
ORDPROB (nonlinear options)
dependent variable list of independent variables ;

Function:

ORDPROB obtains estimates of the linear Ordered Probit model, where the dependent variable takes on only nonnegative integer ordered category values. The scaling of the category values does not matter (although they should be positive and integer for convenience); only information about their order is used in estimation.

Usage:

The basic ORDPROB statement is like the OLSQ statement: first list the dependent variable and then the independent variables. If you wish to have an intercept term in the regression (usually recommended), include the special variable C or CONSTANT in your list of independent variables. You may have as many independent variables as you like subject to the overall limits on the number of arguments per statement and the amount of working space, as well as the number of data observations you have available.

The observations over which the regression is computed are determined by the current sample. If any of the observations have missing values within the current sample, ORDPROB will print a warning message and will drop those observations. ORDPROB also checks that the observations on the dependent variable are integers and are not negative.

The list of independent variables on the ORDPROB command may include variables with explicit lags and leads as well as PDL (Polynomial Distributed Lag) variables. These distributed lag variables are a way to reduce the number of free coefficients when entering a large number of lagged variables in a regression by imposing smoothness on the coefficients. See the PDL section for a description of how to specify such variables.

Options:

Nonlinear options - see the NONLINEAR section of this manual.

Examples:

Ordered Probit regression of patents on lags of log(R&D), science sector dummy, and firm size:

```
ORDPROB PATENTS C LRND LRND(-1) LRND(-2) DSCI SIZE;
```

Output:

The output of ORDPROB begins with an equation title and frequency counts for the lowest 10 values of the dependent variable. Starting values and diagnostic output from the iterations will be printed. Final convergence status is printed. This is followed by the number of observations, mean and standard deviation of the dependent variable, sum of squared residuals, scaled R-squared, likelihood ratio test for zero slopes, log likelihood, and a table of right hand side variable names, estimated coefficients, standard errors and associated t-statistics.

ORDPROB also stores some of these results in data storage for later use. The table below lists the results available after a ORDPROB command.

ORDPROB

Name	Type	Length	Variable Description
@LHV	list	1	Name of dependent variable.
@RNMS	list	#vars	Names of right hand side variables.
@IFCONV	scalar	1	1 if convergence achieved, 0 otherwise.
@YMEAN	scalar	1	Mean of the dependent variable.
@SDEV	scalar	1	Standard deviation of the dependent variable.
@NOB	scalar	1	Number of observations.
@HIST	vector	#yvalues	Frequency counts for each dependent variable value.
@HISTVAL	vector	#yvalues	Each dependent variable value.
@SSR	scalar	1	Sum of squared residuals.
@SRSQ	scalar	1	Scaled R-squared.
@LR	scalar	1	Likelihood ratio test for zero slope coefficients.
%LR	scalar	1	P-value for likelihood ratio test.
@LOGL	scalar	1	Log of likelihood function.
@SBIC	scalar	1	Schwarz Bayesian Information Criterion.
@NCOEF	scalar	1	Number of rhs variables (#vars).
@NCID	scalar	1	Number of identified coefficients.
@COEF	vector	#vars	Coefficient estimates.
@SES	vector	#vars	Standard errors.
@T	vector	#vars	T-statistics.
%T	vector	#vars	p-values for T-statistics.
@GRAD	vector	#vars	Gradient of log L at convergence.
@VCOV	matrix	#vars* #vars	Variance-covariance of estimated coefficients.
@FIT	series	#obs	Fitted probabilities.
@RES	series	#obs	Residuals.

If the regression includes a PDL variable, the following will also be stored:

@SLAG	scalar	1	Sum of the lag coefficients.
@MLAG	scalar	1	Mean lag coefficient (number of time periods).
@LAGF	vector	#lags	Estimated lag coefficients, after "unscrambling".

Method:

Like the binary Probit model, the Ordered Probit model is based on an unobserved continuous dependent variable, $y^* = XB + e$. Instead of y^* , we observe a category value Y , where a larger category value implies a larger value of y^* . In binary Probit, the category values are 0 for $y^* < 0$, and 1 for $y^* > 0$. In Ordered Probit, more than 2 category values are usually involved. The category values need not be consecutive, and the lowest category does not have to be 0. The boundary values between the different categories are estimated parameters (MUs). The lowest effective boundary value (MU1) is normalized to 0, just as in Binary Probit.

For example, suppose there are 3 categories, with category values 0, 1, and 2:

$$\begin{aligned}
 Y = 0 & \text{ if } \text{MU0} \leq XB + e < \text{MU1} && \text{(Note: MU0 = -infinity, and MU1 = 0)} \\
 Y = 1 & \text{ if } \text{MU1} \leq XB + e < \text{MU2} && \text{(MU2 is an estimated parameter)} \\
 Y = 2 & \text{ if } \text{MU2} \leq XB + e < \text{MU3} && \text{(Note: MU3 = infinity)}
 \end{aligned}$$

The MUs are always given names based on the category value for which they are the lower bound (MU2 in the example above is the lower bound for category with value 2). X normally includes a constant term (C), which can be thought of as a replacement for MU1; in this case, the other MUs can be interpreted as being measured relative to the value of C. The estimated MU values are constrained to follow a strict ordering (MU0 < MU1 < MU2, etc.). Negative and non-integer category values are not allowed, simply because they do not easily translate into names for the MUs. Just recode such values to integers (preserving the proper ordering).

ORDPROB uses analytic first and second derivatives to obtain maximum likelihood estimates via the Newton-Raphson algorithm. This algorithm usually converges fairly quickly. TSP uses zeros for starting parameter values, except for the constant term and the MUs. @START can be used to provide different starting values (see NONLINEAR in this manual).

Multicollinearity of the independent variables is handled with generalized inverses, as in the other linear and nonlinear regression procedures in TSP.

If you wish to estimate a nonstandard ordered probit model (e.g. adjusted for heteroskedasticity or with a nonlinear regression function), use the ML command. See the **TSP Users's Guide** for an example.

Before estimation, PROBIT checks for univariate complete and quasi-complete separation of the data and flags this condition. (The model is not identified in this case.) Without this check, one or more RHS variables perfectly predict Y for some observations, and their coefficients would slowly iterate to + or - infinity.

The Scaled R-squared is a measure of goodness of fit relative to a model with just a constant term; it is a nonlinear transformation of the Likelihood Ratio test for zero slopes. See Estrella(1998). Although the paper title mentions dichotomous dependent variables, the scaled R-squared applies to any model with a fixed number of categories, such as Ordered Probit and Multinomial Logit.

References:

Cameron, A. Colin, and Pravid K. Trivedi, **Regression Analysis of Count Data**, Cambridge University Press, New York, 1998, pp. 87-88.

Estrella, Arturo, "A New Measure of Fit for Equations with Dichotomous Dependent Variables," **Journal of Business and Economic Statistics**, April 1998, pp. 198-205.

Maddala, G. S., **Limited-dependent and Qualitative Variables in Econometrics**, Cambridge University Press, New York, 1983, pp. 46-49.