mi impute nbreg — Impute using negative binomial regression

Syntax

`mi impute nbreg` *ivar* [*indepvars*] [*if*] [*weight*] [, *impute_options options*]

impute_options	Description
Main	
* `add(#)`	specify number of imputations to add; required when no imputations exist
* `replace`	replace imputed values in existing imputations
`rseed(#)`	specify random-number seed
`double`	store imputed values in double precision; the default is to store them as `float`
`by(`*varlist*[, *byopts*]`)`	impute separately on each group formed by *varlist*
Reporting	
`dots`	display dots as imputations are performed
`noisily`	display intermediate output
`nolegend`	suppress all table legends
Advanced	
`force`	proceed with imputation, even when missing imputed values are encountered
`noupdate`	do not perform `mi update`; see [MI] **noupdate option**

*`add(#)` is required when no imputations exist; `add(#)` or `replace` is required if imputations exist.
`noupdate` does not appear in the dialog box.

options	Description
Main	
`noconstant`	suppress constant term
`dispersion(mean)`	parameterization of dispersion; the default
`dispersion(constant)`	constant dispersion for all observations
`exposure(`$varname_e$`)`	include ln($varname_e$) in model with coefficient constrained to 1
`offset(`$varname_o$`)`	include $varname_o$ in model with coefficient constrained to 1
`conditional(`*if*`)`	perform conditional imputation
`bootstrap`	estimate model parameters using sampling with replacement
Maximization	
maximize_options	control the maximization process; seldom used

You must mi set your data before using mi impute nbreg; see [MI] **mi set**.

You must mi register *ivar* as imputed before using mi impute nbreg; see [MI] **mi set**.

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

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

Menu

Statistics > Multiple imputation

Description

mi impute nbreg fills in missing values of an overdispersed count variable using a negative binomial regression imputation method. You can perform separate imputations on different subsets of the data by specifying the by() option. You can also account for frequency, importance, and sampling weights.

Options

Main

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

add(), replace, rseed(), double, by(); see [MI] **mi impute**.

dispersion(mean | constant); see [R] **nbreg**.

exposure(*varname*$_e$), offset(*varname*$_o$); see [R] **estimation options**.

conditional(*if*) specifies that the imputation variable be imputed conditionally on observations satisfying *exp*; see [U] **11.1.3 if exp**. That is, missing values in a conditional sample, the sample identified by the *exp* expression, are imputed based only on data in that conditional sample. Missing values outside the conditional sample are replaced with a conditional constant, the value of the imputation variable in observations outside the conditional sample. As such, the imputation variable is required to be constant outside the conditional sample. Also, if any conditioning variables (variables involved in the conditional specification if *exp*) contain soft missing values (.), their missing values must be nested within missing values of the imputation variables. See *Conditional imputation* under *Remarks and examples* in [MI] **mi impute**.

bootstrap specifies that posterior estimates of model parameters be obtained using sampling with replacement; that is, posterior estimates are estimated from a bootstrap sample. The default is to sample the estimates from the posterior distribution of model parameters or from the large-sample normal approximation of the posterior distribution. This option is useful when asymptotic normality of parameter estimates is suspect.

Reporting

dots, noisily, nolegend; see [MI] **mi impute**. noisily specifies that the output from the negative binomial regression fit to the observed data be displayed. nolegend suppresses all legends that appear before the imputation table. Such legends include a legend about conditional imputation that appears when the conditional() option is specified and group legends that may appear when the by() option is specified.

Maximization

maximize_options; see [R] **nbreg**. These options are seldom used.

Advanced

`force`; see [MI] **mi impute**.

The following option is available with `mi impute` but is not shown in the dialog box:

`noupdate`; see [MI] **noupdate option**.

Remarks and examples

stata.com

Remarks are presented under the following headings:

See [MI] **mi impute** for a general description and details about options common to all imputation methods, *impute_options*. Also see [MI] **workflow** for general advice on working with `mi`.

Univariate imputation using negative binomial regression

The negative binomial regression imputation method can be used to fill in missing values of an overdispersed count variable (Royston 2009). It is a parametric method that assumes an underlying negative binomial model (see [R] **nbreg**) for the imputed variable (given other predictors). This method is based on the asymptotic approximation of the posterior predictive distribution of the missing data.

Using mi impute nbreg

In [MI] **mi impute poisson**, we considered a version of the heart attack data containing a count variable, `npreg`, which records the number of pregnancies and is the only variable containing missing values. We imputed its missing values using `mi impute poisson`.

A Poisson model assumes that the mean and the variance are the same. In the presence of overdispersion, when the variance exceeds the mean, a negative binomial model is more appropriate. We can fit a negative binomial model for `npreg` to the observed data to see if there is any indication of overdispersion in the data.

```
. use http://www.stata-press.com/data/r13/mheartpois
(Fictional heart attack data; npreg missing)

. nbreg npreg attack smokes age bmi hsgrad if female==1, nolog

Negative binomial regression                      Number of obs   =         35
                                                  LR chi2(5)      =       1.69
Dispersion     = mean                             Prob > chi2     =     0.8903
Log likelihood = -54.638875                       Pseudo R2       =     0.0152

------------------------------------------------------------------------------
       npreg |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      attack |   .0551929   .4214484     0.13   0.896    -.7708309    .8812166
      smokes |   .0521987   .4182004     0.12   0.901    -.7674591    .8718565
         age |  -.0105877   .0174661    -0.61   0.544    -.0448206    .0236452
         bmi |   .0194787   .0489883     0.40   0.691    -.0765367     .115494
      hsgrad |   .5338139   .4972872     1.07   0.283    -.4408511    1.508479
       _cons |  -.0736959   1.551417    -0.05   0.962    -3.114417    2.967025
-------------+----------------------------------------------------------------
    /lnalpha |  -.7956602   .7987311                     -2.361144     .769824
-------------+----------------------------------------------------------------
       alpha |   .4512832   .3604539                      .0943122    2.159386
------------------------------------------------------------------------------
Likelihood-ratio test of alpha=0:  chibar2(01) =     3.00 Prob>=chibar2 = 0.042
```

The estimate of the overdispersion parameter `alpha` is 0.45 with a 95% confidence interval of [0.094, 2.16]. The confidence interval does not include a value of 0 (no overdispersion), so there is slight overdispersion in the conditional distribution of `nbreg` in the observed data.

We now impute `npreg` using `mi impute nbreg`:

```
. mi set mlong

. mi register imputed npreg
(10 m=0 obs. now marked as incomplete)

. mi impute nbreg npreg attack smokes age bmi hsgrad, add(20)
> conditional(if female==1)

Univariate imputation                       Imputations =       20
Negative binomial regression                      added =       20
Imputed: m=1 through m=20                       updated =        0

Dispersion: mean

Conditional imputation:
  npreg: incomplete out-of-sample obs. replaced with value 0

------------------------------------------------------------------
                   |               Observations per m
                   |----------------------------------------------
          Variable |   Complete   Incomplete   Imputed |     Total
-------------------+-----------------------------------+----------
             npreg |        144           10        10 |       154
------------------------------------------------------------------
(complete + incomplete = total; imputed is the minimum across m
 of the number of filled-in observations.)
```

We specify the `conditional()` option to restrict imputation of `npreg` only to females; see *Conditional imputation* in [MI] **mi impute** for details.

We can analyze these multiply imputed data using logistic regression with `mi estimate`:

```
. mi estimate: logit attack smokes age bmi female hsgrad npreg
  (output omitted)
```

Stored results

`mi impute nbreg` stores the following in `r()`:

Scalars

`r(M)`	total number of imputations
`r(M_add)`	number of added imputations
`r(M_update)`	number of updated imputations
`r(k_ivars)`	number of imputed variables (always 1)
`r(N_g)`	number of imputed groups (1 if `by()` is not specified)

Macros

`r(method)`	name of imputation method (`nbreg`)
`r(ivars)`	names of imputation variables
`r(rseed)`	random-number seed
`r(by)`	names of variables specified within `by()`

Matrices

`r(N)`	number of observations in imputation sample in each group
`r(N_complete)`	number of complete observations in imputation sample in each group
`r(N_incomplete)`	number of incomplete observations in imputation sample in each group
`r(N_imputed)`	number of imputed observations in imputation sample in each group

Methods and formulas

Consider a univariate variable $\mathbf{x} = (x_1, x_2, \ldots, x_n)'$ that follows a negative binomial model

$$\Pr(x_i = x|\mathbf{z}_i) = \frac{\Gamma(m_i + x)}{\Gamma(x+1)\Gamma(m_i)}\, p_i^{m_i}(1-p_i)^x,\ x = 0, 1, 2, \ldots \tag{1}$$

where $m_i = m = 1/\alpha$, $p_i = 1/(1+\alpha\mu_i)$ under mean-dispersion model and $m_i = \mu_i/\delta$, $p_i = p = 1/(1+\delta)$ under constant-dispersion model, $\mu_i = \exp(\mathbf{z}_i'\boldsymbol{\beta} + \text{offset}_i)$, and $\alpha > 0$ and $\delta > 0$ are unknown dispersion parameters; see [R] **nbreg** for details. $\mathbf{z}_i = (z_{i1}, z_{i2}, \ldots, z_{iq})'$ records values of predictors of $\mathbf{x}$ for observation i and $\boldsymbol{\beta}$ is the $q \times 1$ vector of unknown regression coefficients. (When a constant is included in the model—the default—$z_{i1} = 1$, $i = 1, \ldots, n$.)

$\mathbf{x}$ contains missing values that are to be filled in. Consider the partition of $\mathbf{x} = (\mathbf{x}_o', \mathbf{x}_m')$ into $n_0 \times 1$ and $n_1 \times 1$ vectors containing the complete and the incomplete observations. Consider a similar partition of $\mathbf{Z} = (\mathbf{Z}_o, \mathbf{Z}_m)$ into $n_0 \times q$ and $n_1 \times q$ submatrices.

`mi impute nbreg` follows the steps below to fill in $\mathbf{x}_m$:

1. Fit a negative binomial regression model (1) to the observed data $(\mathbf{x}_o, \mathbf{Z}_o)$ to obtain the maximum likelihood estimates, $\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{\beta}}', \ln\widehat{\alpha})'$ under a mean-dispersion model or $\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{\beta}}', \ln\widehat{\delta})'$ under a constant-dispersion model, and their asymptotic sampling variance, $\widehat{\mathbf{U}}$.
2. Simulate new parameters, $\boldsymbol{\theta}_\star$, from the large-sample normal approximation, $N(\widehat{\boldsymbol{\theta}}, \widehat{\mathbf{U}})$, to its posterior distribution, assuming the noninformative prior $\Pr(\boldsymbol{\theta}) \propto \text{const}$.
3. Obtain one set of imputed values, $\mathbf{x}_m^1$, by simulating from a negative binomial distribution (1) with parameters set to their simulated values from step 2.
4. Repeat steps 2 and 3 to obtain M sets of imputed values, $\mathbf{x}_m^1, \mathbf{x}_m^2, \ldots, \mathbf{x}_m^M$.

Steps 2 and 3 above correspond to only approximate draws from the posterior predictive distribution of the missing data, $\Pr(\mathbf{x}_m|\mathbf{x}_o, \mathbf{Z}_o)$, because $\boldsymbol{\theta}_\star$ is drawn from the asymptotic approximation to its posterior distribution.

If weights are specified, a weighted negative binomial regression model is fit to the observed data in step 1 (see [R] **nbreg** for details).

Reference

Royston, P. 2009. Multiple imputation of missing values: Further update of ice, with an emphasis on categorical variables. *Stata Journal* 9: 466–477.

Also see

[MI] **mi impute** — Impute missing values

[MI] **mi impute poisson** — Impute using Poisson regression

[MI] **mi estimate** — Estimation using multiple imputations

[MI] **intro** — Introduction to mi

[MI] **intro substantive** — Introduction to multiple-imputation analysis