Using `bayesmixedlogit` and `bayesmixedlogitwtp` in Stata

Matthew J. Baker
Hunter College and the Graduate Center, CUNY

March 2, 2021

1 Introduction

This document presents an overview of the `bayesmixedlogit` and `bayesmixedlogitwtp` `Stata` packages. It mirrors closely the helpfile obtainable in `Stata` (i.e., through `help bayesmixedlogit` or `help bayesmixedlogitwtp`). Further background for the packages can be found in Baker(2014).

1.1 Description

`bayesmixedlogit` can be used to fit mixed logit models using Bayesian methods – more precisely, `bayesmixedlogit` produces draws from the posterior parameter distribution and then presents summary and other statistics describing the results of the drawing. Detailed analysis of the draws is left to the discretion of the user.

Implementation of `bayesmixedlogit` follows Train (2009, chap. 12), and details of how the algorithm works are described in Baker(2014). A diffuse prior for the mean values of the random coefficients is assumed, and the prior distribution on the covariance matrix of random coefficients is taken to be an identity inverse Wishart. `bayesmixedlogit` uses the `Mata` routines `amcmc()` (if not installed: `ssc install amcmc`) for adaptive Markov chain Monte Carlo sampling from the posterior distribution of individual level coefficients and fixed coefficients. The data setup for `bayesmixedlogit` is the same as for `clogit` (`ssc install clogit`). Much of the syntax follows that used by Hole (2007) in development of the command `mixlogit`.

1.2 Options

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

`identifier(`*varname*`)` identifies coefficient sets (those observations for which a set of coefficients apply). Thus, when a person is observed making choices over multiple occasions, one would use `group(`*varname*`)` to specify the choice occasions, while `identifier(`*varname*`)` would identify the person. `identifier()` is required.

`rand(`*varlist*`)` specifies independent variables with random coefficients. The variables immediately following the dependent variable in the syntax are considered to have fixed coefficients (see the examples below). While a model can be run without any independent variables with fixed coefficients, at least one random-coefficient independent variable is required for bayesmixedlogit to work. `rand()` is required.

`draws(#)` specifies the number of draws that are to be taken from the posterior distribution of the parameters. The default is `draws(1000)`.

`drawsrandom`(#) is an advanced option. The drawing algorithm treats each set of random coefficients as a Gibbs step in sampling from the joint posterior distribution of parameters. In difficult, large-dimensional problems, it might be desirable to let individual Gibbs steps run for more than one draw to achieve better mixing and convergence of the algorithm.

`drawsfixed`(#) is a more advanced option. The drawing algorithm treats fixed coefficients as a Gibbs step in sampling from the joint posterior distribution of parameters. In difficult, large-dimensional problems, it might be desirable to let this step in Gibbs sampling run for more than a single draw. The default is `drawsfixed(1)`.

`burn`(#) specifies the length of the burn-in period; the first # draws are discarded upon completion of the algorithm and before further results are computed.

`thin`(#) specifies that only every #th draw is to be retained, so if thin(3) is specified, only every third draw is retained. This option is designed to help ease autocorrelation in the resulting draws, as is the option jumble, which randomly mixes draws. Both options may be applied.

`araterandom`(#) specifies the desired acceptance rate for random coefficients and should be a number between zero and one. Because an adaptive acceptance-rejection method is used to sample random coefficients, by specifying the desired acceptance rate, the user has some control over adaptation of the algorithm to the problem. The default is `araterandom`(.234).

`aratefixed`(#) specifies the desired acceptance rate for fixed coefficients and works in the same fashion as `araterandom(#)`.

`samplerrandom(`*string*`)` specifies the type of sampler that is to be used when random parameters are drawn. It may be set to either global or mwg. The default is `samplerrandom(`*global*`)`, which means that proposed changes to random parameters are drawn all at once. If `mwg` – an acronym for "Metropolis within Gibbs" – is instead chosen, each random parameter is drawn separately as an independent step conditional on other random parameters in a nested Gibbs step. `mwg` might be useful in situations in which initial values are poorly scaled. The workings of these options are described in greater detail in Baker(2014).

`samplerfixed(`*string*`)` specifies the type of sampler that is used when fixed parameters are drawn. Options are exactly as those described under `samplerrandom(`*string*`)`.

`dampparmfixed`(#) works exactly as option **dampparmrandom**(#) but is applied to drawing fixed parameters.

`dampparmrandom`(#) is a parameter that controls how aggressively the proposal distributions for random parameters are adapted as drawing continues. If the parameter is set close to one, adaptation is aggressive in its early phase of trying to achieve the acceptance rate specified in `araterandom`(#). If the parameter is set closer to zero, adaptation is more gradual.

`from(`*rowvector*`)` specifies a row vector of starting values for all parameters in order. If these are not specified, starting values are obtained via estimation of a conditional logit model via `clogit`**.

`fromvariance(`*matrix*`)` specifies a matrix of starting values for the random parameters.

`jumble` specifies to randomly mix draws.

`noisy` specifies that a dot be produced every time a complete pass through the algorithm is finished. After 50 iterations, a function value `ln_fc(p)` will be produced, which gives the joint log of the value of the posterior choice probabilities evaluated at the latest parameters. While `ln_fc(p)` is not an objective function per se, the author has found that drift in the value of this function indicates that the algorithm has not yet converged or has other problems.

`saving(`*filename*`)` specifies a location to store the draws from the distribution. The file will contain just the draws after any burn-in period or thinning of values is applied.

replace specifies that an existing file is to be overwritten.

append specifies that an existing file is to be appended, which might be useful if multiple runs need to be combined.

`indsave(`*filename*`)` specifies a file to which individual-level random parameters are to be saved. More precisely, `indsave(`*filename*`)` saves the draws of the individual-level parameters. Caution: For long runs and models with large numbers of individuals, specifying this option can cause memory problems. Users should be careful how it is used and consult some of the examples before employing the option.

`indkeep`(#) is for use with indsave and specifies that only the last # draws of the individual-level random parameters be kept. This helps avoid excessive memory consumption.

`indwide` is for use with indsave and affords the user a degree of control over how individual-level parameters are saved. By default, individual-level parameters are saved in a panel form, meaning that each random parameter draw is saved in a row, where draws are marked by the group identifier. If instead the user would prefer that each row contain draws of each parameter, one could specify the indwide option, which saves all draws in a single row, with the first entry of the row being the group identifier. By analogy with `reshape`, by default draws are saved in "long" format, whereas `indwide` stores the draws in "wide" format.

`replaceind` functions in the same way as replace, but in reference to the file specified in `indsave`.

`appendind` functions in the same way as `append`, but in reference to the file specified in `indsave`.

1.3 Examples

1.3.1 Example 1

A single random coefficient, one decision per group. The random parameter acceptance rate is set to 0.4, and a total of 4,000 draws are taken. The first 1,000 draws are dropped, and then every fifth draw is retained. Draws are saved as `choice_draws.dta`:

[1]:
```
webuse choice
describe
```

```
Contains data from http://www.stata-press.com/data/r14/choice.dta
  obs:           885
 vars:             7                          2 Dec 2014 13:25
 size:         9,735
-------------------------------------------------------------------------------
              storage   display    value
```

```
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
id              int     %9.0g
sex             byte    %9.0g      sex
income          float   %9.0g                 income in thousands
car             byte    %9.0g      nation     nationality of car
size            byte    %9.0g
choice          byte    %9.0g                 ID's chosen car
dealer          byte    %9.0g                 number of dealers of each
                                                nationality in ID's city
-------------------------------------------------------------------------------
Sorted by: id
```

[2]:

```
bayesmixedlogit choice, rand(dealer) group(id) id(id) ///
     draws(4000) burn(1000) thin(5) arater(.4) saving(choice_draws) replace
```

```
Bayesian Mixed Logit Model                          Observations    =       885
                                                    Groups          =       295
Acceptance rates:                                   Choices         =       295
 Fixed coefs              =                         Total draws     =      4000
 Random coefs(ave,min,max)= 0.239, 0.186, 0.285     Burn-in draws   =      1000
                                                    *One of every 5 draws kept
------------------------------------------------------------------------------
      choice |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
Random       |
      dealer |   .2150072   .0360706     5.96   0.000     .1441673    .2858471
-------------+----------------------------------------------------------------
Cov_Random   |
  var_dealer |   .1024538   .0322976     3.17   0.002     .0390238    .1658839
------------------------------------------------------------------------------
   Draws saved in choice_draws.dta


   Attention!
   *Results are presented to conform with Stata covention, but
    are summary statistics of draws, not coefficient estimates.
```

1.3.2 Example 2

Fitting a mixed logit model using `bayesmixedlogit`, using the methods as described in Long and Freese (2006, sec. 7.2.4). The data must first be rendered into the correct format, which can be done using the command `case2alt`, which is part of the package `spost9_ado`; if not installed, type `net install spost9_ado, from(https://jslsoc.sitehost.iu.edu/stata)` from the Stata prompt. The example first arranges the data and then generates and summarizes posterior draws from a mixed logit model. The model uses `bangladesh.dta`, which has information on contraceptive choice by a series of families. Coefficients of explanatory variables vary at the district level.

[3]:
```
webuse choice
describe
```

```
Contains data from http://www.stata-press.com/data/r14/choice.dta
  obs:           885
 vars:             7                          2 Dec 2014 13:25
 size:         9,735
--------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
--------------------------------------------------------------------------------
id              int     %9.0g
sex             byte    %9.0g      sex
income          float   %9.0g                 income in thousands
car             byte    %9.0g      nation     nationality of car
size            byte    %9.0g
choice          byte    %9.0g                 ID's chosen car
dealer          byte    %9.0g                 number of dealers of each
                                                nationality in ID's city
--------------------------------------------------------------------------------
Sorted by: id
```

[4]:
```
webuse bangladesh, clear
case2alt, casevars(urban age) choice(c_use) gen(choice)
```

```
(Bangladesh Fertility Survey, 1989)

(note: variable _id used since case() not specified)
(note: variable _altnum used since altnum() not specified)

choice indicated by: choice
case identifier: _id
case-specific interactions: no* yes*
```

[5]:
```
bayesmixedlogit choice, rand(yesXurban yesXage yes) group(_id) id(district) ///
     draws(10000) burn(5000) saving(bdesh_draws) replace
```

```
Bayesian Mixed Logit Model                          Observations    =      3868
                                                    Groups          =        60
Acceptance rates:                                   Choices         =      1934
 Fixed coefs              =                         Total draws     =     10000
```

```
 Random coefs(ave,min,max)= 0.310, 0.231, 0.354    Burn-in draws    =      5000
------------------------------------------------------------------------------
       choice |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
--------------+---------------------------------------------------------------
Random        |
    yesXurban |   .7796584   .1905085     4.09   0.000     .4061781    1.153139
      yesXage |   .0067593   .0329065     0.21   0.837    -.0577519    .0712706
          yes |  -.7777077   .1154274    -6.74   0.000    -1.003996   -.5514193
--------------+---------------------------------------------------------------
Cov_Random    |
var_yesXurban |   .9302604   .4088646     2.28   0.023     .1287065    1.731814
cov_yesXurb~e |  -.0010909    .034572    -0.03   0.975    -.0688672    .0666853
cov_yesXurb~s |  -.3946547    .225488    -1.75   0.080    -.8367101    .0474007
  var_yesXage |   .0611443   .0120586     5.07   0.000     .0375042    .0847844
cov_yesXage~s |   .0051727   .0258404     0.20   0.841    -.0454859    .0558313
      var_yes |   .5352136   .1687242     3.17   0.002     .2044401    .8659871
------------------------------------------------------------------------------
   Draws saved in bdesh_draws.dta


   Attention!
   *Results are presented to conform with Stata covention, but
    are summary statistics of draws, not coefficient estimates.
```

Suppose one wished to save some values of individual-level random parameters, but that the problem has too many individuals or requires too many draws to get to convergence. A useful approach in these circumstances is to complete a long first run without saving parameters, and then do a short second one using starting values. Suppose that the code in the previous example has been run. One can then run something to the effect of the following to get individual parameters:

[6]:
```
mat b = e(b)
mat beta = b[1, 1..3]
mat V = b[1,4], b[1,5], b[1,6] \ b[1,5], b[1,7], b[1,8] \ b[1,6], b[1,7], b[1,9]
```

[7]:
```
bayesmixedlogit choice, rand(yesXurban yesXage yes) group(_id) id(district) ///
     from(beta) fromv(V) draws(100) indsave(randpars) indkeep(50) replaceind
```

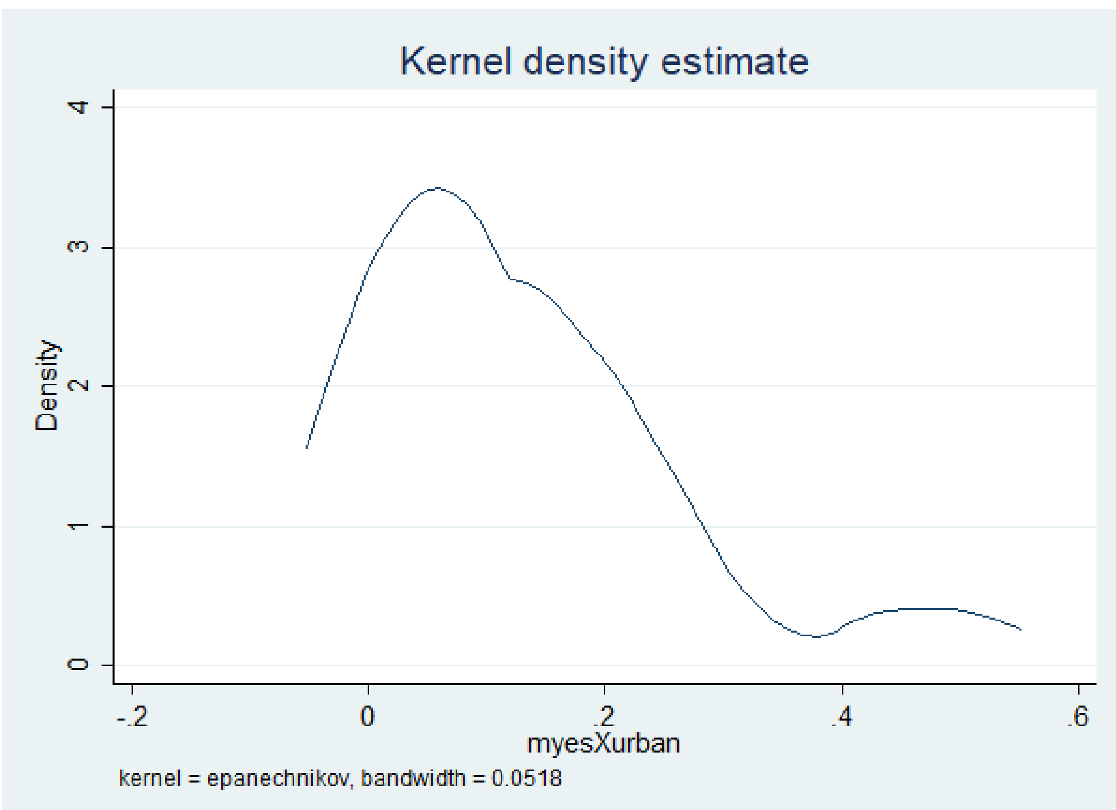
```
Bayesian Mixed Logit Model                          Observations     =      3868
                                                    Groups           =        60
Acceptance rates:                                   Choices          =      1934
 Fixed coefs               =                        Total draws      =       100
 Random coefs(ave,min,max)= 0.297, 0.160, 0.400    Burn-in draws    =         0
------------------------------------------------------------------------------
       choice |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
--------------+---------------------------------------------------------------
Random        |
    yesXurban |   .7763147   .1041361     7.45   0.000     .5697116    .9829178
```

```
       yesXage |   .0078787   .0344768     0.23   0.820    -.0605223    .0762797
           yes |  -.7578993   .1074017    -7.06   0.000    -.9709811   -.5448174
--------------+----------------------------------------------------------------
Cov_Random    |
var_yesXurban |   .3507105   .2150101     1.63   0.106    -.0758635    .7772845
cov_yesXurb~e |  -.0014138   .0203943    -0.07   0.945    -.0418756    .0390479
cov_yesXurb~s |  -.1896818    .140957    -1.35   0.181    -.4693365    .0899729
  var_yesXage |   .0599069   .0127719     4.69   0.000     .0345679    .0852459
cov_yesXage~s |   .0070305   .0200905     0.35   0.727    -.0328285    .0468896
      var_yes |   .3150498   .1372705     2.30   0.024      .042709    .5873906
-------------------------------------------------------------------------------
   50 value(s) of individual-level random parameters saved in randpars.dta


   Attention!
   *Results are presented to conform with Stata covention, but
    are summary statistics of draws, not coefficient estimates.
```

One post-estimation idea is to get the mean for parameter values by individuals, and fit some kernel density to the means to view the distribution of the individual-level parameters:

[8]:
```
bysort district: egen myesXurban = mean(yesXurban)
bysort district: gen last = _n == _N
kdensity myesXurban if last
graph display
```



```
.     global stata_kernel_graph_counter = $stata_kernel_graph_counter + 1
```

1.3.3 Example 3

Of course, it is possible to just do one run and retain all the information. As one final example:

[9]:
```
webuse union, clear
case2alt, casevars(age grade) choice(union) gen(unionmember)
bayesmixedlogit unionmember, rand(y0Xage y0Xgrade y0) group(_id) id(idcode) ///
     draws(1000) burn(800) saving(parm_draws) indsave(indparm_draws) indkeep(20)␣
↪replaceind replace
```

```
(NLS Women 14-24 in 1968)

(note: variable _id used since case() not specified)
(note: variable _altnum used since altnum() not specified)

choice indicated by: unionmember
case identifier: _id
case-specific interactions: y0* y1*



Bayesian Mixed Logit Model                          Observations    =     52400
                                                    Groups          =      4434
Acceptance rates:                                   Choices         =     26200
 Fixed coefs              =                         Total draws     =      1000
 Random coefs(ave,min,max)= 0.221, 0.017, 0.339     Burn-in draws   =       800
------------------------------------------------------------------------------
  unionmember |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
Random       |
       y0Xage |   .0427559   .0045651     9.37   0.000      .033754    .0517578
     y0Xgrade |  -.0475095   .0097271    -4.88   0.000    -.0666903   -.0283288
           y0 |   2.405118   .0067917   354.13   0.000     2.391726    2.418511
-------------+----------------------------------------------------------------
Cov_Random   |
   var_y0Xage |   .0472717   .0030153    15.68   0.000     .0413258    .0532176
cov_y0Xagey~e |   -.084886   .0067274   -12.62   0.000    -.0981517   -.0716203
 cov_y0Xagey0 |  -.0017989   .0019461    -0.92   0.356    -.0056365    .0020387
 var_y0Xgrade |   .2070458   .0164691    12.57   0.000     .1745704    .2395211
```

```
cov_y0Xgrad~0 |  -.0031688   .0034891    -0.91   0.365     -.010049    .0037114
       var_y0 |   .0808648   .0067269    12.02   0.000     .0676001    .0941296
------------------------------------------------------------------------------
   Draws saved in parm_draws.dta
   20 value(s) of individual-level random parameters saved in indparm_draws.dta


   Attention!
   *Results are presented to conform with Stata covention, but
    are summary statistics of draws, not coefficient estimates.
```

1.3.4 Stored results

`bayesmixedlogit` stores the following in e():

Scalars `e(N)` number of observations

`e(df_r)` degrees of freedom for summarizing draws (equal to number of retained draws)

`e(krnd)` number of random parameters

`e(kfix)` number of fixed parameters

`e(draws)` number of draws

`e(burn)` burn-in observations

`e(thin)` thinning parameter

`e(random_draws)` number of draws of each set of random parameters per pass

`e(fixed_draws)` number of draws of fixed parameters per pass

`e(damper_fixed)` damping parameter – fixed parameters

`e(damper_random)` damping parameter – random parameters

`e(opt_arate_fixed)` desired acceptance rate – fixed parameters

`e(opt_arate_random)` desired acceptance rate – random parameters

`e(N_groups)` number of groups

`e(N_choices)` number of choice occasions

`e(arates_fa)` acceptance rate – fixed parameters

`e(arates_ra)` average acceptance rate – random parameters

`e(arates_rmax)` maximum acceptance rate – random parameters

`e(arates_rmin)` minimum acceptance rate – random parameters

`e(inddraws)` draws of individual parameters kept

Macros `e(cmd)` `bayesmixedlogit`

`e(depvar)` name of dependent variable

`e(indepvars)` independent variables

`e(title)` title in estimation output

`e(properties)` b V

`e(saving)` file containing results

`e(fixed_sampler)` sampler type for fixed parameters

`e(random_sampler)` sampler type for random parameters

`e(random)` random parameter names

`e(fixed)` fixed parameter names

`e(identifier)` identifier for individuals

`e(group)` identifier for choice occasions

`e(indsave)` file holding individual-level parameter draws

Matrices `e(b)` mean parameter values

`e(V)` variance-covariance matrix of parameters

`e(V_init)` initial variance-covariance matrix of random parameters

`e(b_init)` initial mean vector of random parameters

`e(arates_fixed)` row vector of acceptance rates of fixed parameters

`e(arates_rand)` vector or matrix of acceptance rates of random parameters

Functions `e(sample)` marks estimation sample

1.4 `bayesmixedlogitwtp`

`bayesmixedlogitwtp` is essentially a wrapper for `bayesmixedlogit`, with a transformation of the coefficient on a price variable. The defining characteristic of the WTP-space mixed logit model is normalization of coefficients using the (random) coefficient on a designated price variable, as described in Train and Weeks (2005), Scarpa, Thiene, and Train (2008), and Hole and Kolstad (2012).

The model assumes that the coefficient on the price variable follows (the negative of) a log-normal distribution. Hence, if the estimated parameter is `b`, the price variable has coefficient `-exp(b)`. The transformed coefficient is saved and displayed as part of the output, but as presented the saved and display value is the negative of the exponentiated average value of b, not the average of the value `-exp(b)`.

1.5 Options

All options for `bayesmixedlogitwtp` are the same as `bayesmixedlogit`, with the following additional option:

`price(varname)` specifies a numeric identifier variable for price occasions. `price()` is required.

1.5.1 Stored results

In addition to all the scalars, macros, and matrices stored by `bayesmixedlogit`, `bayesmixedlogitwtp` adds the following additional macros:

Scalars `e(price_coef)` - exponent of mean of estimated coefficient on price variable

Macros `e(pricevar)` - name of price variable

Example 4

The following example mirrors examples provided of usage of the mixlogitwtp command (with thanks to Arne Rise Hole for allowing use of the example):

[10]:
```
use http://fmwww.bc.edu/repec/bocode/t/traindata.dta, clear
describe

bayesmixedlogitwtp y contract local wknown, group(gid) id(pid) price(price) ///
     rand(seasonal tod) draws(4000) burn(1000) thin(5) arater(.4) saving(draws)␣
 ↪replace
```

```
Contains data from http://fmwww.bc.edu/repec/bocode/t/traindata.dta
  obs:         4,780
 vars:             9                          28 Nov 2006 18:40
 size:        52,580
--------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
--------------------------------------------------------------------------------
y               byte    %8.0g
price           byte    %8.0g
contract        byte    %8.0g
local           byte    %8.0g
wknown          byte    %8.0g
tod             byte    %8.0g
seasonal        byte    %8.0g
gid             int     %8.0g
pid             int     %9.0g
--------------------------------------------------------------------------------
Sorted by:
```

```
Bayesian Mixed Logit Model - WTP Form                Observations    =      4780
                                                     Groups          =       100
Acceptance rates:                                    Choices         =      1195
 Fixed coefs               = 0.290                   Total draws     =      4000
 Random coefs(ave,min,max)= 0.214, 0.164, 0.260      Burn-in draws   =      1000
                                                     *One of every 5 draws kept
------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
Fixed        |
    contract |   -.249242   .0298574    -8.35   0.000    -.3078797   -.1906042
       local |   2.425951   .2125614    11.41   0.000     2.008497    2.843406
      wknown |   1.741476   .1599273    10.89   0.000     1.427391    2.055562
-------------+----------------------------------------------------------------
Random       |
       price |  -.3333228   .0947416    -3.52   0.000    -.5193883   -.1472574
    seasonal |  -9.779766   .3131075   -31.23   0.000    -10.39469   -9.164847
         tod |  -9.612307   .3512882   -27.36   0.000    -10.30221   -8.922403
-------------+----------------------------------------------------------------
Cov_Random   |
   var_price |   .2942746   .0795158     3.70   0.000     .1381115    .4504377
cov_pricese~l |  -.2675019   .2485057    -1.08   0.282    -.7555485    .2205448
 cov_pricetod |  -.4965004   .2658258    -1.87   0.062    -1.018562    .0255617
 var_seasonal |   5.859335   1.728996     3.39   0.001     2.463715    9.254955
cov_seasona~d |   4.173949   1.390441     3.00   0.003     1.443226    6.904672
      var_tod |   7.391194   1.994975     3.70   0.000     3.473211    11.30918
------------------------------------------------------------------------------
   Draws saved in draws.dta
The price variable is price with transformed coef (-exp(b)): -0.779


   Attention!
   *Results are presented to conform with Stata covention, but
    are summary statistics of draws, not coefficient estimates.
```

1.5.2 Example 5

A case in which all coefficients are random:

[11]:
```
bayesmixedlogitwtp y, group(gid) id(pid) price(price) rand(seasonal tod wknown) /
↪//
     draws(2000) burn(1000) thin(5) arater(.4) saving(draws) replace
```

```
Bayesian Mixed Logit Model - WTP Form                    Observations    =      4780
                                                         Groups          =       100
Acceptance rates:                                        Choices         =      1195
 Fixed coefs               =                             Total draws     =      2000
 Random coefs(ave,min,max)= 0.152, 0.068, 0.237          Burn-in draws   =      1000
                                                         *One of every 5 draws kept
------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
Random       |
       price |  -.6533948   .1186226    -5.51   0.000    -.8873062   -.4194833
    seasonal |  -9.684424   .4084787   -23.71   0.000     -10.4899   -8.878947
         tod |  -9.844531   .4323386   -22.77   0.000    -10.69706   -8.992004
      wknown |   .9696596   .2220143     4.37   0.000     .5318704    1.407449
-------------+----------------------------------------------------------------
Cov_Random   |
   var_price |   .7395054   .2301116     3.21   0.002     .2857491    1.193262
cov_pricese~l |  -.4428133   .4570225    -0.97   0.334    -1.344014    .4583876
 cov_pricetod |  -.4947435   .5077538    -0.97   0.331    -1.495981    .5064943
cov_pricewk~n |  -.2005007   .2009999    -1.00   0.320    -.5968516    .1958502
 var_seasonal |   7.062787   2.070746     3.41   0.001     2.979491    11.14608
cov_seasona~d |   5.437463   1.669412     3.26   0.001     2.145556    8.729371
cov_seasona~n |  -.8402783   .5547666    -1.51   0.131     -1.93422    .2536639
     var_tod |   9.310019   2.347582     3.97   0.000      4.68083    13.93921
cov_todwknown |  -1.204763   .5393129    -2.23   0.027    -2.268232   -.1412935
  var_wknown |   .9792376   .2518897     3.89   0.000     .4825372    1.475938
------------------------------------------------------------------------------
   Draws saved in draws.dta
The price variable is price with transformed coef (-exp(b)): -0.520


   Attention!
   *Results are presented to conform with Stata covention, but
    are summary statistics of draws, not coefficient estimates.
```

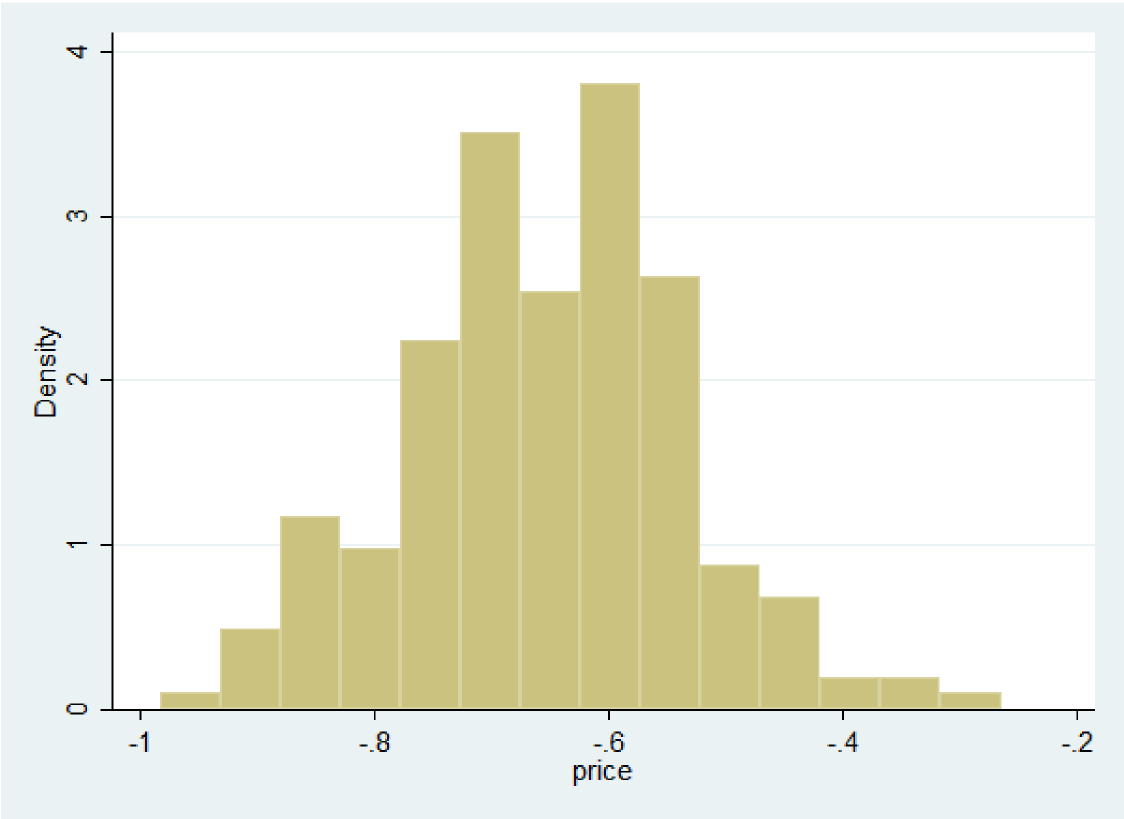
Looking at the distribution of draws for the `price` variable:

[12]:
```
use draws, clear
describe
hist price
graph display
```

```
Contains data from draws.dta
  obs:           200
 vars:            16                          1 Mar 2021 08:53
 size:        12,800
```

```
---------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
---------------------------------------------------------------------------------
price           float   %10.0g
seasonal        float   %10.0g
tod             float   %10.0g
wknown          float   %10.0g
var_price       float   %10.0g
cov_priceseas~l float   %10.0g
cov_pricetod    float   %10.0g
cov_pricewknown float   %10.0g
var_seasonal    float   %10.0g
cov_seasonaltod float   %10.0g
cov_seasonalw~n float   %10.0g
var_tod         float   %10.0g
cov_todwknown   float   %10.0g
var_wknown      float   %10.0g
fun_val         float   %10.0g
t               float   %9.0g
---------------------------------------------------------------------------------
Sorted by:

(bin=14, start=-.98271137, width=.05126763)
```



```
.     global stata_kernel_graph_counter = $stata_kernel_graph_counter + 1
```

1.6 References

Baker, M. J. 2014. *Adaptive Markov chain Monte Carlo sampling and estimation in Mata*. **Stata Journal** 14: 623-61.

Gelman, A., J. B. Carlin, H. S. Stern, and D. B. Rubin. 2009. **Bayesian data analysis**. 2nd. ed. Boca Raton, FL: Chapman & Hall/CRC.

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

Hole, A. R. and J. R. Kolstad. 2012. *Mixed logit estimation of willingness to pay distributions: a comparison of models in preference and WTP space using data from a health-related choice experiment*. **Empirical Economics** 42: 445-469.

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

R. Scarpa, M. Thiene, and K. Train. 2008. *Utility in willingness to pay space: A tool to address confounding random scale effects in destination choice to the Alps*. **American Journal of Agricultural Economics** 90: 994-1010.

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

Train, K. E. and M. Weeks. 2005. *Discrete choice models in preference space and willingness-to-pay space*. In: Scarpa R, Alberini A (eds), **Application of simulation methods in environmental and resource economics**. Springer, Dordrecht, pp 1-16.