

[Description](#)
[Options](#)
[References](#)[Quick start](#)
[Remarks and examples](#)
[Also see](#)[Menu](#)
[Stored results](#)[Syntax](#)
[Methods and formulas](#)

Description

`ologit` fits ordered logit models of ordinal variable *depvar* on the independent variables *indepvars*. The actual values taken on by the dependent variable are irrelevant, except that larger values are assumed to correspond to “higher” outcomes.

Quick start

Ordinal logit model of *y* on *x1* and categorical variables *a* and *b*

```
ologit y x1 i.a i.b
```

Same as above, and include interaction between *a* and *b* and report results as odds ratios

```
ologit y x1 a##b, or
```

With bootstrap standard errors

```
ologit y x1 i.a i.b, vce(bootstrap)
```

Analysis restricted to cases where `catvar = 0` using `svyset` data with replicate weights

```
svy bootstrap, subpop(if catvar==0): ologit y x1 i.a i.b
```

Menu

Statistics > Ordinal outcomes > Ordered logistic regression

Syntax

ologit *depvar* [*indepvars*] [*if*] [*in*] [*weight*] [, *options*]

<i>options</i>	Description
Model	
<code>offset(<i>varname</i>)</code>	include <i>varname</i> in model with coefficient constrained to 1
<code>constraints(<i>constraints</i>)</code>	apply specified linear constraints
SE/Robust	
<code>vce(<i>vcetype</i>)</code>	<i>vcetype</i> may be <code>oim</code> , <code>opg</code> , <code>robust</code> , <code>cluster <i>clustvar</i></code> , <code>bootstrap</code> , or <code>jackknife</code>
Reporting	
<code>level(#)</code>	set confidence level; default is <code>level(95)</code>
<code>or</code>	report odds ratios
<code>nocnsreport</code>	do not display constraints
<code>display_options</code>	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Maximization	
<code>maximize_options</code>	control the maximization process; seldom used
<code>collinear</code>	keep collinear variables
<code>coeflegend</code>	display legend instead of statistics

indepvars may contain factor variables; see [U] 11.4.3 Factor variables.

depvar and *indepvars* may contain time-series operators; see [U] 11.4.4 Time-series varlists.

`bayes`, `bayesboot`, `bootstrap`, `by`, `fmm`, `fp`, `jackknife`, `mfp`, `mi estimate`, `nestreg`, `rolling`, `statsby`, `stepwise`, and `svy` are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] `bayes: ologit` and [FMM] `fmm: ologit`.

`vce(bootstrap)` and `vce(jackknife)` are not allowed with the `mi estimate` prefix; see [MI] `mi estimate`.

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`.

`collinear` and `coeflegend` do not appear in the dialog box.

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

Options

Model
<code>offset(<i>varname</i>)</code> , <code>constraints(<i>constraints</i>)</code> ; see [R] Estimation options.
SE/Robust
<code>vce(<i>vcetype</i>)</code> specifies the type of standard error reported, which includes types that are derived from asymptotic theory (<code>oim</code> , <code>opg</code>), that are robust to some kinds of misspecification (<code>robust</code>), that allow for intragroup correlation (<code>cluster <i>clustvar</i></code>), and that use bootstrap or jackknife methods (<code>bootstrap</code> , <code>jackknife</code>); see [R] <code>vce_option</code> .

Reporting

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

or reports the estimated coefficients transformed to odds ratios, that is, e^b rather than b . Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated. or may be specified at estimation or when replaying previously estimated results.

`nocnsreport`; see [R] [Estimation options](#).

display_options: `nocl`, `nopvalues`, `noomitted`, `vsquish`, `noemptycells`, `baselevels`, `allbaselevels`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(style)`, `cformat(%fmt)`, `pformat(%fmt)`, `sformat(%fmt)`, and `nolstretch`; see [R] [Estimation options](#).

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](#). These options are seldom used.

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

`collinear`, `coeflegend`; see [R] [Estimation options](#).

Remarks and examples

Ordered logit models are used to estimate relationships between an ordinal dependent variable and a set of independent variables. An *ordinal* variable is a variable that is categorical and ordered, for instance, “poor”, “good”, and “excellent”, which might indicate a person’s current health status or the repair record of a car. If there are only two outcomes, see [R] [logistic](#), [R] [logit](#), and [R] [probit](#). This entry is concerned only with more than two outcomes. If the outcomes cannot be ordered (for example, residency in the north, east, south, or west), see [R] [mlogit](#). This entry is concerned only with models in which the outcomes can be ordered.

In ordered logit, an underlying score is estimated as a linear function of the independent variables and a set of cutpoints. The probability of observing outcome i corresponds to the probability that the estimated linear function, plus random error, is within the range of the cutpoints estimated for the outcome:

$$\Pr(\text{outcome}_j = i) = \Pr(\kappa_{i-1} < \beta_1 x_{1j} + \beta_2 x_{2j} + \cdots + \beta_k x_{kj} + u_j \leq \kappa_i)$$

u_j is assumed to be logistically distributed in ordered logit. In either case, we estimate the coefficients $\beta_1, \beta_2, \dots, \beta_k$ together with the cutpoints $\kappa_1, \kappa_2, \dots, \kappa_{k-1}$, where k is the number of possible outcomes. κ_0 is taken as $-\infty$, and κ_k is taken as $+\infty$. All of this is a direct generalization of the ordinary two-outcome logit model.

Example 1

We wish to analyze the 1977 repair records of 66 foreign and domestic cars. The data are a variation of the automobile dataset described in [U] 1.2.2 Example datasets. The 1977 repair records, like those in 1978, take on values “Poor”, “Fair”, “Average”, “Good”, and “Excellent”. Here is a cross-tabulation of the data:

```
. use https://www.stata-press.com/data/r19/fullauto
(Automobile models)
. tabulate rep77 foreign, chi2
```

Repair record 1977	Foreign		Total
	Domestic	Foreign	
Poor	2	1	3
Fair	10	1	11
Average	20	7	27
Good	13	7	20
Excellent	0	5	5
Total	45	21	66

Pearson chi2(4) = 13.8619 Pr = 0.008

Although it appears that foreign takes on the values Domestic and Foreign, it is actually a numeric variable taking on the values 0 and 1. Similarly, rep77 takes on the values 1, 2, 3, 4, and 5, corresponding to Poor, Fair, and so on. The more meaningful words appear because we have attached value labels to the data; see [U] 12.6.3 Value labels.

Because the χ^2 value is significant, we could claim that there is a relationship between foreign and rep77. Literally, however, we can only claim that the distributions are different; the χ^2 test is not directional. One way to model these data is to model the categorization that took place when the data were created. Cars have a true frequency of repair, which we will assume is given by $S_j = \beta \text{foreign}_j + u_j$, and a car is categorized as “poor” if $S_j \leq \kappa_0$, as “fair” if $\kappa_0 < S_j \leq \kappa_1$, and so on:

```
. ologit rep77 foreign
Iteration 0: Log likelihood = -89.895098
Iteration 1: Log likelihood = -85.951765
Iteration 2: Log likelihood = -85.908227
Iteration 3: Log likelihood = -85.908161
Iteration 4: Log likelihood = -85.908161
Ordered logistic regression
Log likelihood = -85.908161
Number of obs = 66
LR chi2(1) = 7.97
Prob > chi2 = 0.0047
Pseudo R2 = 0.0444
```

rep77	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
foreign	1.455878	.5308951	2.74	0.006	.4153425	2.496413
/cut1	-2.765562	.5988208			-3.939229	-1.591895
/cut2	-.9963603	.3217706			-1.627019	-.3657016
/cut3	.9426153	.3136398			.3278925	1.557338
/cut4	3.123351	.5423257			2.060412	4.18629

Our model is $S_j = 1.46 \text{foreign}_j + u_j$; the expected value for foreign cars is 1.46 and, for domestic cars, 0; foreign cars have better repair records.

The estimated cutpoints tell us how to interpret the score. For a foreign car, the probability of a poor record is the probability that $1.46 + u_j \leq -2.77$, or equivalently, $u_j \leq -4.23$. Making this calculation requires familiarity with the logistic distribution: the probability is $1/(1 + e^{4.23}) = 0.014$. On the other hand, for domestic cars, the probability of a poor record is the probability $u_j \leq -2.77$, which is 0.059.

This, it seems to us, is a far more reasonable prediction than we would have made based on the table alone. The table showed that 2 of 45 domestic cars had poor records, whereas 1 of 21 foreign cars had poor records—corresponding to probabilities $2/45 = 0.044$ and $1/21 = 0.048$. The predictions from our model imposed a smoothness assumption—foreign cars should not, overall, have better repair records without the difference revealing itself in each category. In our data, the fractions of foreign and domestic cars in the poor category are virtually identical only because of the randomness associated with small samples.

Thus if we were asked to predict the true fractions of foreign and domestic cars that would be classified in the various categories, we would choose the numbers implied by the ordered logit model:

	tabulate		logit	
	Domestic	Foreign	Domestic	Foreign
Poor	0.044	0.048	0.059	0.014
Fair	0.222	0.048	0.210	0.065
Average	0.444	0.333	0.450	0.295
Good	0.289	0.333	0.238	0.467
Excellent	0.000	0.238	0.043	0.159

See [R] [ologit postestimation](#) for a more complete explanation of how to generate predictions from an ordered logit model.



□ Technical note

Here ordered logit provides an alternative to ordinary two-outcome logistic models with an arbitrary dichotomization, which might otherwise have been tempting. We could, for instance, have summarized these data by converting the five-outcome `rep77` variable to a two-outcome variable, combining cars in the average, fair, and poor categories to make one outcome and combining cars in the good and excellent categories to make the second.

Another even less appealing alternative would have been to use ordinary regression, arbitrarily labeling “excellent” as 5, “good” as 4, and so on. The problem is that with different but equally valid labelings (say, 10 for “excellent”), we would obtain different estimates. We would have no way of choosing one metric over another. That assertion is not, however, true of `ologit`. The actual values used to label the categories make no difference other than through the order they imply.

In fact, our labeling was 5 for “excellent”, 4 for “good”, and so on. The words “excellent” and “good” appear in our output because we attached a value label to the variables; see [U] [12.6.3 Value labels](#). If we were to now go back and type `replace rep77=10 if rep77==5`, changing all the 5s to 10s, we would still obtain the same results when we refit our model.



► Example 2

In the [example](#) above, we used ordered logit as a way to model a table. We are not, however, limited to including only one explanatory variable or to including only categorical variables. We can explore the relationship of rep77 with any of the variables in our data. We might, for instance, model rep77 not only in terms of the origin of manufacture, but also including length (a proxy for size) and mpg:

```
. ologit rep77 foreign length mpg
Iteration 0:  Log likelihood = -89.895098
Iteration 1:  Log likelihood = -78.775147
Iteration 2:  Log likelihood = -78.254294
Iteration 3:  Log likelihood = -78.250719
Iteration 4:  Log likelihood = -78.250719

Ordered logistic regression                                Number of obs =      66
LR chi2(3)      = 23.29
Prob > chi2     = 0.0000
Pseudo R2       = 0.1295

Log likelihood = -78.250719
```

rep77	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
foreign	2.896807	.7906411	3.66	0.000	1.347179	4.446435
length	.0828275	.02272	3.65	0.000	.0382972	.1273579
mpg	.2307677	.0704548	3.28	0.001	.0926788	.3688566
/cut1	17.92748	5.551191			7.047344	28.80761
/cut2	19.86506	5.59648			8.896161	30.83396
/cut3	22.10331	5.708936			10.914	33.29262
/cut4	24.69213	5.890754			13.14647	36.2378

foreign still plays a role—and an even larger role than previously. We find that larger cars tend to have better repair records, as do cars with better mileage ratings.



Stored results

ologit stores the following in e():

Scalars	
e(N)	number of observations
e(N_cd)	number of completely determined observations
e(k_cat)	number of categories
e(k)	number of parameters
e(k_aux)	number of auxiliary parameters
e(k_eq)	number of equations in e(b)
e(k_eq_model)	number of equations in overall model test
e(k_dv)	number of dependent variables
e(df_m)	model degrees of freedom
e(r2_p)	pseudo- R^2
e(ll)	log likelihood
e(ll_0)	log likelihood, constant-only model
e(N_clust)	number of clusters
e(chi2)	χ^2
e(p)	p-value for model test
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise

Macros

<code>e(cmd)</code>	<code>ologit</code>
<code>e(cmdline)</code>	command as typed
<code>e(depvar)</code>	name of dependent variable
<code>e(wtype)</code>	weight type
<code>e(wexp)</code>	weight expression
<code>e(title)</code>	title in estimation output
<code>e(clustvar)</code>	name of cluster variable
<code>e(offset)</code>	linear offset variable
<code>e(chi2type)</code>	Wald or LR; type of model χ^2 test
<code>e(vce)</code>	<i>vcetype</i> specified in <code>vce()</code>
<code>e(vcetype)</code>	title used to label Std. err.
<code>e(opt)</code>	type of optimization
<code>e(which)</code>	max or min; whether optimizer is to perform maximization or minimization
<code>e(ml_method)</code>	type of ml method
<code>e(user)</code>	name of likelihood-evaluator program
<code>e(technique)</code>	maximization technique
<code>e(properties)</code>	b V
<code>e(predict)</code>	program used to implement <code>predict</code>
<code>e(marginsdefault)</code>	default <code>predict()</code> specification for <code>margins</code>
<code>e(asbalanced)</code>	factor variables <code>fvset</code> as <code>asbalanced</code>
<code>e(asobserved)</code>	factor variables <code>fvset</code> as <code>asobserved</code>

Matrices

<code>e(b)</code>	coefficient vector
<code>e(Cns)</code>	constraints matrix
<code>e(ilog)</code>	iteration log (up to 20 iterations)
<code>e(gradient)</code>	gradient vector
<code>e(cat)</code>	category values
<code>e(V)</code>	variance–covariance matrix of the estimators
<code>e(V_modelbased)</code>	model-based variance

Functions

<code>e(sample)</code>	marks estimation sample
------------------------	-------------------------

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

Matrices

<code>r(table)</code>	matrix containing the coefficients with their standard errors, test statistics, <i>p</i> -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

See [Long and Freese \(2014, chap. 7\)](#) for a discussion of models for ordinal outcomes and examples that use Stata. [Cameron and Trivedi \(2005, chap. 15\)](#) describe multinomial models, including the model fit by `ologit`. When you have a qualitative dependent variable, several estimation procedures are available. A popular choice is multinomial logistic regression (see [\[R\] mlogit](#)), but if you use this procedure when the response variable is ordinal, you are discarding information because multinomial logit ignores the ordered aspect of the outcome. Ordered logit and probit models provide a means to exploit the ordering information.

There is more than one “ordered logit” model. The model fit by `ologit`, which we will call the ordered logit model, is also known as the proportional odds model. Another popular choice, not fit by `ologit`, is known as the stereotype model; see [\[R\] slogit](#). All ordered logit models have been derived by starting with a binary logit/probit model and generalizing it to allow for more than two outcomes.

The proportional-odds ordered logit model is so called because, if we consider the odds $\text{odds}(k) = P(Y \leq k)/P(Y > k)$, then $\text{odds}(k_1)$ and $\text{odds}(k_2)$ have the same ratio for all independent variable combinations. The model is based on the principle that the only effect of combining adjoining categories in ordered categorical regression problems should be a loss of efficiency in estimating the regression parameters (McCullagh 1980). This model was also described by McKelvey and Zavoina (1975) and, previously, by Aitchison and Silvey (1957) in a different algebraic form. Brant (1990) offers a set of diagnostics for the model.

Peterson and Harrell (1990) suggest a model that allows nonproportional odds for a subset of the explanatory variables. `ologit` does not allow this, but a model similar to this was implemented by Fu (1998).

The stereotype model rejects the principle on which the ordered logit model is based. Anderson (1984) argues that there are two distinct types of ordered categorical variables: “grouped continuous”, such as income, where the “type a” model applies; and “assessed”, such as extent of pain relief, where the stereotype model applies. Greenland (1985) independently developed the same model. The stereotype model starts with a multinomial logistic regression model and imposes constraints on this model.

Goodness of fit for `ologit` can be evaluated by comparing the likelihood value with that obtained by fitting the model with `mlogit`. Let $\ln L_1$ be the log-likelihood value reported by `ologit`, and let $\ln L_0$ be the log-likelihood value reported by `mlogit`. If there are p independent variables (excluding the constant) and k categories, `mlogit` will estimate $p(k - 1)$ additional parameters. We can then perform a “likelihood-ratio test”, that is, calculate $-2(\ln L_1 - \ln L_0)$, and compare it with $\chi^2\{p(k - 2)\}$. This test is suggestive only because the ordered logit model is not nested within the multinomial logit model. A large value of $-2(\ln L_1 - \ln L_0)$ should, however, be taken as evidence of poorness of fit. Marginally large values, on the other hand, should not be taken too seriously.

The coefficients and cutpoints are estimated using maximum likelihood as described in [R] **Maximize**. In our parameterization, no constant appears, because the effect is absorbed into the cutpoints.

`ologit` and `oprobit` begin by tabulating the dependent variable. Category $i = 1$ is defined as the minimum value of the variable, $i = 2$ as the next ordered value, and so on, for the empirically determined k categories.

The probability of a given observation for ordered logit is

$$\begin{aligned} p_{ij} &= \Pr(y_j = i) = \Pr(\kappa_{i-1} < \mathbf{x}_j\boldsymbol{\beta} + u \leq \kappa_i) \\ &= \frac{1}{1 + \exp(-\kappa_i + \mathbf{x}_j\boldsymbol{\beta})} - \frac{1}{1 + \exp(-\kappa_{i-1} + \mathbf{x}_j\boldsymbol{\beta})} \end{aligned}$$

κ_0 is defined as $-\infty$ and κ_k as $+\infty$.

For ordered probit, the probability of a given observation is

$$\begin{aligned} p_{ij} &= \Pr(y_j = i) = \Pr(\kappa_{i-1} < \mathbf{x}_j\boldsymbol{\beta} + u \leq \kappa_i) \\ &= \Phi(\kappa_i - \mathbf{x}_j\boldsymbol{\beta}) - \Phi(\kappa_{i-1} - \mathbf{x}_j\boldsymbol{\beta}) \end{aligned}$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function.

The log likelihood is

$$\ln L = \sum_{j=1}^N w_j \sum_{i=1}^k I_i(y_j) \ln p_{ij}$$

where w_j is an optional weight and

$$I_i(y_j) = \begin{cases} 1, & \text{if } y_j = i \\ 0, & \text{otherwise} \end{cases}$$

ologit and oprobit support the Huber/White/sandwich estimator of the variance and its clustered version using `vce(robust)` and `vce(cluster clustvar)`, respectively. See [P] [_robust](#), particularly [Maximum likelihood estimators](#) and [Methods and formulas](#).

These commands also support estimation with survey data. For details on VCEs with survey data, see [SVY] [Variance estimation](#).

References

- Aitchison, J., and S. D. Silvey. 1957. The generalization of probit analysis to the case of multiple responses. *Biometrika* 44: 131–140. <https://doi.org/10.2307/2333245>.
- Anderson, J. A. 1984. Regression and ordered categorical variables (with discussion). *Journal of the Royal Statistical Society, B ser.*, 46: 1–30. <https://doi.org/10.1111/j.2517-6161.1984.tb01270.x>.
- Baetschmann, G., A. Ballantyne, K. E. Staub, and R. Winkelmann. 2020. `feologit: A new command for fitting fixed-effects ordered logit models`. *Stata Journal* 20: 253–275.
- Bauldry, S., J. Xu, and A. S. Fullerton. 2018. `gencrm: A new command for generalized continuation-ratio models`. *Stata Journal* 18: 924–936.
- Brant, R. 1990. Assessing proportionality in the proportional odds model for ordinal logistic regression. *Biometrics* 46: 1171–1178. <https://doi.org/10.2307/2532457>.
- Cameron, A. C., and P. K. Trivedi. 2005. *Microeconometrics: Methods and Applications*. New York: Cambridge University Press.
- Fagerland, M. W. 2014. `adjcatlogit`, `ccrlogit`, and `ucrlogit: Fitting ordinal logistic regression models`. *Stata Journal* 14: 947–964.
- Fu, V. K. 1998. `sg88: Estimating generalized ordered logit models`. *Stata Technical Bulletin* 44: 27–30. Reprinted in *Stata Technical Bulletin Reprints*, vol. 8, pp. 160–164. College Station, TX: Stata Press.
- Fullerton, A. S., and J. Xu. 2016. *Ordered Regression Models: Parallel, Partial, and Non-Parallel Alternatives*. Boca Raton, FL: CRC Press.
- Greenland, S. 1985. An application of logistic models to the analysis of ordinal responses. *Biometrical Journal* 27: 189–197. <https://doi.org/10.1002/bimj.4710270212>.
- Liu, X. 2016. *Applied Ordinal Logistic Regression Using Stata*. Thousand Oaks, CA: Sage.
- Long, J. S. 1997. *Regression Models for Categorical and Limited Dependent Variables*. Thousand Oaks, CA: Sage.
- Long, J. S., and J. Freese. 2014. *Regression Models for Categorical Dependent Variables Using Stata*. 3rd ed. College Station, TX: Stata Press.
- McCullagh, P. 1977. A logistic model for paired comparisons with ordered categorical data. *Biometrika* 64: 449–453. <https://doi.org/10.2307/2345320>.
- . 1980. Regression models for ordinal data (with discussion). *Journal of the Royal Statistical Society, B ser.*, 42: 109–142. <https://doi.org/10.1111/j.2517-6161.1980.tb01109.x>.
- McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*. 2nd ed. London: Chapman and Hall/CRC.
- McKelvey, R. D., and W. Zavoina. 1975. A statistical model for the analysis of ordinal level dependent variables. *Journal of Mathematical Sociology* 4: 103–120. <https://doi.org/10.1080/0022250X.1975.9989847>.
- Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. *Stata Journal* 6: 285–308.
- Peterson, B., and F. E. Harrell, Jr. 1990. Partial proportional odds models for ordinal response variables. *Journal of the Royal Statistical Society, C ser.*, 39: 205–217. <https://doi.org/10.2307/2347760>.

- Smith, E. K., M. G. Lacy, and A. Mayer. 2019. Performance simulations for categorical mediation: Analyzing khb estimates of mediation in ordinal regression models. *Stata Journal* 19: 913–930.
- Williams, R. 2006. Generalized ordered logit/partial proportional odds models for ordinal dependent variables. *Stata Journal* 6: 58–82.
- . 2010. Fitting heterogeneous choice models with oglm. *Stata Journal* 10: 540–567.
- Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. *Stata Journal* 5: 537–559.

Also see

- [R] **ologit postestimation** — Postestimation tools for ologit
- [R] **clologit** — Conditional (fixed-effects) logistic regression
- [R] **logistic** — Logistic regression, reporting odds ratios
- [R] **logit** — Logistic regression, reporting coefficients
- [R] **mlogit** — Multinomial (polytomous) logistic regression
- [R] **oprobit** — Ordered probit regression
- [R] **slogit** — Stereotype logistic regression
- [R] **ziologit** — Zero-inflated ordered logit regression
- [BAYES] **bayes: ologit** — Bayesian ordered logistic regression
- [CM] **cmrologit** — Rank-ordered logit choice model
- [CM] **cmroprobit** — Rank-ordered probit choice model
- [FMM] **fmm: ologit** — Finite mixtures of ordered logistic regression models
- [ME] **meologit** — Multilevel mixed-effects ordered logistic regression
- [MI] **Estimation** — Estimation commands for use with mi estimate
- [SVY] **svy estimation** — Estimation commands for survey data
- [XT] **xtologit** — Random-effects ordered logistic model
- [U] **20 Estimation and postestimation commands**

