The Stata Journal (2022)
22, Number 2, pp. 243–268 DOI: 10.1177/1536867X221106368

uirt: A command for unidimensional IRT modeling

Bartosz Kondratek
Educational Research Institute
Warsaw, Poland
b.kondratek@ibe.edu.pl

Abstract. In this article, I introduce the `uirt` command, which allows one to estimate parameters of a variety of unidimensional item response theory models (two-parameter logistic model, three-parameter logistic model, graded response model, partial credit model, and generalized partial credit model). `uirt` has extended item-fit analysis capabilities, features multigroup modeling, allows testing for differential item functioning, and provides tools for generating plausible values with a latent regression conditioning model. I provide examples to illustrate cases where `uirt` can be especially useful in conducting analyses within the item response theory approach.

Keywords: st0670, uirt, uirt_theta, uirt_icc, uirt_dif, uirt_chi2w, uirt_sx2, uirt_esf, uirt_inf, item response theory, item-fit, unidimensional item response theory models, differential item functioning, partial credit model, plausible values

1 Introduction

Item response theory (IRT) is a family of latent-variable models used to explain test behavior by explicitly distinguishing item properties from the properties of test takers. IRT data analysis is a common approach in psychological, educational, medical, and sociological research, wherever the collected data are of the form of responses to a psychometric test. Test construction, computerized adaptive testing, analysis of incomplete testing designs, test equating, and differential item functioning (DIF) analysis are just a few applications where IRT models are especially useful. Versatility of IRT stems from the fact that it naturally handles data missingness and allows controlling for unreliability of the measurement.

The rising popularity of IRT has been accompanied by the rise of Stata commands to conduct IRT-related tasks. In older versions of Stata, users could perform some unidimensional IRT analyses by formulating the IRT model in terms of a generalized linear mixed-effects model (Zheng and Rabe-Hesketh 2007). In Stata 14, a built-in `irt` command was introduced that adapts the `gsem` command to fit different types of IRT models and provides postestimation commands that allow one to plot many IRT-related graphs. In Stata 16, capabilities of `irt` were expanded to include multigroup models and thus allow testing for DIF within the IRT framework.

However, there are some limitations to the native `irt` of Stata. There is currently no item-fit analysis available. The command runs into convergence issues when fitting

the three-parameter logistic model (3PLM). DIF analysis does not provide the effect size measures. Users cannot obtain plausible values (PVs) for the test takers, which are a standard tool to control for measurement error in secondary analyses regarding latent traits (Wu 2005). The `uirt` command addresses all of these issues. Additionally, `uirt` is written in Mata, and it requires Stata 10 to run.

This article will explain some of the inner workings of `uirt` and illustrate its usage with `masc2.dta` (De Boeck and Wilson 2004). This particular dataset is used in many examples presented in the help files for IRT and also in a book by Raykov and Marcoulides (2018), which is dedicated to the topic of IRT in Stata. The presentation is structured in a manner that reflects a natural order of doing IRT analysis. We start with fitting different IRT models in increasing order of complexity: one-parameter logistic model (1PLM), two-parameter logistic model (2PLM), hybrid 2PLM–3PLM, and, finally, a two-group hybrid model. This is accompanied with likelihood-ratio (LR) tests of nested models that verify overall model improvement and item-level model fit analysis. Afterward, we analyze the data for the presence of DIF. And finally, a set of PVs conditioned on latent regression is generated to allow for secondary analysis. Each example is preceded with a brief theoretical introduction.

2 The uirt command

2.1 Description

`uirt` is a command for fitting a variety of unidimensional IRT models (2PLM, 3PLM, graded response model, partial credit model, and generalized partial credit model). It features multigroup modeling, DIF analysis, item-fit analysis, and generating PVs conditioned via latent regression. `uirt` implements the expectation maximization (EM) algorithm (Dempster, Laird, and Rubin 1977) in the form of marginal maximum-likelihood estimation proposed by Bock and Aitkin (1981) with normal Gauss–Hermite quadrature. The LR test is used for DIF testing, and model-based P-DIF effect-size measures are provided (Wainer 1993). Generating PVs is performed by adapting a Markov chain Monte Carlo method developed for IRT models by Patz and Junker (1999). Observed response proportions are plotted against the item characteristic curves to allow for detailed graphical item-fit analysis. Two item-fit statistics are available: $S\text{-}X^2$ by Orlando and Thissen (2000) and χ^2_w developed by the author (Kondratek Forthcoming).

2.2 Syntax

`uirt` [*varlist*] [*if*] [*in*] [, `pcm(`*varlist*`)` `gpcm(`*varlist*`)`
`guessing(`*varlist*[, *opts*]`)` `group(`*varname*[, *opts*]`)` `icc(`*varlist*[, *opts*]`)`
`chi2w(`*varlist*[, *opts*]`)` `sx2(`*varlist*[, *opts*]`)` `theta(`[*nv1 nv2*][, *opts*]`)`
`fix(`[*opts*]`)` `init(`[*opts*]`)` `nip(#)` `nit(#)` `ninrf(#)` `crit_ll(#)`
`crit_par(#)` `errors(`*string*`)` `priors(`*varlist*[, *opts*]`)` `notable` `noheader`
`trace(#)`]

2.3 Options

A short description of `uirt` options is presented in table 1. To see the extended descriptions type, `help uirt`.

Table 1. Options of `uirt`

Option	Description
Models	
`pcm(`*varlist*`)`	items to fit with the partial credit model
`gpcm(`*varlist*`)`	items to fit with the generalized partial credit model
`guessing(`*varlist*[, *opts*]`)`	items to attempt fitting with the 3PLM
opts:	
`attempts(#)`	maximum number of attempts to fit a 3PLM
`lrcrit(#)`	significance level for LR test comparing 2PLM against 3PLM
Multigroup	
`group(`*varname*[, *opts*]`)`	set the group membership variable
opts:	
`reference(#)`	set the value of the reference group
`dif(`*varlist*`)`	items to test for DIF
`free`	free the estimation of parameters of reference group
`slow`	suppress a speedup of EM for the multigroup estimation

Continued on next page

Table 1 (*continued*)

Option, continued	Description, continued
ICC	
`icc(`*varlist*`[ , `*opts*`])`	items to create item characteristic curve (ICC) graphs
opts:	
`bins(`#`)`	number of ability intervals for observed proportions
`noobs`	suppress plotting observed proportions
`pv`	use PVs to compute observed proportions
`pvbin(`#`)`	number of PVs in each bin
`colors(`*string*`)`	list of colors to override default colors of ICC lines
`tw(`*twoway_options*`)`	`graph twoway` options to override default graph layout
`format(`*string*`)`	file format for ICC graphs (`png`, `gph`, `eps`)
`prefix(`*string*`)`	set prefix of filenames
`suffix(`*string*`)`	set suffix of filenames
`cleargraphs`	suppress storing graphs in Stata memory
Item fit	
`chi2w(`*varlist*`[ , `*opts*`])`	items to compute χ^2_w item-fit statistic
opts:	
`bins(`#`)`	number of ability intervals for computation of χ^2_w
`npqmin(`#`)`	minimum expected number of observations in ability intervals (NPQ)
`npqreport`	report information about minimum NPQ in ability intervals
`sx2(`*varlist*`[ , `*opts*`])`	dichotomous items to compute S-X^2 item-fit statistic
opts:	
`minfreq(`#`)`	minimum expected number of observations in ability intervals (NP and NQ)

Continued on next page

Table 1 (*continued*)

Option, continued	Description, continued
Theta and PVs	
`theta(`[*nv1 nv2*] [`,` *opts*]`)`	declare variables to be added to the dataset
opts:	
`eap`	create expected a posteriori (EAP) estimator of θ and its standard error
`nip(`#`)`	number of Gauss–Hermite quadrature points used when calculating EAP and its standard error
`pv(`#`)`	number of PVs added to the dataset
`pvreg(`*str*`)`	define regression for conditioning PVs
`suffix(`*name*`)`	specify a suffix used in naming EAP, PVs, and ICC graphs
`scale(`#`,`#`)`	scale parameters (m, sd) of θ in reference group
`skipnote`	suppress adding notes to newly created variables
Fixing and initiating	
`fix(`[*opts*]`)`	declare parameters to fix
opts:	
`prev`	fix item parameters on estimates from previous `uirt` run (active estimation results)
`from(`*name*`)`	fix item parameters on estimates from `uirt` run stored in memory
`usedist`	fix group parameters on estimates from previous `uirt` run
`imatrix(`*name*`)`	matrix with item parameters to be fixed
`dmatrix(`*name*`)`	matrix with group parameters to be fixed
`cmatrix(`*name*`)`	matrix with item category values
`miss`	allow `imatrix()` to have missing entries
`init(`[*opts*]`)`	declare parameter starting values
opts:	
`prev`	initiate item parameters on estimates from previous `uirt` run (active estimation results)
`from(`*name*`)`	initiate item parameters on estimates from `uirt` run that is stored in memory
`usedist`	initiate group parameters on estimates from previous `uirt` run
`imatrix(`*name*`)`	matrix with starting values of item parameters
`dmatrix(`*name*`)`	matrix with starting values of group parameters
`miss`	allow `imatrix()` to have missing entries

Continued on next page

Table 1 (*continued*)

Option, continued	Description, continued
EM control	
`nip(#)`	number of Gauss–Hermite quadrature points used in EM algorithm
`nit(#)`	maximum number of iterations of EM algorithm
`ninrf(#)`	set the maximum number of iterations of Newton–Raphson–Fisher algorithm within M-step
`crit_ll(#)`	stopping rule—relative change in logL between EM iterations
`crit_par(#)`	stopping rule—maximum absolute change in parameter values between EM iterations
`errors(`*string*`)`	method for computation of standard errors (`cdm`, `rem`, `sem`, `cp`)
`priors(`*varlist*`[ , `*opts*`])`	declare dichotomous items to estimate with priors
opts:	
`anormal(#,#)`	parameters of normal prior for discrimination parameter
`bnormal(#,#)`	parameters of normal prior for difficulty parameter
`cbeta(#,#)`	parameters of beta prior for pseudoguessing parameter
Reporting	
`notable`	suppress coefficient table
`noheader`	suppress model summary
`trace(#)`	control log display after each iteration

2.4 Postestimation

Some `uirt` options are also available as separate postestimation commands (table 3), so it is possible to use them after `uirt` model parameters are estimated. For example, instead of typing

```
. webuse masc1
. uirt q*, sx2(*) icc(q6, bins(200)) theta(, eap)
```

one might split the analysis into four steps with the same result:

```
. uirt q*
. uirt_sx2 *
. uirt_icc q6, bins(200)
. uirt_theta, eap
```

Running these postestimation commands only once after `uirt` may take more time to execute than invoking them as `uirt` options. However, these postestimation commands may become handy when one anticipates using them multiple times after a given `uirt` run or using them according to the results of the intermediate steps of the analysis. For example, a reasonable workflow would be first to check the item-fit statistics and then to inspect the ICC curves plotted against observed response proportions only for those items that produced statistically significant misfit. Such a stepwise approach, which relies on postestimation commands rather than `uirt` options, will be adopted when presenting examples of `uirt` usage throughout the article.

Table 2. Postestimation commands of `uirt`

Command	Description
`uirt_theta`	add EAP estimator of θ or draw PVs
`uirt_icc`	create ICC plots and perform graphical item-fit analysis
`uirt_dif`	perform DIF analysis (two-group models)
`uirt_chi2w`	compute χ^2_w item-fit statistic
`uirt_sx2`	compute S-X^2 item-fit statistic (dichotomous items)
`uirt_esf`	create expected score function plots
`uirt_inf`	create information function plots

3 Item-fit analysis

3.1 Background

Let $\boldsymbol{y} = (y_1, \ldots, y_n)$ be a realized value of an item response vector $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$. In unidimensional multigroup IRT models, the probability of observing $\boldsymbol{y}$ for a test-taker sampled from group g is expressed as

$$P(\boldsymbol{y}|g, \boldsymbol{\beta}) = \int \prod_{j=1}^{n} f_j(y_j, \theta, \boldsymbol{\beta}_j)\Psi_g(\theta)d\theta \tag{1}$$

where f_j is a function that describes the probability of observing an item response y_j conditional on the latent trait level of the test-taker, θ, and Ψ_g is the a priori distribution of θ in group g. The shape of f_j depends on a vector of item parameters $\boldsymbol{\beta}_j$. In some parts of the article, the parameters $\boldsymbol{\beta}_j$ are dropped to simplify the notation.

Validity of inferences made with any model is conditional on the extent to which it fits the data. The structure of data that are gathered in psychometric testing and the form of the model presented in (1) imply an item-by-item strategy of fit assessment in IRT. If a particular item does not fit the data well, a researcher can choose another family of f_j for that item, discard the item from the analyses, or, when it is in a test-development stage, modify the test item.

The `uirt` command provides the user with both graphical and statistical tools that allow to assess item fit. Similarly to other available approaches (see Swaminathan, Hambleton, and Rogers [2007]), `uirt` investigates the concordance between observed item responses and the expectations derived from $\widehat{f}_j$ over some predefined ranges of the measured trait. However, the way this task is accomplished in `uirt` is unique, so it requires some explication.

Let us begin with the graphical item-fit analysis. It is accomplished by computing the observed proportions of responses to each item category c_j, $c_j \in \{0, \ldots, \max(Y_j)\}$, over quantile-based groups of θ, and plotting them against the estimated response functions $\widehat{f}_j(c_j, \theta)$. After option `icc()` (or postestimation command `uirt_icc`) is called, the default behavior of `uirt` is to split the latent trait into $\Delta_1, \ldots, \Delta_r$ intervals that are equiprobable in the reference group ($g = 0$):

$$\Delta_1 \cup \cdots \cup \Delta_r = \mathbb{R}; \quad \Delta_k \cap \Delta_l = \emptyset, \quad \text{for} \quad k \neq l; \quad P(\theta \in \Delta_k | G = 0) = \frac{1}{r}$$

The `bins(#)` option modifies the number of intervals, with $r = 100$ being the default value. The posterior probability that the latent trait of a test-taker i falls into interval Δ_k is computed as

$$\widehat{\tau}_{ki} = P(\theta | \boldsymbol{y}_i, \Delta_k, G = g) = \frac{\int_{\Delta_k} \prod_{j=1}^{n} \widehat{f}_j(y_j, \theta) \widehat{\Psi}_g(\theta) d\theta}{\int \prod_{j=1}^{n} \widehat{f}_j(y_j, \theta) \widehat{\Psi}_g(\theta) d\theta}$$

with numerical integration performed by Gauss–Hermite quadrature used for the denominator and Gauss–Legendre quadrature for the numerator.

The observed proportion for response category c_j in interval Δ_k is obtained as a weighted mean over all m test-takers who responded to item j,

$$_{c_j}\widehat{O}_k = \frac{\sum_{i=1}^{m} \mathbb{1}_{c_j}(y_{ij}) \widehat{\tau}_{ki}}{\sum_{i=1}^{m} \widehat{\tau}_{ki}} \tag{2}$$

where $\mathbb{1}_{c_j}(y_{ij})$ is an indicator function, equal to 1 if $y_{ij} = c_j$ and otherwise 0. The observed proportions (2) are plotted against the response category curves $\widehat{f}_j(c_j, \theta)$, as will be illustrated in the following examples.

To perform a statistical test for an item fit, we can use a similar strategy of weighting by the a posteriori group membership probability. However, instead of category proportions, the item mean is computed,

$$_j\widehat{O}_k = \frac{\sum_{i=1}^{m} y_{ij} \widehat{\tau}_{ki}}{\sum_{i=1}^{m} \widehat{\tau}_{ki}}$$

and it is paired with the model-based expected item mean,

$$_j\widehat{E}_k = \frac{\sum_{i=1}^{m} {}_j\widehat{e}_{ki} \widehat{\tau}_{ki}}{\sum_{i=1}^{m} \widehat{\tau}_{ki}}$$

where ${}_j\widehat{e}_{ki}$ is the fit model-based item mean in interval Δ_k conditional on observing a response vector $\boldsymbol{y}_i$ without the item j:

$${}_j\widehat{e}_{ki} = \frac{\int_{\Delta_k} \left\{ \sum_{c_j} c_j \widehat{f}_j(c_j, \theta) \right\} \prod_{h \neq j} \widehat{f}_h(y_h, \theta) \widehat{\Psi}_g(\theta) d\theta}{\int_{\Delta_k} \prod_{h \neq j} \widehat{f}_h(y_h, \theta) \widehat{\Psi}_g(\theta) d\theta}$$

To test a null hypothesis that the vector of observed means is equal to the expected means vector, one uses a Wald-type test statistic (Kondratek Forthcoming),

$$\chi^2_w = \left({}_j\widehat{\boldsymbol{O}} - {}_j\widehat{\boldsymbol{E}} \right) {}_j\widehat{\boldsymbol{V}}^{-1} \left({}_j\widehat{\boldsymbol{O}} - {}_j\widehat{\boldsymbol{E}} \right)^T$$

with an asymptotic covariance matrix ${}_j\widehat{\mathbf{V}} = [{}_j\widehat{v}_{kl}]_{r \times r}$, where the klth element is

$${}_j\widehat{v}_{kl} = \frac{\sum_{i=1}^{m} \widehat{\tau}_{ki} \widehat{\tau}_{li} \left(y_{ij} - {}_j\widehat{O}_l \right)}{\left(\sum_{i=1}^{m} \widehat{\tau}_{ki} \right) \left(\sum_{i=1}^{m} \widehat{\tau}_{li} \right)}$$

The χ^2_w statistic is assumed to be asymptotically chi-squared distributed with $r - q$ degrees of freedom, where q is the number of estimated parameters of $\widehat{f}_j$. The default `uirt` setting for the number of ability intervals used to compute χ^2_w is either $r = 3$ or a minimal value that leaves a single degree of freedom after accounting for the number of estimated item parameters. The boundaries of intervals Δ_k are constructed individually for each item to obtain a high number of observations relative to the expected item mean within each interval.

The χ^2_w statistic of `uirt` is general: it can be applied to polytomous items and to datasets with missing item responses. If the data are complete and test items are dichotomous, an approach to item-fit testing with grouping over observed scores, rather than θ, can be applied. The S-X^2 item-fit statistic by Orlando and Thissen (2000) is one of the most renowned test statistics for dichotomous items that employs observed score grouping. It is a Pearson X^2 statistic that uses the algorithm of Lord and Wingersky (1984) to obtain the expected proportion of correct responses at each observed score group. S-X^2 can be computed in `uirt` by calling the `sx2()` option or the `uirt_sx2` postestimation command.

3.2 Example

In this section, we will examine the fit of three IRT models using `masc2.dta`. First, a 1PLM will be applied to all items. Then, all items will be modeled with 2PLM. Finally, an attempt to fit a 3PLM to selected items will be performed. These nested models will be compared with an LR test, and specific emphasis will be placed on item-fit analysis.

1PLM is a dichotomous case of a partial credit model, so to fit 1PLM to all items, we have to use the `pcm(*)` option (an asterisk in `uirt` options or postestimation commands is shorthand for a *varlist* consisting of all items declared in the main `uirt` *varlist*):

```
. webuse masc2, clear
(Data from De Boeck & Wilson (2004))
. uirt q*, pcm(*)
ITERATION=  1;logL=     -7801.7990
```

(*output omitted*)

```
Unidimensional item response theory model          Number of obs     =       1500
                                                   Number of items   =       9
                                                   Number of groups  =       1
Log likelihood =      -7791.4953
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
group_1						
mean_theta	0	(omitted)				
sd_theta	1	(constrained)				
q1						
pcm_a	.8444907	.0334272	25.26	0.000	.7789745	.9100068
pcm_b1	-.6346766	.0610899	-10.39	0.000	-.7544106	-.5149426
q2						
pcm_a	.8444907	(constrained)				
pcm_b1	.0247067	.0596748	0.41	0.679	-.0922538	.1416672
q3						
pcm_a	.8444907	(constrained)				
pcm_b1	-1.732418	.0706107	-24.53	0.000	-1.870813	-1.594024
q4						
pcm_a	.8444907	(constrained)				
pcm_b1	.4441906	.0603712	7.36	0.000	.3258653	.562516
q5						
pcm_a	.8444907	(constrained)				
pcm_b1	1.89298	.0728852	25.97	0.000	1.750128	2.035833
q6						
pcm_a	.8444907	(constrained)				
pcm_b1	.9077963	.0626064	14.50	0.000	.7850901	1.030503
q7						
pcm_a	.8444907	(constrained)				
pcm_b1	1.287514	.0656325	19.62	0.000	1.158877	1.416151
q8						
pcm_a	.8444907	(constrained)				
pcm_b1	-2.169308	.0772453	-28.08	0.000	-2.320706	-2.01791
q9						
pcm_a	.8444907	(constrained)				
pcm_b1	-1.183398	.0646675	-18.30	0.000	-1.310144	-1.056653

After the model is fit, we can inspect the item-fit statistics. To compute χ^2_w, we use the following postestimation command:

```
. uirt_chi2w *
```

	chi2W	p-val	df	n_par
q1	13.689	0.0002	1	2
q2	4.019	0.1341	2	1
q3	0.292	0.8640	2	1
q4	2.180	0.3362	2	1
q5	6.397	0.0408	2	1
q6	9.358	0.0093	2	1
q7	10.229	0.0060	2	1
q8	9.590	0.0083	2	1
q9	8.423	0.0148	2	1

Additionally, for a dataset consisting of dichotomous items with no missings, the classical S-X^2 item-fit statistic is also available:

```
. uirt_sx2 *
```

	SX2	p-val	df	n_par
q1	20.510	0.0022	6	2
q2	10.957	0.1405	7	1
q3	1.897	0.9653	7	1
q4	6.505	0.4821	7	1
q5	4.934	0.5523	6	1
q6	16.270	0.0228	7	1
q7	20.302	0.0024	6	1
q8	16.211	0.0127	6	1
q9	14.458	0.0436	7	1

Both χ^2_w and S-X^2 item-fit statistics indicate that responses to five items, `q1` and `q6`–`q9`, significantly deviated from the 1PLM. The χ^2_w statistic is additionally pointing to a misfit for `q5`. It is expected that any statistical model will provide a significant misfit signal when it is used to model real data, given a large enough sample size. To assess the nature of a detected misfit more precisely, we will look at the graphical item-fit information with the `uirt_icc` postestimation command, which will plot the ICCs against observed proportions and save them in the working directory:

```
. uirt_icc *, tw(xtitle({&theta}) scheme(sj)
> title(masc2.dta: Single-group 1PL model)) color(gs2 gs6)
```

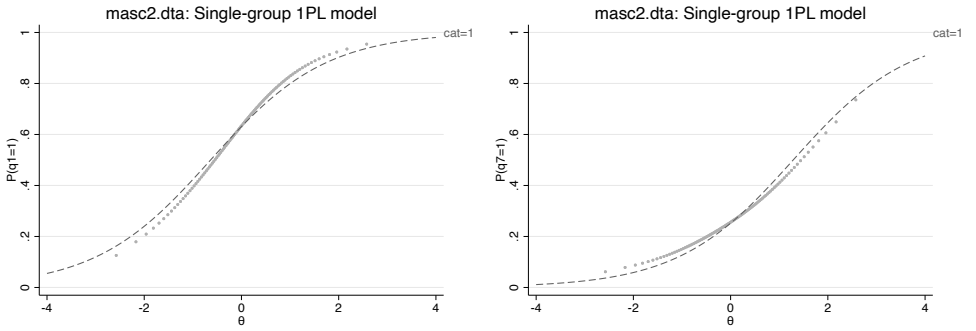

Figure 1. Graphical item-fit analysis of items `q1` and `q7`—1PLM

Graphs for items `q1` and `q7` are presented in figure 1. The patterns of deviance between observed proportions of correct responses and the fit 1PLM curves reveal that the common discrimination constraint that is imposed on items in 1PLM might be responsible for the misfit. It seems that item `q1` would be better fit with a steeper response curve and item `q7` with a flatter one. These patterns of misfit suggest that a 2PLM might be more suitable for the data. Before we proceed to another model, let us store the estimates for future use:

```
. estimates store one_plm_1gr
```

2PLM is the default in `uirt`, so to fit the 2PLM, type

```
. uirt q*
```

(*output omitted*)

Instead of looking at the default results table, which is lengthy, we will inspect the estimated item parameters stored in the `e(item_par)` matrix:

```
. matrix list e(item_par)
e(item_par)[9,2]
                  a           b
q1:2plm   1.2710508    -.480947
q2:2plm     .719883   .02794541
q3:2plm   .87421142  -1.6871188
q4:2plm   .73164289   .49697571
q5:2plm   1.0841777   1.5767331
q6:2plm   .95581212    .8275127
q7:2plm   .54778655   1.8442398
q8:2plm   1.1244723  -1.7600663
q9:2plm   .61388172  -1.5341383
```

There is a noticeable spread in estimated discrimination parameters, with the biggest changes relative to 1PLM happening to the previously discussed pair of items `q1` and `q7`.

To compare the 2PLM with 1PLM, we can conduct an LR test,

```
. estimates store two_plm_1gr
. lrtest one_plm_1gr two_plm_1gr
Likelihood-ratio test
Assumption: one_plm_1gr nested within two_plm_1gr
 LR chi2(8) =  42.06
Prob > chi2 = 0.0000
```

to conclude that, indeed, a model with item-specific discrimination parameters provides a significantly better overall model fit.

Let us now inspect the model fit at an item level for the six items that produced significant misfit under the 1PLM with the χ^2_w statistic (S-X^2 provides similar results):

```
. uirt_chi2w q1 q5-q9
```

	chi2W	p-val	df	n_par
q1	1.071	0.3008	1	2
q5	2.861	0.0907	1	2
q6	7.305	0.0069	1	2
q7	0.798	0.3717	1	2
q8	3.177	0.0747	1	2
q9	2.317	0.1279	1	2

We see that five previously misfitting items do not give significant test results. However, item `q6` still does. We will thus perform a graphical item-fit analysis on item `q6`, also including the previously analyzed pair `q1` and `q7`:

```
. uirt_icc q6 q1 q7, tw(xtitle({&theta}) scheme(sj)
> title(masc2.dta: Single-group 2PL model)) color(gs2 gs6)
```

Resulting graphs are presented in figure 2. Comparison of graphs for `q1` and `q7` (upper panel in figure 2) with their counterparts obtained under 1PLM (figure 1) confirms the improvement of fit that followed after discrimination parameters were freely fit in 2PLM. However, for item `q6` (lower panel in figure 2), we observe a deviance between observed proportions of correct responses and the fit 2PLM curve at the extreme values of the latent trait. The pattern of misfit suggests that a guessing behavior may be present. It is reasonable to expect guessing behavior to occur in a multiple-choice cognitive test. Therefore, we will repeat the analysis trying to fit a 3PLM to the data.

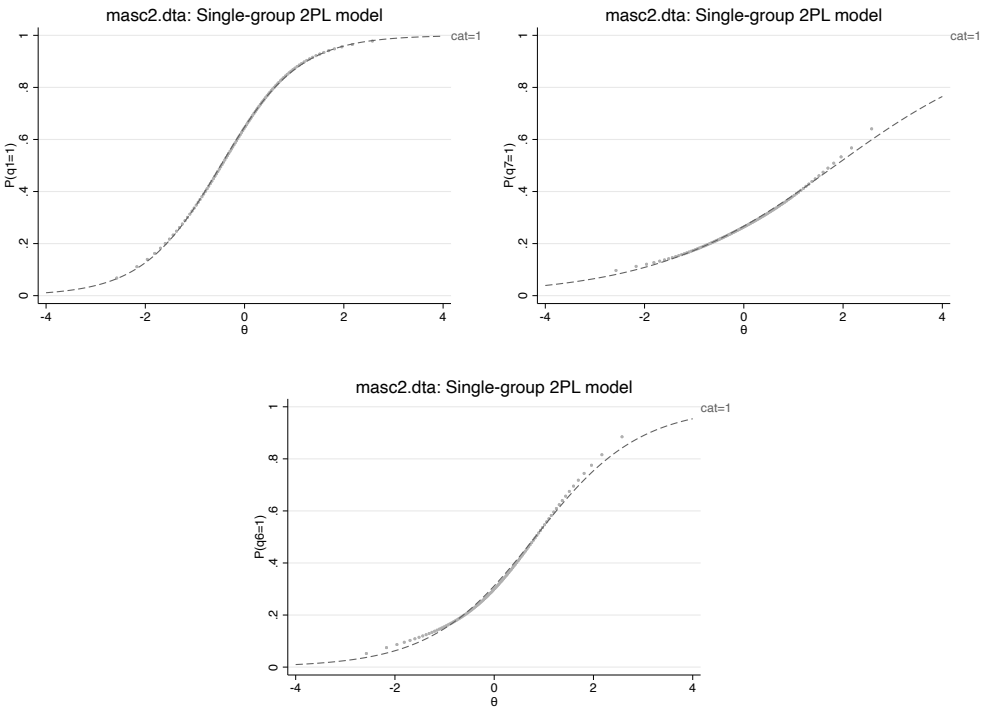

Figure 2. Graphical item-fit analysis of items `q1`, `q6`, and `q7`—2PLM

Fitting a 3PLM to all items of a test may be impossible without imposing priors on the pseudoguessing parameter, because the likelihood function of this parameter is very flat. One can impose prior distributions on parameters of dichotomous items in `uirt` with the `priors()` option. The penalization of likelihood achieved with proper item parameter priors may ascertain convergence of the estimates, but it comes at a cost. Neither the reported log likelihood nor the reported model degrees of freedom account for this penalization. Statistical inference regarding models fit in such a way (item-fit analysis, DIF, LR tests) would thus be prone to bias.

The `uirt` command also introduces an explorative procedure to fit 3PLM without resorting to priors. It results in fitting 3PLM only to those items from `guessing(`*varlist*`)` that converge to 3PLM in a satisfying fashion. The algorithm works as follows. For each 3PLM-candidate item, `uirt` starts with a 2PLM and performs multiple attempts of fitting the 3PLM. The 3PLM attempts are followed by checks on parameter behavior with two criteria to decide whether to keep the item as 2PLM or to go with 3PLM. The first criterion is convergence. An item stays 2PLM if the parameter estimates change too rapidly or if the discrimination or the pseudoguessing parameter turns negative. The second criterion is a result of an "LR test" performed after a single EM iteration. If the model likelihood does not improve significantly, the item stays 2PLM. The maximum

number of attempts of fitting a 3PLM is controlled by `attempts(#)`, and the LR sensitivity is controlled by `lrcrit(#)`.

Let us see how it works with our data:

```
. uirt q*, guessing(q*, lr(0.1))
ITERATION=  1;logL=     -7848.9807
ITERATION=  2;logL=     -7789.6335
ITERATION=  3;logL=     -7775.8430
ITERATION=  4;logL=     -7771.9527
ITERATION=  5;logL=     -7770.8774
ITERATION=  6;logL=     -7770.5808
ITERATION=  7;logL=     -7770.4986
generating starting values for guessing parameters for 9 item(s); attempt=1
Note: did not generate starting values for 8 items:  q1[LR] q2[LR] q3[LR] q4[LR]
> q5[LR] q7[LR] q8[conv] q9[conv]
ITERATION=  8;logL=     -7768.9444
  (output omitted)
ITERATION= 59;logL=     -7765.3681
generating starting values for guessing parameters for 8 item(s); attempt=4
Note: did not generate starting values for 8 items:  q1[LR] q2[LR] q3[LR] q4[LR]
> q5[LR] q7[LR] q8[conv] q9[conv]
  (output omitted)
```

The estimated item parameters in compact form are

```
. matrix list e(item_par)
e(item_par)[9,3]
                  a           b           c
q1:2plm   1.2642306  -.48093941           .
q2:2plm    .7274833   .02852978           .
q3:2plm   .87997814  -1.6786367           .
q4:2plm   .71327773   .50815061           .
q5:2plm    1.042379   1.6203599           .
q6:3plm   2.6676989   1.0292606   .17963811
q7:2plm   .55265401   1.8301782           .
q8:2plm   1.1583755  -1.7259597           .
q9:2plm   .61005359  -1.5420781           .
```

We see that the explorative algorithm has fit the 3PLM model only to the `q6` item, the one that exhibited a misfit pattern typical for the guessing behavior (figure 2). The iteration log informed us that items `q1`–`q5` and `q7` stayed 2PLM because `uirt` did not observe significant increase in likelihood, when trying to fit them as 3PLM, and items `q8`–`q9` ran into convergence issues.

Now let us inspect how the item-fit statistics of q6 have changed under the new model:

```
. uirt_chi2w q6
```

	chi2W	p-val	df	n_par
q6	0.234	0.6284	1	3

```
. uirt_sx2 q6
```

	SX2	p-val	df	n_par
q6	7.726	0.1720	5	3

Indeed, by changing the model of a single item from 2PLM to 3PLM, we have arrived at a hybrid IRT model that does not produce significant item misfit for q6, both with respect to χ^2_w and S-X^2 test statistics. The item characteristic curve for q6 plotted against observed proportions under the 3PLM is presented in figure 3. The item fit in the extreme ranges of latent trait has improved. The LR test that compares the two models also confirms that the hybrid option provides a better fit:

```
. estimates store hybrid_model_1gr
. lrtest hybrid_model_1gr two_plm_1gr
Likelihood-ratio test
Assumption: two_plm_1gr nested within hybrid_model_1gr
 LR chi2(1) =  10.20
Prob > chi2 = 0.0014
```

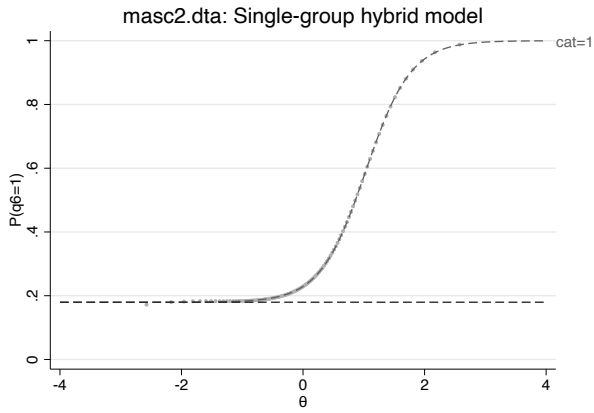

Figure 3. Graphical item-fit analysis of item q6—3PLM

4 DIF

4.1 Background

A general definition of DIF may be formulated as (see Penfield and Camilli [2007])

$$P(y_j|\theta, G) \neq P(y_j|\theta) \tag{3}$$

Equation (3) tells us that the distribution of item response, conditional on the latent trait, is different between groups. Significant DIF is a signal that some additional factors other than the latent trait influence the item response behavior and that these factors covary with the group membership. The presence of DIF leads to questions about test validity. It may indicate that the item is biased toward some group of test takers that pose a threat to test fairness; it may also reveal that the underlying latent trait is of higher dimensionality than assumed.

Because the IRT model is explicitly defined by terms that describe the item response probabilities conditional on the latent trait (1), it provides a straightforward framework for DIF analysis. An IRT model for DIF is obtained by introduction of group-specific parameters for the item that is being tested for DIF:

$$P(\boldsymbol{y}|g) = \int f_j(y_j, \theta, \boldsymbol{\beta}_{j,g}) \left\{ \prod_{h \neq j} f_h\left(y_h, \theta, \boldsymbol{\beta}_h\right) \right\} \Psi_g(\theta) d\theta \tag{4}$$

To test for the presence of DIF, you perform the LR test. It compares the restricted model (1), which has equal item parameters between groups, with the unrestricted model (4), which introduced group-specific parameters for item j. The usual scenario for DIF analysis deals with two groups, the focal and the reference group, $G \in \{f, r\}$. In such case, the null hypothesis is

$$H_0 : \boldsymbol{\beta}_{j,r} = \boldsymbol{\beta}_{j,f}$$

The alternative hypothesis is

$$H_1 : \boldsymbol{\beta}_{j,r} \neq \boldsymbol{\beta}_{j,f}$$

The number of degrees of freedom of the LR statistic is equal to the difference in the number of estimated parameters between the two models.

However, a statistically significant result of an LR test for DIF does not carry with itself any information on the actual degree to which the measurement invariance is violated. Minuscule differences in group-specific item parameters may produce a positive test result if only the sample size is large enough. A proper measure of effect size is necessary to determine whether a statistically significant DIF is of a practical importance. One of the effect size measures used in DIF analysis is the P-DIF index, in which the size of the effect is presented on the scale of the raw score of the item. The P-DIF effect size measure for IRT models was proposed by Wainer (1993):

$$\text{P-DIF}_{j,f} = \int \sum_{c_j} c_j \left\{ f_j\left(c_j, \theta, \boldsymbol{\beta}_{j,f}\right) - f_j(c_j, \theta, \boldsymbol{\beta}_{j,r}) \right\} \Psi_f(\theta) d\theta$$

The P-DIF$_{j,f}$ informs us about the expected difference between the mean of y_j in the focal group based on the item parameters estimated for the focal group and the mean of the same item in the focal group but obtained according to the parameters that are estimated for the reference group. In short, it can be described as an increase in item mean in group f due to the effect of DIF. Positive values of P-DIF$_{j,f}$ indicate that DIF "favors" the focal group (they obtain a higher score on that item after the latent trait is controlled for), and negative values mean the opposite.

Note that P-DIF$_{j,f}$ is not equal to the negative of P-DIF$_{j,r}$. If we compute P-DIF$_{j,r}$, we change not only the order in which the response functions are subtracted but also the latent trait distribution over which the integral is taken. P-DIF$_{j,f}$ and P-DIF$_{j,r}$ weight the local differences between response functions by the density of the distribution of the chosen group, so, in general, they do not produce exactly opposite values.

`uirt` performs the LR test for DIF and computes the P-DIF$_{j,f}$ and P-DIF$_{j,r}$ measures with the `dif(`*varlist*`)` suboption of `group()` option or via the `uirt_dif` postestimation command. When DIF analysis is invoked, graphs that allow comparison of the response functions estimated separately in each group are also created. DIF analysis requires a multigroup model, with a dichotomous grouping variable declared in the `group()` option.

4.2 Example

Let us continue the analysis described in the previous example. `masc2.dta` contains a `female` indicator variable. We want to fit a multigroup IRT model with grouping on the `female` variable and inspect all items for DIF. Let us start with fitting the multigroup model. To speed up the convergence, we can use item parameters from the single-group hybrid model that are stored in memory with the `init()` option:

```
. uirt q*, init(from(hybrid_model_1gr)) group(female)
19 initial parameters of 9 requested items were found in e(item_par) matrix:
 q1 q2 q3 q4 q5 q6 q7 q8 q9
ITERATION=  1;logL=     -7765.3680
```

(*output omitted*)

```
Unidimensional item response theory model          Number of obs     =       1500
                                                   Number of items   =       9
                                                   Number of groups  =       2
Log likelihood =      -7760.0188
```

	Coefficient	Std. err.	z	P>|z|	[95% conf.	interval]
group_0						
mean_theta	0	(omitted)				
sd_theta	1	(constrained)				
group_1						
mean_theta	-.2113879	.0693722	-3.05	0.002	-.3473549	-.0754209
sd_theta	.9383499	.0674802	13.91	0.000	.8060912	1.070609

(*output omitted*)

q9						
2plm_a	.641402	.0961641	6.67	0.000	.4529238	.8298802
2plm_b	-1.576555	.215795	-7.31	0.000	-1.999506	-1.153605

The table that displays model parameters now includes group-specific parameters of latent trait distribution. Parameters of the reference group (`female` = 0) are fixed, and parameters of the focal group (`female` = 1) are estimated freely from the data. The estimates suggest that the mean of latent distribution of females is 0.21 below the mean of the reference group. We can run an LR test to confirm that the multigroup model better explains the data:

```
. estimates store hybrid_model_2gr
. lrtest hybrid_model_1gr hybrid_model_2gr
Likelihood-ratio test
Assumption: hybrid_model_1gr nested within hybrid_model_2gr
 LR chi2(2) =  10.70
Prob > chi2 = 0.0048
```

Now we can proceed to perform DIF analysis with the `uirt_dif` postestimation command:

```
. uirt_dif *, tw(xtitle({&theta}) scheme(sj)
> title(masc2.dta: Two-group hybrid model)) color(gs2 gs6)

-------------------------------------------------------------------------
DIF analysis of item q1 (GR: female=0 , GF: female=1)
                 GR        GF
         a    1.2426    1.4510
         b   -0.4231   -0.8063
     E(parR,GR)     E(parF,GR)     E(parR,GF)     E(parF,GF)
         0.5996         0.7000         0.5269         0.6290
        LR   p-value  P-DIF|GR  P-DIF|GF
   16.0729    0.0003    0.1004   -0.1020

-------------------------------------------------------------------------
DIF analysis of item q2 (GR: female=0 , GF: female=1)
```

(*output omitted*)

	LR	P>LR	P-DIF|GR	P-DIF|GF	E(R|GR)	E(F|GR)	E(R|GF)	E(F|GF)
q1	16.073	0.000	0.100	-0.102	0.600	0.700	0.527	0.629
q2	2.578	0.276	0.022	-0.032	0.501	0.523	0.458	0.490
q3	7.134	0.028	0.052	-0.051	0.773	0.824	0.742	0.793
q4	11.637	0.003	-0.053	0.069	0.464	0.411	0.442	0.373
q5	9.631	0.008	-0.063	0.051	0.237	0.174	0.208	0.157
q6	2.099	0.552	0.037	-0.025	0.354	0.391	0.300	0.325
q7	4.018	0.134	-0.048	0.045	0.313	0.265	0.290	0.245
q8	4.320	0.115	-0.035	0.040	0.859	0.824	0.848	0.808
q9	3.286	0.193	-0.042	0.043	0.736	0.694	0.715	0.672

The `uirt_dif` postestimation command uses active estimates as a null model, fits an alternative model with group-specific item parameters, and compares the two models with the LR test. This is done on an item-by-item basis with detailed item results displayed at each step. At the end, a summarizing table is printed with results of LR tests, the P-DIF effects computed for both the reference and the focal group, and four combinations of marginal means.

We see that DIF was significant for four out of nine items, with the highest effect size for item `q1`. Item `q1` is estimated to be 10 percentage points easier in the focal group because of the DIF effect. The graph illustrating this case is presented in figure 4. `uirt_dif` saves such graphs in the working directory for all items that are declared in the *varlist*.

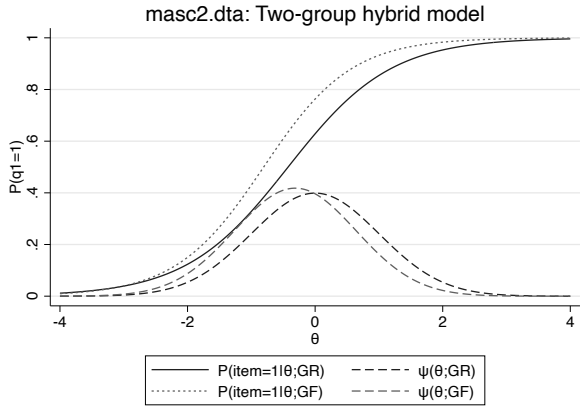

Figure 4. DIF graph for item `q1`

5 PVs

5.1 Background

The a posteriori density of the latent trait upon observing $\boldsymbol{y}$ under the IRT model (1) is

$$P(\theta|\boldsymbol{y}, g, \boldsymbol{\beta}) = \frac{\prod_{j=1}^{n} f_j(y_j, \theta, \boldsymbol{\beta}_j)\Psi_g(\theta)}{\int \prod_{j=1}^{n} f_j(y_j, \theta, \boldsymbol{\beta}_j)\Psi_g(\theta)d\theta} \tag{5}$$

Assuming the model holds, this distribution contains all the information on the latent trait of a test taker sampled from population g, who has responded $\boldsymbol{y}$. One can obtain an EAP point estimate of the latent trait, $\widehat{\theta}$, by taking the expectation from (5) with standard error equal to standard deviation of (5). The EAP estimator and its standard error are added to the dataset by `uirt` with the `theta()` option or with the `uirt_theta` postestimation command.

Using the point estimates $\widehat{\theta}$ to perform secondary data analysis regarding the latent trait is a common practice. However, such Bayesian estimators are biased toward the mean of the a priori distribution. Note that the error of measurement is not constant in IRT models (it is usually highest at the extreme values of θ), and the higher the error of $\widehat{\theta}$, the bigger the shrinkage. Therefore, the distribution of $\widehat{\theta}$ will have smaller variance in comparison with the underlying latent trait distribution, with a complicated, nonlinear relation between $\widehat{\theta}$ and θ. Furthermore, any statistic computed on $\widehat{\theta}$ will have its standard error biased toward 0 because the error of measurement is being ignored.

The drawbacks associated with using point estimates of a latent trait can be overcome by adopting a multiple imputation method described by Rubin (1987) that was developed to deal with missing data. Within the IRT approach, the missing data are the latent trait variable θ, and the multiple imputations of θ are random draws from

(5). These are called PVs. When PVs are used for analyses that relate the latent trait to some ancillary variables $\boldsymbol{x}$, these ancillary variables must be properly incorporated into the imputation model (Wu 2005). This is accomplished by a latent regression,

$$\theta = \boldsymbol{x}^T\boldsymbol{\xi} + \epsilon \tag{6}$$

where $\boldsymbol{\xi}$ is the vector of regression coefficients. This extends the IRT model (1) into

$$P(\boldsymbol{y}|g, \boldsymbol{x}, \boldsymbol{\beta}, \boldsymbol{\xi}) = \int \prod_{j=1}^{n} f_j(y_j, \theta, \boldsymbol{\beta}_j)\Psi_g(\theta, \boldsymbol{x}, \boldsymbol{\xi})d\theta$$

and the a posteriori distribution (5) becomes

$$P(\theta|\boldsymbol{y}, g, \boldsymbol{x}, \boldsymbol{\beta}, \boldsymbol{\xi}) = \frac{\prod_{j=1}^{n} f_j(y_j, \theta, \boldsymbol{\beta}_j)\Psi_g(\theta, \boldsymbol{x}, \boldsymbol{\xi})}{\int \prod_{j=1}^{n} f_j(y_j, \theta, \boldsymbol{\beta}_j)\Psi_g(\theta, \boldsymbol{x}, \boldsymbol{\xi})d\theta} \tag{7}$$

Patz and Junker (1999) and de la Torre (2009) developed Markov chain Monte Carlo techniques that enable drawing PVs from (7). These algorithms are designed to estimate all model parameters. They include chains not only for θ_i but also for the item parameters $\boldsymbol{\beta}_j$ and the structural parameters $\boldsymbol{\xi}$. Each iteration t involves three separate steps: i) sampling $\boldsymbol{\xi}^t$, ii) sampling $\boldsymbol{\beta}_j^t$ item by item, and iii) sampling θ_i^t observation by observation. In `uirt`, the Markov chains are constructed only for the last part. The chains for $\boldsymbol{\beta}_j$ are replaced by sampling from $N(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\Sigma}})$, where $\widehat{\boldsymbol{\Sigma}}$ is the estimated covariance matrix of item parameter estimates $\boldsymbol{\beta}$. The chains for $\boldsymbol{\xi}$ are replaced by their maximum likelihood estimates obtained after fitting the latent regression (6) to θ_i^t. Such modifications of the original algorithms speed up the procedure and allow one to fix the scale of the latent trait at the estimates of item and distribution parameters provided by the EM algorithm of `uirt`.

5.2 Example

After fitting a two-group model to the `masc2.dta` data in the previous example, we have learned that when the latent trait distribution of males is fixed at $N(0, 1)$, the mean of the latent trait in the group of females is -0.211 with standard error of 0.069. To illustrate the benefits of using PVs to perform secondary analyses, we will now estimate the difference in mean between the two groups with PVs using the single-group model and compare it with results obtained with the EAP point estimates of θ.

The following syntax will add the EAP estimator of the latent trait, its standard error, and a set of 10 PVs that are conditioned by a latent regression on the `female` variable:

```
. set seed 314
. estimates restore hybrid_model_1gr
(results hybrid_model_1gr are active now)
. uirt_theta, eap pv(10) pvreg(i.female)
Added variables: theta, se_theta
Generating PVs: 0%....10%....20%....30%....40%....50%....60%....70%....80%....
> 90%....100%
Added variables: pv_1 - pv_10
```

The summary statistics reveal that the EAP is shrunk toward the mean of latent distribution, while the means and standard deviations of PVs are in accordance with the underlying latent distribution:

```
. summarize pv* theta, sep(10)
```

Variable	Obs	Mean	Std. dev.	Min	Max
pv_1	1,500	.0195623	.9994535	-3.606971	2.995973
pv_2	1,500	-.0454238	1.007159	-3.429081	4.218628
pv_3	1,500	-.0202424	1.027965	-3.367804	4.4001
pv_4	1,500	.0156488	.9935948	-3.105897	3.321138
pv_5	1,500	-.0465479	.9823118	-3.339587	3.731291
pv_6	1,500	.0559545	1.00571	-2.830438	3.815922
pv_7	1,500	-.0518064	.9902592	-3.637417	2.976331
pv_8	1,500	.0095771	.9869397	-3.720683	3.302024
pv_9	1,500	-.045652	1.011767	-3.350186	3.189074
pv_10	1,500	.0113273	.9733658	-3.172166	3.395834
theta	1,500	5.30e-06	.7535171	-1.848152	1.770002

We must consider that the scale of the single-group model is different from the scale of the two-group model. The first is fixed at $N(0,1)$ globally, and the second is fixed at $N(0,1)$ within the male group. We have to rescale the current PVs, so the pooled mean and standard deviation of males align with the two-group model:

```
. local m_male=0
. local sd_male=0
. foreach pv of varlist pv* {
  2. quietly summarize `pv' if female==0
  3. local m_male=`m_male' + r(mean)
  4. local sd_male=`sd_male' + r(sd)
  5. }
. local m_male=`m_male'/10
. local sd_male=`sd_male'/10
```

```
. foreach pv of varlist pv* {
  2. quietly replace `pv' = (`pv' - `m_male')/`sd_male'
  3. }
```

Data analysis with PVs involves a two-step procedure. In the first step, each PV is analyzed separately, and in the second step, the estimates and error variances are combined into a single result according to rules provided by Rubin (1987). To perform this task, we will use the `pv` command (Macdonald 2008), available from the Statistical Software Components Archive, together with `regress` to estimate the effect for females:

```
. pv, pv(pv*): regress @pv female
command(s) run for each plausible value:
     regress @pv female
Estimates for pv_1   complete
  (output omitted)
Estimates for pv_10   complete
Number of observations: 1500
Average R-Squared: .0121505834733326

              Coef     Std Err           t     t Param       P>|t|
female  -.21461321   .07213509   -2.975157   36.521336   .00516712
 _cons  -4.293e-16   .06463344  -6.643e-15   18.879588           1
```

The effect for the `female` variable obtained with PVs is very close to the mean of latent distribution estimated directly by the EM algorithm in the two-group model. The standard error of the effect is also in accordance with the standard error obtained in the two-group model.

Let us now run a similar analysis with the EAP estimate of ability:

```
. quietly summarize theta if female==0
. quietly replace theta=(theta-r(mean))/r(sd)
. regress theta female
```

Source	SS	df	MS	Number of obs	=	1,500
				F(1, 1498)	=	9.54
Model	9.21249391	1	9.21249391	Prob > F	=	0.0020
Residual	1445.87352	1,498	.965202617	R-squared	=	0.0063
				Adj R-squared	=	0.0057
Total	1455.08601	1,499	.970704479	Root MSE	=	.98245

theta	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
female	-.1567544	.0507388	-3.09	0.002	-.256281	-.0572277
_cons	-2.78e-16	.0356137	-0.00	1.000	-.069858	.069858

We can see that when simple point estimates of a latent trait are used to infer about the difference between the groups, the estimate is considerably shrunk toward 0 (−0.16 instead of −0.21). This is accompanied with a drop in standard error (from 0.07 to 0.05), so the inference about statistical significance of the effect is not affected much. However, it is clear that if the actual size of the effect would be of importance to the researcher, the bias that results from using point estimates is not negligible.

6 Acknowledgment

Preparation of this article was made possible by the National Science Centre research grant number 2015/17/N/HS6/02965.

7 Programs and supplemental materials

To install a snapshot of the corresponding software files as they existed at the time of publication of this article, type

```
. net sj 22-2
. net install st0670     (to install program files, if available)
. net get st0670         (to install ancillary files, if available)
```

8 References

Bock, R. D., and M. Aitkin. 1981. Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika* 46: 443–459. https://doi.org/10.1007/BF02293801.

De Boeck, P., and M. Wilson. 2004. A framework for item response models. In *Explanatory Item Response Models: A Generalized Linear and Nonlinear Approach*, ed. P. De Boeck and M. Wilson, 3–41. New York: Springer. https://doi.org/10.1007/978-1-4757-3990-9_1.

Dempster, A. P., N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B* 39: 1–38. https://doi.org/10.1111/j.2517-6161.1977.tb01600.x.

Kondratek, B. Forthcoming. Item-fit statistic based on posterior probabilities of membership in ability groups. *Applied Psychological Measurement*.

Lord, F. M., and M. S. Wingersky. 1984. Comparison of IRT true-score and equipercentile observed-score "equatings". *Applied Psychological Measurement* 8: 453–461. https://doi.org/10.1177/014662168400800409.

Macdonald, K. 2008. pv: Stata module to perform estimation with plausible values. Statistical Software Components S456951, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s456951.html.

Orlando, M., and D. Thissen. 2000. Likelihood-based item-fit indices for dichotomous item response theory models. *Applied Psychological Measurement* 24: 50–64. https://doi.org/10.1177/01466216000241003.

Patz, R. J., and B. W. Junker. 1999. A straightforward approach to Markov chain Monte Carlo methods for item response models. *Journal of Educational and Behavioral Statistics* 24: 146–178. https://doi.org/10.2307/1165199.

Penfield, R. D., and G. Camilli. 2007. Differential item functioning and item bias. In *Psychometrics*, ed. C. R. Rao and S. Sinharay. Vol. 26 of *Handbook of Statistics*, 125–167. New York: Elsevier. https://doi.org/10.1016/S0169-7161(06)26005-X.

Raykov, T., and G. A. Marcoulides. 2018. *A Course in Item Response Theory and Modeling with Stata*. College Station, TX: Stata Press.

Rubin, D. B. 1987. *Multiple Imputation for Nonresponse in Surveys*. New York: Wiley. https://doi.org/10.1002/9780470316696.

Swaminathan, H., R. K. Hambleton, and H. J. Rogers. 2007. Assessing the fit of item response theory models. In *Psychometrics*, ed. C. R. Rao and S. Sinharay. Vol. 26 of *Handbook of Statistics*, 683–718. New York: Elsevier. https://doi.org/10.1016/S0169-7161(06)26021-8.

de la Torre, J. 2009. Improving the quality of ability estimates through multidimensional scoring and incorporation of ancillary variables. *Applied Psychological Measurement* 33: 465–485. https://doi.org/10.1177/0146621608329890.

Wainer, H. 1993. Model-based standardized measurement of an item's differential impact. In *Differential Item Functioning*, ed. P. W. Holland and H. Wainer, 123–136. Hillsdale, NJ: Lawrence Erlbaum.

Wu, M. 2005. The role of plausible values in large-scale surveys. *Studies in Educational Evaluation* 31: 114–128. https://doi.org/10.1016/j.stueduc.2005.05.005.

Zheng, X., and S. Rabe-Hesketh. 2007. Estimating parameters of dichotomous and ordinal item response models with gllamm. *Stata Journal* 7: 313–333. https://doi.org/10.1177/1536867X0700700302.

About the author

Bartosz Kondratek specializes in the field of psychometrics and statistical analysis of educational data. For many years, he has worked for the Central Examination Board and the Educational Research Institute, both in Warsaw.