The probability of alternative 1 is $P_{n1} = Prob(\tilde{U}_{n21} < 0$ and $\tilde{U}_{n31} < 0) = Prob(\tilde{V}_{n21} + \tilde{\varepsilon}_{n21} < 0$ and $\tilde{V}_{n31} + \tilde{\varepsilon}_{n31} < 0)$. This probability is hard to evaluate numerically in terms of the $\tilde{\varepsilon}$'s because they are correlated. However, using the transformation based on the Choleski factor, the probability can be written in a way that involves independent random terms. The probability becomes a function of the one dimensional standard cumulative normal distribution:

$$\begin{aligned} P_{n1} &= Prob(\tilde{V}_{n21} + c_{aa}\eta_1 < 0 \text{ and } \tilde{V}_{n31} + c_{ab}\eta_1 + c_{bb}\eta_2 < 0) \\ &= Prob(\tilde{V}_{n21} + c_{aa}\eta_1 < 0) \times Prob(\tilde{V}_{n31} + c_{ab}\eta_1 + c_{bb}\eta_2 < 0 \mid \tilde{V}_{n21} + c_{aa}\eta_1 < 0) \\ &= Prob(\eta_1 < -\tilde{V}_{n21}/c_{aa}) \times Prob(\eta_2 < -(\tilde{V}_{n31} + c_{ab}\eta_1)/c_{bb} \mid \eta_1 < -\tilde{V}_{n21}/c_{aa}) \\ &= \Phi(-\tilde{V}_{n21}/c_{aa}) \times \int_{\eta_1=-\infty}^{-\tilde{V}_{n21}/c_{aa}} \Phi(-(\tilde{V}_{n31} + c_{ab}\eta_1)/c_{bb})\phi(\eta_1)d\eta_1 \end{aligned}$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution evaluated at the point in the parentheses and $\phi(\cdot)$ is the standard normal density. The first term, $\Phi(-\tilde{V}_{n21}/c_{aa})$ is easy to calculate: it is simply the cumulative standard normal evaluated at $-\tilde{V}_{n21}/c_{aa}$. Computer packages contain fast routines for calculating the cumulative standard normal. The second term is an integral. As we know, computers cannot integrate, and we use simulation to approximate integrals. This is the heart of the GHK procedure: using simulation to approximate the integral in the second term in P_{n1}.

Let us examine this integral more closely. It is the integral over a truncated normal, namely, over η_1 up to $-\tilde{V}_{n21}/c_{aa}$. The simulation proceeds as follows. Draw a value of η_1 from a standard normal density truncated above at $-\tilde{V}_{n21}/c_{aa}$. For this draw, calculate the term $\Phi(-(\tilde{V}_{n31} + c_{ab}\eta_1)/c_{bb})$. Repeat this process for many draws, and average the results. This average is a simulated approximation to $\int_{\eta_1=-\infty}^{-\tilde{V}_{n21}/c_{aa}} \Phi(-(\tilde{V}_{n31} + c_{ab}\eta_1)/c_{bb})\phi(\eta_1)d\eta_1$. The simulated probability is then obtained by multiplying this average by the value of $\Phi(-\tilde{V}_{n21}/caa)$, which is calculated exactly. Simple enough!

The question arises: how do we take a draw from a truncated normal? We describe how to take draws from truncated univariate distributions in section (9.2.4). The reader might want to jump ahead and quickly review that section. For truncated normals, the process is: Take a draw from a standard uniform, labeled μ. Then calculate $\eta = \Phi^{-1}(\mu\Phi(-\tilde{V}_{n21}/c_{aa}))$. The resulting η is a draw from a normal density truncated from above at $-\tilde{V}_{n21}/c_{aa}$.

We can now put this all together to give the explicit steps that are used for the GHK simulator in our three-alternative case. The probability of alternative 1 is:

$$P_{n1} = \Phi(-\tilde{V}_{n21}/c_{aa}) \times \int_{\eta_1=-\infty}^{-\tilde{V}_{n21}/c_{aa}} \Phi(-(\tilde{V}_{n31} + c_{ab}\eta_1)/c_{bb})\phi(\eta_1)d\eta_1.$$

This probability is simulated as follows:

1. Calculate $k = \Phi(-\tilde{V}_{n21}/c_{aa})$.
2. Draw a value of η_1, labeled η_1^r, from a truncated standard normal truncated at $-\tilde{V}_{n21}/c_{aa}$. This is accomplished as follows:
   (a) Draw a standard uniform μ^r.
   (b) Calculate $\eta_1^r = \Phi^{-1}(\mu^r\Phi(-\tilde{V}_{n21}/c_{aa})$.
3. Calculate $g^r = \Phi(-(\tilde{V}_{n31} + c_{ab}\eta_1^r)/c_{bb})$.
4. The simulated probability for this draw is $\check{P}_{n1}^r = k \times g^r$.
5. Repeat steps 1-4 R times and average the results. This average is the simulated probability: $\check{P}_{n1} = (1/R)\sum \check{P}_{n1}^r$.

A graphical depiction is perhaps useful. Figure 5.3 shows the probability for alternative 1 in the space of independent errors η_1 and η_2. The x-axis is the value of η_1, and the y-axis is the value of η_2. The line labeled A is where η_1 is equal to $-\tilde{V}_{n21}/c_{aa}$. The condition that η_1 is below $-\tilde{V}_{n21}/c_{aa}$ is met in the striped area to the left of line A. The line labeled B is where $\eta_2 = -(\tilde{V}_{n31} + c_{ba}\eta_1)/c_{bb}$. Note that the y-intercept is where $\eta_1 = 0$ such that $\eta_2 = -\tilde{V}_{n31}/c_{bb}$ at this point. The slope of the line is $-c_{ba}/c_{bb}$. The condition that $\eta_2 < -(\tilde{V}_{n31} + c_{ba}\eta_1)/c_{bb}$ is satisfied below line B. The shaded area is where η_1 is to the left of line A and η_2 is below line B. The mass of density in the shaded area is therefore the probability that alternative 1 is chosen.

The probability (i.e., the shaded mass) is the mass of the striped area times the proportion of this striped mass that is below line B. The striped area has mass $\Phi(-\tilde{V}_{n21}/c_{aa})$. This is easy to calculate. For any given value of η_1, the portion of the striped mass that is below line B is also easy to calculate. For example, in Figure 5.4, when η_1 takes the value η_1^r, then the probability that η_2 is below line B is the share of line C's mass that is below line B. This share is simply

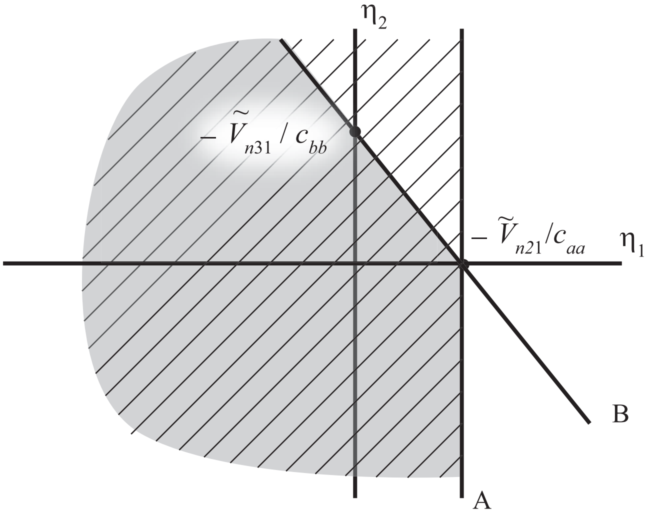

Figure 5.3: Probability of alternative 1.

$\Phi(-(\tilde{V}_{n31}+c_{ab}\eta_1^r)/c_{bb})$. The portion of the striped mass that is below line B is therefore the average of $\Phi(-(\tilde{V}_{n31}+c_{ab}\eta_1^r)/c_{bb})$ over all values of η_1 that are to the left of line A. This average is simulated by taking draws of η_1 to the left of line A, calculating $\Phi(-(\tilde{V}_{n31}+c_{ab}\eta_1^r)/c_{bb})$ for each draw, and averaging the results. The probability is then this average times the mass of the striped area, $\Phi(-\tilde{V}_{n21}/c_{aa})$.

General model

We can now describe the GHK simulator in general terms quickly, since the basic logic has already been discussed. This succinct expression serves to reinforce the concept that the GHK simulator is actually easier than it might at first appear.

Utility is expressed as:

$$U_{nj} = V_{nj} + \varepsilon_{nj} \qquad j = 1, ..., J$$

$$\varepsilon_n =< \varepsilon_{n1}, ..., \varepsilon_{nJ} > \qquad \varepsilon_n : J \times 1$$

$$\varepsilon_n \sim N(0, \Omega)$$

Transform to utility differences against alternative i:

$$\tilde{U}_{nji} = \tilde{V}_{nji} + \tilde{\varepsilon}_{nji}, \quad j \neq i$$

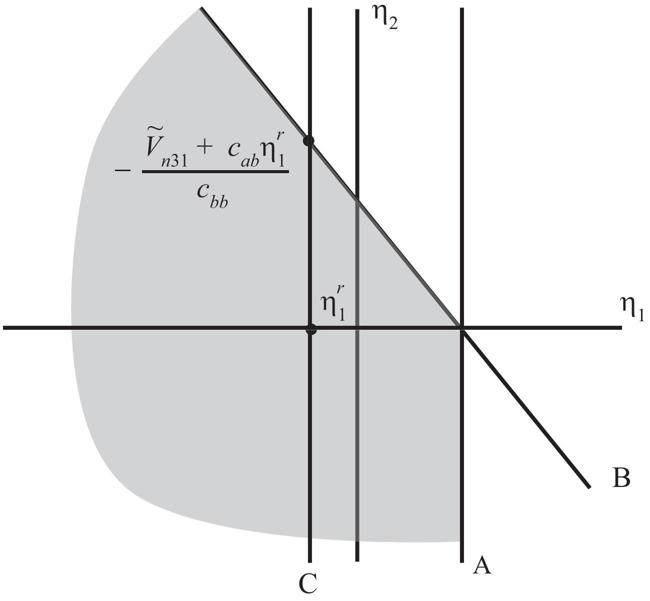

Figure 5.4: Probability that η_2 is in the correct range given η_1^r.

$$\tilde{\varepsilon}_{ni} =< \tilde{\varepsilon}_{n1}, ..., \tilde{\varepsilon}_{nJ} > \qquad \text{where ... is over all except } \; i$$

$$\tilde{\varepsilon}_{ni} : (J-1) \times 1$$

$$\tilde{\varepsilon}_{ni} \sim N(0, \tilde{\Omega}_i)$$

where $\tilde{\Omega}_i$ is derived from Ω.

Re-express the errors as a Choleski transformation of iid standard normal deviates

$$L_i \;\text{ s.t. }\; L_i L_i' = \Omega_i$$

$$L_i = \begin{pmatrix} c_{11} & 0 & \dots & \dots & \dots & 0 \\ c_{21} & c_{22} & 0 & \dots & \dots & 0 \\ c_{31} & c_{32} & c_{33} & 0 & \dots & 0 \\ \vdots & & & & & . \end{pmatrix}$$

Then stacking utilities $\tilde{U}_{ni} = (\tilde{U}_{n1i}, ..., \tilde{U}_{nJi})$, we get the vector form of the model

$$\tilde{U}_{ni} = \tilde{V}_{ni} + L_i \eta_n$$

where $\eta_n =< \eta_{1n}, ..., \eta_{J-1,n} >$ is a vector of iid standard normal deviates: $\eta_{nj} \sim N(0,1) \; \forall \; j$. Written explicitly the model is

$$\tilde{U}_{n1} = \tilde{V}_{n1} + c_{11}\eta_1$$

$$\tilde{U}_{n2} = \tilde{V}_{n2} + c_{21}\eta_1 + c_{22}\eta_2$$

$$\tilde{U}_{n3} = \tilde{V}_{n3} + c_{31}\eta_1 + c_{32}\eta_2 + c_{33}\eta_3$$

and so on. The choice probabilities are

$$\begin{aligned} P_{ni} &= Prob(\tilde{U}_{nji} < 0 \;\; \forall \; j \neq i) \\ &= Prob\left(\eta_1 < \frac{-\tilde{V}_{n1i}}{c_{11}}\right) \\ &\times \; Prob\left(\eta_2 < \frac{-(\tilde{V}_{n2i} + c_{21}\eta_1)}{c_{22}} \mid \eta_1 < \frac{-\tilde{V}_{n1i}}{c_{11}}\right) \\ &\times \; Prob\left(\eta_3 < \frac{-(\tilde{V}_{n3i} + c_{31}\eta_1 + c_{32}\eta_2)}{c_{33}} \mid \eta_1 < \frac{-\tilde{V}_{n1i}}{c_{11}} \;\text{ and }\; \eta_2 < \frac{-(\tilde{V}_{n2i} + c_{21}\eta_1)}{c_{22}}\right) \\ &\times \quad \text{and so on} \end{aligned}$$

The GHK simulator is calculated as follows:

1. Calculate
$$Prob\left(\eta_1 < \frac{-\tilde{V}_{n1i}}{c_{11}}\right) = \Phi\left(\frac{-\tilde{V}_{n1i}}{c_{11}}\right)$$

2. Draw a value of η_1, labeled η_1^r, from a truncated standard normal truncated at $-\tilde{V}_{1in}/c_{11}$.This draw is obtained as follows:

   (a) Draw a standard uniform μ_1^r

   (b) Calculate $\eta_1^r = \Phi^{-1}(\mu_1^r \Phi(-\tilde{V}_{n1i}/c_{11}))$

3. Calculate
$$Prob\left(\eta_2 < \frac{-(\tilde{V}_{n2i} + c_{21}\eta_1)}{c_{22}} \mid \eta_1 = \eta_1^r\right) = \Phi\left(\frac{-(\tilde{V}_{n2i} + c_{21}\eta_1^r)}{c_{22}}\right)$$

4. Draw a value of η_2, labeled η_2^r, from a truncated standard normal truncated at $-(\tilde{V}_{n2i} + c_{21}\eta_1^r)/c_{22}$. This draw is obtained as follows:

   (a) Draw a standard uniform μ_1^r

   (b) Calculate $\eta_2^r = \Phi^{-1}(\mu_2^r \Phi(-(\tilde{V}_{n2i} + c_{21}\eta_1^r)/c_{22}))$

5. Calculate
$$Prob\left(\eta_3 < \frac{-(\tilde{V}_{n3i} + c_{31}\eta_1 + c32\eta_2)}{c_{33}} \mid \eta_1 = \eta_1^r \ , \ \eta_2 = \eta_2^r\right) = \Phi\left(\frac{-(\tilde{V}_{n3i} + c_{31}\eta_1^r + c_{32}\eta_2^r)}{c_{33}}\right)$$

6. And so on for all alternatives but i.

7. The simulated probability for this r^{th} draw of $\eta_1, \ \eta_2, ...$ is calculated as
$$\begin{array}{rcl} \check{P}_{ni}^r & = & \Phi\left(\dfrac{-\tilde{V}_{n1i}}{c_{11}}\right) \\ & \times & \Phi\left(\dfrac{-(\tilde{V}_{n2i} + c_{21}\eta_1^r)}{c_{22}}\right) \\ & \times & \Phi\left(\dfrac{-(\tilde{V}_{n3i} + c_{31}\eta_1^r + c_{32}\eta_2^r)}{c_{33}}\right) \\ & \times & \text{etc.} \end{array}$$

8. Repeat steps 1–7 many times, for $r = 1, ..., R$.

9. The simulated probability is
$$\check{P}_{in} = \frac{1}{R}\sum_r \check{P}_{in}^r.$$

GHK simulator with maximum likelihood estimation

There are several issues that need to be addressed when using the GHK simulator in maximum likelihood estimation. First, in the log-likelihood function, we use the probability of the decision-maker's chosen alternative. Since different decision-makers choose different alternatives, P_{ni} must be calculated for different i's. The GHK simulator takes utility differences against the alternative for which the probability is being calculated, and so different utility differences must be taken for decision-makers who chose different alternatives. Second, for a person who chose alternative i, the GHK simulator uses the covariance matrix $\tilde{\Omega}_i$, while for a person who chose alternative j, the matrix $\tilde{\Omega}_j$ is used. Both of these matrices are derived from the same covariance matrix Ω of the original errors. We must assure that the parameters in $\tilde{\Omega}_i$ are consistent with those in $\tilde{\Omega}_j$, in the sense that they both are derived from a common Ω. Third, we need to assure that the parameters that are estimated by maximum likelihood imply covariance matrices $\Omega_j\ \forall\, j$ that are positive definite, as a covariance matrix must be. Fourth, as always, we must make sure that the model is normalized for scale and level of utility, so that the parameters are identified.

Researchers use various procedures to address these issues. I will describe the procedure that I use.

To assure that the model is identified, I start with the covariance matrix of scaled utility differences with the differences taken against the first alternative. This is matrix $\tilde{\Omega}_1$ which is $(J-1)\times(J-1)$. To assure that the covariance matrix is positive definite, I parameterize the model in terms of the Choleski factor of $\tilde{\Omega}_1$. That is, I start with a lower-triangular matrix that is $(J-1)\times(J-1)$ whose top-left element is 1:
$$L_1 = \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 \\ c_{21} & c_{22} & 0 & \dots & \dots & 0 \\ c_{31} & c_{32} & c_{33} & 0 & \dots & 0 \\ \vdots & & & & & . \end{pmatrix}$$

The elements $c_{k\ell}$ of this Choleski factor are the parameters that are estimated in the model. Any matrix that is the product of a lower-triangular full-rank matrix multiplied by itself is positive definite. So by using the elements of L_1 as the parameters, I am assured that $\tilde{\Omega}_1$ is positive definite for any estimated values of these parameters.

The matrix Ω for the J non-differenced errors is created from L_1. I create a $J \times J$ Choleski factor for Ω by adding a row of zeros at the top of L_1 and a column of zeros at the left. The resulting matrix is

$$L = \begin{pmatrix} 0 & 0 & \ldots & \ldots & \ldots & \ldots & 0 \\ 0 & 1 & 0 & \ldots & \ldots & \ldots & 0 \\ 0 & c_{21} & c_{22} & 0 & \ldots & \ldots & 0 \\ 0 & c_{31} & c_{32} & c_{33} & 0 & \ldots & 0 \\ 0 & \vdots & & & & \ldots & \end{pmatrix}.$$

Then Ω is calculated as LL'. With this Ω, I can derive $\tilde{\Omega}_j$ for any j. Note that Ω constructed in this way is fully general (i.e., allows any substitution pattern), since it utilizes all the parameters in the normalized $\tilde{\Omega}_1$.

Utility is expressed in vector form stacked by alternatives: $U_n = V_n + \varepsilon_n$, $\varepsilon_n \sim N(0, \Omega)$. Consider a person who has chosen alternative i. For the log-likelihood function, we want to calculate P_{ni}. Recall the matrix M_i that we introduced in section (5.1). Utility differences are taken using this matrix: $\tilde{U}_{ni} = M_i U_n, \tilde{V}_{ni} = M_i V_n$, and $\tilde{\varepsilon}_{ni} = M_i \varepsilon_n$. The covariance of the error differences $\tilde{\varepsilon}_{ni}$ is calculated as $\tilde{\Omega}_i = M_i \Omega M_i'$. The Choleski factor of $\tilde{\Omega}_i$ is taken and labeled L_i. (Note that L_1 obtained here will necessarily be the same as the L_1 that we used at the beginning to parameterize the model.) The person's utility is expressed as: $\tilde{U}_{ni} = \tilde{V}_{ni} + L_i \eta_n$ where η_n is a $(J-1)$ vector of iid standard normal deviates. The GHK simulator is applied to this expression.

This procedure satisfies all of our requirements. The model is necessarily normalized for scale and level since we parameterize the model in terms of the Choleski factor L_1 of the covariance of *scaled* error *differences*, $\tilde{\Omega}_1$. Each $\tilde{\Omega}_i$ is consistent with each other $\tilde{\Omega}_j$ for $j \neq i$, because they are both derived from the same Ω (which is constructed from L_1.) Each $\tilde{\Omega}_i$ is positive definite for any values of the parameters, because the parameters are the elements of L_1. As stated above, any matrix that is the product of a lower triangular matrix multiplied by itself is positive definite, and so $\tilde{\Omega}_1 = LL'$ is positive definite. And each of the other $\tilde{\Omega}_j$'s, for $j = 2, \ldots, J$, is also positive definite, since they are constructed to be consistent with Ω_1, which is positive definite.

GHK as importance sampling

As I described above in the three-alternative case, the GHK simulator provides a simulated approximation of the integral

$$\int_{\eta_1=-\infty}^{-\tilde{V}_{n21}/caa} \Phi(-(\tilde{V}_{n31}+c_{ab}\eta_1)/c_{bb})\phi(\eta_1)d\eta_1.$$

The GHK simulator can be interpreted in another way that is often useful.

Importance sampling is a way of transforming an integral to be more convenient to simulate. The procedure is described in section (9.2.7), and readers might find it useful to jump ahead to review that section. Importance sampling can be summarized following. Consider any integral $\bar{t}=\int t(\varepsilon)g(\varepsilon)d\varepsilon$ over density g. Suppose that another density exists from which it is easy to draw. Label this other density $f(\varepsilon)$. The density g is called the target density and f is the generating density. The integral can be rewritten as $\bar{t}=\int[t(\varepsilon)g(\varepsilon)/f(\varepsilon)]f(\varepsilon)d\varepsilon$. This integral is simulated by taking draws from f, calculating $[t(\varepsilon)g(\varepsilon)/f(\varepsilon)]$ for each draw, and averaging the results. This procedure is called importance sampling because each draw from f is weighted by g/f when taking the average of t; the weight g/f is the "importance" of the draw from f. This procedure is advantageous if (1) it is easier to draw from f than g, and/or (2) the simulator based on $[t(\varepsilon)g(\varepsilon)/f(\varepsilon)]$ has better properties (e.g., smoothness) than the simulator based on $t(\varepsilon)$.

The GHK simulator can be seen as making this type of transformation, and hence as being a type of importance sampling. Let η be a vector of $J-1$ iid standard normal deviates. The choice probability can be expressed as

$$P_{ni}=\int I(\eta\in B)g(\eta)d\eta \tag{5.7}$$

where $B=\{\eta \text{ s.t } \tilde{U}_{nji}<0\ \forall\ j\neq i\}$ is the set of η's that result in i being chosen; $g(\eta)=\phi(\eta_1)...\phi(\eta_{J-1})$ is the density where ϕ denotes the standard normal density; and utilities are:

$$\tilde{U}_{n1}=\tilde{V}_{n1}+c_{11}\eta_1$$

$$\tilde{U}_{n2}=\tilde{V}_{n2}+c_{21}\eta_1+c_{22}\eta_2$$

$$\tilde{U}_{n3}=\tilde{V}_{n3}+c_{31}\eta_1+c_{32}\eta_2+c_{33}\eta_3$$

etc.

The direct way to simulate this probability is to take draws of η, calculate $I(\eta \in B)$ for each draw and average the results. This is the A-R simulator. This simulator has the unfortunate properties that it can be zero and is not smooth.

For GHK we draw η from a different density, not from $g(\eta)$. Recall that for GHK, we draw η_1 from a standard normal density truncated at $-\tilde{V}_{n1i}/c_{11}$. The density of this truncated normal is $\frac{\phi(\eta_1)}{\Phi(-\tilde{V}_{n1i}/c_{11})}$, that is: the standard normal density normalized by the total probability below the truncation point. Draws of η_2, η_3, and so on are also taken from truncated densities, but with different truncation points. Each of these truncated densities takes the form $\frac{\phi(\eta_j)}{\Phi(\cdot)}$ for some truncation point in the denominator. The density from which we draw for the GHK simulator is therefore:

$$f(\eta) = \begin{cases} \frac{\phi(\eta_1)}{\Phi(-\tilde{V}_{n1i}/c_{11})} \times \frac{\phi(\eta_2)}{\Phi(-(\tilde{V}_{n2i}+c_{21}\eta_1)/c_{22})} \times \text{ etc. for } \eta \in B \\ \\ 0 \text{ for } \eta \notin B \end{cases} \tag{5.8}$$

Note that we only take draws that are consistent with the person choosing alternative i (since we draw from the correctly truncated distributions). So $f(\eta) = 0$ for $\eta \notin B$.

Recall that for a draw of η within the GHK simulator, we calculate:

$$\begin{aligned} \check{P}_{in}(\eta) &= \Phi\left(\frac{-\tilde{V}_{n1i}}{c_{11}}\right) \\ &\times \Phi\left(\frac{-(\tilde{V}_{n2i} + c_{21}\eta_1)}{c_{22}}\right) \\ &\times \text{ etc.} \end{aligned} \tag{5.9}$$

Note that this expression is the denominator of $f(\eta)$ for $\eta \in B$, given in equation (5.8). Using this fact, we can re-write the density $f(\eta)$ as:

$$f(\eta) = \begin{cases} \frac{g(\eta)}{\check{P}_{ni}(\eta)} \text{ for } \eta \in B \\ \\ 0 \text{ for } \eta \notin B \end{cases}$$

With this expression for $f(\eta)$, we can prove that the GHK simulator, $\check{P}_{in}(\eta)$, is unbiased for $P_{ni}(\eta)$:

$$E(\check{P}_{in}(\eta)) = \int \check{P}_{in}(\eta) f(\eta) d\eta$$

$$\begin{aligned}
&= \int_{\eta \in B} \check{P}_{in}(\eta) \frac{g(\eta)}{\check{P}_{in}(\eta)} d\eta \quad \text{by (5.6.3)} \\
&= \int_{\eta \in B} g(\eta) d\eta \\
&= \int I(\eta \in B) g(\eta) d\eta \\
&= P_{in}.
\end{aligned}$$

The interpretation of GHK as an importance sampler is also obtained from this expression:

$$\begin{aligned}
P_{in} &= \int I(\eta \in B) g(\eta) d\eta \\
&= \int I(\eta \in B) g(\eta) \frac{f(\eta)}{f(\eta)} d\eta \\
&= \int I(\eta \in B) \frac{g(\eta)}{g(\eta)/\check{P}_{in}(\eta)} f(\eta) d\eta \quad \text{by (5.6.3)} \\
&= \int I(\eta \in B) \check{P}_{in}(\eta) f(\eta) d\eta \\
&= \int \check{P}_{in}(\eta) f(\eta) d\eta
\end{aligned}$$

where the last equality is because $f(\eta) > 0$ only when $\eta \in B$. The GHK procedure takes draws from $f(\eta)$, calculates $\check{P}_{in}(\eta)$ for each draw, and averages the results. Essentially, GHK replaces the 0-1 $I(\eta \in B)$ with smooth $\check{P}_{in}(\eta)$ and makes the corresponding change in the density from $g(\eta)$ to $f(\eta)$.

Chapter 6

Mixed Logit

6.1 Choice probabilities

Mixed logit is a highly flexible model that can approximate any random utility model (McFadden and Train, 2000). It resolves the three limitations of standard logit by allowing for random taste variation, unrestricted substitution patterns, and correlation in unobserved factors over time. Unlike probit, it is not restricted to normal distributions. Its derivation is straightforward, and simulation of its choice probabilities is computationally simple.

Like probit, the mixed logit model has been known for many years but has only become fully applicable since the advent of simulation. The first application of mixed logit was apparently the automobile demand models created jointly by Boyd and Mellman (1980) and Cardell and Dunbar (1980). In these studies, the explanatory variables did not vary over decision-makers and the observed dependent variable was market shares rather than individual customer's choices. As a result, the computationally intensive integration that is inherent in mixed logit (as explained below) needed to be performed only once for the market as a whole, rather than for each decision-maker in a sample. Early applications on customer-level data, such as Train et al. (1987) and Ben-Akiva et al. (1993), included only one or two dimensions of integration, which could be calculated by quadrature. Improvements in computer speed and in our understanding of simulation methods have allowed the full power of mixed logits to be utilized. Among the studies to evidence this power are those by Bhat (1998*a*) and Brownstone and Train (1999) on cross-sectional data and Erdem (1996), Revelt and Train (1998), and Bhat (2000) on panel data. The description in the current chapter draws heavily from Train (1999).

Mixed logit models can be derived under a variety of different behavioral specifications, and each derivation provides a particular interpretation. The mixed logit model is *defined* on the basis of the functional form for its choice probabilities. Any behavioral specification whose derived choice probabilities take this particular form is called a mixed logit model.

Mixed logit probabilities are the integral of standard logit probabilities over a density of parameters. Stated more explicitly, a mixed logit model is any model whose choice probabilities can be expressed in the form:

$$P_{ni} = \int L_{ni}(\beta) f(\beta) d\beta$$

where $L_{ni}(\beta)$ is the logit probability evaluated at parameters β:

$$L_{ni}(\beta) = \frac{e^{V_{ni}(\beta)}}{\sum_{j=1}^{J} e^{V_{nj}(\beta)}}$$

and $f(\beta)$ is a density function. $V_{ni}(\beta)$ is a portion of utility, which depends on parameters β. If utility is linear in β, then $V_{ni}(\beta) = \beta' x_{ni}$. In this case, the mixed logit probability takes its usual form:

$$P_{ni} = \int \left(\frac{e^{\beta' x_{ni}}}{\sum_j e^{\beta' x_{nj}}} \right) f(\beta) d\beta. \tag{6.1}$$

The mixed logit probability is a weighted average of the logit formula evaluated at different values of β, with the weights given by density $f(\beta)$. In the statistics literature, the weighted average of several functions is called a mixed function, and the density that provides the weights is called the mixing distribution. Mixed logit is a mixture of the logit function evaluated at different β's with $f(\beta)$ as the mixing distribution.

Standard logit is a special case where the mixing distribution $f(\beta)$ is degenerate at fixed parameters b: $f(\beta) = 1$ for $\beta = b$ and zero for $\beta \neq b$. The choice probability (6.1) then becomes the simple logit formula

$$P_{ni} = \frac{e^{b' x_{ni}}}{\sum_j e^{b' x_{nj}}}.$$

The mixing distribution $f(\beta)$ can be discrete, with β taking a finite set of distinct values. Suppose β takes M possible values labeled

$b_1, \ldots, b_M$ with probability s_m that $\beta = b_m$. In this case, the mixed logit becomes the "latent class model" that has long been popular in psychology and marketing; examples include Kamakura and Russell (1989) and Chintagunta, Jain and Vilcassim (1991). The choice probability is:

$$P_{ni} = \sum_{m=1}^{M} s_m \left(\frac{e^{b_m' x_{ni}}}{\sum_j e^{b_m' x_{nj}}} \right).$$

This specification is useful if there are M segments in the population, each of which has its own choice behavior or preferences. The share of the population in segment m is s_m, which the researcher can estimate within the model along with the b's for each segment.

In most applications that have actually been called mixed logit (such as those cited in the introductory paragraphs above), $f(\beta)$ is specified to be continuous. For example, the density of β can be specified to be normal with mean b and covariance W. The choice probability under this density becomes:

$$P_{ni} = \int \left(\frac{e^{\beta' x_{ni}}}{\sum_j e^{\beta' x_{nj}}} \right) \phi(\beta \mid b, W) d\beta$$

where $\phi(\beta \mid b, W)$ is the normal density with mean b and covariance W. The researcher estimates b and W. Lognormal, uniform, triangular, gamma, or any other distribution can be used. As will be shown in section 6.5, by specifying the explanatory variables and density appropriately, the researcher can represent any utility-maximizing behavior by a mixed logit model, as well as many forms of non-utility-maximizing behavior.

Tests for the need for a non-degenerate mixing distribution, as well as the adequacy of any given distribution, have been developed by McFadden and Train (2000) and Chesher and Santos-Silva (2002). Several studies have compared discrete and continuous mixing distributions within the context of mixed logit; see, for example, Wedel and Kamakura (2000) and Ainslie, Andrews and Currim (2001).

An issue of terminology arises with mixed logit models. There are two sets of parameters in a mixed logit model. First, we have the parameters β, which enter the logit formula. These parameters have density $f(\beta)$. The second set are parameters that describe this density. For example, if β is normally distributed with mean b and covariance W, then b and W are parameters that describe the density

$f(\beta)$. Usually (though not always, as noted below), the researcher is interested in estimating the parameters of f.

Denote the parameters that describe the density of β as θ. The more appropriate way to denote this density is $f(\beta \mid \theta)$. The mixed logit choice probabilities do not depend on the values of β. These probabilities are $P_{ni} = \int L_{ni}(\beta) f(\beta \mid \theta) d\beta$, which is a function of θ. The parameters β are integrated out. In this sense, the β's are similar to the ε_{nj}'s, in that both are random terms that are integrated out to obtain the choice probability.

Under some derivations of the mixed logit model, the values of β have interpretable meaning as representing the tastes of individual decision-makers. In these cases, the researcher might want to obtain information about the β's for each sampled decision-maker, as well as the θ that describes the distribution of β's across decision-makers. In Chapter 11, we describe how the researcher can obtain this information based on estimates of θ and the observed choices of each decision-maker. In the current chapter, we describe the estimation and interpretation of θ, using classical estimation procedures. In Chapter 12, we describe Bayesian procedures that provide information about θ and each decision-maker's β simultaneously.

6.2 Random coefficients

The mixed logit probability can be derived from utility-maximizing behavior in several ways that are formally equivalent but provide different interpretations. The most straightforward derivation, and most widely used in recent applications, is based on random coefficients. The decision-maker faces a choice among J alternatives. The utility of person n from alternative j is specified as

$$U_{nj} = \beta_n' x_{nj} + \varepsilon_{nj}$$

where x_{nj} are observed variables that relate to the alternative and decision-maker, β_n is a vector of coefficients of these variables for person n representing that person's tastes, and ε_{nj} is a random term that is iid extreme value. The coefficients vary over decision-makers in the population with density $f(\beta)$. This density is a function of parameters θ that represent, for example, the mean and covariance of the β's in the population. This specification is the same as for standard logit except that β varies over decision-makers rather than being fixed.

The decision-maker knows the value of his own β_n and ε_{nj}'s for all j and chooses alternative i if and only if $U_{ni} > U_{nj}$ $\forall j \neq i$. The researcher observes the x_{nj}'s but not β_n or the ε_{nj}'s. If the researcher observed β_n, then the choice probability would be standard logit, since the ε_{nj}'s are iid extreme value. That is, the probability *conditional* on β_n is

$$L_{ni}(\beta_n) = \frac{e^{\beta_n' x_{ni}}}{\sum_j e^{\beta_n' x_{nj}}}.$$

However, the researcher does not know β_n and therefore cannot condition on β. The unconditional choice probability is therefore the integral of $L_{ni}(\beta_n)$ over all possible variables of β_n:

$$P_{ni} = \int \left(\frac{e^{\beta' x_{ni}}}{\sum_j e^{\beta' x_{nj}}} \right) f(\beta) d\beta$$

which is the mixed logit probability (6.1).

The researcher specifies a distribution for the coefficients and estimates the parameters of that distribution. In most applications, such as Revelt and Train (1998), Mehndiratta (1996), Ben-Akiva and Bolduc (1996), $f(\beta)$ has been specified to be normal or lognormal: $\beta \sim N(b, W)$ or $ln(\beta) \sim N(b, W)$ with parameters b and W that are estimated. The lognormal distribution is useful when the coefficient is known to have the same sign for every decision-maker, such as a price coefficient that is known to be negative for everyone. Revelt and Train (2000), Hensher and Greene (2001), and Train (2001) have used triangular and uniform distributions. With the uniform density, β is distributed uniformly between $b - s$ and $b + s$, where the mean b and spread s are estimated. The triangular distribution has positive density that starts at $b - s$, rises linearly to b, and then drops linearly to $b + s$, taking the form of a tent or triangle. The mean b and spread s is estimated, like with the uniform, but the density is peaked instead of flat. These densities have the advantage of being bounded on both sides, thereby avoiding the problem that can arise with normals and lognormals of unreasonably large coefficients for some share of decision-makers. By constraining $s = b$, the researcher can assure that the coefficients have the same sign for all decision-makers. Siikamaki (2001) and Siikamaki and Layton (2001) use the Rayleigh distribution (Johnson, Kotz and Balakrishnan, 1994), which is on one side of zero like the lognormal but, as these researchers found, can be easier for estimation than the lognormal. Revelt (1999) used truncated normals. As these examples indicate, the researcher is free to specify a distribution that satisfies his concepts about behavior in his own application.

Variation in tastes that are related to observed attributes of the decision-maker are captured through specification of the explanatory variables and/or the mixing distribution. For example, cost might be divided by the decision-maker's income to allow the value or relative importance of cost to decline as income rises. The random coefficient of this variable then represents the variation over people with the same income in the value that they place on cost. The mean valuation of cost declines with income while the variance around the mean is fixed. Observed attributes of the decision-maker can also enter $f(\beta)$ so that higher-order moments of taste variation can also depend on attributes of the decision-maker. For example, Bhat (1998*a*) and Bhat (2000) specify $f(\beta)$ to be log-normal with the mean and variance being a function of decision-maker characteristics.

6.3 Error-components

A mixed logit model can be used without a random-coefficients interpretation, as simply representing error components that create correlations among the utilities for different alternatives. Utility is specified as

$$U_{nj} = \alpha' x_{nj} + \mu_n' z_{nj} + \varepsilon_{nj}$$

where x_{nj} and z_{nj} are vectors of observed variables relating to alternative j, α is a vector of fixed coefficients, μ is a vector of random terms with zero mean, and ε_{nj} is distributed iid extreme value. The terms in z_{nj} are error components that, along with ε_{nj}, define the stochastic portion of utility. That is, the unobserved (random) portion of utility is $\eta_{nj} = \mu_n' z_{nj} + \varepsilon_{nj}$, which can be correlated over alternatives depending on the specification of z_{nj}. For the standard logit model, z_{nj} is identically zero, such that there is no correlation in utility over alternatives. This lack of correlation gives rise to the IIA property and its restrictive substitution patterns. With non-zero error components, utility is correlated over alternatives: $Cov(\eta_{ni}, \eta_{nj}) = E(\mu_n' z_{ni} + \varepsilon_{ni})(\mu_n' z_{nj} + \varepsilon_{nj}) = z_{ni}' W z_{nj}$ where W is the covariance of μ_n. Utility is correlated over alternatives even when, as in most specifications, the error components are independent such that W is diagonal.

Various correlation patterns, and hence substitution patterns, can be obtained by appropriate choice of variables to enter as error components. For example, an analog to nested logit is obtained by specifying a dummy variable for each nest that equals 1 for each alternative in the nest and zero for alternatives outside the nest. With K non-overlapping nests, the error components are $\mu_n' z_{nj} = \sum_{k=1}^K \mu_{nk} d_{jk}$, where $d_{jk} = 1$ if j is in nest k and zero otherwise. It is convenient in this situation to specify the error components to be independently normally distributed: μ_{nk} iid $N(0, \sigma_k)$. The random term μ_{nk} enters the utility of each alternative in nest k, inducing correlation among these alternatives. It does not enter any of the alternatives in other nests, thereby not inducing correlation between alternatives in the nest with those outside the nest. The variance σ_k captures the magnitude of the correlation. It plays an analogous role as the "inclusive value coefficient" of nested logit models.

To be more precise, the covariance between two alternatives in nest k is $Cov(\eta_{ni}, \eta_{nj}) = E(\mu_k + \varepsilon_{ni})(\mu_k + \varepsilon_{nj}) = \sigma_k$. The variance for each of the alternatives in nest k is $Var(\eta_{ni}) = E(\mu_k + \varepsilon_{ni})^2 = \sigma_k + \pi^2/6$, since the variance of the extreme value term, ε_{ni}, is $\pi^2/6$ (see section 3.1). The correlation between any two alternatives within nest k is therefore $\sigma_k/(\sigma_k + \pi^2/6)$. Constraining the variance of each nest's error component to be the same for all nests (i.e., constraining $\sigma_k = \sigma,\ k = 1, \ldots, K$), is analogous to constraining the log-sum coefficient to be the same for all nests in a nested logit. This constraint also assures that the mixed logit model is normalized for scale and level.

Allowing different variances for the random terms for different nests is analogous to allowing the inclusive value coefficient to differ across nests in a nested logit. An analog to overlapping nests is captured with dummies that identify overlapping sets of alternatives, as in Bhat (1998*a*). An analog to heteroskedastic logit (discussed in section 4.5) is obtained by entering an error component for each alternative. Ben-Akiva, Bolduc and Walker (2001) provide guidance on how to specify these variables appropriately.

Error-components and random-coefficients specifications are formally equivalent. Under the random-coefficient motivation, utility is specified as $U_{nj} = \beta_n' x_{nj} + \varepsilon_{nj}$ with random β_n. The coefficients β_n can be decomposed into their mean denoted α and deviations denoted μ_n such that $U_{nj} = \alpha' x_{nj} + \mu_n' x_{nj} + \varepsilon_{nj}$, which has error components defined by $z_{nj} = x_{nj}$. Conversely, under an error-components motivation, utility is $U_{nj} = \alpha' x_{nj} + \mu_n' z_{nj} + \varepsilon_{nj}$, which is equivalent to a random parameters model with fixed coefficients for variables x_{nj} and random coefficients with zero means for variables z_{nj}. If x_{nj} and z_{nj} overlap (in the sense that some of the same variables enter x_{nj} and z_{nj},) the coefficients of these variables can be considered to vary randomly with mean α and the same distribution as μ_n around their mean.

Though formally equivalent, the way a researcher thinks about the model affects the specification of the mixed logit. For example, when thinking in terms of random parameters, it is natural to allow each variable's coefficient to vary and perhaps even to allow correlations among the coefficients. This is the approach pursued by Revelt and Train (1998). However, when the primary goal is to represent substitution patterns appropriately through the use of error components, the emphasis is placed on specifying variables that can induce correlations over alternatives in a parsimonious fashion so as to provide sufficiently realistic substitution patterns. This is the approach taken by Brownstone and Train (1999). The goals differed in these studies, with Revelt and Train being interested in the pattern of tastes, while Brownstone and Train were more concerned with prediction. The number of explanatory variables also differed, with Revelt and Train examining six variables, such that estimating the joint distribution of their coefficients was a reasonable goal, while Brownstone and Train included 26 variables. Expecting to estimate the distribution of 26 coefficients might be unreasonable, and yet thinking in terms of random parameters instead of error components can lead the researcher to such unreasonable expectations. It is important to remember that the mixing distribution, whether motivated by random parameters or error components, captures variance and correlations in unobserved factors. There is a natural limit on how much one can learn about things that are not seen.

6.4 Substitution patterns

Mixed logit does not exhibit independence from irrelevant alternatives (IIA) or the restrictive substitution patterns of logit. The ratio of mixed logit probabilities P_{ni}/P_{nj} depends on all the data, including attributes of alternatives other than i or j. The denominators of the logit formula are inside the integral and therefore do not cancel. The percent change in the probability for one alternative given a change in the m-th attribute of another alternative is:

$$\begin{aligned} E_{nix_{nj}^m} &= -\frac{1}{P_{ni}} \int \beta^m L_{ni}(\beta) L_{nj}(\beta) f(\beta) d\beta \\ &= -\int \beta^m L_{nj}(\beta) \left[\frac{L_{ni}(\beta)}{P_{ni}} \right] f(\beta) d\beta, \end{aligned}$$

where β^m is the m-th element of β. This elasticity is different for each alternative i. A 10 percent reduction for one alternative need not imply (as with logit) a 10 percent reduction in each other alternative. Rather, the substitution pattern depends on the specification of the variables and mixing distribution, which can be determined empirically.

Note that the percent change in probability depends on the correlation between $L_{ni}(\beta)$ and $L_{nj}(\beta)$ over different values of β, which is determined by the researcher's specification of variables and mixing distribution. For example, to represent a situation where an improvement in alternative j draws more proportionately from alternative i than alternative k, the researcher can specify an element of x that is positively correlated between i and j but uncorrelated or negatively correlated between k and j, with a mixing distribution that allows the coefficient of this variable to vary.

6.5 Approximation to any random utility model

McFadden and Train (2000) show that any random utility model (RUM) can be approximated to any degree of accuracy by a mixed logit with appropriate choice of variables and mixing distribution. This proof is analogous to the RUM-consistent approximations provided by Dagsvik (1994). An intuitive explanation can easily be provided. Suppose the true model is $U_{nj} = \alpha_n' z_{nj}$ where z_{nj} are variables related to alternative j and α follows any distribution $f(\alpha)$. Any random utility model can be expressed in this form. (The more traditional notation of $U_{nj} = \beta_n' x_{nj} + \varepsilon_{nj}$ is obtained by letting $z_{nj}' = \langle x_{nj}', d_j \rangle$ and $\alpha' = \langle \beta_n', \varepsilon_{nj} \rangle$ and $f(\alpha)$ the joint density of β_n and ε_{nj} $\forall j$.) Conditional on α, the person's choice is fully determined since U_{nj} is then known for each j. The conditional probability is therefore:

$$q_{ni}(\alpha) = I(\alpha_n' z_{ni} > \alpha_n' z_{nj} \; \forall j \neq i)$$

where $I(\cdot)$ is the 1-0 indicator of whether the event in parentheses occurs. This conditional probability is deterministic in the sense that the probability is either zero or one: conditional on all the unknown random terms, the decision-maker's choice is completely determined. The unconditional choice probability is the integral of $q_{ni}(\alpha)$ over α:

$$Q_{ni} = \int I(\alpha_n' z_{ni} > \alpha_n' z_{ni} \ \forall\, j \neq i) f(\alpha) d\alpha.$$

We can approximate this probability with a mixed logit. Scale utility by λ, such that $U_{nj}^* = (\alpha/\lambda) z_{nj}$. This scaling does not change the model since behavior is unaffected by the scale of utility. Then add an iid extreme value term: ε_{nj}. The addition of the extreme value term *does* change the model, since it changes the utility of each alternative. We add it because doing so gives us a mixed logit. And, as we will show (this is the purpose of the proof), adding the extreme value term is innocuous. The mixed logit probability based on this utility is:

$$P_{ni} = \int \left(\frac{e^{(\alpha/\lambda)' z_{ni}}}{\sum_j e^{(\alpha/\lambda)' z_{nj}}} \right) f(\alpha) d\beta.$$

As λ approaches zero, the coefficients α/λ in the logit formula grow large and P_{ni} approaches a 1-0 indicator for the alternative with the highest utility. That is, the mixed logit probability P_{ni} approaches the true probability Q_{ni} as λ approaches zero. By scaling the coefficients upwards sufficiently, the mixed logit based on these scaled coefficients is arbitrarily close to the true model. Srinivasan and Mahmassani (2000) use this concept of raising the scale of coefficients to show that a mixed logit can approximate a probit model; the concept applies generally to approximate any random utility model.

Recall that we added an iid extreme value term to the true utility of each alternative. These terms change the model because the alternative with highest utility before the terms are added might not have highest utility after adding them (since a different amount is added to each utility). However, by raising the scale of utility sufficiently, we can be essentially sure that the addition of the extreme value terms has no effect. Consider a two alternative example. Suppose, using the true model with its original scaling, that the utility of alternative 1 is 0.5 units higher than the utility of alternative 2, such that alternative 1 is chosen. Suppose we add an extreme value term to each alternative. There's a good chance, given the variance of these random terms, that the value obtained for alternative 2 will exceed that for alternative 1 by at least half a unit, such that alternative 2 obtains the highest utility instead of 1 after adding them. The addition of the extreme value terms changes the model since it changes which alternative has the highest utility. Suppose, however, that we scale up the original utility by a factor of 10 (i.e., $\lambda = 0.10$). The utility for alternative 1 now exceeds the utility for alternative 2 by 5 units. It is highly unlikely that adding extreme value terms to these utilities will reverse this difference. That is, it is highly unlikely, in fact next to impossible, that the value of ε_{n2} that is added to the utility of alternative 2 is larger by 5 than the ε_{n1} that is added to the utility of alternative 1. If scaling up by 10 is not sufficient to assure that adding the extreme value term has no effect, then the original utilities can be scaled up by 100 or 1000. At some point, a scale will be found for which the addition of the extreme value terms has no effect. Stated succinctly: adding an extreme value term to true utility, which makes the model into a mixed logit, does not change utility in any meaningful way when the scale of the utility is sufficiently large. A mixed logit can approximate any random utility model simply by scaling up utility sufficiently.

This demonstration is not intended to suggest that raising the scale of utility is actually how the researcher would proceed in specifying a mixed logit as an approximation to the true model. Rather, the demonstration simply indicates that if no other means for specifying a mixed logit to approximate the true model can be found, then this re-scaling procedure can be used to attain the approximation. Usually, a mixed logit can be specified that adequately reflects the true model without needing to resort to an upward scaling of utility. For example, the true model will usually contain some iid term that is added to the utility of each alternative. Assuming an extreme value distribution for this term is perhaps close enough to reality to be empirically indistinguishable from other distributional assumptions for the iid term. In this case, the scale of utility is determined naturally by the variance of this iid term. The researcher's task is simply to find variables and a mixing distribution that capture the other parts of utility, namely, the parts that are correlated over alternatives or heteroskedastic.

6.6 Simulation

Mixed logit is well suited to simulation methods for estimation. Utility is $U_{nj} = \beta_n' x_{nj} + \varepsilon_{nj}$ where the the coefficients β_n are distributed with density $f(\beta \mid \theta)$ where θ refers collectively to the parameters of this distribution (such as the mean and covariance of β). The researcher specifies the functional form $f(\cdot)$ and wants to estimate the parameters θ. The choice probabilities are

$$P_{ni} = \int L_{ni}(\beta) f(\beta \mid \theta) d\beta$$

where:

$$L_{ni}(\beta) = \frac{e^{\beta' x_{ni}}}{\sum_{j=1}^{J} e^{\beta' x_{nj}}}.$$

The probabilities are approximated through simulation for any given value of θ. (1) Draw a value of β from $f(\beta \mid \theta)$ and label it β^r with the superscript $r = 1$ referring to the first draw. (2) Calculate the logit formula $L_{ni}(\beta^r)$ with this draw. (3) Repeat steps 1 and 2 many times, and average the results. This average is the simulated probability:

$$\check{P}_{ni} = \frac{1}{R} \sum_{r=1}^{R} L_{ni}(\beta^r)$$

where R is the number of draws. $\check{P}_{ni}$ is an unbiased estimator of P_{ni} by construction. Its variance decreases as R increases. It is strictly positive, such that $ln(\check{P}_{ni})$ is defined, which is useful for approximating the log-likelihood function below. $\check{P}_{ni}$ is smooth (twice differentiable) in the parameters θ and the variables x, which facilitates the numerical search for the maximum of the likelihood function and the calculation of elasticities. And $\check{P}_{ni}$ sums to one over alternatives, which is useful in forecasting.

The simulated probabilities are inserted into the log-likelihood function to give a simulated log-likelihood:

$$SLL = \sum_{n=1}^{N} \sum_{j=1}^{J} d_{nj} ln \check{P}_{nj}$$

where $d_{nj} = 1$ if n chose j and zero otherwise. The maximum simulated likelihood estimator (MSLE), is the value of θ that maximizes SLL. The properties of this estimator are discussed in Chapter 10. Usually, different draws are taken for each observation. This procedure maintains independence over decision-makers of the simulated probabilities that enter SLL. Lee (1992) describes the properties of MSLE when the same draws are used for all observations.

The simulated mixed logit probability can be related to accept-reject (A-R) methods of simulation. A-R simulation is described in section (5.6) for probit models, but it is applicable more generally. For any random utility model, the A-R simulator is constructed as follows: (1) A draw of the random terms is taken. (2) The utility of each alternative is calculated based on this draw, and the alternative with the highest utility is identified. (3) Steps 1 and 2 are repeated many times. (4) The simulated probability for an alternative is calculated as the proportion of draws for which that alternative has the highest utility. The A-R simulator is unbiased by construction. However, it is not strictly positive for any finite number of draws. It is also not smooth but rather a step-function: constant within ranges of parameter for which the identity of the alternative with the highest utility does not change for any draws, and with jumps where changes in the parameters change the identity of the alternative with the highest utility. Numerical methods for maximization based on the A-R simulator are hampered by these characteristics. To address these numerical problems, the A-R simulator can be smoothed by replacing the 0-1 indicator with the logit formula. As discussed in section (5.6.2), the logit-smoothed A-R simulator can approximate the A-R simulator arbitrarily closely by scaling utility appropriately.

The mixed logit simulator can be seen as a logit-smoothed A-R simulator of any random utility model: draws of the random terms are taken, utility is calculated for these draws, the calculated utilities are inserted into the logit formula, and the results are averaged. The theorem that a mixed logit can approximate any random utility model (section 6.5) can be viewed from this perspective. We know from section (5.6.2) that the logit-smoothed A-R simulator can be arbitrarily close to the A-R simulator for any model, with sufficient scaling of utility. Since the mixed logit simulator is equivalent to a logit-smoothed A-R simulator, the simulated mixed logit model can be arbitrarily close to the A-R simulator of any model.

6.7 Panel data

labelsec:mxpanel

The specification is easily generalized to allow for repeated choices by each sampled decision-maker. The simplest specification treats the coefficients that enter utility as varying over people but being constant