# A Short Introduction into Multinomial Probit Models

**Univ. Prof. Dr. Reinhard Hujer**
Chair of Statistics and Econometrics
University of Frankfurt/M.

# The Multinomial Probit Model (1)

Nested Logit models introduced in the previous lecture are one way to avoid the IIA assumption. Another one is the use of multinomial Probit models.

Recall once again how we introduced multinomial Logit models. Our aim was to model probabilities for the $M$ different outcomes of the dependent variable $y_i$ in such a way that they sum up to unity:

$$p(y_i = 1) + p(y_i = 2) + \ldots + p(y_i = M) = 1$$

To meet this restriction we used the following function:

$$p(y_i = j) = F_j(\zeta_{ij}) = \frac{\exp(\zeta_{ij})}{\sum_{i=1}^{M} \exp(\zeta_{ij})}$$

where assuming $\zeta_{ij} = x_i'\beta_j$ leads to the conditional and $\zeta_{ij} = x_{ij}'\beta$ to the multinomial Logit model.

# The Multinomial Probit model (2)

It can be shown that assuming a latent threshold model:

$$y_{ij}^* = \zeta_{ij} + \epsilon_i$$

where the $\epsilon$'s are independent distributed according to a type I (Gumbel) extreme value distribution and where the observable dependent variable $y_i$ is linked with its latent counterpart $y_i^*$ via:

$$y_i = \begin{cases} j \text{ if } y_{ij}^* = \max(y_{i1}^*, y_{i2}^*, \ldots, y_{iM}^*) \\ 0 \text{ otherwise} \end{cases}$$

implies that the probability for choosing category $j$ is given by:

$$p(y_i = j) = F_j(\zeta_{ij}) = \frac{\exp(\zeta_{ij})}{\sum_{i=1}^{M} \exp(\zeta_{ij})}.$$

In this sense these two model formulations are equivalent and lead to the same choice probabilities.

# The Multinomial Probit Model (3)

One problem with multinomial Logit models was the IIA assumption imposed by them. This is due to the fact that the $\epsilon$'s were assumed to be independent distributed from each other, i.e. from each other, i.e. the covariance matrix $E(\epsilon\epsilon')$ restricted to be a diagonal matrix.

Although this independence has the advantage that the likelihood function is quite easy to compute, in most of the cases the IIA assumption leads to unrealistic predictions (recall the famous "red-bus-blue-bus" example). One alternative to break down the IIA assumption therefore consists in allowing the $\epsilon$'s to be correlated with each other and that is exactly what the **multinomial Probit model** does.

Assume again the following model for the latent variable $y_{ij}^*$:

$$y_{ij}^* = x_i'\beta_j + \epsilon_{ij}.$$

In the multinomial Probit model it is assumed that the $\epsilon_i$'s follows a multivariate normal distribution with covariance matrix $\Sigma$ where $\Sigma$ now is **not** restricted to be a diagonal matrix.

# The Multinomial Probit Model (4)

Multinomial Probit models assume that the $\epsilon_i$'s follow a multivariate normal distribution and are correlated across choices:

$$\epsilon \sim MND(0, \Omega), \text{ with } \Omega = I_N \otimes \Sigma \text{ and } \Sigma = E(\epsilon_i \epsilon_i') = \begin{pmatrix} \sigma_{11} & \dots & \sigma_{1M} \\ & \ddots & \\ \sigma_{M1} & \dots & \sigma_{MM} \end{pmatrix}.$$

Category $j$ is chosen if $y_{ij}^*$ is highest for $j$, i.e.:

$$y_i = j \text{ if } y_{ij}^* = \begin{cases} j \text{ if } y_{ij}^* = \max(y_{i1}^*, y_{i2}^*, \dots, y_{iM}^*) \\ 0 \text{ otherwise.} \end{cases}$$

The probability to choose category $j$ can then be written as:

$$\begin{aligned} p(y_i = j|x_i) &= p(y_{ij}^* > y_{i1}^*, \dots, y_{ij}^* > y_{i(j-1)}^*, y_{ij}^* > y_{i(j+1)}^*, \dots, y_{ij}^* > y_{iM}^*) \\ &= p\left((\epsilon_{ij} - \epsilon_{i1}) > x_i'(\beta_1 - \beta_j), \dots, (\epsilon_{ij} - \epsilon_{i(j-1)}) > x_i'(\beta_{(j-1)} - \beta_j),\right. \\ &\qquad \left.(\epsilon_{ij} - \epsilon_{i(j+1)}) > x_i'(\beta_{(j+1)} - \beta_j), \dots, (\epsilon_{ij} - \epsilon_{iM}) > x_i'(\beta_M - \beta_j)\right). \end{aligned}$$

# The Multinomial Probit Model (5)

Looking at this probability one can see that only the differences between the $y_{ij}^*$'s are identified and hence a reference category has to be assigned as it was the case for the Logit model. As a consequence the covariance matrix also reduced in its dimension from $(M \times M)$ to $(M-1) \times (M-1)$.

If we define $\tilde{\epsilon}_{il} \equiv \epsilon_{ij} - \epsilon_{il}$ and $\xi_{il} \equiv x_i'(\beta_l - \beta_j)$ for $l = 1, \ldots, (j-1), (j+1), \ldots, M$ then the probability $p(y_i = j|x_i)$ is given by:

$$\int_{\xi_{i1}}^{\infty} \cdots \int_{\xi_{i(j-1)}}^{\infty} \int_{\xi_{i(j+1)}}^{\infty} \cdots \int_{\xi_{iM}}^{\infty} \phi(\tilde{\epsilon}_{1i}, \ldots, \tilde{\epsilon}_{i(j-1)}, \tilde{\epsilon}_{i(j+1)}, \ldots, \tilde{\epsilon}_{iM}) d\tilde{\epsilon}_{i1} \ldots d\tilde{\epsilon}_{i(j-1)} d\tilde{\epsilon}_{i(j+1)} \ldots d\tilde{\epsilon}_{iM}$$

And here lies the **practical obstacle** with multinomial Probit models. There are no closed form expressions for such high dimensional integrals and hence for $M \geq 3$ one has to simulate them using monte carlo simulation techniques.

# Evaluating Integrals Using Monte Carlo Techniques (1)

In the following we will illustrate the basic idea of **monte carlo simulation techniques**. Assume for simplicity that we want to calculate the following one-dimensional integral:

$$I = \int_a^b f(x)dx.$$

If we rewrite this expression as follows ...

$$\frac{1}{b-a}I = \int_a^b \frac{1}{b-a} \cdot f(x)dx = E(f(x)).$$

... $f(x)$ can be treated as a random variable which is uniformly distributed in the interval $[a, b]$ and hence the right hand side of the above expression represents the expected value of $f(x)$.

# Evaluating Integrals Using Monte Carlo Techniques (2)

We can therefore rewrite the integral as:

$$I = (b-a)E(f(x)).$$

Next we take $D$ draws $u_i$ from an uniform distribution in the interval $[0,1]$ and transform it according to $x_i = a + (b-a) \cdot u_i$. $x_i$ will then follow an uniform distribution in the interval $[a,b]$.

Using this series of $x_i$ draws we can approximate the expected value of $f(x)$ by averaging the function $f(x)$ evaluated at the different $x_i$'s:

$$E(f(x)) \approx \frac{1}{D} \sum_{i=1}^{D} f(x_i).$$

Multiplying this expression with $(b-a)$ finally we get an approximation for the integral $I$:

$$I \approx (b-a) \cdot \frac{1}{D} \sum_{i=1}^{D} f(x_i).$$

# Evaluating Integrals Using Monte Carlo Techniques (3)

Consider as an example the following simple integral:

$$I = \int_1^2 e^x dx.$$

The exact value for this integral is 4.67. In the following we approximated this integral for $D = 10, 20, 50$ and $100$ draws, respectively. The following table contains the so obtained values and the absolute deviations from the true value. One can see that with more draws the approximation gets better.

| Draws | $I^{approx.}$ | $I^{exact}$ | Absolute deviation |
|---|---|---|---|
| 10 | 4.58 | 4.67 | 0.09 |
| 20 | 4.38 | 4.67 | 0.29 |
| 50 | 4.79 | 4.67 | 0.12 |
| 100 | 4.75 | 4.67 | 0.08 |

# Evaluating Integrals Using Monte Carlo Techniques (4)

The previously presented proceeding was a simple simulator applied to an univariate integral. However, estimating a multinomial probit model amounts to evaluate a multidimensional integral like the following:

$$\int_{\xi_1}^{\infty} \cdots \int_{\xi_{(j-1)}}^{\infty} \int_{\xi_{(j+1)}}^{\infty} \cdots \int_{\xi_M}^{\infty} \phi(\tilde{\epsilon}_1, \ldots, \tilde{\epsilon}_{(j-1)}, \tilde{\epsilon}_{(j+1)}, \ldots, \tilde{\epsilon}_M) d\tilde{\epsilon}_1 \ldots d\tilde{\epsilon}_{(j-1)} d\tilde{\epsilon}_{(j+1)} \ldots d\tilde{\epsilon}_M$$

Note that we have skipped the individual index for notational convenience. Such integrals reflect the probabilities for choosing a certain category. The above integral e.g. is the probability for choosing category $j$, i.e. $p(y_i = j|x_i)$.

There are number of different **simulators** which can be used in this context. Examples include the Accept-Reject Procedure, Importance Sampling, Gibbs Sampling and the Metropolis-Hastings Algorithm. One algorithm which has been found to be fast and accurate is the Geweeke-Hajivassiliou-Keane smooth recursive simulator (**GHK-simulator**) which will be presented in the following.

# Evaluating Integrals Using Monte Carlo Techniques (5)

We will illustrate the GHK-simulator using a multinomial Probit model with 4 categories, i.e. $y \in \{0, 1, 2, 3\}$. Since only utility differences matter, the dimension is reduced by one so that the probability for $p(y_i = 1|x_i)$ e.g. is given by the following three-fold integral:

$$\begin{aligned} p(y_i = 1|x_i) &= p(\tilde{\epsilon}_0 > \xi_0, \tilde{\epsilon}_2 > \xi_2, \tilde{\epsilon}_3 > \xi_3) \\ &= \int_{\xi_0}^{+\infty} \int_{\xi_2}^{+\infty} \int_{\xi_3}^{+\infty} \phi_3(\tilde{\epsilon}_0, \tilde{\epsilon}_2, \tilde{\epsilon}_3|x_i, \Sigma) d\tilde{\epsilon}_0 d\tilde{\epsilon}_2 d\tilde{\epsilon}_3 \end{aligned}$$

with $\tilde{\epsilon}_l \equiv \tilde{\epsilon}_j - \tilde{\epsilon}_l$ and $\xi_l \equiv x_i'(\beta_l - \beta_j)$ for $l = 0, 2, 3$ where the $\epsilon$'s are distributed according to a threefold normal distribution with covariance matrix $\Sigma$.

The probability $p(y_i = 1|x_i) = p(\tilde{\epsilon}_0 > \xi_0, \tilde{\epsilon}_2 > \xi_2, \tilde{\epsilon}_3 > \xi_3)$ can be rewritten as a product of one unconditional and two conditional probabilities according to:

$$p(\tilde{\epsilon}_0 > \xi_0) p(\tilde{\epsilon}_2 > \xi_2|\tilde{\epsilon}_0 > \xi_0) p(\tilde{\epsilon}_3 > \xi_3|\tilde{\epsilon}_2 > \xi_2, \tilde{\epsilon}_0 > \xi_0).$$

# Evaluating Integrals Using Monte Carlo Techniques (6)

Since $\Sigma$ is a positive definite matrix using the Cholesky decomposition, a lower triangular matrix $\Lambda$ can be found such that: $\Lambda\Lambda' = \Sigma$. Defining:

$$\nu = \begin{pmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{pmatrix} \text{ with } \nu \sim N(0,1)$$

we can rewrite $\tilde{\epsilon}$ as $\tilde{\epsilon} = \Lambda\nu$ and arrive at the following and arrive at the following relation between the $\tilde{\epsilon}$'s and the $\nu$'s:

$$\begin{aligned} \tilde{\epsilon}_0 &= \lambda_{11}\nu_1 \\ \tilde{\epsilon}_2 &= \lambda_{12}\nu_1 + \lambda_{22}\nu_2 \\ \tilde{\epsilon}_3 &= \lambda_{13}\nu_1 + \lambda_{23}\nu_2 + \lambda_{33}\nu_3 \end{aligned}$$

where $\lambda_{ij}$ is the $(i,j)$-element in the $\Lambda$ matrix.

# Evaluating Integrals Using Monte Carlo Techniques (7)

Using these relations, the product of unconditional and conditional probabilities can equivalently be written as:

$$
\begin{aligned}
p(y_i = j|x_i) \quad = \quad & p\left(\nu_1 > \frac{\xi_0}{\lambda_{11}}\right) \\
& p\left(\nu_2 > \frac{\xi_2 - \lambda_{12}\nu_1}{\lambda_{22}} \middle| \nu_1 > \frac{\xi_0}{\lambda_{11}}\right) \\
& p\left(\nu_3 > \frac{\xi_3 - \lambda_{13}\nu_1 - \lambda_{23}\nu_2}{\lambda_{33}} \middle| \nu_1 > \frac{\xi_0}{\lambda_{11}}, \nu_2 > \frac{\xi_2 - \lambda_{12}\nu_1}{\lambda_{22}}\right).
\end{aligned}
$$

The advantage of this expression is the fact that the $\nu$'s are independent normal distributed random variables and hence the probability $p(y_i = j|x_i)$ which we want to evaluate can be equivalently expressed as a product of independent but conditioned univariate cumulative density functions.

# Evaluating Integrals Using Monte Carlo Techniques (8)

Assume now that $\nu_1^*$ and $\nu_2^*$ are realizations taken from truncated normal distributions with lower truncation points $\frac{\xi_0}{\lambda_{11}}$ and $\frac{\xi_2 - \lambda_{12}\nu_1}{\lambda_{22}}$, respectively.

Drawing samples from these truncated normal distributions ensures that the conditioning of the probabilities is taken into account. Plugging $\nu_1^*$ and $\nu_2^*$ into the previous expression, we can rewrite it as a product of only unconditional, univariate and independent probabilities:

$$\begin{aligned} p(y_i = j|x_i) &= p\left(\nu_1 > \frac{\xi_0}{\lambda_{11}}\right) \\ &\quad p\left(\nu_2 > \frac{\xi_2 - \lambda_{12}\nu_1^*}{\lambda_{22}}\right) \\ &\quad p\left(\nu_3 > \frac{\xi_3 - \lambda_{13}\nu_1^* - \lambda_{23}\nu_2^*}{\lambda_{33}}\right). \end{aligned}$$

# Evaluating Integrals Using Monte Carlo Techniques (9)

The GHK simulator now generates a series of $d = 1, \ldots, D$ random observations of $\nu_1^{*d}$ and $\nu_2^{*d}$ so that the probability $p(y_i = j|x_i)$ that we seek for can be approximated by:

$$\hat{p}(y_i = j|x_i) = \frac{1}{D} \sum_{i=1}^{N} \Phi\left(\frac{\xi_0}{\lambda_{11}}\right) \Phi\left(\frac{\xi_2 + \lambda_{12}\nu_1^{*d}}{\lambda_{22}}\right) \Phi\left(\frac{\xi_3 + \lambda_{13}\nu_1^{*d} + \lambda_{23}\nu_2^{*d}}{\lambda_{33}}\right).$$

All other probabilities can be calculated in an analogous way. Plugging them into the likelihood function standard maximization procedures can be applied to get estimates for the parameters.

**Quote of the day:** *"As we know, computers cannot integrate."* Kenneth E. Train (2003)