

irt constraints — Specifying constraints

[Description](#)[Quick start](#)[Syntax](#)[Remarks and examples](#)[Also see](#)

Description

Constraints are imposed on the estimated parameters of a model. `irt` allows you to constrain a parameter to a fixed value or to constrain two or more parameters to be equal.

Quick start

2PL model for binary items `b1` to `b5`, with the discrimination parameters for items `b1` and `b2` constrained to be equal using symbolic constraints

```
irt ///
    (2pl b1, cns(a@k)) ///
    (2pl b2, cns(a@k)) ///
    (2pl b3-b5)
```

As above, but with the discrimination and difficulty parameters for items `b1` and `b2` constrained to fixed values

```
irt ///
    (2pl b1, cns(a@0.9 b@-1)) ///
    (2pl b2, cns(a@1.2 b@1.5)) ///
    (2pl b3-b5)
```

2PL model for binary items `b1` to `b5` and GRM for ordinal items `o1` to `o5` with discrimination parameters constrained to be equal for all items

```
irt ///
    (2pl b1-b5, cns(a@k)) ///
    (grm o1-o5, cns(a@k))
```

GRM for ordinal items `o1` to `o5` with the parameters of item `o1` constrained to fixed values

```
irt ///
    (grm o1, cns(a@1 b1@-1 b2@0 b3@1)) ///
    (grm o2-05)
```

1PL model for two groups with group-common items `b3-b7` and group-specific items `b1`, `b2`, `b8`, `b9` with the discrimination parameter constrained to be the same for all items

```
irt ///
    (0: 1pl b1 b2, cns(a@k)) ///
    ( 1pl b3-b7, cns(a@k)) ///
    (1: 1pl b8 b9, cns(a@k)) ///
    , group(female)
```

Syntax

```
irt ... [ , cns(spec [spec ...]) ... ]
```

where *spec* is *parm@#* or *parm@symbol*.

In 1PL and 2PL models, *parm* is one of **a** or **b**, which corresponds to the discrimination or difficulty parameter in the IRT parameterization, or *parm* is one of **alpha** or **beta**, which corresponds to the slope or intercept in the slope-intercept parameterization.

In 3PL models, *parm* is one of **a**, **b**, or **c**, which corresponds to the discrimination, difficulty, or guessing parameter in the IRT parameterization, or *parm* is one of **alpha** or **beta**, which corresponds to the slope or intercept in the slope-intercept parameterization.

In nominal response models, *parm* is one of **a1**, **a2**, ... for the multiple discrimination parameters per item, or *parm* is one of **b1**, **b2**, ... for the multiple difficulty parameters per item.

In graded response, partial credit, and rating scale models, *parm* is **a** for the discrimination parameter, or *parm* is one of **b1**, **b2**, ... for the multiple difficulty parameters per item.

a is a synonym for **a1**, and **b** is a synonym for **b1**.

Remarks and examples

[stata.com](https://www.stata.com)

Remarks are presented under the following headings:

[Overview](#)
[Constraints in 1PL, 2PL, and 3PL models](#)
[Constraints in graded response models](#)
[Constraints in nominal response models](#)
[Constraints in partial credit models](#)
[Constraints in rating scale models](#)

Overview

Two types of constraints can be specified in IRT models:

1. Fixed-value constraints set a parameter to a specific value. These constraints are specified by using **@** and the value of the constraint, for example, **a@1.5**.
2. Symbolic constraints set parameters to be equal to each other. Symbolic constraints are specified by using **@** and a name, for example, **a@k1**. Symbolic names are just names from 1 to 32 characters in length.

Constraints in 1PL, 2PL, and 3PL models

IRT models can be written using the IRT or the slope-intercept parameterization. Constraints in 1PL, 2PL, and 3PL models are applied to *a*, *b*, and *c* of the IRT parameterization. For instance, in [Methods and formulas](#) in [IRT] **irt 2pl**, we show that the 2PL model can be written in the IRT parameterization as

$$\Pr(Y_{ij} = 1 | a_i, b_i, \theta_j) = \frac{\exp\{a_i(\theta_j - b_i)\}}{1 + \exp\{a_i(\theta_j - b_i)\}}$$

where a_i represents discrimination of item i and b_i represents the difficulty of item i . For 2PL models, we can specify constraints on these a_i and b_i parameters. Similarly, constraints on 1PL models can be applied to a and b_i parameters. The 3PL model also allows constraints on the c parameter; see [Methods and formulas](#) in [IRT] **irt 3pl** for information on the IRT parameterization of this model.

The rules for specifying constraints in the IRT parameterization are the following:

1. Both fixed-value and symbolic constraints are allowed on the discrimination parameter, a .

For instance, to constrain the discrimination parameter to 0.8 in a 1PL model, we could type

```
. irt 1pl q1-q10, cns(a@0.8)
```

To constrain all discrimination parameters in a 2PL model to be equal, reducing it to a 1PL model, we type

```
. irt 2pl q1-q10, cns(a@k1)
```

2. Fixed-value constraints are allowed on the difficulty parameter b when a fixed-value constraint is also set on the corresponding a .

For instance, in a 2PL model, to constrain the discrimination parameter to 0.8 and the difficulty parameter to -2 for item $q1$, we could type

```
. irt                                     ///
  (2pl q1, cns(a@0.8 b@-2)) ///
  (2pl q2-q10)
```

Notice that we use the hybrid syntax to separate the item on which we are placing constraints from items that have no parameter constraints; see [IRT] **irt hybrid**.

3. Both fixed-value and symbolic constraints are allowed on the guessing parameter, c , in the 3PL model.

To constrain the guessing parameter for all items to 0.1, we could type

```
. irt 3pl q1-q10, cns(c@0.1)
```

To constrain the guessing parameter for $q1$ and $q2$ to be equal (and different from the common guessing parameter for $q3$ - $q10$), we could type

```
. irt                                     ///
  (3pl q1, cns(c@k1)) ///
  (3pl q2, cns(c@k1)) ///
  (3pl q3-q10)
```

We can also set constraints on the α and β parameters in the slope-intercept parameterization. For instance, in [Methods and formulas](#) in [IRT] **irt 2pl**, we show that the slope-intercept parameterization of the 2PL model is

$$\Pr(Y_{ij} = 1 | \alpha_i, \beta_i, \theta_j) = \frac{\exp(\alpha_i \theta_j + \beta_i)}{1 + \exp(\alpha_i \theta_j + \beta_i)}$$

where α_i is the slope for item i and β_i is the intercept for item i . 1PL and 3PL models have corresponding slope-intercept parameterizations.

Constraints on α and β are most often used in group IRT models to test for differential item functioning.

The rules for specifying constraints in the slope-intercept parameterization are the following:

1. Symbolic constraints are allowed on the slope parameter, α .

To constrain the slope of item q1 to be equal across groups and let the intercepts vary across groups, we could type

```
. irt                                     ///  
  (0: 2pl q1, cns(alpha@k1)) ///  
  (1: 2pl q1, cns(alpha@k1)) ///  
  (2pl q2-q10)
```

2. Symbolic constraints are allowed on the intercept parameter, β .

To constrain the intercept of item q1 to be equal across groups and let the slopes vary across groups, we could type

```
. irt                                     ///  
  (0: 2pl q1, cns(beta@k1)) ///  
  (1: 2pl q1, cns(beta@k1)) ///  
  (2pl q2-q10)
```

▷ Example 1: Fixed-value constraints in a 2PL model

Using `masc1.dta`, we could fit a 2PL model by typing

```
. use https://www.stata-press.com/data/r17/masc1  
(Data from De Boeck & Wilson (2004))  
. irt (2pl q1) (2pl q2 q3 q4)
```

Notice that we used a hybrid specification of our 2PL model with `q1` separated from the other items. This hybrid syntax is commonly used when fitting models with constraints because it allows us to apply constraints to a chosen item or set of items; see [\[IRT\] irt hybrid](#) for information on this syntax.

We now constrain the discrimination to 1.5 and the difficulty to -0.5 for q_1 by adding the `cns(a@1.5 b@-0.5)` option.

```
. irt (2pl q1, cns(a@1.5 b@-.5)) (2pl q2 q3 q4)
Fitting fixed-effects model:
Iteration 0:   log likelihood = -2042.1777
Iteration 1:   log likelihood = -2041.492
Iteration 2:   log likelihood = -2041.4917
Iteration 3:   log likelihood = -2041.4917
Fitting full model:
Iteration 0:   log likelihood = -2000.6715
Iteration 1:   log likelihood = -1995.1431
Iteration 2:   log likelihood = -1995.0644
Iteration 3:   log likelihood = -1995.0643
Hybrid IRT model                               Number of obs = 800
Log likelihood = -1995.0643
```

		Coefficient	Std. err.	z	P> z	[95% conf. interval]	
2pl							
q1	Discrim	1.5	(constrained)				
	Diff	-.5	(constrained)				
2pl							
q2	Discrim	.70149	.1469275	4.77	0.000	.4135175	.9894626
	Diff	-.1455087	.1131457	-1.29	0.198	-.3672702	.0762528
q3	Discrim	.9136072	.1960573	4.66	0.000	.5293419	1.297873
	Diff	-1.726684	.2977765	-5.80	0.000	-2.310316	-1.143053
q4	Discrim	.742854	.1556356	4.77	0.000	.4378138	1.047894
	Diff	.3541431	.1220373	2.90	0.004	.1149543	.5933319

We see that the fixed-value constraints on q_1 appear in the first table in the output.



► Example 2: Constraining slopes in a group 2PL model

IRT models are reported using the IRT parameterization where $b_i = -\beta_i/\alpha_i$ and $a_i = \alpha_i$. Typically, constraints are set on the a 's and b 's as in the [previous example](#). Note, however, that to set a constraint on one of the b 's, we need to constrain both the underlying α and β parameters. Therefore, we are required to set a constraint on a any time we set a constraint on b .

Sometimes, we instead set constraints directly on the β parameters of the slope-intercept metric. Constraints on β do not require constraints on corresponding α parameters. Constraints specified in this slope-intercept metric are most often used in the context of multiple-group models.

For example, in a group 2PL model, we may want to constrain the intercepts to be the same across groups while allowing the slopes to differ across groups. Using `masc2.dta`, we can fit a group 2PL model by typing

```
. use https://www.stata-press.com/data/r17/masc2
(Data from De Boeck & Wilson (2004))
. irt (0: 2pl q1) (1: 2pl q1) (2pl q2 q3 q4), group(female)
```

This model allows the slope and intercept for `q1` to differ across groups and constrains the slopes and intercepts for all other items across groups; see [\[IRT\] irt, group\(\)](#) for information on group IRT models.

Now we can constrain the intercept for item `q1` to be equal across the two groups and allow only the slope of `q1` to vary across groups by typing

```
. irt (0: 2pl q1, cns(beta@k1)) (1: 2pl q1, cns(beta@k1)) (2pl q2 q3 q4),
> group(female)
Fitting fixed-effects model:
Iteration 0: log likelihood = -3848.9104
Iteration 1: log likelihood = -3845.4263
Iteration 2: log likelihood = -3845.4252
Iteration 3: log likelihood = -3845.4252
Group: Male
Group: Female
Fitting full model:
Iteration 0: log likelihood = -3786.3332
Iteration 1: log likelihood = -3779.6101
Iteration 2: log likelihood = -3778.4145
Iteration 3: log likelihood = -3778.3648
Iteration 4: log likelihood = -3778.3646
Hybrid IRT model                               Number of obs = 1,500
Log likelihood = -3778.3646
Group: Male
```

		Coefficient	Std. err.	z	P> z	[95% conf. interval]	
2pl							
q1	Discrim	1.756291	.6093519	2.88	0.004	.5619834	2.950599
	Diff	-.3976132	.0983202	-4.04	0.000	-.5903172	-.2049092
2pl							
q2	Discrim	.7065962	.160211	4.41	0.000	.3925883	1.020604
	Diff	-.0184657	.1007856	-0.18	0.855	-.2160018	.1790704
q3	Discrim	.7360524	.1741027	4.23	0.000	.3948173	1.077287
	Diff	-2.006872	.4360328	-4.60	0.000	-2.861481	-1.152264
q4	Discrim	.602399	.1225595	4.92	0.000	.3621868	.8426112
	Diff	.5431222	.1520137	3.57	0.000	.2451809	.8410635
mean(Theta)		0 (omitted)					
var(Theta)		1 (constrained)					

Group: Female

		Coefficient	Std. err.	z	P> z	[95% conf. interval]	
2pl							
q1							
	Discrim	1.000017	.2965882	3.37	0.001	.4187151	1.581319
	Diff	-.6983125	.2424141	-2.88	0.004	-1.173435	-.2231896
2pl							
q2							
	Discrim	.7065962	.160211	4.41	0.000	.3925883	1.020604
	Diff	-.0184657	.1007856	-0.18	0.855	-.2160018	.1790704
q3							
	Discrim	.7360524	.1741027	4.23	0.000	.3948173	1.077287
	Diff	-2.006872	.4360328	-4.60	0.000	-2.861481	-1.152264
q4							
	Discrim	.602399	.1225595	4.92	0.000	.3621868	.8426112
	Diff	.5431222	.1520137	3.57	0.000	.2451809	.8410635
	mean(Theta)	-.0903098	.1215655	-0.74	0.458	-.3285738	.1479542
	var(Theta)	1.295406	.4927367			.6146562	2.730106

We do not see the estimates of β in this output, but we can use the `estmetric` option to display the results in the slope-intercept parameterization.

8 irt constraints — Specifying constraints

. irt, estmetric

Hybrid IRT model

Number of obs = 1,500

Log likelihood = -3778.3646

```
( 1) [q1]0bn.female - [q1]1.female = 0
( 2) [q2]0bn.female - [q2]1.female = 0
( 3) [q2]0bn.female#c.Theta - [q2]1.female#c.Theta = 0
( 4) [q3]0bn.female - [q3]1.female = 0
( 5) [q3]0bn.female#c.Theta - [q3]1.female#c.Theta = 0
( 6) [q4]0bn.female - [q4]1.female = 0
( 7) [q4]0bn.female#c.Theta - [q4]1.female#c.Theta = 0
( 8) [/]mean(Theta)#0bn.female = 0
( 9) [/]var(Theta)#0bn.female = 1
```

Group: Male

Number of obs = 761

	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
q1						
Theta	1.756291	.6093519	2.88	0.004	.5619834	2.950599
_cons	.6983246	.1370853	5.09	0.000	.4296424	.9670067
q2						
Theta	.7065962	.160211	4.41	0.000	.3925883	1.020604
_cons	.0130478	.0709493	0.18	0.854	-.1260102	.1521058
q3						
Theta	.7360524	.1741027	4.23	0.000	.3948173	1.077287
_cons	1.477163	.0933292	15.83	0.000	1.294241	1.660085
q4						
Theta	.602399	.1225595	4.92	0.000	.3621868	.8426112
_cons	-.3271763	.0668258	-4.90	0.000	-.4581524	-.1962002
mean(Theta)	0 (omitted)					
var(Theta)	1 (constrained)					

Group: Female

Number of obs = 739

	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
q1						
Theta	1.000017	.2965882	3.37	0.001	.4187151	1.581319
_cons	.6983246	.1370853	5.09	0.000	.4296424	.9670067
q2						
Theta	.7065962	.160211	4.41	0.000	.3925883	1.020604
_cons	.0130478	.0709493	0.18	0.854	-.1260102	.1521058
q3						
Theta	.7360524	.1741027	4.23	0.000	.3948173	1.077287
_cons	1.477163	.0933292	15.83	0.000	1.294241	1.660085
q4						
Theta	.602399	.1225595	4.92	0.000	.3621868	.8426112
_cons	-.3271763	.0668258	-4.90	0.000	-.4581524	-.1962002
mean(Theta)	-.0903098	.1215655	-0.74	0.458	-.3285738	.1479542
var(Theta)	1.295406	.4927367			.6146562	2.730106

Now we see that the equality constraints on the β parameters for q1 were imposed across groups. The β 's are the intercepts labeled `_cons` with a value of 0.698.



Constraints in graded response models

Constraints for the GRM differ from the 1PL, 2PL, and 3PL models because each item in a GRM has multiple b parameters. In the `cns()` option, we refer to these parameters as `b1`, `b2`, `b3`, ...

The rules for constraints in the IRT parameterization of the GRM are the following:

1. Both fixed-value and symbolic constraints are allowed on the discrimination parameter, a .

For instance, to constrain the discrimination parameters for all items to 0.8, we could type

```
. irt grm q1-q5, cns(a@0.8)
```

To constrain all discrimination parameters to be equal, we type

```
. irt grm q1-q5, cns(a@k1)
```

2. Fixed-value constraints are allowed on the difficulty parameters, b_1 , b_2 , ..., when a fixed-value constraint is also set on the corresponding a .

For instance, to constrain the discrimination parameter to 0.8 and the difficulty parameters to -2 , -1 , and 0 for item `q1`, we could type

```
. irt                                     ///
  (grm q1, cns(a@0.8 b1@-2 b2@-1 b3@0)) ///
  (grm q2-q5)
```

Constraints in nominal response models

Constraints for the NRM differ from the GRM models because each item in an NRM has multiple a parameters in addition to multiple b parameters. In the `cns()` option, we refer to these a parameters as `a1`, `a2`, `a3`, ...

The rules for constraints in the IRT parameterization of the NRM are the following:

1. Both fixed-value and symbolic constraints are allowed on the discrimination parameters, a_1 , a_2 , ...

For instance, to constrain the discrimination parameters for item `q1` to 1 and 1.1, we could type

```
. irt                                     ///
  (nrm q1, cns(a1@1 a2@1.1)) ///
  (nrm q2-q5)
```

To constrain discrimination parameters for `q1` and `q2` to be equal, we type

```
. irt                                     ///
  (nrm q1 q2, cns(a1@k1 a2@k2)) ///
  (nrm q3-q5)
```

- Fixed-value constraints are allowed on the difficulty parameters, b_1, b_2, \dots , when a fixed-value constraint is also set on the corresponding discrimination parameters a_1, a_2, \dots .

For instance, to constrain the discrimination parameters to 1 and 1.1 and constrain the difficulty parameters to -1 and 1 for item q1, we could type

```
. irt                                     ///
  (nrm q1, cns(a1@1 a2@1.1 b1@-1 b2@1)) ///
  (nrm q2-q5)
```

Constraints in partial credit models

The PCM and GPCM have multiple difficulty parameters for each item. In the `cns()` option, we refer to these parameters as b_1, b_2, b_3, \dots .

The rules for constraints in the IRT parameterization of the PCM and GPCM are the following:

- Both fixed-value and symbolic constraints are allowed on the discrimination parameter, a .

For instance, to constrain the discrimination parameter in a PCM to 0.8, we could type

```
. irt pcm q1-q5, cns(a@0.8)
```

To constrain discrimination parameters on q1 and q2 to be equal in a GPCM, we type

```
. irt                                     ///
  (gpcm q1 q2, cns(a@k1)) ///
  (gpcm q3-q5)
```

- Fixed-value constraints are allowed on the difficulty parameters, b_1, b_2, \dots , when a fixed-value constraint is also set on the corresponding a .

For instance, to constrain the discrimination parameter to 0.8 and the difficulty parameters to $-2, -1$, and 0 for item q1 in a GPCM, we could type

```
. irt                                     ///
  (gpcm q1, cns(a@0.8 b1@-2 b2@-1 b3@0)) ///
  (gpcm q2-q5)
```

Constraints in rating scale models

The RSM has multiple difficulty parameters for each item. In the `cns()` option, we refer to these parameters as b_1, b_2, b_3, \dots .

The rules for constraints in the IRT parameterization of the PCM are the following:

- Both fixed-value and symbolic constraints are allowed on the discrimination parameter, a .

For instance, to constrain the discrimination parameter to 0.8, we could type

```
. irt rsm q1-q5, cns(a@0.8)
```

2. Fixed-value constraints are allowed on the difficulty parameters, b_1, b_2, \dots , when a fixed-value constraint is also set on the corresponding a and on all b 's for the item.

For instance, to constrain the discrimination parameter to 0.8 and the difficulty parameters to $-2, -1$, and 0 for item `q1`, we could type

```
. irt                                     ///
   (rsm q1, cns(a@0.8 b1@-2 b2@-1 b3@0)) ///
   (rsm q2-q5)
```

Also see

- [IRT] [irt](#) — Introduction to IRT models
- [IRT] [irt 1pl](#) — One-parameter logistic model
- [IRT] [irt 2pl](#) — Two-parameter logistic model
- [IRT] [irt 3pl](#) — Three-parameter logistic model
- [IRT] [irt grm](#) — Graded response model
- [IRT] [irt, group\(\)](#) — IRT models for multiple groups
- [IRT] [irt hybrid](#) — Hybrid IRT models
- [IRT] [irt nrm](#) — Nominal response model
- [IRT] [irt pcm](#) — Partial credit model
- [IRT] [irt rsm](#) — Rating scale model