The Stata Journal (*yyyy*) *vv*, Number *ii*, pp. 1–13

Estimating mixed logit models using maximum simulated likelihood

Arne Risa Hole
National Primary Care Research and Development Centre
Centre for Health Economics
University of York
ah522@york.ac.uk

Abstract. This paper describes the `mixlogit` Stata command for estimating mixed logit models using maximum simulated likelihood.

Keywords: st0001, mixed logit, maximum simulated likelihood

1 Introduction

In a recent issue of the Stata Journal devoted to maximum simulated likelihood estimation Haan and Uhlendorff (2006) showed an example of how a multinomial logit model with unobserved heterogeneity can be implemented in Stata. This paper describes the `mixlogit` Stata command which can be used to estimate models of the type considered by Haan and Uhlendorff as well as other types of mixed logit models (Train 2003).

The paper is organised as follows. Section 2 gives a brief overview of the mixed logit model, section 3 describes the `mixlogit` syntax and options and section 4 presents some examples.

2 The mixed logit model

Following Revelt and Train (1998) we assume a sample of N respondents with the choice of J alternatives on T choice occasions. The utility that individual n derives from choosing alternative j on choice occasion t is given by $U_{njt} = \beta_n' x_{njt} + \varepsilon_{njt}$ where β_n is a vector of individual-specific coefficients, x_{njt} is a vector of observed attributes relating to individual n and alternative j on choice occasion t and ε_{njt} is a random term which is assumed to be distributed IID extreme value. The density for β is denoted as $f(\beta|\theta)$ where θ are the parameters of the distribution. Conditional on knowing β_n the probability of respondent n choosing alternative i on choice occasion t is given by:

$$L_{nit}(\beta_n) = \frac{\exp(\beta_n' x_{nit})}{\sum_{j=1}^{J} \exp(\beta_n' x_{njt})} \tag{1}$$

which is the conditional logit formula (McFadden 1974). The probability of the observed sequence of choices conditional on knowing β_n is given by:

$$S_n(\beta_n) = \prod_{t=1}^{T} L_{ni(n,t)t}(\beta_n) \tag{2}$$

where $i(n,t)$ denotes the alternative chosen by individual n on choice occasion t. The *unconditional* probability of the observed sequence of choices is the conditional probability integrated over the distribution of β:

$$P_n(\theta) = \int S_n(\beta) f(\beta|\theta) d\beta \tag{3}$$

The unconditional probability is thus a weighted average of a product of logit formulas evaluated at different values of β, with the weights given by the density f.

Note that this specification is quite general as it allows models with both 'individual-specific' and 'alternative-specific' explanatory variables to be estimated. This is analogous to the way that the `clogit` command (see [R] **clogit**) can be used to estimate multinomial logit models as well as McFadden's choice model. In the examples section it is shown how `mixlogit` can be used to estimate various models, including the multinomial logit model with unobserved heterogeneity considered by Haan and Uhlendorff (2006).

The log likelihood for the model is given by $LL(\theta) = \sum_{n=1}^{N} \ln P_n(\theta)$. This expression cannot be solved analytically, and it is therefore approximated using simulation methods (see Train 2003). The simulated log likelihood is given by:

$$SLL(\theta) = \sum_{n=1}^{N} \ln \left[\frac{1}{R} \sum_{r=1}^{R} S_n(\beta^r) \right] \tag{4}$$

where R is the number of replications and β^r is the the r-th draw from $f(\beta|\theta)$.

3 The mixlogit command

3.1 Syntax

`mixlogit` is implemented as a `d0 ml` evaluator. The command allows correlated and uncorrelated normal and log-normal distributions for the coefficients. The pseudo-random draws used in the estimation process are generated using the Mata function `halton()` (Drukker and Gates 2006). The generic syntax for the command is as follows:

`mixlogit` *depvar* [*varlist*] [*if*] [*in*] , `group(`*varname*`)` `rand(`*varlist*`)` [`options`]

The command `mixlpred` can be used following `mixlogit` to obtain predicted probabilities. The predictions are available both in and out of sample; type `mixlpred ... if e(sample) ...` if predictions are wanted for the estimation sample only.

`mixlpred` *newvarname* [*if*] [*in*] , [`options`]

The command `mixlcov` can be used following `mixlogit` to obtain the elements in the coefficient covariance matrix along with their standard errors. This command is only relevant when the coefficients are specified to be correlated, see the `corr` option below. `mixlcov` is a wrapper for `nlcom` (see [R] **nlcom**).

`mixlcov` , [`options`]

3.2 Mixlogit options

`group(`*varname*`)` is required; it specifies a numeric identifier variable for the choice occasions.

`rand(`*varlist*`)` is required; it specifies the independent variables whose coefficients are random. The random coefficients can be specified to be normally or log-normally distributed (see the `ln()` option). Note that the variables immediately following the dependent variable in the syntax are specified to have fixed coefficients.

`id(`*varname*`)` specifies a numeric identifier variable for the decision-makers. This option needs only be specified when each individual performs several choices, i.e. the data is a panel.

`ln(#)` specifies that the last # variables in `rand()` have log-normal rather than normally distributed coefficients; default is `ln(0)`.

`corr` specifies that the random coefficients are correlated; the default is that they are independent. When the corr option is specified the estimated parameters are the means of the (fixed and random) coefficients plus the elements of the lower-triangular matrix L, where the covariance matrix for the random coefficients is given by V = LL'. The estimated parameters are reported in the following order: the means of the fixed coefficients, the means of the random coefficients and the elements of the L matrix. Note that the `mixlcov` command can be used post-estimation to obtain the elements in the V matrix along with their standard errors.

If the `corr` option is *not* specified the estimated parameters are the means of the fixed coefficients and the means and standard deviations of the random coefficients, reported in that order. The sign of the estimated standard deviations is irrelevant. While it may happen in practice that the estimates are negative they should be interpreted as being positive.

The sequence of the parameters is important to bear in mind when specifying starting values.

`nrep(#)` specifies the number of Halton draws used for the simulation; default is `nrep(50)`.

`burn(#)` specifies the number of initial sequence elements to drop when creating the Halton sequences. The default is `burn(15)`. Specification of this option helps reduce the correlation between the sequences in each dimension. Train (2003, p230) recommends that # should be at least as large as the prime number used to generate the sequences. If there are K random coefficients, mixlogit uses the first K primes to generate the Halton draws.

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

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

maximize_options; technique(*algorithm_spec*), iterate(#), trace, gradient, showstep, hessian, tolerance(#), ltolerance(#), gtolerance(#), nrtolerance(#), from(*init_specs*), difficult; see [R] **maximize**. Note that technique(*bhhh)* is not allowed.

3.3 Mixlpred options

nrep(#) specifies the number of Halton draws used for the simulation; default is nrep(50).

burn(#) specifies the number of initial sequence elements to drop when creating the Halton sequences. The default is burn(15).

3.4 Mixlcov options

sd report the standard deviations of the correlated coefficients instead of the covariance matrix.

4 Examples

To illustrate how the mixlogit command can be used to estimate mixed logit models with alternative-specific explanatory variables we use a subset of the data from Huber and Train (2001) on households' choice of electricity supplier[1]. A sample of residential electricity customers were presented with a series of experiments in which they were presented with four alternative electricity suppliers. The suppliers differed in terms of the following characteristics: price per kWh, lenght of contract, whether the company is local and whether it is 'well-known'. Depending on the experiment the price is either fixed or a variable rate that depends on the time of day (TOD) or the season. The following explanatory variables enter the model:

- Price in cents per kWh if fixed price, 0 if TOD or seasonal rates
- Contract length in years
- Whether a local company (0-1 dummy)
- Whether a well-known company (0-1 dummy)

1. The complete data set can be downloaded from Kenneth Train's website as part of his excellent distant-learning course on discrete choice methods (http://elsa.berkeley.edu/~train/).

- TOD rates (0-1 dummy)
- Seasonal rates (0-1 dummy)

The data setup for `mixlogit` is identical to that required by `clogit` when used to estimate McFadden's choice model. To get an impression of how the data are structured the first 12 observations are listed below. Each observation corresponds to an alternative and the dependent variable `y` is 1 for the chosen alternative in each choice situation and 0 otherwise. `gid` identifies the alternatives in a choice situation, `pid` identifies the choice situations faced by a given individual and the remaining variables are the alternative attributes described earlier. In the listed data there are three choice situations faced by the same individual.

```
. use traindata.dta, clear
. list in 1/12, sepby(gid)
```

	y	price	contract	local	wknown	tod	seasonal	gid	pid
1.	0	7	5	0	1	0	0	1	1
2.	0	9	1	1	0	0	0	1	1
3.	0	0	0	0	0	0	1	1	1
4.	1	0	5	0	1	1	0	1	1
5.	0	7	0	0	1	0	0	2	1
6.	0	9	5	0	1	0	0	2	1
7.	1	0	1	1	0	1	0	2	1
8.	0	0	5	0	0	0	1	2	1
9.	0	9	5	0	0	0	0	3	1
10.	0	7	1	0	1	0	0	3	1
11.	0	0	0	0	1	1	0	3	1
12.	1	0	0	1	0	0	1	3	1

We begin by estimating a model in which the coefficient for price is fixed and the remaining coefficients normally distributed[2]. `mixlogit` uses the coefficients from a conditional logit model estimated using the same data as starting values for the means of the coefficients and sets the starting values for the standard deviations to 0.1. The model is estimated using 50 Halton draws; while the accuracy of the results increases with the number of draws so does the estimation time and the choice of draws therefore represents a tradeoff between the two. One possible strategy is to use a relatively small number of draws (say 50) when doing the specification search and a larger number (say 500) for the final model. Train (2003), Cappellari and Jenkins (2006) and Haan and Uhlendorff (2006) discuss the issue of accuracy in greater detail.

(*Continued on next page*)

2. Note that the estimated models have no alternative-specific constants. This is common practice when the data come from so-called unlabelled choice experiments, where the alternatives have no utility over and above the characteristics attributed to them in the experiment.

```
. global randvars "contract local wknown tod seasonal"
. mixlogit y price, rand($randvars) group(gid) id(pid) nrep(50)
Iteration 0:   log likelihood = -1320.2214  (not concave)
  (output omitted)
Iteration 12:  log likelihood = -1137.7962
Mixed logit model                               Number of obs   =       4780
                                                LR chi2(5)      =     437.18
Log likelihood = -1137.7962                     Prob > chi2     =     0.0000
```

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
Mean						
price	-.871424	.0587205	-14.84	0.000	-.9865141	-.7563339
contract	-.2337225	.0362325	-6.45	0.000	-.304737	-.1627081
local	1.93945	.1736135	11.17	0.000	1.599173	2.279726
wknown	1.480568	.1427073	10.37	0.000	1.200867	1.76027
tod	-8.334531	.5066989	-16.45	0.000	-9.327642	-7.341419
seasonal	-8.449154	.5167855	-16.35	0.000	-9.462035	-7.436273
SD						
contract	.2959921	.0305113	9.70	0.000	.2361911	.3557931
local	1.798181	.2129429	8.44	0.000	1.38082	2.215541
wknown	1.114259	.2248277	4.96	0.000	.6736051	1.554913
tod	1.560564	.1666313	9.37	0.000	1.233973	1.887156
seasonal	1.684006	.1799348	9.36	0.000	1.33134	2.036671

```
. *Save coefficients for later use
. matrix b = e(b)
```

It can be seen from the results that on average consumers prefer lower costs, shorter contract lenght, a local and well-known provider and fixed rather than variable rates. Further, there is significant preference heterogeneity for all the attributes. Looking at the magnitudes of the standard deviations relative to the mean coefficients it can be seen that while practically all consumers prefer fixed to variable rates 21% prefer longer contracts, 14% prefer a provider which is not local and 9% prefer a provider which is not well-known. These figures are given by $100 \times \Phi(-b_k/s_k)$, where Φ is the cumulative standard normal distribution and b_k and s_k is the mean and standard deviation of the kth coefficient.

A likelihood ratio test for the joint significance of the standard deviations is reported in the upper-right corner of the table. The associated p-value is very small, implying that the null hypothesis that all the standard deviations are equal to zero is rejected.

In some cases it may be desirable to restrict the sign of the coefficients to be either positive or negative for all individuals. If this is the case the log-normal distribution provides an alternative to the normal distribution. While specifying a coefficient to be log-normally distributed implies that it is positive for all individuals, negative coefficients can be accomodated by entering the attribute multiplied by minus one in the model. The following example demonstrates by specifying the price coefficient to be log-normally distributed:

```
. gen mprice=-1*price
. global lnrandv "contract local wknown tod seasonal mprice"
. mixlogit y, rand($lnrandv) group(gid) id(pid) ln(1) nrep(50)
Iteration 0:   log likelihood = -1277.6348  (not concave)
  (output omitted)
Iteration 7:   log likelihood = -1130.7054
Mixed logit model                                 Number of obs   =       4780
                                                  LR chi2(6)      =     451.36
Log likelihood = -1130.7054                       Prob > chi2     =     0.0000

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
Mean         |
    contract |  -.2464902    .035744    -6.90   0.000    -.3165472   -.1764332
       local |   2.196089   .2192703    10.02   0.000     1.766327    2.625851
      wknown |    1.47136   .1279782    11.50   0.000     1.220527    1.722193
         tod |  -8.604945   .5067274   -16.98   0.000    -9.598112   -7.611777
    seasonal |  -8.903156    .525998   -16.93   0.000    -9.934093   -7.872219
      mprice |  -.0695899   .0681758    -1.02   0.307    -.2032121    .0640323
-------------+----------------------------------------------------------------
SD           |
    contract |   .2791734   .0294738     9.47   0.000     .2214058     .336941
       local |   1.656502   .2948766     5.62   0.000     1.078555     2.23445
      wknown |   .6732312   .1638918     4.11   0.000     .3520092    .9944533
         tod |   .8999234   .2082437     4.32   0.000     .4917733    1.308074
    seasonal |   1.102238   .2370827     4.65   0.000     .6375644    1.566911
      mprice |   .2367958   .0256924     9.22   0.000     .1864395     .287152
------------------------------------------------------------------------------
```

The estimated price parameters in the above model are the mean (b_p) and standard deviation (s_p) of the natural logarithm of the price coefficient. The median, mean and standard deviation of the coefficient itself is given by $\exp(b_p)$, $\exp(b_p + s_p^2/2)$ and $\exp(b_p + s_p^2/2) \times \sqrt{\exp(s_p^2) - 1}$, respectively (Train 2003). The standard errors of the mean, median and standard deviation of the coefficient can be conveniently calculated using `nlcom`:

```
. nlcom (mean_price: -1*exp([Mean]_b[mprice]+0.5*[SD]_b[mprice]^2)) ///
>        (med_price: -1*exp([Mean]_b[mprice])) ///
>        (sd_price: exp([Mean]_b[mprice]+0.5*[SD]_b[mprice]^2) ///
>                * sqrt(exp([SD]_b[mprice]^2)-1))
  mean_price:  -1*exp([Mean]_b[mprice]+0.5*[SD]_b[mprice]^2)
   med_price:  -1*exp([Mean]_b[mprice])
    sd_price:  exp([Mean]_b[mprice]+0.5*[SD]_b[mprice]^2)                 * sqr
> t(exp([SD]_b[mprice]^2)-1)

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  mean_price |  -.9592978   .0634787   -15.11   0.000    -1.083714   -.8348819
   med_price |  -.9327763   .0635928   -14.67   0.000    -1.057416   -.8081367
    sd_price |   .2303795   .0258277     8.92   0.000     .1797581    .2810008
------------------------------------------------------------------------------
```

Note that the mean and median estimates have been multiplied by minus one to undo the sign change introduced in the estimation process.

The next example demonstrates how `mixlogit` can be used to fit a model with correlated normally distributed coefficients. Here the `from()` option is used to specify the starting values, which are taken from the model with uncorrelated normal coefficients. The final 15 coefficients are the elements of the lower-triangular matrix L, where the covariance matrix for the random coefficients is given by V = LL' (the L matrix is the Cholesky factorisation of the covariance matrix V).

```
. *Starting values
. matrix b = b[1,1..7],0,0,0,0,b[1,8],0,0,0,b[1,9],0,0,b[1,10],0,b[1,11]
. mixlogit y price, rand($randvars) group(gid) id(pid) nrep(50) ///
> corr from(b, copy)
Iteration 0:   log likelihood = -1137.7962  (not concave)
  (output omitted)
Iteration 11:  log likelihood = -1060.8267
Mixed logit model                                 Number of obs   =       4780
                                                  LR chi2(15)     =     591.12
Log likelihood = -1060.8267                       Prob > chi2     =     0.0000
```

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
price	-.8886557	.0604114	-14.71	0.000	-1.00706	-.7702517
contract	-.2283449	.0354989	-6.43	0.000	-.2979216	-.1587683
local	2.526601	.2448636	10.32	0.000	2.046677	3.006524
wknown	1.994449	.1883359	10.59	0.000	1.625318	2.363581
tod	-8.680891	.5628243	-15.42	0.000	-9.784006	-7.577776
seasonal	-8.480597	.5405835	-15.69	0.000	-9.540122	-7.421073
/l11	.3242159	.0327134	9.91	0.000	.2600988	.388333
/l21	.5076903	.1918853	2.65	0.008	.1316021	.8837786
/l31	.5164186	.1574543	3.28	0.001	.2078138	.8250233
/l41	-.5622626	.2119887	-2.65	0.008	-.9777528	-.1467725
/l51	.2008202	.1936121	1.04	0.300	-.1786527	.580293
/l22	2.63833	.2709845	9.74	0.000	2.10721	3.169449
/l32	1.69457	.2366776	7.16	0.000	1.23069	2.158449
/l42	.5041141	.2377614	2.12	0.034	.0381102	.9701179
/l52	.619007	.2024404	3.06	0.002	.2222311	1.015783
/l33	.4146704	.1683534	2.46	0.014	.0847038	.744637
/l43	1.13526	.2551701	4.45	0.000	.6351363	1.635385
/l53	.3854606	.2379867	1.62	0.105	-.0809848	.851906
/l44	2.003162	.2427179	8.25	0.000	1.527444	2.47888
/l54	1.346629	.2146774	6.27	0.000	.9258688	1.767389
/l55	1.57518	.1856906	8.48	0.000	1.211233	1.939127

The joint significance of the off-diagonal elements of the covariance matrix can be tested using a likelihood ratio test. The test statistic, which is Chi-square distributed with 10 degrees of freedom under the null of uncorrelated coefficients, is given by 2×(1137.7962-1060.8267)=153.939 implying that the null hypothesis is rejected.

The covariance matrix and standard deviations of the random coefficients can conveniently be calculated using `mixlcov`:

```
. mixlcov
```
(*output omitted*)

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
v11	.105116	.0212124	4.96	0.000	.0635404	.1466915
v21	.1646013	.066446	2.48	0.013	.0343696	.294833
v31	.1674311	.055532	3.02	0.003	.0585903	.2762719
v41	-.1822945	.0772516	-2.36	0.018	-.3337049	-.0308841
v51	.0651091	.0622507	1.05	0.296	-.0568999	.1871181
v22	7.218532	1.407761	5.13	0.000	4.459371	9.977694
v32	4.733014	1.031262	4.59	0.000	2.711777	6.754251
v42	1.044564	.6297303	1.66	0.097	-.189685	2.278813
v52	1.735099	.5491032	3.16	0.002	.6588763	2.811321
v33	3.310206	.812972	4.07	0.000	1.71681	4.903602
v43	1.034652	.4864579	2.13	0.033	.0812125	1.988092
v53	1.312497	.3707543	3.54	0.000	.5858317	2.039162
v44	5.871744	1.390638	4.22	0.000	3.146144	8.597343
v54	3.33425	.8074523	4.13	0.000	1.751672	4.916827
v55	4.866679	.9491096	5.13	0.000	3.006458	6.7269

```
. mixlcov, sd
```
(*output omitted*)

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
contract	.3242159	.0327134	9.91	0.000	.2600988	.388333
local	2.686733	.2619839	10.26	0.000	2.173254	3.200212
wknown	1.819397	.223418	8.14	0.000	1.381506	2.257288
tod	2.423168	.2869462	8.44	0.000	1.860764	2.985572
seasonal	2.206055	.2151147	10.26	0.000	1.784438	2.627672

To illustrate how the `mixlogit` command can be used to estimate a multinomial logit model with unobserved heterogeneity we use the data from Haan and Uhlendorff (2006) on teachers' ratings of pupils' behaviour. The first step is to rearrange the data so that they are in the form required by `mixlogit`. This is analogous to the example in Long and Freese (2006), section 7.2.4, where it is shown how `clogit` can be used to estimate a multinomial logit model. The first four observations in the dataset are listed below:

(*Continued on next page*)

```
. use jspmix.dta, clear
. list scy3 id tby sex in 1/4
```

	scy3	id	tby	sex
1.	1	280	1	0
2.	1	281	2	1
3.	1	282	1	0
4.	1	283	1	1

The next step is to expand the data. As there are 3 alternatives (low, medium and high performance) we create 3 duplicate records using the `expand 3` command. Then we create a variable `alt` which identifies the alternatives, and this variable is used to generate alternative-specific constants as well as interactions with the gender variable:

```
. expand 3
(2626 observations created)
. bysort id: gen alt = _n
. gen mid = (alt == 2)
. gen low = (alt == 3)
. gen sex_mid = sex*mid
. gen sex_low = sex*low
```

Finally we generate the new dependent variable `choice` which equals 1 if `tby == alt` and 0 otherwise:

```
. gen choice = (tby == alt)
```

The observations corresponding to the first four records in the original dataset are listed below:

(*Continued on next page*)

```
. sort scy3 id alt

. list scy3 id choice mid low sex_mid sex_low in 1/12, sepby(id)

     +--------------------------------------------------------+
     | scy3    id   choice   mid   low   sex_mid   sex_low |
     |--------------------------------------------------------|
  1. |    1   280        1     0     0         0         0 |
  2. |    1   280        0     1     0         0         0 |
  3. |    1   280        0     0     1         0         0 |
     |--------------------------------------------------------|
  4. |    1   281        0     0     0         0         0 |
  5. |    1   281        1     1     0         1         0 |
  6. |    1   281        0     0     1         0         1 |
     |--------------------------------------------------------|
  7. |    1   282        1     0     0         0         0 |
  8. |    1   282        0     1     0         0         0 |
  9. |    1   282        0     0     1         0         0 |
     |--------------------------------------------------------|
 10. |    1   283        1     0     0         0         0 |
 11. |    1   283        0     1     0         1         0 |
 12. |    1   283        0     0     1         0         1 |
     +--------------------------------------------------------+
```

In order to replicate the results from Haan and Uhlendorff (2006) we begin by estimating a model with random but uncorrelated intercepts:

```
. mixlogit choice sex_mid sex_low, group(id) id(scy3) rand(mid low) nrep(50)
Iteration 0:   log likelihood = -1329.3862  (not concave)
  (output omitted)
Iteration 4:   log likelihood = -1315.5573

Mixed logit model                                 Number of obs   =       3939
                                                  LR chi2(2)      =      32.73
Log likelihood = -1315.5573                       Prob > chi2     =     0.0000

------------------------------------------------------------------------------
      choice |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
Mean         |
     sex_mid |   .4797342   .1419879     3.38   0.001     .2014431    .7580253
     sex_low |   1.019558   .1699843     6.00   0.000     .6863945    1.352721
         mid |    .531875   .1143519     4.65   0.000     .3077495    .7560006
         low |  -.6773662   .1503376    -4.51   0.000    -.9720225     -.38271
-------------+----------------------------------------------------------------
SD           |
         mid |    .514834   .1095761     4.70   0.000     .3000688    .7295992
         low |   .5778391   .1126084     5.13   0.000     .3571307    .7985474
------------------------------------------------------------------------------

. matrix b = e(b)
```

The next step is to use the coefficients from the above model as starting values for the final model specification with correlated intercepts:

(*Continued on next page*)

```
. matrix b = b[1,1..5],0,b[1,6]
. mixlogit choice sex_mid sex_low, group(id) id(scy3) rand(mid low) corr nrep(5
> 0) from(b, copy)
Iteration 0:   log likelihood = -1315.5573
  (output omitted)
Iteration 5:   log likelihood = -1300.1117
Mixed logit model                                 Number of obs   =       3939
                                                  LR chi2(3)      =      63.62
Log likelihood = -1300.1117                       Prob > chi2     =     0.0000
```

choice	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
sex_mid	.5494836	.1456751	3.77	0.000	.2639657	.8350014
sex_low	1.101967	.1747535	6.31	0.000	.7594559	1.444477
mid	.6278598	.1425238	4.41	0.000	.3485182	.9072013
low	-.5204487	.1806557	-2.88	0.004	-.8745273	-.16637
/l11	.7321526	.119431	6.13	0.000	.4980721	.9662332
/l21	.809698	.1564731	5.17	0.000	.5030165	1.11638
/l22	-.346577	.1106231	-3.13	0.002	-.5633942	-.1297597

It can be seen that the results are similar to those reported by Haan and Uhlendorff but not identical. The main reason is that the Halton draws are generated differently in the two applications: while Haan and Uhlendorff base their draws on primes 7 and 11 `mixlogit` uses primes 2 and 3 (see Drukker and Gates, 2006, for a description of how Halton draws are generated). Simulation-based estimators will generally produce slightly different results unless the draws are generated in exactly the same way.

As before, the covariance matrix and standard deviations of the random coefficients can conveniently be calculated using `mixlcov`:

```
. mixlcov
  (output omitted)
```

choice	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
v11	.5360475	.1748835	3.07	0.002	.1932821	.8788129
v21	.5928225	.1889485	3.14	0.002	.2224903	.9631547
v22	.7757264	.2540111	3.05	0.002	.2778738	1.273579

```
. mixlcov, sd
  (output omitted)
```

choice	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
mid	.7321526	.119431	6.13	0.000	.4980721	.9662332
low	.8807533	.1442011	6.11	0.000	.5981245	1.163382

5 Acknowledgements

I am very grateful to Kenneth Train for helpful comments and for giving me permission to use his data. Thanks also to Peter Haan for providing me with the data on pupil behaviour, Peter Sivey for helping to test out the code and an anonymous referee for helpful comments and suggestions for how to improve the paper. NPCRDC receives funding from the Department of Health. The views expressed are not necessarily those of the funders.

6 References

Cappellari, L., and S. P. Jenkins. 2006. Calculation of multivariate normal probabilities by simulation, with applications to maximum simulated likelihood estimation. *The Stata Journal* 6(2): 156–189.

Drukker, D. M., and R. Gates. 2006. Generating Halton sequences using Mata. *The Stata Journal* 6(2): 214–228.

Haan, P., and A. Uhlendorff. 2006. Estimation of multinomial logit models with unobserved heterogeneity using maximum simulated likelihood. *The Stata Journal* 6(2): 229–245.

Huber, J., and K. Train. 2001. On the Similarity of Classical and Bayesian Estimates of Individual Mean Partworths. *Marketing Letters* 12: 259–269.

Long, J. S., and J. Freese. 2006. *Regression Models for Categorical Dependent Variables Using Stata*. 2nd ed. College Station, TX: Stata Press.

McFadden, D. 1974. Conditional logit analysis of qualitative choice behavior. In *Frontiers in Econometrics*, ed. P. Zarembka, 105–142. New York: Academic Press.

Revelt, D., and K. Train. 1998. Mixed logit with repeated choices: Households' choices of appliance efficiency level. *Review of Economics and Statistics* 80: 647–657.

Train, K. 2003. *Discrete choice methods with simulation*. Cambridge: Cambridge University Press.

About the author

Arne Risa Hole is a research fellow at the Centre for Health Economics, York, UK.