estimates obtained with MM-M as starting values:

```
R> eq_lik <- multiec(mods_extract, method = "lik", start = eq_mm,
+    obsinf = FALSE)
R> summary(eq_lik)

Equating coefficients:
 EQ Form Estimate  StdErr
  A   T1  1.00000 0.00000
  A   T2  0.89833 0.21352
  A   T3  0.41085 0.19520
  A   T4  1.55587 0.31619
  A   T5  0.91122 0.15105
  B   T1  0.00000 0.00000
  B   T2 -0.67320 0.15599
  B   T3 -0.28756 0.15962
  B   T4 -0.22401 0.14758
  B   T5 -0.82224 0.10925
```

The function `item.common` extracts the item parameters on the common scale, namely $\hat{a}_j^*$ and $\hat{b}_j^*$. The formula used to estimate the item parameters on the common scale depends on the equating method used. In case of the MM-GM method, $\log a_j^*$ and b_j^* are estimated when the models (6) and (7) are fitted, and a_j^* is then obtained by taking the exponential. The MM-M method estimates a_j^* as $\frac{\sum_{s \in U_j} \hat{a}_{js}}{\sum_{s \in U_j} \hat{A}_s}$, while b_j^* is estimated as in the MM-GM method. Using the MIRF or the MTRF methods the items on the common metric are estimated using Equations 9. Finally, if the likelihood-based approach is followed, a_j^* and b_j^* are obtained after the estimation of the equating coefficients from the maximum likelihood estimators. For example, likelihood-based method leads to:

```
R> items.com <- item.common(eq_lik)
R> items.com[1:4, ]

            Item  Estimate    StdErr
1 Dscrmn.M000112 2.3465202 0.3632949
2   Dscrmn.M0095 0.6545708 0.1735455
3   Dscrmn.M0101 1.4180386 0.1750849
4   Dscrmn.M0102 1.3305638 0.1657058

R> items.com[42:45,]

             Item  Estimate    StdErr
42 Dffclt.M000112 -1.979155 0.1407583
43   Dffclt.M0095 -1.450017 0.1460135
44   Dffclt.M0101 -1.557044 0.1132841
45   Dffclt.M0102 -1.034990 0.1020986
```

Here, the first four discrimination and difficulty parameters are printed.

The function `eqc` may be used to extract the equating coefficients. Additionally, a table containing the item parameters for all administrations, as well as the item parameters converted to the scale of the base form, can be obtained using the function `itm`.

3.4. Scoring

Once the equating coefficients are computed, it is possible to obtain the equated scores. The main methods for this task are true score equating and observed score equating. True score equating proceeds by finding the value of the latent variable θ such that the expected value of the number of correct responses, which is the score, is equal to the one observed on the base form. Then the expected value of the score is computed for all other forms, using the item parameters converted to the scale of the base form. Instead, the method of observed score equating is based on computing the score of non-base forms at the same percentile of the score of the base form. To this end, it is first necessary to compute the cumulative distribution functions of the scores in each form using the item parameters converted to the common metric. This requires to marginalize over the distribution of the latent variable, which can be performed using the distribution of the latent variable of both the group administered the

base form and the group administered the non-base form, leading to two distributions. Then, a mixture of them is obtained with fixed weights. A common practice is to use continuous approximations of the discrete score distribution, called continuization. The **equateMultiple** package performs this step through linear interpolation. We refer to Kolen and Brennan (2014, Chapter 6.5 and 6.6) for more details. The function `score` implements both methods, adopting true score equating as default:

```
R> sc.eq.tse <- score(eq_lik)

The following scores are not attainable: 0

R> round(sc.eq.tse, 3)[1:6,]

   theta T1 T2.as.T1 StdErr_T2.as.T1 T3.as.T1 StdErr_T3.as.T1
1 -2.920  1    2.108           0.781    0.451           0.664
2 -2.173  2    3.629           0.911    1.177           1.149
3 -1.698  3    4.920           0.740    2.114           1.437
4 -1.321  4    5.955           0.517    3.300           1.506
5 -0.985  5    6.796           0.354    4.763           1.305
6 -0.660  6    7.517           0.273    6.488           0.871
  T4.as.T1 StdErr_T4.as.T1 T5.as.T1 StdErr_T5.as.T1
1    3.327           0.586    1.795           0.385
2    4.421           0.472    2.771           0.323
3    5.176           0.364    3.758           0.274
4    5.789           0.269    4.793           0.205
5    6.332           0.192    5.824           0.161
6    6.847           0.145    6.825           0.164
```

Here the output is shown for scores from 1 to 6 and it is rounded to three decimal places. A score equal to zero cannot be attained since, even for extremely low values of ability, the expected score in Form 1 is greater than zero. For example, a value of θ equal to -0.985 leads to an expected score of 5 in Form 1, and to an expected score of 6.796 in Form 2, showing that Form 1 is more difficult at this level of ability. By default, all attainable scores are equated, but it is possible to specify which ones to equate using the `scores` argument. The standard errors of the equated scores are also computed, unless `se` is set to `FALSE`.

To perform observed score equating, the `method` argument may be set to `OSE`:

```
R> sc.eq.ose <- score(eq_lik, method = "OSE")
R> round(sc.eq.ose, 3)[1:7,]

  T1 T2.as.T1 StdErr_T2.as.T1 T3.as.T1 StdErr_T3.as.T1 T4.as.T1
1  0    1.089           0.652   -0.294           0.319    1.735
2  1    2.566           0.532    0.539           0.758    3.077
3  2    3.783           0.599    1.506           1.097    4.023
4  3    4.871           0.517    2.684           1.212    4.789
5  4    5.823           0.394    4.076           1.079    5.491
```

```
6  5    6.677           0.299    5.559           0.675    6.070
7  6    7.462           0.321    7.040           0.555    6.703
  StdErr_T4.as.T1 T5.as.T1 StdErr_T5.as.T1
1           0.342    0.737           0.217
2           0.473    1.828           0.254
3           0.381    2.835           0.236
4           0.264    3.827           0.202
5           0.261    4.831           0.163
6           0.170    5.834           0.143
7           0.143    6.828           0.157
```

It is possible to observe that, for example, a score of 6 in the first form is equivalent to a score of 7.46 in the second and to a score of 7.04 in the third, and so on. This indicates that it is more difficult to attain a score equal to 6 in the first form compared to the other two. By default, the marginal distributions of the scores are obtained by Gauss-Hermite quadrature with 30 points. The number of points can be specified by the `nq` argument. Alternatively to Gaussian quadrature, it is possible to specify a vector of points using the `theta` argument and the corresponding weights with the `weights` argument. Argument `w` is the weight given to the group that was administered the base form in the computation of the mixture among the groups that took the base and the non-base forms. By default, it is equal to 0.5.

4. Package overview

In this section, a comprehensive overview of the functionality of the **equateMultiple** package is provided. The package focuses on performing equating of multiple forms of a test administered to nonequivalent groups of subjects. Hence, a different dataset is available for each form. The parameters of the IRT model need to be estimated previously for each dataset, using other R packages or external software. Then, the parameters must be extracted and eventually converted to the parameterization adopted in Equation 1. This can be accomplished using the function `modIRT`, as explained is Section 2. At this point it is possible to estimate the equating coefficients. After the estimation of the equating coefficient it is possible to obtain equivalent scores. This process is shown in Figure 2.

In the **equateMultiple** package the equating coefficients for multiple forms can be estimated using the function `multiec`, which is the main function of the package. The arguments of the function are the following:

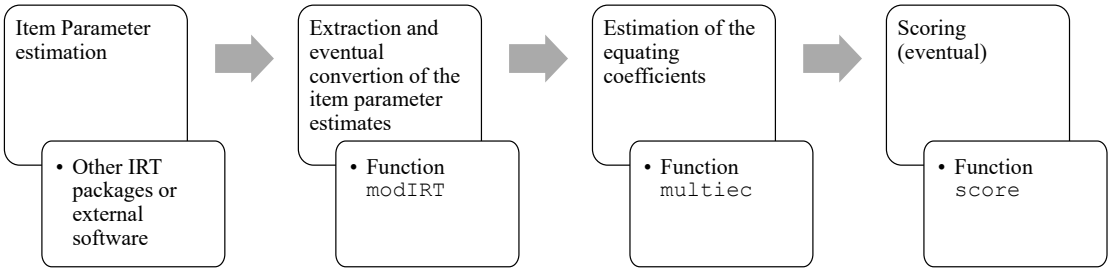

Figure 2: Steps of the equating process.

- `mods`: An object of class `modIRT`, obtained as output of the `modIRT` function.
- `base`: The base form specified as an integer value. By default it is the first form.
- `method`: The equating method for the estimation of the equating coefficients. The default is `mean-mean` for the MM-M method, while it needs to be set to `mean-gmean` for MM-GM, `irf` for MIRF, `trf` for MTRF or `lik` for the likelihood-based approach.
- `se`: Logical value indicating if the standard errors of the equating coefficients and of the item parameters on a common scale should be computed. The default is `TRUE`.
- `nq`: Number of quadrature points used for the Gauss-Hermite quadrature to approximate the integrals for methods MIRF and MTRF. By default it is set to 30.
- `start`: Initial values to be used in the optimization procedure for methods MIRF, MTRF and the likelihood-based approach. It can be specified as the output of the function `multiec` previously called, or as a vector containing $T-1$ numeric values for the A_t equating coefficients followed by other $T-1$ values for the B_t equating coefficients, since the equating coefficients of the base form are fixed.
- `iter.max`: Maximum number of iterations allowed in the optimization procedure.
- `obsinf`: Logical value indicating if the observed information matrix should be used for the computation of the standard errors in the likelihood-based method. The default is `TRUE`.
- `trace`: Logical value indicating if tracing information should be produced. The default is `TRUE`.

The output of function `multiec` is an object of class `mlteqc`, which includes the following components.

- `A`: Vector of equating coefficients A_t.
- `B`: Vector of equating coefficients B_t.
- `se.A`: Vector of standard errors of the equating coefficients A_t.
- `se.B`: Vector of standard errors of the equating coefficients B_t.
- `varAB`: Covariance matrix of the equating coefficients.
- `as`: Estimates of the discrimination parameters on a commons scale, namely $\hat{a}_j^*$.
- `bs`: Estimates of the discrimination parameters on a commons scale, namely $\hat{b}_j^*$.
- `se.as`: Standard errors of $\hat{a}_j^*$.
- `se.bs`: Standard errors of $\hat{b}_j^*$.
- `tab`: Data frame containing item parameter names (Item), item parameter estimates across all the forms (e.g. T1, ..., T3), and item parameter estimates of all the forms converted in the scale of the base form (e.g. T3.as.T1).

- `varFull`: List of covariance matrices of the item parameter estimates of every form.
- `partial`: Partial derivatives of estimates of the equating coefficients with respect to the item parameters.
- `itmp`: Type of IRT model used, `1` for 1PL, `2` for 2PL and `3` for 3PL.
- `method`: The equating method used.
- `basename`: The name of the base form.
- `convergence`: An integer code. `0` indicates successful convergence. Returned only with MIRF, MTRF and the likelihood-based methods.

The methods implemented for the class `mtleqc` are

- `print` that prints essential information, as the method used.
- `summary` that provides estimates and standard values.
- `plot` that produces a plot of the item parameter estimates of one form against the base form before and after conversion.
- `item.common` that extracts the item parameter estimates on a common scale.
- `eqc` that extracts the equating coefficients.
- `itm` that extracts the item parameter estimates before and after conversion.
- `score` that computes the equated scores.

5. An illustration with simulated data

Making use of simulated data, this section aims to provide a better understanding of the effect of fixing the mean and the variance of the latent variable on the estimates and at showing the capabilities of the methods implemented in the **equateMultiple** package.

Let the true equating coefficients be the following:

```
R> A <- seq(1, 2, length = 5)
R> B <- seq(0, 2, length = 5)
R> A

[1] 1.00 1.25 1.50 1.75 2.00

R> B

[1] 0.0 0.5 1.0 1.5 2.0
```

which are also the standard deviation and the mean of the abilities, generated as normally distributed:

```
R> set.seed(1)
R> n <- 100000
R> theta <- list()
R> for (i in 1:5) theta[[i]] <- rnorm(n, B[i], A[i])
```

A quite large number of subjects (i.e. 100,000) per form is employed to reduce the sample variability of the estimates that are later obtained, so that it is possible to observe more clearly the effect of the different parameters of the ability distributions. The discrimination and difficulty parameters are generated as follows:

```
R> set.seed(1)
R> as <- runif(20, 0.7, 1.3)
R> bs <- sort(rnorm(20, 1, 1))
```

where the difficulty parameters are ordered. Then, a list of 5 datasets is generated from a 2PL model. To disentangle the effect of different means and standard deviations of the abilities from the effect of differences in the item parameters that compose the test forms, these test forms are composed of the same 20 items.

```
R> gen.resp <- function(theta, a, b) {
+    lp <- a * (theta - b)
+    pr <- plogis(lp)
+    rn <- runif(length(theta))
+    (pr > rn) * 1
+  }
R> itms <- paste("I", formatC(1:20, width = 2, format = "d", flag = "0"),
+    sep = "")
R> set.seed(1)
R> resp <- list()
R> for (i in 1:5) {
+    resp_i <- matrix(NA, n, 20)
+    for (j in 1:20) resp_i[, j] <- gen.resp(theta[[i]], as[j], bs[j])
+    colnames(resp_i) <- itms
+    resp[[i]] <- resp_i
+  }
```

Due to the increasing mean of the abilities, the scores obtained in subsequent forms tend to be higher, as can be observed in the histograms in Figure 3 that represent the distribution of the scores for each form:

```
R> row.scores <- lapply(resp, rowSums)
R> par(mfrow = c(1, 5), mar = c(2, 2, 2, 1))
R> for (i in 1:5) hist(row.scores[[i]], main = paste("Form", i),
+    xlab = "", col = 5)
```

The item parameters are estimated using the **mirt** package

```
R> mods_mirt_sim <- list()
R> for (i in 1:5) mods_mirt_sim[[i]] <- mirt(resp[[i]], 1,
+    itemtype = "2PL", SE = TRUE)
```

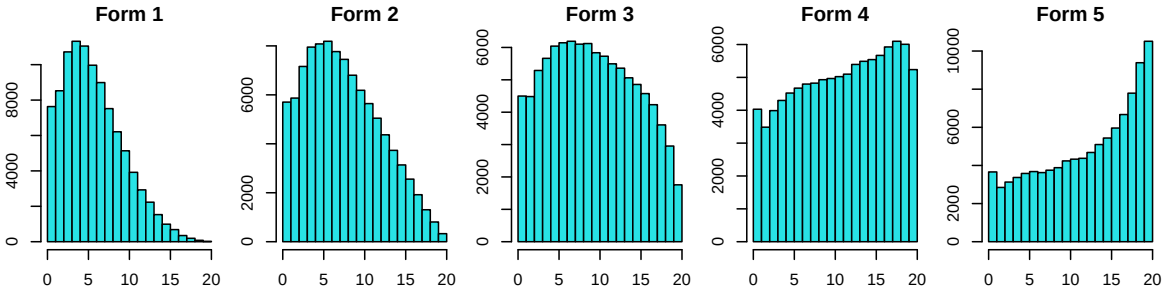

Figure 3: Distribution of row scores across the forms.

and the `modIRT` function is again used to extract the item parameters and their covariance matrix:

```
R> mods_extract_sim <- modIRT(mods_mirt_sim, display = FALSE)
```

The estimation of the equating coefficients is here performed using the MIRF method

```
R> eq_irf_sim <- multiec(mods_extract_sim, method = "irf", se = TRUE)
R> summary(eq_irf_sim)

Equating coefficients:
 EQ Form Estimate    StdErr
  A   T1  1.00000 0.0000000
  A   T2  1.24391 0.0054581
  A   T3  1.48863 0.0064841
  A   T4  1.74392 0.0076522
  A   T5  1.99551 0.0088832
  B   T1  0.00000 0.0000000
  B   T2  0.50260 0.0058033
  B   T3  0.99824 0.0068284
  B   T4  1.49458 0.0082097
  B   T5  1.99465 0.0098106
```

It is possible to observe that the estimated equating coefficients are very close to the true ones, due to the large samples employed to estimate the item parameters. We can now compare the item parameter estimates before and after the conversion. To this end, the `plot` function produces a scatter plot of the difficulty and discrimination parameter estimates of a specified form against the corresponding estimates of the base form, for the items in common between the two forms. In the following, Form 5 is selected by name, though it is possible to select the form by an integer value:

```
R> par(mfrow = c(2, 2))
R> plot(eq_irf_sim, form = "T5")
```

It is possible to observe in Figure 4 that the estimates of the difficulty parameters are lower in Form 5 than in Form 1, due to the higher average ability of the examinees who took Form 5,

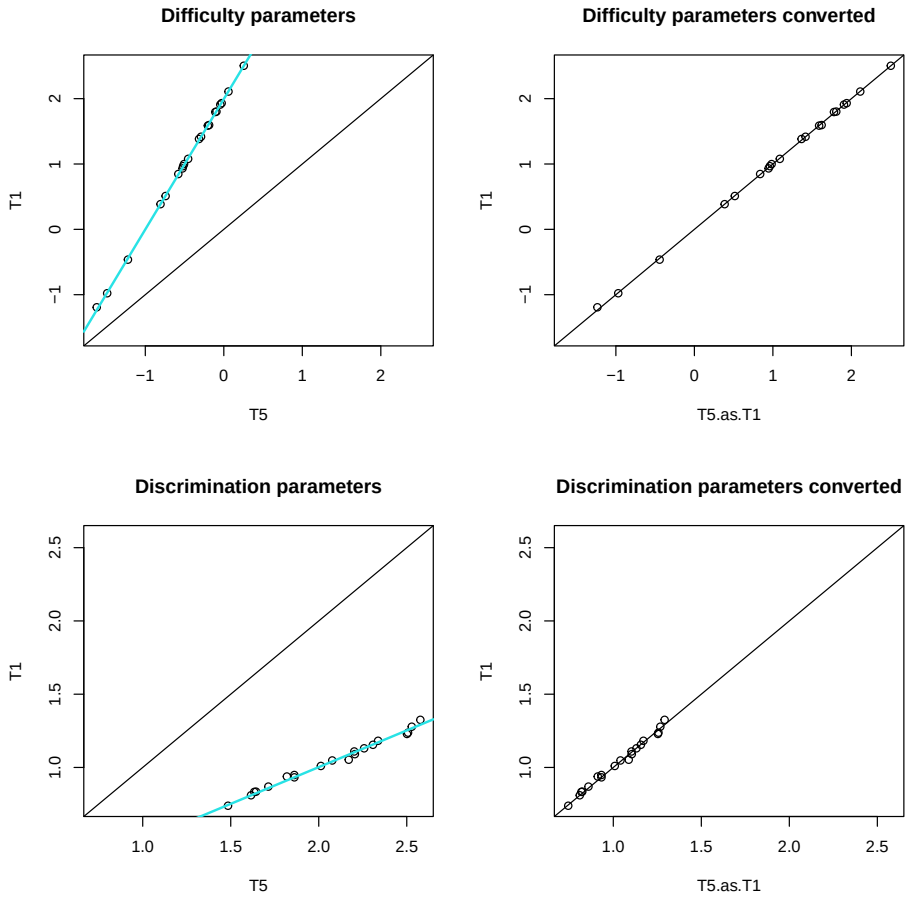

Figure 4: Comparison of item parameter estimates before and after conversion.

which makes seem the items easier. They are also shrunk, due to the higher variability of the abilities. Instead, the discrimination parameters are amplified in Form 5, because of the larger variability of the abilities. Once the parameters are converted, they are very close to the bisecting line.

Since the forms are composed of the same items, the scores here are already comparable. In fact, the equated scores (obtained by applying true score equating) show that there are substantially no differences across the test forms:

```
R> sc.eq.tse.sim <- score(eq_irf_sim)

The following scores are not attainable: 0

R> round(sc.eq.tse.sim, 3)[1:6,]

   theta T1 T2.as.T1 StdErr_T2.as.T1 T3.as.T1 StdErr_T3.as.T1
1 -2.425  1    1.002           0.007    1.014           0.007
2 -1.532  2    2.000           0.005    2.009           0.006
3 -0.966  3    2.999           0.003    3.002           0.004
4 -0.536  4    3.999           0.003    3.997           0.003
```

```
5 -0.182  5    5.000           0.005    4.994           0.004
6  0.125  6    6.002           0.005    5.992           0.005
  T4.as.T1 StdErr_T4.as.T1 T5.as.T1 StdErr_T5.as.T1
1    1.003           0.007    1.013           0.008
2    2.001           0.006    2.007           0.006
3    2.999           0.004    2.999           0.004
4    3.998           0.004    3.995           0.004
5    4.999           0.004    4.994           0.004
6    6.000           0.005    5.995           0.005
```

A score equal to zero is not attainable for these data with true score equating, since the expected score is higher than zero even for extremely low values of the ability. Observed score equating is not performed since it gives very similar results.

Computing the equivalent scores is actually meaningful when the forms are not composed of the same items. Hence, the data are now modified by deleting the second half of the items in Form 1 and the first half in Form 5. Since the difficulties were ordered, Form 1 is now composed of the easiest items, while Form 5 is composed of the more difficult ones. Note also that these forms do not have items in common, so the methods developed for two test forms cannot be applied, while the multiple equating methods permit to link them.

```
R> resp[[1]] <- resp[[1]][, 1:10]
R> resp[[5]] <- resp[[5]][, 11:20]
```

The 2PL model is fitted on the modified datasets, and the item parameters are extracted

```
R> for (i in c(1, 5))
+    mods_mirt_sim[[i]] <- mirt(resp[[i]], 1, itemtype = "2PL", SE = TRUE)
R> mods_extract_sim <- modIRT(mods_mirt_sim, display = FALSE)
```

The equating coefficients, which depend on the distribution of the abilities, are substantially unchanged

```
R> eq_irf_sim <- multiec(mods_extract_sim, method = "irf", se = TRUE)
R> summary(eq_irf_sim)

Equating coefficients:
 EQ Form Estimate    StdErr
  A   T1  1.00000 0.0000000
  A   T2  1.24544 0.0061978
  A   T3  1.49046 0.0074040
  A   T4  1.74600 0.0087616
  A   T5  1.98875 0.0112424
  B   T1  0.00000 0.0000000
  B   T2  0.50408 0.0061408
  B   T3  1.00044 0.0074075
  B   T4  1.49744 0.0090994
  B   T5  2.00075 0.0113124
```

while the equated scores show that the forms now have different difficulties

```
R> sc.eq.tse.sim <- score(eq_irf_sim)

The following scores are not attainable: 0

R> round(sc.eq.tse.sim, 3)

    theta T1 T2.as.T1 StdErr_T2.as.T1 T3.as.T1 StdErr_T3.as.T1
1  -2.196  1    1.204           0.007    1.216           0.007
2  -1.236  2    2.482           0.004    2.489           0.005
3  -0.595  3    3.848           0.005    3.846           0.005
4  -0.081  4    5.314           0.006    5.306           0.006
5   0.376  5    6.903           0.007    6.891           0.007
6   0.817  6    8.657           0.007    8.644           0.007
7   1.283  7   10.648           0.007   10.638           0.007
8   1.841  8   13.017           0.011   13.014           0.011
9   2.680  9   16.024           0.017   16.034           0.016
10 48.358 10   20.000           0.000   20.000           0.000
   T4.as.T1 StdErr_T4.as.T1 T5.as.T1 StdErr_T5.as.T1
1     1.205           0.007    0.203           0.002
2     2.483           0.005    0.482           0.003
3     3.847           0.005    0.847           0.003
4     5.312           0.007    1.312           0.004
5     6.900           0.007    1.901           0.004
6     8.653           0.007    2.654           0.005
7    10.646           0.007    3.646           0.006
8    13.017           0.011    5.016           0.008
9    16.028           0.016    7.030           0.013
10   20.000           0.000   10.000           0.000
```

A score equal to 6 on the first form (which includes only the 10 easiest items) is equivalent to a score of about 8.6 on Forms 2, 3 and 4 (that include all 20 items) and to a score of about 2.6 on Form 10 (composed of the 10 more difficult items). In other words, a person who scored 6 in Form 1 is expected to score 2.6 in Form 5, due to the higher difficulty of the last form.

6. Conclusions

Although testing programs develop several forms of a test, whose scores need to be comparable, most of the IRT equating methods proposed in the literature deal with the case of two forms. Some recent works proposed methods to equate a large number of test forms. The **equateMultiple** package implements them, thus making accessible the use of such methods to researchers and practitioners. This paper provides a brief review of the equating methods for multiple forms and shows how the package can be used to apply them to a real dataset. The effect of different ability levels across groups of examinees on the estimation of the item parameters was further inspected employing simulated data. The simulated data were also used

to show the effect of composing forms of different items. This paper showed the use of the **equateMultiple** package to adjust for such effects and to obtain comparable item parameter estimates and comparable scores, even when two forms do not share any items.

The **equateMultiple** package implements the methodology to equate multiple forms that was developed only for the case of unidimensional models for binary items. Hence, the extension of the methods to polytomous items or multidimensional models remains a subject for future research.

Acknowledgments

This research was funded by the European Union-NextGenerationEU (Italian Ministry for Universities and Research DM funds 737/2021).

References

Albano A (2022). ***equate**: Statistical Methods for Test Score Equating*. doi:10.32614/CRAN.package.equate. R package version 2.0.8.

Andersson B, Bränberg K, Wiberg M (2013). "Performing the Kernel Method of Test Equating with the Package **kequate**." *Journal of Statistical Software*, **55**(6), 1–25. doi:10.18637/jss.v055.i06.

Battauz M (2015). "**equateIRT**: An R Package for IRT Test Equating." *Journal of Statistical Software*, **68**(7), 1–22. doi:10.18637/jss.v068.i07.

Battauz M (2017). "Multiple Equating of Separate IRT Calibrations." *Psychometrika*, **82**(3), 610–636. doi:10.1007/s11336-016-9517-x.

Battauz M (2019). "On Wald Tests for Differential Item Functioning Detection." *Statistical Methods & Applications*, **28**, 103–118. doi:10.1007/s10260-018-00442-w.

Battauz M (2023). "Testing for Differences in Chain Equating." *Statistica Neerlandica*, **77**(2), 134–145. doi:10.1111/stan.12277.

Battauz M (2026). ***equateMultiple**: Equating of Multiple Forms*. doi:10.32614/CRAN.package.equatemultiple. R package version 1.1.3.

Battauz M, Leôncio W (2023). "A Likelihood Approach to Item Response Theory Equating of Multiple Forms." *Applied Psychological Measurement*, **47**(3), 200–220. doi:10.1177/01466216231151702.

Bock RD, Aitkin M (1981). "Marginal Maximum Likelihood Estimation of Item Parameters: Application of an EM Algorithm." *Psychometrika*, **46**(4), 443–459. doi:10.1007/bf02293801.

Brennan RL, Wang T, Kim S, Seol J (2009). ***Equating Recipes***. CASMA, University of Iowa, URL https://education.uiowa.edu/casma/computer-programs.

Butts CT (2008). "Social Network Analysis with **sna**." *Journal of Statistical Software*, **24**(6), 1–51. doi:10.18637/jss.v024.i06.

Butts CT (2024). ***sna**: Tools for Social Network Analysis*. doi:10.32614/CRAN.package.sna. R package version 2.8.

Chalmers RP (2012). "**mirt**: A Multidimensional Item Response Theory Package for the R Environment." *Journal of Statistical Software*, **48**(6), 1–29. doi:10.18637/jss.v048.i06.

Donoghue JR, Isham SP (1998). "A Comparison of Procedures to Detect Item Parameter Drift." *Applied Psychological Measurement*, **22**(1), 33–51. doi:10.1177/01466216980221002.

Haberman SJ (2009). "Linking Parameter Estimates Derived From an Item-Response Model Through Separate Calibrations." *ETS Research Report Series*, **2009**, i–9. doi:10.1002/j.2333-8504.2009.tb02197.x.

Haebara T (1980). "Equating Logistic Ability Scales by a Weighted Least Squares Method." *Japanese Psychological Research*, **22**(3), 144–149. doi:10.4992/psycholres1954.22.144.

Han KT (2009). "**IRTEQ**: Windows Application That Implements Item Response Theory Scaling and Equating." *Applied Psychological Measurement*, **33**(6), 491–493. doi:10.1177/0146621608319513.

Kolen MJ, Brennan RL (2014). *Test Equating, Scaling, and Linking: Methods and Practices*. 3rd edition. Springer-Verlag, New York.

Loyd BH, Hoover HD (1980). "Vertical Equating Using the Rasch Model." *Journal of Educational Measurement*, **17**(3), 179–193. doi:10.1111/j.1745-3984.1980.tb00825.x.

Magis D, Béland S, Tuerlinckx F, De Boeck P (2010). "A General Framework and an R Package for the Detection of Dichotomous Differential Item Functioning." *Behavior Research Methods*, **42**(3), 847–862. doi:10.3758/brm.42.3.847.

Mislevy RJ, Bock RD (1990). ***BILOG** 3. Item Analysis and Test Scoring with Binary Logistic Models*. Mooresville.

Ogasawara H (2001). "Standard Errors of Item Response Theory Equating/Linking by Response Function Methods." *Applied Psychological Measurement*, **25**, 53–67. doi:10.1177/01466216010251004.

R Core Team (2025). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. doi:10.32614/R.manuals. URL https://www.R-project.org/.

Rizopoulos D (2006). "**ltm**: An R Package for Latent Variable Modelling and Item Response Theory Analyses." *Journal of Statistical Software*, **17**(5), 1–25. doi:10.18637/jss.v017.i05.

Robitzsch A (2025). ***sirt**: Supplementary Item Response Theory Models*. doi:10.32614/CRAN.package.sirt. R package version 4.2-133.

Robitzsch A, Kiefer T, Wu M (2025). ***TAM**: Test Analysis Modules*. doi:10.32614/CRAN.package.tam. R package version 4.3-25.

Stocking ML, Lord FM (1983). "Developing a Common Metric in Item Response Theory." *Applied Psychological Measurement*, **7**(2), 201–210. doi:10.1177/014662168300700208.

Weeks JP (2010). "**plink**: An R Package for Linking Mixed-Format Tests Using IRT-Based Methods." *Journal of Statistical Software*, **35**(12), 1–33. doi:10.18637/jss.v035.i12.

Affiliation:

Michela Battauz
Departimento di Scienze Economiche e Statistiche
Università degli Studi di Udine
via Tomadini 30/A
33100 Udine, Italy
E-mail: michela.battauz@uniud.it

Journal of Statistical Software https://www.jstatsoft.org/
published by the Foundation for Open Access Statistics https://www.foastat.org/
December 2025, Volume 115, Issue 11 *Submitted:* 2024-10-18
doi:10.18637/jss.v115.i11 *Accepted:* 2025-05-19