Mixed Logit

Three limitations of standard logit:

1. Random taste variation
2. Unrestricted substitution patterns
3. Correlation in unobserved factors over time

Choice probabilities are given by:

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

is the probability of respondent n choosing product i.

$L_{ni}(\beta) = \frac{\exp(V_{ni}(\beta))}{\sum_j \exp(V_{nj}(\beta))}$ is called the logit probability at β.

We could have $V_{ni}(\beta) = \beta' x_{ni}$.

This form of P_{ni} is called a **mixed** form, with $f(.)$ as the **mixing distribution**. It is like taking the weighted average of the logit probabilities, L_{ni} at different βs, with $f(\beta)$ as the weights. Hence, the name **mixed logit**.

Note that we get standard logit when $f(\beta) = I(\beta = b)$. So this is more general.

We could take $f(\beta) = \phi(\beta | b, W)$. This is normal density with mean vector b and covariance matrix W.

In Hierarchical Bayesian estimation techniques, we assume prior distributions for b and W. There is a hierarchy of priors. And we use Bayes' Theorem. Hence the name **Hierarchical Bayes**.

Motivations

Random Coefficients

Decision maker n faces a choice task among J alternatives. We model utility that n gets from alternative j as

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

where ϵ_{nj} are i.i.d extreme value.

Decision maker knows β_n and ϵ_{nj} for all j, and chooses alternative i that maximises his utility from i, $U_{ni} > U_{nj}$ for all $j \neq i$.

Researcher doesn't observe β_n and ϵ_{nj}, but observes x_{nj}. So he does the math and finds out:

$$P_{ni} = P(U_{ni} > U_{nj} \forall j \neq i) = \int L_{ni}(\beta) f(\beta) d\beta.$$

Error Components

Utility is modeled as

$$U_{ni} = \alpha' x_{nj} + \eta_{nj}$$

where η_{nj} is unobserved part of the utility. It is natural that errors η_{nj} won't be independent over respondents, n and also over the alternatives j, because obviously we cannot observe *everything*. So, we introduce correlation,

$$\eta_{nj} = \mu_n' z_{nj} + \epsilon_{nj}$$

where μ_n are random terms with mean 0 and ϵ_{nj} are i.i.d. extreme value.

α is a vector of fixed coefficients. z_{nj} could be x_{nj}.

Some math:

$$Cov(\eta_{ni}, \eta_{nj}) = \mathbb{E}[(\mu_n' z_{ni} + \epsilon_{ni})(\mu_n' z_{nj} + \epsilon_{nj})] = z_{ni}' W z_{nj}$$

where W is the covariance matrix of μ_n. This shows that the unobserved utility components are correlated over alternatives.

See that even when W is diagonal, i.e. the μ_{nj} are independent over j, the covariance above is not 0.

You get the standard logit when you take $z_{nj} = 0$. There is no correlation in utility among alternatives. This gives rise to IIA property and its restrictive substitution patterns.

To get nested logit with K nests, take $\mu_n' z_{nj} = \sum_k \mu_{nk} I(j \in k)$. That is z_{nj} will be a K dimensional binary vector, with only one 1 at the position corresponding to the nest it belongs to. We take μ_{nk} to be $N(0, \sigma_k)$, all independent over n, k. This μ_{nk} enters into utility of each alternative in nest k inducing correlation among alternatives in k.

But μ_{nk} does not enter into utility of any alternative in other nests. So we have IIN and its restrictive substitution patterns.

So mixed logit is more general.

Substitution Patterns

There is no IIA or IIN.

Percentage change in the probability for one alternative i given a percentage change in the mth attribute of another alternative j is

$$E_{nix_{nj}^m} = -x_{nj}^m \int \beta^m L_{nj}(\beta) \frac{L_{ni}(\beta)}{P_{ni}} f(\beta) d\beta.$$

This elasticity is different for each i. A 10% reduction in probability for one alternative need not imply a 10% reduction in probability for each other alternative, unlike logit.

Simulation

The choice probabilities are:

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

and

$$L_{ni}(\beta) = \frac{\exp(V_{ni}(\beta))}{\sum_j \exp(V_{nj}(\beta))}.$$

For any given value of θ.

1. Draw a value of β from $f(\beta|\theta)$. Call it β^r for rth draw.
2. Calculate $L_{ni}(\beta^r)$.
3. Repeat steps 1 and 2 many times, say R times, and average the results.

So we are using Law of Large Numbers.

The average is the simulated probability:

$$\hat{P}_{ni} = \frac{1}{R} \sum_r L_{ni}(\beta^r).$$

We now maximise the SLL, the simulated log-likelihood:

$$SLL = \sum_n \sum_j d_{nj} \log \hat{P}_{nj}$$

where d_{nj} is 1 if n chose alternative j, otherwise 0.

The θ which maximises the SLL, is the MSLE, Maximum Simulated Likelihood Estimator.

But we cannot use this estimation for our projects. Think about what would happen when we consider new price scenarios.

Read the case study (page 147, page 14 of the PDF) here, for a practical example.

Hierarchical Bayesian estimation

Let the utility that person n obtains from alternative j in time period t be

$$U_{njt} = \beta_n' x_{njt} + \epsilon_{njt}$$

where ϵ_{njt} is i.i.d. extreme value and β_n follows $N(b, W)$.

There is a normal prior on b with large variance. There is an inverted wishart prior on W. b and W are called hyper-parameters.

We observe a sample of N people. The chosen alternatives for person n is $y_n' = (y_{n1}, ..., y_{nT})$. The choices of the entire sample is $Y = (y_1, ..., y_N)$.

The probability of person n's observed choices, conditional on β is

$$L(y_n|\beta) = \prod_t \frac{exp(\beta' x_{ny_{nt}t})}{\sum_j exp(\beta' x_{njt})}.$$

The unconditional probability of person n's observed choices is

$$L(y_n|b, W) = \int L(y_n|\beta)\phi(\beta|b, W)d\beta.$$

This is the mixed logit probability, with $f(.)$ being the normal density.

Now, use Bayes' Theorem,

$$K(b, W|Y) \propto \prod_n L(y_n|b, W)k(b, W)$$

where $k(b, W)$ is the product of normal density for b and the inverted wishart for W. That is, the product of priors.

We want to draw from the **posterior** distribution $K(b, W|Y)$. So that we can average the draws to get the estimates for b and W, using Law of Large Numbers.

But $L(y_n|b, W)$ is an expression involving an integral and it does not have a closed form and therfore, must be approximated through simulation. This makes estimation using MCMC algorithms like Metropolis-Hastings very time consuming.

That is why we use this set up:

$$K(b, W, \beta_n \forall n|Y) \propto \prod_n L(y_n|\beta_n)\phi(\beta_n|b, W)k(b, W).$$

Here, $L(y_n|\beta_n)$ is the logit formula.

Then we use Gibbs sampling. We keep drawing $b, W, \beta_n \forall n$, one at a time, conditioning the others. The resulting draws converge to draws from the joint posterior $K(b, W, \beta_n \forall n|Y)$.

Once the converged draws are obtained, the mean and standard deviation of the draws can be calculated. This gives the estimates and standard errors of the parameters.

This procedure also gives $\beta_n \forall n$. So this estimation technique can be used for our projects where we keep changing pricing scenarios.

Thumb rule: A total of 20000 iterations can be used for 1000 draws: 10000 for burn-in and 10000 after convergence, of which every 10th draw is retained to calculate the estimates and standard errors to remove correlations over iterations.

Read this for the prior used in `bayesm` package.

Remaining stuff:

Implementation of the following have not yet been done:

- Determining whether convergence has achieved
- MCMC diagnostics like Gelman-Rubin test statistic
- More performance diagnostics (as mentioned here)
