irt nrm — Nominal response model

Description

`irt nrm` fits nominal response models (NRMs) to categorical items. In the NRM, items vary in their difficulty and discrimination.

Quick start

NRM for nominal items `n1` to `n5`

```
irt nrm n1-n5
```

Plot CCCs for `n1`

```
irtgraph icc n1
```

Menu

Statistics > IRT (item response theory)

Syntax

`irt nrm` *varlist* [*if*] [*in*] [*weight*] [, *options*]

options	Description
group(*varname*)	fit model for different groups
Model	
cns(*spec*)	apply specified parameter constraints
listwise	drop observations with any missing items
SE/Robust	
vce(*vcetype*)	*vcetype* may be oim, robust, cluster *clustvar*, bootstrap, or jackknife
Reporting	
level(#)	set confidence level; default is level(95)
notable	suppress coefficient table
noheader	suppress output header
display_options	control columns and column formats
Integration	
intmethod(*intmethod*)	integration method
intpoints(#)	set the number of integration points; default is intpoints(7)
Maximization	
maximize_options	control the maximization process; seldom used
startvalues(*svmethod*)	method for obtaining starting values
noestimate	do not fit the model; show starting values instead
estmetric	show parameter estimates in the estimation metric
dnumerical	use numerical derivative techniques
coeflegend	display legend instead of statistics

intmethod	Description
mvaghermite	mean–variance adaptive Gauss–Hermite quadrature; the default
mcaghermite	mode-curvature adaptive Gauss–Hermite quadrature
ghermite	nonadaptive Gauss–Hermite quadrature

bootstrap, by, collect, jackknife, statsby, and svy are allowed; see [U] **11.1.10 Prefix commands**.

Weights are not allowed with the bootstrap prefix; see [R] **bootstrap**.

vce() and weights are not allowed with the svy prefix; see [SVY] **svy**.

fweights, iweights, and pweights are allowed; see [U] **11.1.6 weight**.

startvalues(), noestimate, estmetric, dnumerical, and coeflegend do not appear in the dialog box.

See [U] **20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

group(*varname*) specifies that the model be fit separately for the different values of *varname*; see [IRT] **irt, group()** for details.

Model

cns(*spec*) constrains item parameters to a fixed value or constrains two or more parameters to be equal; see [IRT] **irt constraints** for details.

listwise handles missing values through listwise deletion, which means that the entire observation is omitted from the estimation sample if any of the items are missing for that observation. By default, all nonmissing items in an observation are included in the likelihood calculation; only missing items are excluded.

SE/Robust

vce(*vcetype*) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster *clustvar*), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] ***vce_option***.

Reporting

level(*#*); see [R] **Estimation options**.

notable suppresses the estimation table, either at estimation or upon replay.

noheader suppresses the output header, either at estimation or upon replay.

display_options: noci, nopvalues, cformat(*%fmt*), pformat(*%fmt*), sformat(*%fmt*), and nolstretch; see [R] **Estimation options**.

Integration

intmethod(*intmethod*) specifies the integration method to be used for computing the log likelihood. mvaghermite performs mean and variance adaptive Gauss–Hermite quadrature; mcaghermite performs mode and curvature adaptive Gauss–Hermite quadrature; and ghermite performs nonadaptive Gauss–Hermite quadrature.

The default integration method is mvaghermite.

intpoints(*#*) sets the number of integration points for quadrature. The default is intpoints(7), which means that seven quadrature points are used to compute the log likelihood.

The more integration points, the more accurate the approximation to the log likelihood. However, computation time increases with the number of integration points.

Maximization

maximize_options: difficult, technique(*algorithm_spec*), iterate(*#*), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(*#*), ltolerance(*#*), nrtolerance(*#*), nonrtolerance, and from(*init_specs*); see [R] **Maximize**. Those that require special mention for irt are listed below.

from() accepts a properly labeled vector of initial values or a list of coefficient names with values. A list of values is not allowed.

The following options are available with irt but are not shown in the dialog box:

startvalues() specifies how starting values are to be computed. Starting values specified in from() override the computed starting values.

startvalues(zero) specifies that all starting values be set to 0. This option is typically useful only when specified with the from() option.

startvalues(constantonly) builds on startvalues(zero) by fitting a constant-only model for each response to obtain estimates of intercept and cutpoint parameters.

startvalues(fixedonly) builds on startvalues(constantonly) by fitting a full fixed-effects model for each response variable to obtain estimates of coefficients along with intercept and cutpoint parameters. You can also add suboption iterate(#) to limit the number of iterations irt allows for fitting the fixed-effects model.

startvalues(ivloadings) builds on startvalues(fixedonly) by using instrumental-variable methods with the generalized residuals from the fixed-effects models to compute starting values for latent-variable loadings. This is the default behavior.

noestimate specifies that the model is not to be fit. Instead, starting values are to be shown (as modified by the above options if modifications were made), and they are to be shown using the coeflegend style of output. An important use of this option is before you have modified starting values at all; you can type the following:

```
. irt ..., ... noestimate
. matrix b = e(b)
. ... (modify elements of b) ...
. irt ..., ... from(b)
```

estmetric displays parameter estimates in the slope-intercept metric that is used for estimation.

dnumerical specifies that during optimization, the gradient vector and Hessian matrix be computed using numerical techniques instead of analytical formulas. By default, irt uses analytical formulas for computing the gradient and Hessian for all integration methods.

coeflegend; see [R] **Estimation options**.

Remarks and examples stata.com

Remarks are presented under the following headings:

Overview
Video example

Overview

The following discussion is about how to use irt to fit NRMs to categorical items. If you are new to the IRT features in Stata, we encourage you to read [IRT] **irt** first.

The NRM is used for nominally scored responses. The responses are allocated to mutually exclusive, exhaustive, and nonordered categories. For example, responses to a behavioral question may be recorded as “yes”, “no”, “maybe”, and “refused to say”, and the actual score has no meaning other than to designate the response category. If there are only two outcomes, the NRM is equivalent to the 2PL model; see [IRT] **irt 2pl**. If the item responses are ordered, see [IRT] **irt grm**, [IRT] **irt pcm**, or [IRT] **irt rsm**.

The NRM allows the categories to vary between items; however, to keep the following discussion from being overly complicated, we will assume the outcome levels for all items are given by $k = 1, \dots, K$.

In the NRM, the items are modeled each with their own collection of discrimination and "difficulty" parameters using the multinomial logistic model. We quote "difficulty" here to acknowledge that in the context of the NRM as implemented in `irt nrm`, the difficulty parameter measures the propensity to choose a given item category instead of the base outcome. For item i, the probability of person j choosing category k on item i is

$$\Pr(Y_{ij} = k|\theta_j) = \frac{\exp\{a_{ik}(\theta_j - b_{ik})\}}{\sum_{h=1}^{K} \exp\{a_{ih}(\theta_j - b_{ih})\}} \qquad \theta_j \sim N(0,1)$$

where a_{ik} represents the discrimination of category k for item i, b_{ik} represents the difficulty of category k for item i, and θ_j is the latent trait of person j. `irt nrm` assigns the first outcome as the base outcome with which other parameters will be compared; this implies the constraint $a_{i1} = 0$ and $b_{i1} = 0$ for each item i. With this constraint, a_{ik} and b_{ik} are the discrimination and difficulty to choose category k relative to the first category.

The NRM was proposed by Bock (1972). The slope-intercept parameterization of the NRM was proposed by Baker and Kim (2004). When no latent variable is present, the model for a single item is known as the multinomial logistic model; see [R] **mlogit**.

▷ Example 1: Fitting an NRM

To illustrate the NRM, we use the data from de Ayala (2022). `science.dta` contains four multiple-choice questions from a physical science test, `q1` through `q4`, with each response assigned to an unordered category of 1, 2, 3, or 4.

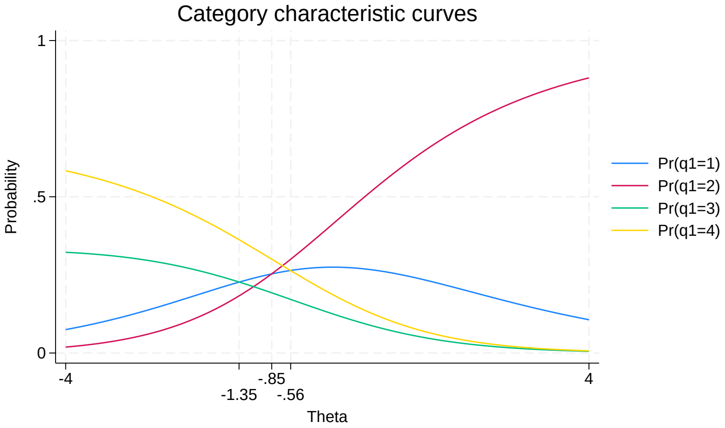
We fit an NRM as follows:

```
. use https://www.stata-press.com/data/r18/science
(Physical science data from de Ayala (2009))

. irt nrm q1-q4

Fitting fixed-effects model:

Iteration 0:  Log likelihood = -9256.1514
Iteration 1:  Log likelihood = -9256.1514

Fitting full model:

Iteration 0:  Log likelihood = -9287.6878  (not concave)
Iteration 1:  Log likelihood = -9221.5198
Iteration 2:  Log likelihood = -9207.5015  (not concave)
Iteration 3:  Log likelihood = -9165.7851
Iteration 4:  Log likelihood =  -9154.166
Iteration 5:  Log likelihood = -9152.2717
Iteration 6:  Log likelihood =  -9152.244
Iteration 7:  Log likelihood =  -9152.244

Nominal response model                                  Number of obs = 1,799
Log likelihood = -9152.244

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
q1           |
     Discrim |
      2 vs 1 |   .4349366   .1221761     3.56   0.000     .1954759    .6743973
      3 vs 1 |  -.5492982   .1630795    -3.37   0.001    -.8689282   -.2296683
      4 vs 1 |  -.5967016   .1459728    -4.09   0.000     -.882803   -.3106002
        Diff |
      2 vs 1 |  -.8531528   .3228014    -2.64   0.008    -1.485832   -.2204737
      3 vs 1 |  -1.346445   .3511587    -3.83   0.000    -2.034703   -.6581867
      4 vs 1 |  -.5634533   .1473554    -3.82   0.000    -.8522645   -.2746422
-------------+----------------------------------------------------------------
q2           |
     Discrim |
      2 vs 1 |  -.2226277   .2065114    -1.08   0.281    -.6273825    .1821272
      3 vs 1 |   .6076979   .2000752     3.04   0.002     .2155576    .9998381
      4 vs 1 |  -.1144097    .194821    -0.59   0.557    -.4962518    .2674324
        Diff |
      2 vs 1 |   5.169253   5.017017     1.03   0.303    -4.663919    15.00242
      3 vs 1 |   -3.20606   1.043754    -3.07   0.002    -5.251781   -1.160339
      4 vs 1 |   14.36817   24.79236     0.58   0.562    -34.22397     62.9603
-------------+----------------------------------------------------------------
q3           |
     Discrim |
      2 vs 1 |   -.214686    .214803    -1.00   0.318    -.6356921    .2063201
      3 vs 1 |   .7354083   .2053428     3.58   0.000     .3329438    1.137873
      4 vs 1 |   1.272605   .2287827     5.56   0.000     .8241987     1.72101
        Diff |
      2 vs 1 |     1.3132   2.006775     0.65   0.513    -2.620008    5.246407
      3 vs 1 |  -1.752087   .3919143    -4.47   0.000    -2.520224   -.9839486
      4 vs 1 |  -1.145029    .177832    -6.44   0.000    -1.493573   -.7964847
-------------+----------------------------------------------------------------
q4           |
     Discrim |
      2 vs 1 |   .9090316   .1734278     5.24   0.000     .5691194    1.248944
      3 vs 1 |   .6275533   .1588602     3.95   0.000     .3161931    .9389136
      4 vs 1 |   1.387606   .2034136     6.82   0.000     .9889224    1.786289
        Diff |
      2 vs 1 |  -.9150171   .1377816    -6.64   0.000    -1.185064   -.6449703
      3 vs 1 |  -1.445922    .269392    -5.37   0.000     -1.97392   -.9179229
      4 vs 1 |  -.4153831   .0870474    -4.77   0.000    -.5859929   -.2447734
------------------------------------------------------------------------------
```

Looking at item q1, we see that the second category, labeled 2 vs 1, is the most discriminating among the respondents. The difficulty parameters represent the points at which the base outcome intersects with the other outcomes. For item q1, the estimated difficulty for category 2 is -0.85; this means that a person with $\theta = -0.85$ would be equally likely to select response 1 or response 2 on this item. Likewise, category 3 has a estimated difficulty of -1.35, so a person with $\theta = -1.35$ would be equally likely to select responses 1 or 3.

For an NRM, it is easiest to interpret the parameters by plotting the CCCs. The curves trace the probability of choosing each category as a function of θ using the estimated NRM parameters. Here we plot the probabilities for item q1 using irtgraph icc; see [IRT] **irtgraph icc** for details.

```
. irtgraph icc q1, xlabel(-4 -1.35 -.85  -.56 4, alt)
```



We see that respondents with the latent trait level below approximately -0.7 tend to endorse category 4, and respondents with the latent trait level above that point tend to choose category 2.

Although the other two alternatives are dominated by category 2 and 4, this does not mean that the dominated alternatives are not viable responses. In the discussion above, we purposefully used “tend” rather than “most probable” because we can consider the probability of choosing category 4 in two ways: the probability of choosing category 4 over any other single category and the probability of choosing category 4 over all other categories combined. For values of the latent trait below about -0.7, the probability of choosing category 4 is larger than the probability of choosing any other individual category. However, for values of θ below approximately -2.7, we can also say that the probability of choosing category 4 is greater than the probability of choosing all other categories combined. In the range of $(-2.7, -0.7)$, the sum of the probabilities of categories 1, 2, and 3 together exceeds the probability of a response in category 4. Thus, in this range, it is more probable that a respondent chooses “anything other than 4” rather than “4”. We can use the same argument for category 2 in the range $(-0.7, 0.6)$.

◁

Video example

Item response theory using Stata: Nominal response models (NRMs)

Stored results

irt nrm stores the following in e():

Scalars

e(N)	number of observations
e(k)	number of parameters
e(k_eq)	number of equations in e(b)
e(k_dv)	number of dependent variables
e(k_rc)	number of covariances
e(k_rs)	number of variances
e(irt_k_eq)	number of IRT equations
e(k_items1)	number of items in first IRT equation
e(k_out#)	number of outcomes for the #th item, nominal
e(ll)	log likelihood
e(N_clust)	number of clusters
e(N_groups)	number of groups
e(n_quad)	number of integration points
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if target model converged, 0 otherwise

Macros

e(cmd)	gsem
e(cmd2)	irt
e(cmdline)	command as typed
e(model1)	nrm
e(items1)	names of items in first IRT equation
e(depvar)	names of all item variables
e(wtype)	weight type
e(wexp)	weight expression
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(groupvar)	name of group variable
e(family#)	family for the #th *item*
e(link#)	link for the #th *item*
e(intmethod)	integration method
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. err.
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization
e(method)	estimation method: ml
e(ml_method)	type of ml method
e(user)	name of likelihood-evaluator program
e(technique)	maximization technique
e(datasignature)	the checksum
e(datasignaturevars)	variables used in calculation of checksum
e(properties)	b V
e(estat_cmd)	program used to implement estat
e(predict)	program used to implement predict
e(covariates)	list of covariates
e(footnote)	program used to implement the footnote display

Matrices	
`e(_N)`	sample size for each item
`e(b)`	coefficient vector, slope-intercept parameterization
`e(b_pclass)`	parameter class
`e(out#)`	outcomes for the #th item, nominal
`e(Cns)`	constraints matrix
`e(ilog)`	iteration log (up to 20 iterations)
`e(gradient)`	gradient vector
`e(V)`	variance–covariance matrix of the estimators
`e(V_modelbased)`	model-based variance
`e(groupvalue)`	vector of group values in `e(groupvar)`
`e(nobs)`	vector with number of observations per group
Functions	
`e(sample)`	marks estimation sample

In addition to the above, the following is stored in `r()`:

Matrices	
`r(table)`	matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

Let Y_{ij} represent the (yet to be observed) outcome for item i from person j. Without loss of generality, we will assume all items take on the unordered categories, $k = 1, \dots, K$.

Using the IRT parameterization, we see that the probability of person j with latent trait level θ_j (the latent trait) providing response k for item i is given by

$$\Pr(Y_{ij} = k|\mathbf{a}_i, \mathbf{b}_i, \theta_j) = \frac{\exp\{a_{ik}(\theta_j - b_{ik})\}}{\sum_{h=1}^{K} \exp\{a_{ih}(\theta_j - b_{ih})\}}$$

where $\mathbf{a}_i = (a_{i1}, \dots, a_{iK})$ represent the discrimination for each category of item i and $\mathbf{b}_i = (b_{i1}, \dots, b_{iK})$ represent the difficulties for each category of item i. `irt nrm` fits the model using the slope-intercept form, so the probability for providing response k is parameterized as

$$\Pr(Y_{ij} = k|\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \theta_j) = \frac{\exp(\alpha_{ik}\theta_j + \beta_{ik})}{\sum_{h=1}^{K} \exp(\alpha_{ih}\theta_j + \beta_{ih})}$$

The transformation between these two parameterizations is

$$a_{ik} = \alpha_{ik} \qquad b_{ik} = -\frac{\beta_{ik}}{\alpha_{ik}}$$

`irt nrm` uses baseline constraints to ensure the model is identified; the baseline constraints are set on the slope and intercept for the first outcome for each item, for example, $a_{i1} = 0$ and $b_{i1} = 0$. This baseline outcome is necessary for the model to be identified.

Let y_{ij} be the observed response for Y_{ij} and $p_{ij} = \Pr(Y_{ij} = y_{ij}|\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \theta_j)$. Conditionally on θ_j, the item responses are assumed independent, so the conditional density for person j is given by

$$f(\mathbf{y}_j|\boldsymbol{B}, \theta_j) = \prod_{i=1}^{I} p_{ij}$$

where $\mathbf{y}_j = (y_{1j}, \dots, y_{Ij})$, $\boldsymbol{B} = (\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_I, \boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_I)$, and I is the number of items.

Missing items are skipped over in the above product by default. When the `listwise` option is specified, persons with any missing items are dropped from the estimation sample.

The likelihood for person j is computed by integrating out the latent variable from the joint density

$$L_j(\boldsymbol{B}) = \int_{-\infty}^{\infty} f(\mathbf{y}_j|\boldsymbol{B},\theta_j)\,\phi(\theta_j)\,d\theta_j$$

where $\phi(\cdot)$ is the density function for the standard normal distribution. The log likelihood for the estimation sample is simply the sum of the log likelihoods from the N persons in the estimation sample.

$$\log L(\boldsymbol{B}) = \sum_{j=1}^{N} \log L_j(\boldsymbol{B})$$

The integral in the formula for $L_j(\boldsymbol{B})$ is generally not tractable, so we must use numerical methods.

Models for multiple groups, Gauss–Hermite quadrature, and adaptive quadrature are documented in *Methods and formulas* of [IRT] **irt hybrid**.

References

Baker, F. B., and S.-H. Kim. 2004. *Item Response Theory: Parameter Estimation Techniques*. 2nd ed, revised and expanded. Boca Raton, FL: CRC Press.

Bock, R. D. 1972. Estimating item parameters and latent ability when responses are scored in two or more nominal categories. *Psychometrika* 37: 29–51. https://doi.org/10.1007/BF02291411.

de Ayala, R. J. 2022. *The Theory and Practice of Item Response Theory*. 2nd ed. New York: Guilford Press.

Also see

[IRT] **irt nrm postestimation** — Postestimation tools for irt nrm

[IRT] **irt** — Introduction to IRT models

[IRT] **irt constraints** — Specifying constraints

[IRT] **irt grm** — Graded response model

[IRT] **irt pcm** — Partial credit model

[IRT] **irt rsm** — Rating scale model

[SEM] **gsem** — Generalized structural equation model estimation command

[SVY] **svy estimation** — Estimation commands for survey data

[U] **20 Estimation and postestimation commands**

For suggested citations, see the FAQ on citing Stata documentation.