**Description**

`estat greport` displays the estimated group IRT parameters.

**Quick start**

Group 2PL model for binary items b1 to b10

```
irt 2pl b1-b10, group(female)
```

Report results in a compact format

```
estat greport
```

Report standard errors in addition to coefficients

```
estat greport, se
```

Report results grouped by parameter type

```
estat greport, byparm
```

As above, and sort items by estimated difficulty

```
estat greport, byparm sort(b)
```

**Menu**

Statistics > IRT (item response theory)
estat greport — Report estimated group IRT parameters

Syntax

```
estat greport [, options]
```

```
sort(p[, descending])
```
sort items by the estimated p parameters; p may be a, b, or c

```
byparm
```
arange table rows by parameter rather than by item

Main

```
alabel(string)
```
specify the a parameter label; the default is Discrim

```
blabel(string)
```
specify the b parameter label; the default is Diff

```
clabel(string)
```
specify the c parameter label; the default is Guess

```
seqlabel
```
label parameters in sequential order

```
post
```
post estimated IRT parameters and their VCE as estimation results

Reporting

```
b(\%fmt)
```
how to format coefficients, which are always reported

```
se(\%fmt)
```
report standard errors and use optional format

```
t(\%fmt)
```
report t or z statistics and use optional format

```
p(\%fmt)
```
report p-values and use optional format

```
parmwidth(#)
```
use # characters to display variable and parameter names

```
grwidth(#)
```
use # characters to display group names and statistics

```
style(oneline)
```
display vertical line after variable names; the default

```
style(columns)
```
display vertical lines separating columns

```
style(noline)
```
suppress all vertical lines

```
grlabel(string)
```
column labels for groups

```
title(string)
```
title for table

Options

```
sort(p[, descending])
```
requests that items be sorted according to parameter p, where p is one of a, b, or c.

```
sort(a)
```
specifies that items be sorted according to the estimated discrimination parameters.

```
sort(b)
```
specifies that items be sorted according to the estimated difficulty parameters.

```
sort(c)
```
specifies that items be sorted according to the estimated pseudoguessing parameters. It is relevant only for a 3PL model when option sepguessing is specified.

```
descending
```
requests that the sorted items be reported in descending order. Sorted items are reported in ascending order by default.

```
byparm
```
requests that the table rows be grouped by parameter rather than by item.

Main

```
alabel(string)
```
labels the discrimination parameters with string. The default label is Discrim.

```
blabel(string)
```
labels the difficulty parameters with string. The default label is Diff.

```
clabel(string)
```
labels the pseudoguessing parameters with string. The default label is Guess. This option applies only to 3PL models.
seqlabel labels the estimated difficulty parameters within each categorical item sequentially, starting from 1. In NRM, seqlabel also labels the estimated discrimination parameters within each item sequentially, starting from 1. This option applies only to categorical models.

post causes estat greport to behave like a Stata estimation (e-class) command. estat greport posts the vector of estimated IRT parameters along with the corresponding variance–covariance matrix to e(), so that you can treat the estimated IRT parameters just as you would results from any other estimation command. For example, you could use test to perform simultaneous tests of hypotheses on the parameters, or you could use lincom to create linear combinations.

Reporting

b(%fmt) specifies how the coefficients are to be displayed. You might specify b(%9.2f) to make decimal points line up. There is also a b option, which specifies that coefficients be displayed, but that is just included for consistency with the se, t, and p options. Coefficients are always displayed.

se, t, and p specify that standard errors, t or z statistics, and p-values be displayed. The default is not to display them. se(%fmt), t(%fmt), and p(%fmt) specify that each be displayed and specify the display format to be used.

parmwidth(#) specifies the number of character positions used to display the names of the variables and parameters. The default is parmwidth(12).

grwidth(#) specifies the number of character positions used to display the names of the groups and statistics. The default is grwidth(12).

style(stylespec) specifies the style of the coefficient table.

style(oneline) specifies that a vertical line be displayed after the variables but not between the groups. This is the default.

style(columns) specifies that vertical lines be displayed after each column.

style(noline) specifies that no vertical lines be displayed.

grouplabel(string) specifies the labels for the group columns. The default is to use value labels of the group variable or, if the group variable has no value labels, a factor-variable indicator for each level of the group variable.

title(string) specifies the title to appear above the table.

Remarks and examples

The following discussion is about how to use estat greport with irt estimation results. If you are new to the IRT features in Stata, we encourage you to read [IRT] irt first.
Example 1: Sorting binary items

We illustrate the features of `estat greport` on a group 2PL model we fit in example 1 of [IRT] `irt, group()`. First, we refit the model.

```
. use https://www.stata-press.com/data/r16/masc2
(Data from De Boeck & Wilson (2004))
. irt 2pl q1-q5, group(female)
Fitting fixed-effects model:
Iteration 0: log likelihood = -4594.5412
Iteration 1: log likelihood = -4590.4516
Iteration 2: log likelihood = -4590.4502
Iteration 3: log likelihood = -4590.4502
Group: male
Group: female
Fitting full model:
Iteration 0: log likelihood = -4503.5396 (not concave)
Iteration 1: log likelihood = -4479.7967
Iteration 2: log likelihood = -4476.3965
Iteration 3: log likelihood = -4476.3448
Iteration 4: log likelihood = -4476.3447
Two-parameter logistic model Number of obs = 1,500
Log likelihood = -4476.3447
Group: male
```

| Item | Coef. | Std. Err. | z     | P>|z| | [95% Conf. Interval] |
|------|-------|-----------|-------|-----|----------------------|
| q1   |       |           |       |     |                      |
| Discrim | 1.187923 | .1804778 | 6.58  | 0.000 | .8341933 1.541653    |
| Diff   | -.5507796 | .0894536 | -6.16 | 0.000 | -.7261054 -.3754538 |
| q2   |       |           |       |     |                      |
| Discrim | .90663 | .1318739 | 6.87  | 0.000 | .6481618 1.165098    |
| Diff   | -.0450698 | .0761722 | -0.59 | 0.554 | -.1943645 .104225   |
| q3   |       |           |       |     |                      |
| Discrim | .8828704 | .1462984 | 6.03  | 0.000 | .5961307 1.16961    |
| Diff   | -1.703158 | .2385734 | -7.14 | 0.000 | -2.170753 -1.235563 |
| q4   |       |           |       |     |                      |
| Discrim | .8196789 | .1221824 | 6.71  | 0.000 | .5802057 1.059152    |
| Diff   | .3770973 | .0993197 | 3.80  | 0.000 | .1824342 .5717603   |
| q5   |       |           |       |     |                      |
| Discrim | 1.439933 | .2218141 | 6.49  | 0.000 | 1.005185 1.874681    |
| Diff   | 1.197739 | .1437481 | 8.33  | 0.000 | .9159978 1.47948    |
| mean(Theta) | 0 | (omitted) |       |     |                      |
| var(Theta) | 1 | (constrained) |       |     |                      |
estat greport — Report estimated group IRT parameters

| Parameter | Coef. Std. Err. z P>|z|  [95% Conf. Interval] |
|-----------|-----------------------------|
| q1        |                             |
| Discrim   | 1.187923 (.1804778 6.58 0.000 .8341933 1.541653) |
| Diff      | -.5507796 (.0894536 -6.16 0.000 -.7261054 -.3754538) |
| q2        |                             |
| Discrim   | .90663 (.1318739 6.87 0.000 .6481618 1.165098) |
| Diff      | -.0450698 (.0761722 -0.59 0.554 -.1943645 .104225) |
| q3        |                             |
| Discrim   | .8828704 (.1462984 6.03 0.000 .5961307 1.16961) |
| Diff      | -1.703158 (.2385734 -7.14 0.000 -2.170753 -1.235563) |
| q4        |                             |
| Discrim   | .8196789 (.1221824 6.71 0.000 .5802057 1.059152) |
| Diff      | .3770973 (.0993197 3.80 0.000 .1824342 .5717603) |
| q5        |                             |
| Discrim   | 1.439933 (.2218141 6.49 0.000 1.005185 1.874681) |
| Diff      | 1.197739 (.1437481 8.33 0.000 .9159978 1.47948) |
| mean(Theta)| -.1348222 (.0721434 -1.87 0.062 -.2762206 .0065763) |
| var(Theta)| .6239155 (.1239068 4.227474 .9208111) |

Group IRT models always produce a lot of output. The estimated parameters are reported separately for each group, and it is hard to visually compare estimates across groups. estat greport makes such comparisons a breeze.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>male</th>
<th>female</th>
</tr>
</thead>
<tbody>
<tr>
<td>q1</td>
<td>1.1879233</td>
<td>1.1879233</td>
</tr>
<tr>
<td>Diff</td>
<td>-.55077963</td>
<td>-.55077963</td>
</tr>
<tr>
<td>q2</td>
<td>.90662997</td>
<td>.90662997</td>
</tr>
<tr>
<td>Diff</td>
<td>-.04506976</td>
<td>-.04506976</td>
</tr>
<tr>
<td>q3</td>
<td>.88287037</td>
<td>.88287037</td>
</tr>
<tr>
<td>Diff</td>
<td>-1.7031579</td>
<td>-1.7031579</td>
</tr>
<tr>
<td>q4</td>
<td>.81967885</td>
<td>.81967885</td>
</tr>
<tr>
<td>Diff</td>
<td>.37709727</td>
<td>.37709727</td>
</tr>
<tr>
<td>q5</td>
<td>1.4399331</td>
<td>1.4399331</td>
</tr>
<tr>
<td>Diff</td>
<td>1.1977389</td>
<td>1.1977389</td>
</tr>
<tr>
<td>mean(Theta)</td>
<td>0</td>
<td>-.13482217</td>
</tr>
<tr>
<td>var(Theta)</td>
<td>1</td>
<td>.6239155</td>
</tr>
</tbody>
</table>
For models with many items, it is often convenient to arrange the output according to highest or lowest difficulty ($b$) or discrimination ($a$). `estat greport` makes sorting in a desired order easy. Below, we specify option `sort(b)` to cause `estat greport` to display the items in ascending order of the estimated difficulty parameter. The sort is performed on the first group.

```
. estat greport, sort(b)
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>male</th>
<th>female</th>
</tr>
</thead>
<tbody>
<tr>
<td>q3</td>
<td>0.88287037</td>
<td>0.88287037</td>
</tr>
<tr>
<td></td>
<td>-1.7031579</td>
<td>-1.7031579</td>
</tr>
<tr>
<td>q1</td>
<td>1.1879233</td>
<td>1.1879233</td>
</tr>
<tr>
<td></td>
<td>-0.55077963</td>
<td>-0.55077963</td>
</tr>
<tr>
<td>q2</td>
<td>0.90662997</td>
<td>0.90662997</td>
</tr>
<tr>
<td></td>
<td>-0.04506976</td>
<td>-0.04506976</td>
</tr>
<tr>
<td>q4</td>
<td>0.81967885</td>
<td>0.81967885</td>
</tr>
<tr>
<td></td>
<td>0.37709727</td>
<td>0.37709727</td>
</tr>
<tr>
<td>q5</td>
<td>1.4399331</td>
<td>1.4399331</td>
</tr>
<tr>
<td></td>
<td>1.1977389</td>
<td>1.1977389</td>
</tr>
<tr>
<td>mean(Theta)</td>
<td>0</td>
<td>-0.13482217</td>
</tr>
<tr>
<td>var(Theta)</td>
<td>1</td>
<td>0.6239155</td>
</tr>
</tbody>
</table>

Finally, we add the `byparm` option to cause `estat greport` to arrange the table rows by parameter type and then by item difficulty.

```
. estat greport, sort(b) byparm
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>male</th>
<th>female</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrim</td>
<td></td>
<td></td>
</tr>
<tr>
<td>q3</td>
<td>0.88287037</td>
<td>0.88287037</td>
</tr>
<tr>
<td>q1</td>
<td>1.1879233</td>
<td>1.1879233</td>
</tr>
<tr>
<td>q2</td>
<td>0.90662997</td>
<td>0.90662997</td>
</tr>
<tr>
<td>q4</td>
<td>0.81967885</td>
<td>0.81967885</td>
</tr>
<tr>
<td>q5</td>
<td>1.4399331</td>
<td>1.4399331</td>
</tr>
<tr>
<td>Diff</td>
<td></td>
<td></td>
</tr>
<tr>
<td>q3</td>
<td>-1.7031579</td>
<td>-1.7031579</td>
</tr>
<tr>
<td>q1</td>
<td>-0.55077963</td>
<td>-0.55077963</td>
</tr>
<tr>
<td>q2</td>
<td>-0.04506976</td>
<td>-0.04506976</td>
</tr>
<tr>
<td>q4</td>
<td>0.37709727</td>
<td>0.37709727</td>
</tr>
<tr>
<td>q5</td>
<td>1.1977389</td>
<td>1.1977389</td>
</tr>
<tr>
<td>mean(Theta)</td>
<td>0</td>
<td>-0.13482217</td>
</tr>
<tr>
<td>var(Theta)</td>
<td>1</td>
<td>0.6239155</td>
</tr>
</tbody>
</table>
Stored results

`estat greport` stores the following in `r()`:

Macros
- `r(names)` labels used for group names

Matrices
- `r(b)` vector of estimated IRT parameters
- `r(b_pclass)` parameter class
- `r(V)` variance–covariance matrix of the estimated IRT parameters

`estat greport` with the `post` option also stores the following in `e()`:

Macros
- `e(properties) b V`

Matrices
- `e(b)` vector of estimated IRT parameters
- `e(V)` variance–covariance matrix of the estimated IRT parameters

Also see

- `[IRT] irt` — Introduction to IRT models
- `[IRT] irt, group()` — IRT models for multiple groups
- `[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 hybrid` — Hybrid IRT models
- `[IRT] irt nrm` — Nominal response model
- `[IRT] irt pcm` — Partial credit model
- `[IRT] irt rsm` — Rating scale model