asmixlogit — Alternative-specific mixed logit regression

Description

`asmixlogit` fits an alternative-specific mixed logit model, also known as a mixed multinomial logit model or random-parameter logit model, that uses random coefficients to model the correlation of choices across alternatives. The random coefficients are on variables that vary across both cases and alternatives known as alternative-specific variables. The correlation of choices across alternatives relaxes the independence of irrelevant alternatives (IIA) property imposed by the conventional multinomial logit model fit by `mlogit` and the alternative-specific conditional logit model fit by `asclogit`.

Quick start

Mixed logit regression of `y` on `x1`, where the coefficients on `x1` are assumed random normal, cases are identified by `idvar`, and alternatives are identified by `altvar`

```
asmixlogit y, random(x1) case(idvar) alternatives(altvar)
```

As above, and include case-specific factor covariate `a`

```
asmixlogit y, random(x1) case(idvar) alternatives(altvar)  ///
    casevars(i.a)
```

As above, and add covariate `x2`, whose coefficients are random triangular

```
asmixlogit y, random(x1) random(x2, triangle) case(idvar)  ///
    alternatives(altvar) casevars(i.a)
```

Mixed logit model of `y` on `x1`, `x2`, and `x3`, where the random coefficients for `x2` and `x3` are correlated

```
asmixlogit y x1, random(x2 x3, correlated) case(idvar)     ///
    alternatives(altvar)
```

As above, but omit the alternative-specific constants

```
asmixlogit y x1, random(x2 x3, correlated) case(idvar)     ///
    alternatives(altvar) noconstant
```

Menu

Statistics > Categorical outcomes > Mixed logit model

Syntax

asmixlogit *depvar* [*indepvars*] [*if*] [*in*] [*weight*] , case(*caseid*) [*options*]

depvar equal to 1 identifies the outcome or chosen alternative, whereas a 0 indicates the alternatives that were not chosen. Only one alternative may be chosen for each case.

indepvars specifies the alternative-specific covariates that have fixed coefficients.

options	Description
Model	
* case(*caseid*)	use variable *caseid* to identify cases
* alternatives(*altvar*)	use *altvar* to identify the alternatives available for each case
casevars(*varlist*)	case-specific variables
noconstant	suppress the alternative-specific constant terms
random(*varlist* [, *distribution*])	specify variables that are to have random coefficients and the coefficients' distribution
constraints(*constraints*)	apply specified linear constraints
collinear	keep collinear variables
Model 2	
corrmetric(*metric*)	correlation metric for correlated random coefficients
basealternative(# | *lbl* | *str*)	alternative used for normalizing location
altwise	use alternativewise deletion instead of casewise deletion
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 columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Integration	
intmethod(*seqspec*)	specify point set for Monte Carlo integration
intpoints(#)	specify number of points in each sequence
intburn(#)	specify starting index in the Hammersley or Halton sequence
intseed(#)	specify random-number seed for pseudorandom sequence
favor(speed | space)	favor speed or space when generating integration points
Maximization	
maximize_options	control the maximization process
coeflegend	display legend instead of statistics

metric	Description
`correlation`	standard deviation and correlation; the default
`covariance`	variance and covariance
`cholesky`	Cholesky factor

distribution	Description
`normal`	Gaussian-distributed random coefficients; the default
`correlated`	correlated Gaussian-distributed random coefficients
`lnormal`	lognormal distributed random coefficients
`tnormal`	truncated normal distributed random coefficients
`uniform`	uniform distributed random coefficients
`triangle`	triangular distributed random coefficients

seqspec is

seqtype [, `antithetics` | `mantithetics`]

seqtype	Description
`hammersley`	Hammersley point set; the default
`halton`	Halton point set
`random`	uniform pseudorandom point set

* `case(`*casevar*`)` is required. `alternatives(`*altvar*`)` is required to estimate alternative-specific constants or if case-specific variables are specified.

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

`bootstrap`, `by`, `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**.

`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.

Options

Model

`case(`*caseid*`)` specifies the variable that identifies each case. This variable identifies the individuals or entities making a choice. `case()` is required.

`alternatives(`*altvar*`)` specifies the variable that identifies the alternatives for each case. The number of alternatives can vary with each case. `alternatives()` is required to estimate alternative-specific constants or if case-specific variables are specified in `casevars()`.

`casevars(`*varlist*`)` specifies the case-specific variables that are constant for each `case()`. If there are a maximum of A alternatives, there will be $A-1$ sets of coefficients associated with `casevars()`.

`noconstant` suppresses the $A-1$ alternative-specific constant terms.

random(*varlist* [, *distribution*]) specifies the alternative-specific variables that are to have random coefficients and optionally the assumed distribution of the random coefficients. The default distribution is normal, meaning Gaussian-distributed random coefficients. *distribution* may also be correlated, lnormal, tnormal, uniform, or triangle. random() may be specified more than once to specify different sets of variables that correspond to different coefficient distributions.

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

Model 2

corrmetric(*metric*) specifies the estimation metric for correlated random coefficients. corrmetric(correlation), the default, estimates the standard deviations and correlations of the random coefficients. corrmetric(covariance) estimates variances and covariances, and corrmetric(cholesky) estimates Cholesky factors. corrmetric() is allowed only when random(*varlist*, correlated) is specified.

basealternative(*#* | *lbl* | *str*) specifies the alternative used to normalize the latent-variable location (also referred to as the level of utility). The base alternative may be specified as a number, label, or string. The default is the most frequent alternative. This option is ignored if neither alternative-specific constants nor case-specific variables are specified.

altwise specifies that alternativewise deletion be used when marking out observations due to missing values in your variables. The default is to use casewise deletion; that is, the entire group of observations making up a case is deleted if any missing values are encountered. This option does not apply to observations that are marked out by the if or in qualifier or the by prefix.

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***.

Specifying vce(robust) is equivalent to specifying vce(cluster *caseid*).

If specifying vce(bootstrap) or vce(jackknife), you must also specify basealternative().

Reporting

level(*#*), nocnsreport; see [R] **estimation options**.

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

Integration

intmethod(*seqtype* [, antithetics | mantithetics]) specifies the method of generating the point sets used in the Monte Carlo integration. intmethod(hammersley), the default, uses the Hammersley sequence; intmethod(halton) uses the Halton sequence; and intmethod(random) uses a sequence of uniform random numbers.

antithetics and mantithetics specify that a unidimensional antithetic sequence or a multidimensional antithetic sequence, respectively, be generated instead of the standard implementation of the requested *seqtype*. These methods improve the accuracy of the Monte Carlo integration at the cost of additional computation time; see *Methods and formulas*.

`intpoints(#)` specifies the number of raw points to use in the Monte Carlo integration. The default number of points is a function of model complexity and integration method. If `intmethod(hammersley)` or `intmethod(halton)` is used and there are r uncorrelated random coefficients in the model, the default is $50 \times \text{floor}(\sqrt{r})$. If there are also correlated random coefficients in the model and c is the number of correlation parameters, another $50 \times \text{floor}(\sqrt{c})$ points are added. If `intmethod(random)` is used, the number of points is the above times 5. Larger values of `intpoints()` provide better approximations of the log likelihood at the cost of additional computation time.

`intburn(#)` specifies where in the Hammersley or Halton sequence to start, which helps reduce the correlation between the sequences of each dimension. The default is to discard the first n initial elements from each sequence, where n is the largest prime used to generate the sequences. This option may not be specified with `intmethod(random)`.

`intseed(#)` specifies the seed to use for generating uniform pseudorandom sequences. This option may be specified only with `intmethod(random)`. # must be an integer between 0 and $2^{31} - 1$. The default is to use the current seed value from Stata's uniform random-number generator; see [R] **set seed**.

`favor(speed | space)` instructs `asmixlogit` to favor either speed or space when generating the integration points. `favor(speed)` is the default. When favoring speed, the integration points are generated once and stored in memory, thus increasing the speed of evaluating the likelihood. This speed increase can be seen when there are many cases or when the user specifies a large number of integration points in `intpoints(#)`. When favoring space, the integration points are generated repeatedly with each likelihood evaluation.

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**.

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

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

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

Remarks and examples stata.com

`asmixlogit` fits an alternative-specific mixed logit model, in the following simply referred to as mixed logit model. The mixed logit model is most frequently used to model the probability that an individual chooses one of several unordered alternatives. It is also known as the mixed multinomial logit model (McFadden and Train 2000), the random-parameters logit model (Cameron and Trivedi 2005), the logit kernel model (Ben-Akiva, Bolduc, and Walker 2001), or the hybrid logit model (Ben-Akiva et al. 1997).

The mixed logit model is often used in the context of discrete choice models. These models represent how decision-makers choose among a countable set of alternatives. The decision-maker, called a case, is often an individual. The mixed logit model can incorporate attributes that vary across individuals, known as case-specific variables. Income, educational attainment, and age are examples of case-specific variables.

It can also incorporate observed attributes that vary by alternative or by alternative and individual, known as alternative-specific variables. The size of the lake at a fishing site is an example of an alternative-specific covariate that varies only by alternative. The travel distance to any given fishing site is an example of an alternative-specific covariate that varies by alternative and individual.

In the mixed logit model, the coefficients on alternative-specific variables may be treated as fixed or random. Specifying random coefficients accounts for correlation of choices across alternatives, thereby relaxing the IIA assumption that is imposed by the multinomial logit models discussed in [R] **mlogit**, [R] **clogit**, and [R] **asclogit**. In this sense, the mixed logit model fit by `asmixlogit` is more flexible than the models fit by `mlogit`, `clogit`, and `asclogit`.

McFadden and Train (2000) show that the mixed logit model can approximate a wide class of choice representations. Although the mixed logit model was derived under a utility framework and is most often used for these applications, it also can be applied in contexts that lack this individual-choice motivation, for example, classification problems. For an introduction to mixed logit models, see Cameron and Trivedi (2005) and Train (2009). See Hole (2007) for a previous implementation of mixed logit models via the community-contributed `mixlogit` command. You can also use this command to fit mixed logit models to panel data.

In discrete choice, an individual chooses the alternative that yields the highest value of an unobserved ranking index known as utility. Utility is a latent variable that is a function of observed attributes of the individuals, the alternatives, random coefficients, and a random component. In other contexts, such as classification analysis, utility is just an unobserved random index. We call it utility here because this model is most frequently applied to discrete choice data.

For the mixed logit model, the utility that individual i receives from alternative a, denoted by U_{ia}, is

$$U_{ia} = \mathbf{x}_{ia}\boldsymbol{\beta}_i + \mathbf{w}_{ia}\boldsymbol{\alpha} + \mathbf{z}_i\boldsymbol{\delta}_a + \epsilon_{ia} \qquad a = 1, \dots, A$$

$\boldsymbol{\beta}_i$ are random coefficients that vary over individuals in the population, and $\mathbf{x}_{ia}$ is a vector of alternative-specific variables. $\boldsymbol{\alpha}$ are fixed coefficients on $\mathbf{w}_{ia}$, a vector of alternative-specific variables. $\boldsymbol{\delta}_a$ are fixed, alternative-specific coefficients on $\mathbf{z}_i$, a vector of case-specific variables. ϵ_{ia} is a random term that follows a type I extreme value distribution.

`asmixlogit` estimates the fixed coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\delta}_a$ and the parameters of $f(\boldsymbol{\beta})$, the distribution of the random coefficients. The mixed logit model can only estimate the parameters of $f(\boldsymbol{\beta})$, not the $\boldsymbol{\beta}_i$ per se. For example, if the random coefficients $\boldsymbol{\beta}_i$ follow a normal distribution, $\boldsymbol{\beta}_i \sim N(\mu, \boldsymbol{\Sigma})$, then the mixed logit model estimates μ and $\boldsymbol{\Sigma}$.

Note that only the rank order of the utilities for each alternative matters; that is, the location and scale of utility are irrelevant. The data only reveal the chosen alternative, so we must normalize for location by taking differences with respect to a base alternative k. The assumed type I extreme value distribution implies that the difference in the errors for alternative a and base k, $\epsilon_{ia} - \epsilon_{ik}$, follows a logistic distribution. Assuming a standard logistic distribution normalizes for scale.

The choice probabilities are the standard logistic probabilities integrated over the density $f(\boldsymbol{\beta})$. That is, the probability of choosing alternative a for individual i is

$$P_{ia} = \int P_{ia}(\boldsymbol{\beta}) f(\boldsymbol{\beta}) d\boldsymbol{\beta} \tag{1}$$

where $P_{ia}(\boldsymbol{\beta}) = e^{\mathbf{x}_{ia}\boldsymbol{\beta}_i + \mathbf{w}_{ia}\boldsymbol{\alpha} + \mathbf{z}_i\boldsymbol{\delta}_a} / \sum_{a=1}^{A} e^{\mathbf{x}_{ia}\boldsymbol{\beta}_i + \mathbf{w}_{ia}\boldsymbol{\alpha} + \mathbf{z}_i\boldsymbol{\delta}_a}$ are the logistic probabilities, evaluated at parameters $\boldsymbol{\beta}$.

The integral in (1) must be approximated because it has no closed-form solution. `asmixlogit` approximates (1) by simulation and estimates the model parameters by maximum simulated likelihood (MSL). Consistency of the MSL estimator requires that the number of random draws in the simulation method be sufficiently large. More draws will produce more precise estimates by reducing approximation error, at the cost of increased computation time. See *Methods and formulas* for further details, and see Cameron and Trivedi (2005) for an introduction to MSL estimation.

▷ Example 1: Mixed logit model with fixed and random parameters

inschoice.dta records information about available insurance plans and the selected plan for 250 individuals. Each individual selected an insurance plan from the five alternatives that are recorded in the insurance variable. The binary variable choice records the chosen alternative; choice is 1 for the chosen alternative and 0 otherwise. For each individual, we have one observation for each alternative. Here are the data for the first two individuals:

```
. use http://www.stata-press.com/data/r15/inschoice
(Fictional health insurance data)

. list in 1/10, sepby(id) abbreviate(10)

     +--------------------------------------------------------------------+
     | id   premium   deductible   income    insurance   choice |
     |--------------------------------------------------------------------|
  1. |  1      2.87         1.70     5.74       Health        1 |
  2. |  1      3.13         2.14     5.74        HCorp        0 |
  3. |  1      2.03         2.26     5.74      SickInc        0 |
  4. |  1      1.65         2.94     5.74       MGroup        0 |
  5. |  1      0.87         3.56     5.74   MoonHealth        0 |
     |--------------------------------------------------------------------|
  6. |  2      3.52         1.24     2.89       Health        0 |
  7. |  2      3.23         1.52     2.89        HCorp        0 |
  8. |  2      2.81         2.31     2.89      SickInc        0 |
  9. |  2      1.04         2.58     2.89       MGroup        1 |
 10. |  2      0.93         3.17     2.89   MoonHealth        0 |
     +--------------------------------------------------------------------+
```

Insurance premiums (premium) and deductibles (deductible) vary over alternatives and are thus alternative-specific variables. In this example, they also vary over individuals. Income (income) varies only over individuals and is thus a case-specific variable.

We wish to estimate the effect of health insurance premiums, insurance deductibles, and personal income on the choice of health insurance plans. We assume that preferences with respect to deductibles vary over individuals in the population but that preferences with respect to premiums are constant over individuals in the population.

We fit a model for this outcome by using asmixlogit. We specify random(deductible) to include random coefficients on deductible, and we include premium as *indepvar* to include a fixed coefficient on premium. We specify id in the required case() option because it is the variable that identifies the cases, in this case, individuals. The alternatives(insurance) option specifies that insurance identifies the insurance plans available and that alternative-specific constants are to be estimated for these plans.

```
. asmixlogit choice premium, case(id) alternatives(insurance) random(deductible)
Fitting fixed parameter model:
Fitting full model:
Iteration 0:   log simulated likelihood = -296.14935  (not concave)
Iteration 1:   log simulated likelihood = -295.69689
Iteration 2:   log simulated likelihood = -295.03152
Iteration 3:   log simulated likelihood =   -295.029
Iteration 4:   log simulated likelihood =   -295.029
Alternative-specific mixed logit             Number of obs      =      1,250
Case variable: id                            Number of cases    =        250
Alternative variable: insurance              Alts per case: min =          5
                                                            avg =        5.0
                                                            max =          5
Integration sequence:      Hammersley
Integration points:                50           Wald chi2(2)    =      99.66
Log simulated likelihood =   -295.029           Prob > chi2     =     0.0000

              choice        Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]

insurance
     premium          -2.67349    .2692025    -9.93   0.000    -3.201118   -2.145863
  deductible          -1.11102    .3376753    -3.29   0.001    -1.772852    -.449189

Normal
sd(deductible)        .8978139    .3605058                      .4086929    1.972312

Health
       _cons          .5203306    .2979809     1.75   0.081    -.0637012    1.104363

HCorp               (base alternative)

SickInc
       _cons         -.8428823    .2910199    -2.90   0.004    -1.413271   -.2724937

MGroup
       _cons         -2.108393    .4435735    -4.75   0.000    -2.977781   -1.239005

MoonHealth
       _cons         -3.363821    .6785549    -4.96   0.000    -4.693764   -2.033877

LR test vs. fixed parameters: chibar2(01) =     3.02  Prob >= chibar2 = 0.0411
```

The estimated fixed coefficient on `premium` is -2.67, so an increase in a plan's premium reduces the probability that it is chosen. The estimated mean of the normally distributed coefficients on `deductible` is -1.11. The estimated standard deviation of these random coefficients is 0.90, indicating heterogeneity across individuals in the population with respect to the effect of a plan's deductible.

The likelihood-ratio (LR) test in the footer shows the result of a test against a model with only fixed parameters and indicates that we can reject the null hypothesis that the coefficients on `deductible` are fixed.

◁

❑ Technical note

The LR `test vs. fixed parameters` is a test of `sd(deductible)` $= 0$. This is a boundary test and thus requires careful consideration concerning the calculation of its p-value. In particular, the null distribution of the LR test statistic is not the usual χ^2_1 but rather is a 50:50 mixture of a χ^2_0 (point mass at 0) and a χ^2_1, denoted as $\overline{\chi}^2_{01}$. See Gutierrez, Carter, and Drukker (2001) for more details. ❑

▷ Example 2: Correlated random parameters

Continuing with example 1, we now assume that preferences for deductibles also vary, and we estimate the parameters of a model with random coefficients on `premium` and `deductible`. We allow the random coefficients to be correlated, assuming a multivariate normal distribution, by specifying `random(deductible premium, correlated)`.

```
. asmixlogit choice, case(id) alternatives(insurance)
> random(deductible premium, correlated)
Fitting fixed parameter model:
Fitting full model:
Iteration 0:   log simulated likelihood = -295.87207  (not concave)
Iteration 1:   log simulated likelihood = -295.76182
Iteration 2:   log simulated likelihood =  -294.4461  (not concave)
Iteration 3:   log simulated likelihood = -294.29192
Iteration 4:   log simulated likelihood = -294.17441
Iteration 5:   log simulated likelihood = -294.07048
Iteration 6:   log simulated likelihood = -294.06834
Iteration 7:   log simulated likelihood = -294.06832
Refining estimates:
Iteration 0:   log simulated likelihood = -294.06832
Iteration 1:   log simulated likelihood = -294.06832
Alternative-specific mixed logit               Number of obs      =      1,250
Case variable: id                              Number of cases    =        250
Alternative variable: insurance                Alts per case: min =          5
                                                              avg =        5.0
                                                              max =          5
Integration sequence:      Hammersley
Integration points:               100             Wald chi2(2)    =      44.89
Log simulated likelihood = -294.06832             Prob > chi2     =     0.0000

------------------------------------------------------------------------------
         choice |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
----------------+-------------------------------------------------------------
insurance       |
     deductible |  -1.287217   .4588254    -2.81   0.005    -2.186499   -.3879361
        premium |  -3.005814   .4533996    -6.63   0.000    -3.894461   -2.117167
----------------+-------------------------------------------------------------
Normal          |
sd(deductible)  |   1.187556   .7181368                       .363009    3.885001
corr(deduct~e,  |
       premium) |   .4372473    .729373     0.60   0.549    -.8613711    .9774217
   sd(premium)  |    .980627   .6046964                      .2928308    3.283908
----------------+-------------------------------------------------------------
Health          |
          _cons |   .4812461   .3267912     1.47   0.141    -.1592529    1.121745
----------------+-------------------------------------------------------------
HCorp           |  (base alternative)
----------------+-------------------------------------------------------------
SickInc         |
          _cons |  -.8806237   .3072683    -2.87   0.004    -1.482859   -.2783889
----------------+-------------------------------------------------------------
MGroup          |
          _cons |  -2.297077   .5045889    -4.55   0.000    -3.286053   -1.308101
----------------+-------------------------------------------------------------
MoonHealth      |
          _cons |  -3.715646   .8084699    -4.60   0.000    -5.300217   -2.131074
------------------------------------------------------------------------------
LR test vs. fixed parameters: chi2(3) =       4.94        Prob > chi2 = 0.1760
Note: LR test is conservative and provided only for reference.
```

The estimated means of the random coefficients on `deductible` and `premium` are −1.29 and −3.01, respectively. Beneath the estimated means, we see the estimated standard deviations of the random coefficients and their estimated correlation, which are 1.19, 0.98, and 0.44, respectively. The high standard errors on these parameters indicates that they are not precisely estimated.

◁

▷ Example 3: Lognormal random parameters and case-specific variables

In the previous example, we assumed that the coefficients on `premium` are normally distributed. Assuming a normal distribution implies that the random coefficients could be both positive and negative. However, it is typically more plausible to assume that increasing prices do not have positive effects on the probability of choosing a corresponding alternative. In this example, we assume a lognormal distribution for the `premium` coefficients, which allows us to constrain the premium coefficients to be negative.

Because the lognormal distribution is only defined over positive real values, we need to reverse the sign of the `premium` variable to constrain the random coefficients to be negative.

```
. generate negpremium = -1*premium
```

We again assume a normal distribution for random coefficients on `deductible`, and we include `income` as a case-specific variable in the `casevars()` option.

```
. asmixlogit choice, case(id) alternatives(insur) random(deductible)
> random(negpremium, lnormal) casevars(income)

Fitting fixed parameter model:

Fitting full model:

Iteration 0:   log simulated likelihood = -291.16296
Iteration 1:   log simulated likelihood = -289.45565
Iteration 2:   log simulated likelihood = -289.40847
Iteration 3:   log simulated likelihood = -289.40817
Iteration 4:   log simulated likelihood = -289.40817

Alternative-specific mixed logit              Number of obs      =      1,250
Case variable: id                             Number of cases    =        250

Alternative variable: insurance               Alts per case: min =          5
                                                             avg =        5.0
                                                             max =          5

Integration sequence:      Hammersley
Integration points:                50            Wald chi2(6)    =      85.13
Log simulated likelihood = -289.40817            Prob > chi2     =     0.0000

------------------------------------------------------------------------------
      choice |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
insurance    |
   deductible |     -1.142   .3643336    -3.13   0.002    -1.856081    -.427919
   negpremium |   1.058024   .1160747     9.12   0.000     .8305215    1.285526
-------------+----------------------------------------------------------------
Normal       |
sd(deductible)|    .858745    .440354                      .3143247    2.346119
-------------+----------------------------------------------------------------
Lognormal    |
sd(negpremium)|   .2863518   .1568402                      .0978771    .8377583
-------------+----------------------------------------------------------------
Health       |
      income |   .1441049   .1650493     0.87   0.383    -.1793857    .4675955
       _cons |  -.2085157   .8954954    -0.23   0.816    -1.963654    1.546623
-------------+----------------------------------------------------------------
HCorp        |  (base alternative)
-------------+----------------------------------------------------------------
SickInc      |
      income |  -.3055032   .1553252    -1.97   0.049    -.6099349   -.0010714
       _cons |   .5882824   .7980491     0.74   0.461    -.9758651     2.15243
-------------+----------------------------------------------------------------
MGroup       |
      income |   -.337676   .2001726    -1.69   0.092    -.7300071    .0546552
       _cons |  -.7273225   .9995525    -0.73   0.467    -2.686409    1.231764
-------------+----------------------------------------------------------------
MoonHealth   |
      income |  -.4928405   .2461211    -2.00   0.045     -.975229    -.010452
       _cons |  -1.465562   1.247985    -1.17   0.240    -3.911567     .980443
------------------------------------------------------------------------------
LR test vs. fixed parameters: chi2(2) =      3.05       Prob > chi2 = 0.2179

Note: LR test is conservative and provided only for reference.
```

The estimated parameters for `income` (our case-specific variable) show that individuals are more likely to choose insurance plan `Health` over `HCorp` as income increases. The remaining alternatives are less likely to be chosen over `HCorp` as income increases.

Because we assumed a lognormal distribution for the random coefficients on `negpremium`, we have to transform the estimated mean and standard deviation before we can interpret them. The lognormal distribution is parameterized in terms of the underlying normal distribution. The estimates shown above are the mean and standard deviation of the natural logarithm of the premium coefficients. The mean of the coefficients is $e^{\mu+\sigma^2/2}$, and their standard deviation is $\sqrt{e^{2*\mu+\sigma^2}(e^{\sigma^2}-1)}$. We can calculate point estimates and standard errors by using nlcom. To account for the reversed sign of premium, we multiply the mean by -1.

```
. local m _b[insurance:negpremium]

. local s _b[Lognormal:sd(negpremium)]

. nlcom mean: -1*exp(‘m’+‘s’^2/2)

        mean:  -1*exp(_b[insurance:negpremium]+_b[Lognormal:sd(negpremium)]^2/2)

------------------------------------------------------------------------------
      choice |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        mean |   -3.00123   .4058243    -7.40   0.000    -3.796631   -2.205829
------------------------------------------------------------------------------

. nlcom sd: sqrt(exp(2*‘m’+‘s’^2)*(exp(‘s’^2)-1)), nopval

          sd:  sqrt(exp(2*_b[insurance:negpremium]+_b[Lognormal:sd(negpremium)]^
> 2)*(exp(_b[Lognormal:sd(negpremium)]^2-1))

-------------------------------------------------------------
      choice |      Coef.   Std. Err.     [95% Conf. Interval]
-------------+-----------------------------------------------
          sd |   .8773296    .575614    -.2508532    2.005512
-------------------------------------------------------------
```

We estimate a mean of -3.00, which in this case is the same result as in example 2 where we assumed a normal distribution for premium coefficients.

◁

Stored results

asmixlogit stores the following in e():

Scalars

e(N)	number of observations
e(N_case)	number of cases
e(k)	number of parameters
e(k_alt)	number of alternatives
e(k_casevars)	number of case-specific variables
e(k_eq)	number of equations in e(b)
e(k_eq_model)	number of equations in overall model test
e(df_m)	model degrees of freedom
e(ll)	log simulated-likelihood
e(N_clust)	number of clusters
e(const)	constant indicator
e(intpoints)	number of raw integration points
e(lsequence)	length of each integration sequence
e(intburn)	starting sequence index
e(chi2)	χ^2
e(p)	model test p-value
e(ll_c)	log likelihood, comparison model
e(chi2_c)	χ^2, comparison test
e(df_c)	degrees of freedom, comparison test
e(p_c)	p-value for comparison test
e(alt_min)	minimum number of alternatives
e(alt_avg)	average number of alternatives
e(alt_max)	maximum number of alternatives
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise

Macros	
`e(cmd)`	`asmixlogit`
`e(cmdline)`	command as typed
`e(depvar)`	name of dependent variable
`e(casevars)`	case-specific variables
`e(case)`	variable defining cases
`e(altvar)`	variable defining alternatives
`e(alteqs)`	alternative equation names
`e(alt#)`	alternative labels
`e(base)`	base alternative
`e(corrmetric)`	correlation metric for correlated random coefficients
`e(wtype)`	weight type
`e(wexp)`	weight expression
`e(title)`	title in estimation output
`e(clustvar)`	name of cluster variable
`e(chi2type)`	type of χ^2
`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(ml_method)`	type of `ml` method
`e(intmethod)`	technique used to generate sequences
`e(sequence)`	type of sequences
`e(mc_rngstate)`	random-number state used
`e(user)`	name of likelihood-evaluator program
`e(technique)`	maximization technique
`e(properties)`	`b V`
`e(predict)`	program used to implement `predict`
`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

`asmixlogit` estimates the parameters of the mixed logit model by MSL. The probability that case i chooses alternative a, conditional on the random parameter $\boldsymbol{\beta}_i$, is

$$P_{ia}(\boldsymbol{\beta}) = e^{\mathbf{x}_{ia}\boldsymbol{\beta}_i + \mathbf{w}_{ia}\boldsymbol{\alpha} + \mathbf{z}_i\boldsymbol{\delta}_a} \Big/ \sum_{a=1}^{A} e^{\mathbf{x}_{ia}\boldsymbol{\beta}_i + \mathbf{w}_{ia}\boldsymbol{\alpha} + \mathbf{z}_i\boldsymbol{\delta}_a}$$

We get the unconditional choice probability, P_{ia}, by integrating over the mixing distribution $f(\boldsymbol{\beta})$:

$$P_{ia} = \int P_{ia}(\boldsymbol{\beta}) f(\boldsymbol{\beta}) d\boldsymbol{\beta} \tag{2}$$

This integral of dimension d, where d equals the number of random parameters, is approximated by simulation because it has no closed-form solution. The simulated likelihood for the ith case is

$$L_i = \sum_{a=1}^{A} d_{ia} \widehat{P}_{ia}$$

where d_{ia} is an indicator that takes on the value 1 for the chosen alternative and 0 otherwise. The overall log simulated-likelihood is then $\sum_{i=1}^{N} \ln(L_i)$. $\widehat{P}_{ia}$ are the simulated probabilities

$$\widehat{P}_{ia} = 1/M \sum_{m=1}^{M} P_{ia}(\boldsymbol{\beta}^m) \tag{3}$$

where $\boldsymbol{\beta}^m$ are the random parameters drawn from $f(\boldsymbol{\beta})$, and M is the number of random draws. Equation (3) is the computation used to approximate the probabilities in (2).

Computation of $\widehat{P}_{ia}$ is based on integration sequences where each point of the sequence is a draw from density $f(\boldsymbol{\beta})$. The underlying uniform sequences are either pseudorandom draws from the uniform density or deterministic sequences such as a Halton sequence. Using deterministic sequences leads to better coverage of the probability space and lower variance of the simulator, and thus having a smaller approximation error than pseudorandom sequences, given the same number of draws. `asmixlogit` supports pseudorandom, Halton, and Hammersley sequences; see Drukker and Gates (2006) for details.

Using a higher M in (3) will produce a better approximation to the probabilities in (2), at the cost of increased computation time. M is a function of the number of raw integration points q, which may be specified using the `intpoints()` option. In the default method, $M = q$ is the number of draws used in the approximation in (3). In addition to the default method, `asmixlogit` supports methods in which the draws are symmetric around a midpoint, known as unidimensional and multidimensional antithetic draws. These antithetic methods produce a better approximation to the probabilities in (2), at the cost of additional computation time; see Train (2009, sec. 9.3.1). For unidimensional antithetics, $M = 2q$ draws are used. For multidimensional antithetics on a problem with d random coefficients, $M = 2^d q$ draws are used.

Random coefficients with mean μ and scale parameter σ are simulated as follows:

$$\begin{aligned}
\boldsymbol{\beta}^m_{\text{normal}} &= \mu + \sigma\eta_i, \eta_i \sim N(0,1) \\
\boldsymbol{\beta}^m_{\text{lognormal}} &= \exp(\mu + \sigma\eta_i), \eta_i \sim N(0,1) \\
\boldsymbol{\beta}^m_{\text{truncated\_normal}} &= \mu + \sigma\eta_i, \eta_i \sim \text{TN}(0,1,-1.96,1.96) \\
\boldsymbol{\beta}^m_{\text{uniform}} &= \mu + \sigma\eta_i, \eta_i \sim U(-1,1) \\
\boldsymbol{\beta}^m_{\text{triangular}} &= \mu + \sigma\eta_i, \eta_i \sim \triangle(-1,1)
\end{aligned}$$

where $N(\mu,\sigma^2)$ is the normal distribution, $\text{TN}(\mu,\sigma^2,a,b)$ is the truncated normal distribution with lower truncation point a and upper truncation point b, $U(a,b)$ is uniform over $[a,b]$, and $\triangle(a,b)$ is the triangular distribution over $[a,b]$.

Correlated random parameters drawn from the multivariate normal distribution are generated as $\boldsymbol{\beta}^m_{\text{MVN}} = \mathbf{M} + \mathbf{L}\eta_i$, where $\mathbf{M}$ is a vector of means, $\eta_i \sim N(\mathbf{0},\mathbf{I})$, and $\mathbf{L}$ is the Cholesky factor such that the variance–covariance matrix $\mathbf{V} = \mathbf{LL}'$.

This command supports the clustered version of the Huber/White/sandwich estimator of the variance using `vce(robust)` and `vce(cluster` *clustvar*`)`. See [P] **_robust**, particularly *Maximum likelihood estimators* and *Methods and formulas*. Specifying `vce(robust)` is equivalent to specifying `vce(cluster` *caseid*`)`.

References

Ben-Akiva, M., D. Bolduc, and J. Walker. 2001. Specification, identification, and estimation of the logit kernel (or continuous mixed logit) model. Manuscript, University of California, Berkeley. http://eml.berkeley.edu/reprints/misc/multinomial2.pdf.

Ben-Akiva, M., D. L. McFadden, M. Abe, U. Böckenholt, D. Bolduc, D. Gopinath, T. Morikawa, V. Ramaswamy, V. Rao, D. Revelt, and D. Steinberg. 1997. Modeling methods for discrete choice analysis. *Marketing Letters* 8: 273–286.

Brownstone, D., and K. E. Train. 1998. Forecasting new product penetration with flexible substitution patterns. *Journal of Econometrics* 89: 109–129.

Cameron, A. C., and P. K. Trivedi. 2005. *Microeconometrics: Methods and Applications*. New York: Cambridge University Press.

Drukker, D. M., and R. Gates. 2006. Generating Halton sequences using Mata. *Stata Journal* 6: 214–228.

Gutierrez, R. G., S. L. Carter, and D. M. Drukker. 2001. sg160: On boundary-value likelihood-ratio tests. *Stata Technical Bulletin* 60: 15–18. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 269–273. College Station, TX: Stata Press.

Hole, A. R. 2007. Fitting mixed logit models by using maximum simulated likelihood. *Stata Journal* 7: 388–401.

McFadden, D. L., and K. E. Train. 2000. Mixed MNL models for discrete response. *Journal of Applied Econometrics* 15: 447–470.

Train, K. E. 2009. *Discrete Choice Methods with Simulation*. 2nd ed. New York: Cambridge University Press.

Also see

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

[R] **asclogit** — Alternative-specific conditional logit (McFadden's choice) model

[R] **asmprobit** — Alternative-specific multinomial probit regression

[R] **asroprobit** — Alternative-specific rank-ordered probit regression

[R] **mlogit** — Multinomial (polytomous) logistic regression

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

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