

Maximum Likelihood Estimation of Discrete Choice Models Using PROC NLP

Donald M. Perry
GTE Business Development & Integration

OVERVIEW

- Illustrate how PROC NLP can be used to perform ML estimation of a wide range of Discrete Choice Models
- Why we chose to use PROC NLP ?
- Brief Review of Discrete Choice Models
 - model types
 - likelihood functions
 - SAS code used for estimation

Discrete Choice Models

- Probit or Logit specifications
- Binomial
- Multinomial
- Rank-ordered
- Pairwise (nested)

Why PROC NLP?

- Flexible
 - specify the equation to be maximized
 - can modify underlying probability function (normal or Weibull)
 - can modify functional form (binomial to rank-ordered)
- Direct estimates of variance/covariance
 - unlike PROC LP

ML Estimation in NLP

- Two “tricks”
 - minimizing $-L(B)$ is the same as maximizing $L(B)$
 - NLP using a non-linear least squares optimization
 - specify square root of $-L(B)$ as the objective function
 - NLP is minimizing sum squared residuals

Review of Discrete Choice Models

Binomial Logit Models

- Binomial Logit
 - Random Utility Maximization
 - Choice: GTE or some other TelCo
 - Choose GTE if $U(G) > U(N)$
 - $\Pr(i=GTE) = \Pr(U(G) > U(N))$
 - assuming iid extreme value
- Likelihood function is simply the product of all individuals
- SAS Code

Multinomial Logit Models

- Individual chooses one from among many alternatives
 - choose GTE if
 - $U(GTE) > U(AT\&T)$
 - $U(GTE) > U(MCI)$
 - $U(GTE) > U(Sprint)$
 - $U(GTE) > U(USWest)$
 - $U(GTE) > U(CableTV)$
- For an individual - choosing GTE
- Likelihood function
- SAS Code

Rank-Ordered Models

- Individual not only chooses one alternative from many; but also ranks all alternatives
 - $GTE > AT\&T > MCI > \dots$
- Random utility model for this ranking is
 - $U(G) > U(A) > U(M) > U(S) > U(USW)$
- Underlying probability model
- SAS Code