intreg — Interval regression

Syntax

`intreg` *depvar*$_1$ *depvar*$_2$ [*indepvars*] [*if*] [*in*] [*weight*] [, *options*]

options	Description
Model	
`noconstant`	suppress constant term
`het(`*varlist* [, `noconstant`]`)`	independent variables to model the variance; use `noconstant` to suppress constant term
`offset(`*varname*`)`	include *varname* in model with coefficient constrained to 1
`constraints(`*constraints*`)`	apply specified linear constraints
`collinear`	keep collinear variables
SE/Robust	
`vce(`*vcetype*`)`	*vcetype* may be `oim`, `robust`, `cluster` *clustvar*, `opg`, `bootstrap`, or `jackknife`
Reporting	
`level(`*#*`)`	set confidence level; default is `level(95)`
`nocnsreport`	do not display constraints
display_options	control column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Maximization	
maximize_options	control the maximization process; seldom used
`coeflegend`	display legend instead of statistics

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

depvar$_1$, *depvar*$_2$, *indepvars*, and *varlist* may contain time-series operators; see [U] **11.4.4 Time-series varlists**.

`bootstrap`, `by`, `fp`, `jackknife`, `mfp`, `nestreg`, `rolling`, `statsby`, `stepwise`, and `svy` are allowed; see [U] **11.1.10 Prefix commands**.

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

`aweight`s are not allowed with the `jackknife` prefix; see [R] **jackknife**.

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

`aweight`s, `fweight`s, `iweight`s, and `pweight`s are allowed; see [U] **11.1.6 weight**.

`coeflegend` does not appear in the dialog box.

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

Menu

Statistics > Linear models and related > Censored regression > Interval regression

Description

intreg fits a model of $y = [\,depvar_1, depvar_2\,]$ on *indepvars*, where y for each observation is point data, interval data, left-censored data, or right-censored data.

depvar$_1$ and *depvar*$_2$ should have the following form:

Type of data		*depvar*$_1$	*depvar*$_2$
point data	$a = [\,a, a\,]$	a	a
interval data	$[\,a, b\,]$	a	b
left-censored data	$(\,-\infty, b\,]$	.	b
right-censored data	$[\,a, +\infty\,)$	a	.

Options

Model

noconstant; see [R] **estimation options**.

het(*varlist* [, noconstant]) specifies that *varlist* be included in the specification of the conditional variance. This *varlist* enters the variance specification collectively as multiplicative heteroskedasticity.

offset(*varname*), constraints(*constraints*), collinear; see [R] **estimation options**.

SE/Robust

vce(*vcetype*) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), 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**.

nocnsreport; see [R] **estimation options**.

display_options: 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.

Setting the optimization type to technique(bhhh) resets the default *vcetype* to vce(opg).

The following option is available with intreg but is not shown in the dialog box:

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

Remarks and examples

intreg is a generalization of the models fit by tobit. Cameron and Trivedi (2010, 548–550) discuss the differences among censored, truncated, and interval data. If you know that the value for the jth individual is somewhere in the interval $[\,y_{1j},\,y_{2j}\,]$, then the likelihood contribution from this individual is simply $\Pr(y_{1j} \le Y_j \le y_{2j})$. For censored data, their likelihoods contain terms of the form $\Pr(Y_j \le y_j)$ for left-censored data and $\Pr(Y_j \ge y_j)$ for right-censored data, where y_j is the observed censoring value and Y_j denotes the random variable representing the dependent variable in the model.

Hence, intreg can fit models for data where each observation represents interval data, left-censored data, right-censored data, or point data. Regardless of the type of observation, the data should be stored in the dataset as interval data; that is, two dependent variables, *depvar*$_1$ and *depvar*$_2$, are used to hold the endpoints of the interval. If the data are left-censored, the lower endpoint is $-\infty$ and is represented by a missing value, '.', or an extended missing value, '.a, .b, ..., .z', in *depvar*$_1$. If the data are right-censored, the upper endpoint is $+\infty$ and is represented by a missing value, '.' (or an extended missing value), in *depvar*$_2$. Point data are represented by the two endpoints being equal.

Type of data		*depvar*$_1$	*depvar*$_2$
point data	$a = [\,a, a\,]$	a	a
interval data	$[\,a, b\,]$	a	b
left-censored data	$(\,-\infty, b\,]$	.	b
right-censored data	$[\,a, +\infty\,)$	a	.

Truly missing values of the dependent variable must be represented by missing values in both *depvar*$_1$ and *depvar*$_2$.

Interval data arise naturally in many contexts, such as wage data. Often you know only that, for example, a person's salary is between \$30,000 and \$40,000. Below we give an example for wage data and show how to set up *depvar*$_1$ and *depvar*$_2$.

▷ Example 1

We have a dataset that contains the yearly wages of working women. Women were asked via a questionnaire to indicate a category for their yearly income from employment. The categories were less than 5,000, 5,001–10,000, ..., 25,001–30,000, 30,001–40,000, 40,001–50,000, and more than 50,000. The wage categories are stored in the wagecat variable.

```
. use http://www.stata-press.com/data/r13/womenwage
(Wages of women)

. tabulate wagecat

       Wage
   category
   ($1000s)        Freq.     Percent        Cum.
------------------------------------------------
          5           14        2.87        2.87
         10           83       17.01       19.88
         15          158       32.38       52.25
         20          107       21.93       74.18
         25           57       11.68       85.86
         30           30        6.15       92.01
         40           19        3.89       95.90
         50           14        2.87       98.77
         51            6        1.23      100.00
------------------------------------------------
      Total          488      100.00
```

A value of 5 for `wagecat` represents the category less than 5,000, a value of 10 represents 5,001–10,000, . . . , and a value of 51 represents greater than 50,000.

To use `intreg`, we must create two variables, `wage1` and `wage2`, containing the lower and upper endpoints of the wage categories. Here is one way to do it. We first create a dataset containing the nine wage categories, lag the wage categories into `wage1`, and match-merge this dataset with nine observations back into the main one.

```
. by wagecat: keep if _n==1
(479 observations deleted)

. generate wage1 = wagecat[_n-1]
(1 missing value generated)

. keep wagecat wage1

. save lagwage
file lagwage.dta saved

. use http://www.stata-press.com/data/r13/womenwage
(Wages of women)

. merge m:1 wagecat using lagwage

    Result                           # of obs.
    -----------------------------------------
    not matched                             0
    matched                               488  (_merge==3)
    -----------------------------------------
```

Now we create the upper endpoint and list the new variables:

```
. generate wage2 = wagecat

. replace wage2 = . if wagecat == 51
(6 real changes made, 6 to missing)

. sort age, stable
```

```
. list wage1 wage2 in 1/10

     ┌───────────────┐
     │ wage1   wage2 │
     ├───────────────┤
  1. │     .       5 │
  2. │     5      10 │
  3. │     5      10 │
  4. │    10      15 │
  5. │     .       5 │
     ├───────────────┤
  6. │     .       5 │
  7. │     .       5 │
  8. │     5      10 │
  9. │     5      10 │
 10. │     5      10 │
     └───────────────┘
```

We can now run `intreg`:

```
. intreg wage1 wage2 age c.age#c.age nev_mar rural school tenure
Fitting constant-only model:
Iteration 0:   log likelihood = -967.24956
Iteration 1:   log likelihood =  -967.1368
Iteration 2:   log likelihood =  -967.1368
Fitting full model:
Iteration 0:   log likelihood = -856.65324
Iteration 1:   log likelihood = -856.33294
Iteration 2:   log likelihood = -856.33293
Interval regression                               Number of obs   =        488
                                                  LR chi2(6)      =     221.61
Log likelihood = -856.33293                       Prob > chi2     =     0.0000

-----------------------------------------------------------------------------
             |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+---------------------------------------------------------------
         age |   .7914438   .4433604     1.79   0.074    -.0775265    1.660414
             |
 c.age#c.age |  -.0132624   .0073028    -1.82   0.069    -.0275757    .0010509
             |
     nev_mar |  -.2075022   .8119581    -0.26   0.798    -1.798911    1.383906
       rural |  -3.043044   .7757324    -3.92   0.000    -4.563452   -1.522637
      school |   1.334721   .1357873     9.83   0.000     1.068583    1.600859
      tenure |   .8000664   .1045077     7.66   0.000     .5952351    1.004898
       _cons |  -12.70238   6.367117    -1.99   0.046     -25.1817   -.2230583
-------------+---------------------------------------------------------------
    /lnsigma |   1.987823   .0346543    57.36   0.000     1.919902    2.055744
-------------+---------------------------------------------------------------
       sigma |   7.299626   .2529634                       6.82029     7.81265
-----------------------------------------------------------------------------
  Observation summary:         14  left-censored observations
                                0     uncensored observations
                                6 right-censored observations
                              468       interval observations
```

We could also model these data by using an ordered probit model with `oprobit` (see [R] **oprobit**):

```
. oprobit wagecat age c.age#c.age nev_mar rural school tenure
Iteration 0:   log likelihood =  -881.1491
Iteration 1:   log likelihood = -764.31729
Iteration 2:   log likelihood = -763.31191
Iteration 3:   log likelihood = -763.31049
Iteration 4:   log likelihood = -763.31049
Ordered probit regression                         Number of obs   =        488
                                                  LR chi2(6)      =     235.68
                                                  Prob > chi2     =     0.0000
Log likelihood = -763.31049                       Pseudo R2       =     0.1337
```

wagecat	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age	.1674519	.0620333	2.70	0.007	.0458689	.289035
c.age#c.age	-.0027983	.0010214	-2.74	0.006	-.0048001	-.0007964
nev_mar	-.0046417	.1126737	-0.04	0.967	-.225478	.2161946
rural	-.5270036	.1100449	-4.79	0.000	-.7426875	-.3113196
school	.2010587	.0201189	9.99	0.000	.1616263	.2404911
tenure	.0989916	.0147887	6.69	0.000	.0700063	.127977
/cut1	2.650637	.8957245			.8950495	4.406225
/cut2	3.941018	.8979167			2.181134	5.700903
/cut3	5.085205	.9056582			3.310148	6.860263
/cut4	5.875534	.9120933			4.087864	7.663204
/cut5	6.468723	.918117			4.669247	8.268199
/cut6	6.922726	.9215455			5.11653	8.728922
/cut7	7.34471	.9237628			5.534168	9.155252
/cut8	7.963441	.9338881			6.133054	9.793828

We can directly compare the log likelihoods for the `intreg` and `oprobit` models because both likelihoods are discrete. If we had point data in our `intreg` estimation, the likelihood would be a mixture of discrete and continuous terms, and we could not compare it directly with the `oprobit` likelihood.

Here the `oprobit` log likelihood is significantly larger (that is, less negative), so it fits better than the `intreg` model. The `intreg` model assumes normality, but the distribution of wages is skewed and definitely nonnormal. Normality is more closely approximated if we model the log of wages.

```
. generate logwage1 = log(wage1)
(14 missing values generated)

. generate logwage2 = log(wage2)
(6 missing values generated)

. intreg logwage1 logwage2 age c.age#c.age nev_mar rural school tenure

Fitting constant-only model:

Iteration 0:   log likelihood = -889.23647
Iteration 1:   log likelihood = -889.06346
Iteration 2:   log likelihood = -889.06346

Fitting full model:

Iteration 0:   log likelihood = -773.81968
Iteration 1:   log likelihood = -773.36566
Iteration 2:   log likelihood = -773.36563

Interval regression                             Number of obs     =        488
                                                LR chi2(6)        =     231.40
Log likelihood = -773.36563                     Prob > chi2       =     0.0000
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age	.0645589	.0249954	2.58	0.010	.0155689	.1135489
c.age#c.age	-.0010812	.0004115	-2.63	0.009	-.0018878	-.0002746
nev_mar	-.0058151	.0454867	-0.13	0.898	-.0949674	.0833371
rural	-.2098361	.0439454	-4.77	0.000	-.2959675	-.1237047
school	.0804832	.0076783	10.48	0.000	.0654341	.0955323
tenure	.0397144	.0058001	6.85	0.000	.0283464	.0510825
_cons	.7084023	.3593193	1.97	0.049	.0041495	1.412655
/lnsigma	-.906989	.0356265	-25.46	0.000	-.9768157	-.8371623
sigma	.4037381	.0143838			.3765081	.4329373

```
  Observation summary:         14  left-censored observations
                                0     uncensored observations
                                6 right-censored observations
                              468       interval observations
```

The log likelihood of this `intreg` model is close to the `oprobit` log likelihood, and the z statistics for both models are similar. ◁

❑ Technical note

`intreg` has two parameterizations for the log-likelihood function: the transformed parameterization $(\beta/\sigma,\ 1/\sigma)$ and the untransformed parameterization $(\beta,\ \ln(\sigma))$. By default, the log likelihood for `intreg` is parameterized in the transformed parameter space. This parameterization tends to be more convergent, but it requires that any starting values and constraints have the same parameterization, and it prevents the estimation with multiplicative heteroskedasticity. Therefore, when the `het()` option is specified, `intreg` switches to the untransformed log likelihood for the fit of the conditional-variance model. Similarly, specifying `from()` or `constraints()` causes the optimization in the untransformed parameter space to allow constraints on (and starting values for) the coefficients on the covariates without reference to σ.

The estimation results are all stored in the $(\beta,\ \ln(\sigma))$ metric. ❑

Stored results

intreg stores the following in e():

Scalars	
e(N)	number of observations
e(N_unc)	number of uncensored observations
e(N_lc)	number of left-censored observations
e(N_rc)	number of right-censored observations
e(N_int)	number of interval observations
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(ll)	log likelihood
e(ll_0)	log likelihood, constant-only model
e(ll_c)	log likelihood, comparison model
e(N_clust)	number of clusters
e(chi2)	χ^2
e(p)	p-value for model χ^2 test
e(sigma)	sigma
e(se_sigma)	standard error of sigma
e(rank)	rank of e(V)
e(rank0)	rank of e(V) for constant-only model
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise
Macros	
e(cmd)	intreg
e(cmdline)	command as typed
e(depvar)	names of dependent variables
e(wtype)	weight type
e(wexp)	weight expression
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(offset)	linear offset variable
e(chi2type)	Wald or LR; type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. Err.
e(het)	heteroskedasticity, if het() specified
e(ml_score)	program used to implement scores
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization
e(ml_method)	type of ml method
e(user)	name of likelihood-evaluator program
e(technique)	maximization technique
e(properties)	b V
e(predict)	program used to implement predict
e(footnote)	program and arguments to display footnote
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved
Matrices	
e(b)	coefficient vector
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
Functions	
e(sample)	marks estimation sample

Methods and formulas

See Wooldridge (2013, sec. 17.4) or Davidson and MacKinnon (2004, sec. 11.6) for an introduction to censored and truncated regression models.

The likelihood for `intreg` subsumes that of the `tobit` models.

Let $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ be the model. $\mathbf{y}$ represents continuous outcomes—either observed or not observed. Our model assumes $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2\mathbf{I})$.

For observations $j \in \mathcal{C}$, we observe y_j, that is, point data. Observations $j \in \mathcal{L}$ are left-censored; we know only that the unobserved y_j is less than or equal to $y_{\mathcal{L}j}$, a censoring value that we do know. Similarly, observations $j \in \mathcal{R}$ are right-censored; we know only that the unobserved y_j is greater than or equal to $y_{\mathcal{R}j}$. Observations $j \in \mathcal{I}$ are intervals; we know only that the unobserved y_j is in the interval $[\,y_{1j},\, y_{2j}\,]$.

The log likelihood is

$$\begin{aligned}
\ln L = & -\frac{1}{2}\sum_{j\in\mathcal{C}} w_j \left\{ \left(\frac{y_j - \mathbf{x}\boldsymbol{\beta}}{\sigma} \right)^2 + \log 2\pi\sigma^2 \right\} \\
& + \sum_{j\in\mathcal{L}} w_j \log \Phi \left(\frac{y_{\mathcal{L}j} - \mathbf{x}\boldsymbol{\beta}}{\sigma} \right) \\
& + \sum_{j\in\mathcal{R}} w_j \log \left\{ 1 - \Phi \left(\frac{y_{\mathcal{R}j} - \mathbf{x}\boldsymbol{\beta}}{\sigma} \right) \right\} \\
& + \sum_{j\in\mathcal{I}} w_j \log \left\{ \Phi \left(\frac{y_{2j} - \mathbf{x}\boldsymbol{\beta}}{\sigma} \right) - \Phi \left(\frac{y_{1j} - \mathbf{x}\boldsymbol{\beta}}{\sigma} \right) \right\}
\end{aligned}$$

where $\Phi()$ is the standard cumulative normal and w_j is the weight for the jth observation. If no weights are specified, $w_j = 1$. If `aweights` are specified, $w_j = 1$, and σ is replaced by $\sigma/\sqrt{a_j}$ in the above, where a_j are the `aweights` normalized to sum to N.

Maximization is as described in [R] **maximize**; the estimate reported as `_sigma` is $\widehat{\sigma}$.

See Amemiya (1973) for a generalization of the tobit model to variable, but known, cutoffs.

This command supports 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*.

`intreg` also supports estimation with survey data. For details on VCEs with survey data, see [SVY] **variance estimation**.

References

Amemiya, T. 1973. Regression analysis when the dependent variable is truncated normal. *Econometrica* 41: 997–1016.

Cameron, A. C., and P. K. Trivedi. 2010. *Microeconometrics Using Stata*. Rev. ed. College Station, TX: Stata Press.

Conroy, R. M. 2005. Stings in the tails: Detecting and dealing with censored data. *Stata Journal* 5: 395–404.

Davidson, R., and J. G. MacKinnon. 1993. *Estimation and Inference in Econometrics*. New York: Oxford University Press.

——. 2004. *Econometric Theory and Methods*. New York: Oxford University Press.

Goldberger, A. S. 1983. Abnormal selection bias. In *Studies in Econometrics, Time Series, and Multivariate Statistics*, ed. S. Karlin, T. Amemiya, and L. A. Goodman, 67–84. New York: Academic Press.

Hurd, M. 1979. Estimation in truncated samples when there is heteroscedasticity. *Journal of Econometrics* 11: 247–258.

Long, J. S. 1997. *Regression Models for Categorical and Limited Dependent Variables*. Thousand Oaks, CA: Sage.

Stewart, M. B. 1983. On least squares estimation when the dependent variable is grouped. *Review of Economic Studies* 50: 737–753.

Wooldridge, J. M. 2013. *Introductory Econometrics: A Modern Approach*. 5th ed. Mason, OH: South-Western.

Also see

[R] **intreg postestimation** — Postestimation tools for intreg

[R] **regress** — Linear regression

[R] **tobit** — Tobit regression

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

[XT] **xtintreg** — Random-effects interval-data regression models

[XT] **xttobit** — Random-effects tobit models

[U] 20 Estimation and postestimation commands