Nested Logit

Chris Conlon

Spring 2023

NYU Stern: Applied Econometrics

Reading

Today's reading is Chapters 4 from:

Ken Train's Discrete Choice Methods with Simulation

Multinomial Logit: IIA

The multinomial logit is frequently criticized for producing unrealistic substitution patterns

- Suppose we got rid of a product k then $s_j^{(1)} = s_j^{(0)} \frac{1}{1-s_k}$.
- Substitution is just proportional to your pre-existing shares s_j
- No concept of "closeness" of competition!

Relaxing IIA

Let's make ε_{ij} more flexible than IID. Hopefully still have our integrals work out.

$$u_{ij} = V_{ij} + \varepsilon_{ij}$$

- One approach is to allow for a block structure on ε_{ij} (and consequently on the elasticities).
- We assign products into groups g and add a group specific error term

$$u_{ij} = V_{ij} + \eta_g + \varepsilon_{ij}$$

- The trick putting a distribution on $\eta_g + \varepsilon_{ij}$ so that the integrals still work out.
- Do not try this at home: it turns out the required distribution is known as GEV and the resulting model is known as the nested logit.

Nested Logit

A traditional (and simple) relaxation of the IIA property is the Nested Logit. This model is often presented as two sequential decisions.

- First consumers choose a category (following an IIA logit).
- Within a category consumers make a second decision (following the IIA logit).
- This leads to a situation where while choices within the same nest follow the IIA property (do not depend on attributes of other alternatives) choices among different nests do not!

Alternative Interpretation

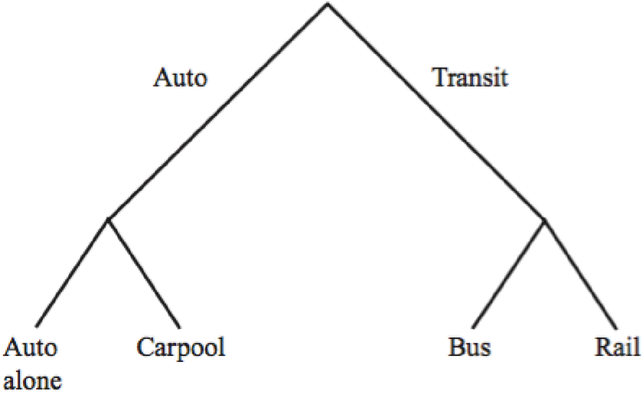

Figure 4.1. Tree diagram for mode choice.

Nested Logit

Utility looks basically the same as before:

$$U_{ij} = V_{ij} + \underbrace{\eta_{ig} + \widetilde{\varepsilon_{ij}}}_{\varepsilon_{ij}(\lambda_g)}$$

- We add a new term that depends on the group g but not the product j and think about it as varying unobservably over individuals i just like ε_{ij}.
- Now $\varepsilon_i \sim F(\varepsilon)$ where $F(\varepsilon) = \exp[-\sum_{g=G}^{G}\left(\sum_{j\in J_g}\exp[-\varepsilon_{ij}/\lambda_g]\right)^{\lambda_g}$. This is no longer Type I EV but GEV.
- The key is the addition of the λ_g parameters which govern (roughly) the within group correlation.
- This distribution is a bit cooked up to get a closed form result, but for $\lambda_g \in [0, 1]$ for all g it is consistent with random utility maximization.

Nested Logit

The nested logit choice probabilities are:

$$s_{ij} = \frac{e^{V_{ij}/\lambda_g} \left(\sum_{k \in J_g} e^{V_{ik}/\lambda_g} \right)^{\lambda_g - 1}}{\sum_{h=1}^{G} \left(\sum_{k \in J_h} e^{V_{ik}/\lambda_h} \right)^{\lambda_h}}$$

Within the same group g we have IIA and proportional substitution

$$\frac{s_{ij}}{s_{ik}} = \frac{e^{V_{ij}/\lambda_g}}{e^{V_{ik}/\lambda_g}}$$

But for different groups we do not:

$$s_{ij} = \frac{e^{V_{ij}/\lambda_g} \left(\sum_{k \in J_g} e^{V_{ik}/\lambda_g} \right)^{\lambda_g - 1}}{e^{V_{ik}/\lambda_h} \left(\sum_{k \in J_h} e^{V_{ik}/\lambda_h} \right)^{\lambda_h - 1}}$$

Nested Logit

We can take the probabilities and re-write them slightly with the substitution that $\log\left(\sum_{k\in J_g} e^{V_{ik}}\right) \equiv IV_{ig}$:

$$\begin{aligned}
s_{ij} &= \frac{e^{V_{ij}/\lambda_g}}{\left(\sum_{k\in J_g} e^{V_{ik}/\lambda_g}\right)} \cdot \frac{\left(\sum_{k\in J_g} e^{V_{ik}/\lambda_g}\right)^{\lambda_g}}{\sum_{h=1}^{G}\left(\sum_{k\in J_h} e^{V_{ik}/\lambda_h}\right)^{\lambda_h}} \\
&= \underbrace{\frac{e^{V_{ij}/\lambda_g}}{\left(\sum_{k\in J_g} e^{V_{ik}/\lambda_g}\right)}}_{s_{ij|g}} \cdot \underbrace{\frac{e^{\lambda_g IV_{ig}}}{\sum_{h=1}^{G} e^{\lambda_h IV_{ih}}}}_{s_{ig}}
\end{aligned}$$

This is the decomposition into two logits that leads to the “sequential logit” story.

Nested Logit : Notes

- $\lambda_g = 1$ is the simple logit case (IIA)
- $\lambda_g \to 0$ implies that all consumers stay within the nest.
- $\lambda < 0$ or $\lambda > 1$ can happen and usually means something is wrong. These models are not generally consistent with RUM. (If you report one in your paper I will reject it).
- λ is often interpreted as a correlation parameter and this is almost true but not exactly!
- There are other extensions: overlapping nests, or three level nested logit.
- In general the hard part is understanding what the appropriate nesting structure is ex ante. Maybe for some problems this is obvious but for many not.

Nested Logit

In practice we end up with the following:

$$s_{ij} = s_{ij|g}(\theta) s_{ig}(\theta)$$

- Because the nested logit can be written as the within group share $s_{ij|g}$ and the share of the group s_{ig} we often explain this model as sequential choice
- First you pick a category, then you pick a product within a category.
- This is a sometimes helpful (sometimes unhelpful) way to think about this.
- We can also think about this imposing a block structure on the covariance matrix of ε_i
- You need to assign products to categories before you estimate and you can't make mistakes!

Convexity and Computation

Convexity

An optimization problem is convex if

$$\min_x f(\mathbf{x}) \quad s.t. \quad h(\mathbf{x}) \leq 0 \quad A\mathbf{x} = 0$$

- $f(\mathbf{x}), h(\mathbf{x})$ are convex (PSD second derivative matrix)
- Equality Constraint is affine

Some helpful identities about convexity

- Compositions and sums of convex functions are convex.
- Norms $||$ are convex, max is convex, log is convex
- $\log(\sum_{i=1}^{n} \exp(x_i))$ is convex.
- Fixed Points can introduce non-convexities.
- Globally convex problems have a unique optimum

Properties of Convex Optimization

- If a program is globally convex then it has a unique minimizer that will be found by convex optimizers.
- If a program is not globally convex, but is convex over a region of the parameter space, then most convex optimization routines find any local minima in the convex hull
- Convex optimization routines are unlikely to find local minima (including the global minimum) if they do not begin in the same convex hull as the optimum (starting values matter!).
- Most good commercial routines are clever about dealing with multiple starting values and handling problems that are well approximated by convex functions.
- Good Routines use information about sparseness of Hessian – this generally determines speed.

FIML Nested Logit Model is Non-Convex

$$\min_{\theta} \sum_{j} q_j \ln S_j(\theta) \quad \text{s.t.} \quad S_j(\theta) = \frac{e^{x_j\beta/\lambda}(\sum_{k\in g_l} e^{x_j\beta/\lambda})^{\lambda-1}}{\sum_{\forall l'}(\sum_{k\in g_l'} e^{x_j\beta/\lambda})^{\lambda}}$$

This is a pain to show but the problem is with the cross term $\frac{\partial^2 S_j}{\partial\beta\partial\lambda}$ because $\exp[x_j\beta/\lambda]$ is not convex.

A Simple Substitution Saves the Day: let $\gamma = \beta/\lambda$

$$\min_{\theta} \sum_{j} q_j \ln S_j(\theta) \quad \text{s.t.} \quad S_j(\theta) = \frac{e^{x_j\gamma}(\sum_{k\in g_l} e^{x_j\gamma})^{\lambda-1}}{\sum_{\forall l'}(\sum_{k\in g_l'} e^{x_j\gamma})^{\lambda}}$$

This is much better behaved and easier to optimize.

Nested Logit Model

	Original	**Substitution**	**No Derivatives**
Parameters	49	49	49
Nonlinear λ	5	5	5
Likelihood	2.279448	2.279448	2.27972
Iterations	197	146	352
Time	59.0 s	10.7 s	192s

Discuss Nelder-Meade

Computing Derivatives

A key aspect of any optimization problem is going to be computing the derivatives (first and second) of the model. There are some different approaches

- Numerical: Often inaccurate and error prone (why?) $f'(x) \approx \frac{f(x+h)-f(x-h)}{2h}$
- Pencil and Paper: this tends to be mistake prone – but often actually the fastest
- Automatic: Software brute forces through a chain rule calculation at every step (limited language). See `jax` in `Python` or `Optimization.jl` in `Julia`.
- Symbolic (Maple/Mathematica): software "knows" derivatives of certain objects and can do its own simplification. (limited language).

Thanks