需要関数の推定 #2

Takahiro KONNO

BLP (1995)

Berry, Levinsohn, and Pakes (1995)

- 効用関数を以下のように設定。

$$u_{ij} = X_j \beta_i - \alpha_i p_j + \xi_j + \epsilon_{ij}$$

- 係数が消費者によって異なる。
- 消費者間で、α_i, β_i はiidな確率変数であることを仮定。
- よく使う仮定は、確率変数が多変量正規分布:

$$(\alpha_i, \beta_i)' \sim N\left((\bar{\alpha}, \bar{\beta})', \Sigma\right)$$

- $\bar{\alpha}, \bar{\beta}, \Sigma$ も推定しなければならない。

- Mean utility が $\delta_j = X_i\bar{\beta} - \bar{\alpha}p_j + \xi_j$ であることを用いると、

$$u_{ij} = \delta_j + \epsilon_{ij} + (\beta_j - \bar{\beta})X_j - (\alpha_i - \bar{\alpha})p_j$$

- このとき、Logit inversion はうまくいかない。
- Berry, Levinsohn, Pakes(1995)がこの解決方法を提示した。

- α_i, β_i に対して、消費者 *i* の選択確率は、

$$Pr(i,j) = \frac{exp(X_j\beta_i - \alpha_i p_j + \xi_j)}{1 + \sum_{j'=1}^{J} exp(X_{j'}\beta_i - \alpha_i p_{j'} + \xi_{j'})}$$

- 集計シェアは、

$$\tilde{s_j} = \iint Pr(i,j) dG(\alpha_i, \beta_i) = \iint \frac{exp(X_j\beta_i - \alpha_i p_j + \xi_j)}{1 + \sum_{j'=1}^{J} exp(X_{j'}\beta_i - \alpha_i p_{j'} + \xi_{j'})} dG(\alpha_i, \beta_i)$$

$$= \iint \frac{exp(\delta_j + (\beta_i - \bar{\beta})X_j - (\alpha_i - \bar{\alpha})p_j)}{1 + \sum_{j'=1}^{J} exp(\delta_{j'} + (\beta_i - \bar{\beta})X_{j'} - (\alpha_i - \bar{\alpha})p_{j'})} dG(\alpha_i, \beta_i)$$

$$\equiv \tilde{s_j}^{RC}(\delta_1, ..., \delta_J; \bar{\alpha}, \bar{\beta}, \Sigma)$$

- 集計シェアは、(α, β)の分布によってウェイト付された全消費者の $Pr(i,j)$ の積分となる。

- 今回は J 本の方程式に対し未知のパラメータが J+3 個ある。

  よって今回はBerry(1994)の Inversion を行うことはできない。

- 多重積分は計算が難しい。

  BLPではシミュレーションを用いて積分計算を行う。

- すべての$\bar{\alpha}, \bar{\beta}, \Sigma$に対して$\tilde{s_j}^{RC}$を計算できると仮定する。

- GMMで推定を行う。

  使用するモーメント条件は$E(\xi Z_m) = 0$

- $\theta \equiv (\bar{\alpha}, \bar{\beta}, \Sigma)$ とすると、標本モーメント条件は

$$m_{m,J}(\theta) \equiv \frac{1}{J}\sum_{i=1}^{J}(\delta_j - X_j\bar{\beta} + \bar{\alpha}p_j)Z_{mj}$$

- θは標本モーメント関数の二次式の最小化問題で

  推定できる。

$$\min_{\theta} Q_J(\theta) \equiv [m_{m,J}(\theta)]'_m W_J [m_{m,J}(\theta)]_m$$

- W_j は、M×Mのウェイト行列。

- $\delta_1, ..., \delta_J$ を計算するために "inner loop" を "outer loop" でネストした "nested" estimation algorithm を使う。

- outer loopでは、異なる値のパラメータで繰り返し計算を行う。$\hat{\theta}$ は、その時に検討されている値。

- inner loopでは、与えられた$\hat{\theta}$に対して$Q(\hat{\theta})$を求める。

1. まず、所与の $\hat{\theta}$ に対して連立方程式を解き、
   $\delta_1(\hat{\theta}), ..., \delta_J(\hat{\theta})$ の値を得る。

$$\begin{aligned} \hat{s_1} &= \tilde{s_1}^{RC}(\delta_1, ..., \delta_J; \hat{\theta}) \\ &\vdots \quad \vdots \\ \hat{s_J} &= \tilde{s_J}^{RC}(\delta_1, ..., \delta_J; \hat{\theta}) \end{aligned}$$

2. $\delta_1(\hat{\theta}), ..., \delta_J(\hat{\theta})$ の結果から、
   $Q(\hat{\theta}) \equiv [m_{m,J}(\hat{\theta})]'_m W_J [m_{m,J}(\hat{\theta})]_m$ を計算する。

- その後、outer loop に戻る。
  outer loop では、上式を最小にする $\hat{\theta}$ の値を見つけるまでサーチする。

Note: 識別のための必要条件

- 識別のための必要条件は、

$$M = dim(\vec{Z}) \geq dim(\theta) > dim(\alpha, \beta) = dim([X, p])$$

- Coefficients Σ がランダム係数と独立でないので、

  もし価格の内生性の問題がないとしても、

  パラメータを推定するためには操作変数が必要。

Estimation Equilibrium:
Both demand and supply side

- Random coefficient モデルに供給サイドを導入する。
- 限界費用は一定。コスト要因 γ に対して線形を仮定する。

$$C_1^j = c^j \equiv w_j \gamma + \omega_j$$

- 最適反応の条件は、

$$D^j + \sum_{j' \in \mathcal{K}} \frac{\partial D^{j'}}{\partial p_j} (p_{j'} - c^j) = 0$$

- $E(\omega U) = 0$ となるような操作変数 U_j を持っていると仮定。この操作変数はDemand shifter である。

- Objective function に供給サイドのモーメント条件を追加。

$$Q(\theta,\gamma) = G_J(\theta,\gamma)' W_j G_J(\theta,\gamma)$$

- G_J は、M+N 次元ベクトル。

$$G_J(\theta,\gamma) \equiv \begin{bmatrix} \frac{1}{J}\sum_{i=1}^{J}(\delta_j - X_j\bar{\beta} + \bar{\alpha}p_j)z_{1j} \\ \vdots \\ \frac{1}{J}\sum_{i=1}^{J}(\delta_j - X_j\bar{\beta} + \bar{\alpha}p_j)z_{Mj} \\ \frac{1}{J}\sum_{i=1}^{J}(c_j(\theta) - w_j\gamma)u_{1j} \\ \vdots \\ \frac{1}{J}\sum_{i=1}^{J}(c_j(\theta) - w_j\gamma)u_{Nj} \end{bmatrix}$$

- Mは需要サイドの操作変数の数。

  Nは供給サイドの操作変数の数。

$$M + N \geq dim(\theta) + dim(\gamma)$$

- Inner loop では、所与の $\hat{\theta}, \hat{\gamma}$ に対して、$Q(\hat{\theta}, \hat{\gamma})$ を評価する。

1. まず、所与の $\hat{\theta}$ に対して、$\delta_1(\hat{\theta}), ..., \delta_J(\hat{\theta})$ を求める。
2. $\delta_1(\hat{\theta}), ..., \delta_J(\hat{\theta})$ で、

$$\vec{s_j}^{RC}(\hat{\theta}) \equiv \left(\tilde{s_1}^{RC}(\delta(\hat{\theta})), ..., \tilde{s_J}^{RC}(\delta(\hat{\theta}))\right)'$$

と、偏微係数行列

$$\mathrm{D}(\hat{\theta}) = \begin{pmatrix} \frac{\partial {s_1}^{RC}(\delta(\hat{\theta}))}{\partial p_1} & \frac{\partial {s_1}^{RC}(\delta(\hat{\theta}))}{\partial p_2} & \cdots & \frac{\partial {s_1}^{RC}(\delta(\hat{\theta}))}{\partial p_J} \\ \frac{\partial {s_2}^{RC}(\delta(\hat{\theta}))}{\partial p_1} & \frac{\partial {s_2}^{RC}(\delta(\hat{\theta}))}{\partial p_2} & \cdots & \frac{\partial {s_2}^{RC}(\delta(\hat{\theta}))}{\partial p_J} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial {s_J}^{RC}(\delta(\hat{\theta}))}{\partial p_1} & \frac{\partial {s_J}^{RC}(\delta(\hat{\theta}))}{\partial p_2} & \cdots & \frac{\partial {s_J}^{RC}(\delta(\hat{\theta}))}{\partial p_J} \end{pmatrix}$$

を計算する。

- Multinomial Logit ならばこれらの偏微係数は、

$$\frac{\partial s_j}{\partial p_k} = \begin{cases} -\alpha s_j(1 - s_j) & \text{for } j = k \\ -\alpha s_j s_k & \text{for } j \neq k \end{cases}$$

3. 供給サイドの最適反応の下式から $c_1(\hat{\theta}), \ldots, c_J(\hat{\theta})$ を求める。

$$\vec{s_j}^{RC}(\hat{\theta}) + \mathrm{D}(\hat{\theta}) * \begin{pmatrix} p_1 - c^1 \\ \vdots \\ p_J - c^J \end{pmatrix} = 0$$

4. $G(\hat{\theta}, \hat{\gamma})$ を計算する。

- *s = 1,2,...,S* 個のシミュレーション

1. u_1^s, u_2^s を独立に $N(0,1)$ からドローする。
2. 所与の推定パラメータ $\hat{\alpha}, \hat{\beta}, \hat{\Sigma}$ に対して、
   以下の変換を行い (u_1^s, u_2^s) を $N((\hat{\alpha}, \hat{\beta}), \hat{\Sigma})$ からのドローに
   変換する。

$$\begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} = \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \end{pmatrix} + \hat{\Sigma}^{1/2} \begin{pmatrix} u_1^s \\ u_2^s \end{pmatrix}$$

- $\hat{\Sigma}^{1/2}$は、行列 $\hat{\Sigma}$ のCholesky分解。

  正則対称行列 Γ のCholesky分解は、$G'G = \Gamma$ を満たすような三角行列である。ここでは下側の三角行列を使う。

- ドローを通して、積分を標本平均で近似することができる。

$$\mathcal{E}(\hat{\hat{\alpha}}, \hat{\hat{\beta}}, \hat{\Sigma}) \approx \frac{1}{S}\sum_{s=1}^{S} \frac{exp(\delta_j + (\beta^s - \hat{\hat{\beta}})X_j - (\alpha^s - \hat{\hat{\alpha}})p_j)}{1 + \sum_{j'=1}^{J} exp(\delta_{j'} + (\beta^s - \hat{\hat{\beta}})X_{j'} - (\alpha^s - \hat{\hat{\alpha}})p_{j'}}$$

- 所与の $\hat{\hat{\alpha}}, \hat{\hat{\beta}}, \hat{\Sigma}$ に対してこの近似が正確であることは、大数の法則によって保証される。

積分シミュレーション

- シミュレーションの原則：

  期待値は、標本平均で近似することができる。

$$\mathcal{E} \equiv E_G\left[\frac{exp(\delta_j + (\beta_i - \bar{\beta})X_j - (\alpha_i - \bar{\alpha})p_j)}{1 + \sum_{j'=1}^{J} exp(\delta_{j'} + (\beta_i - \bar{\beta})X_{j'} - (\alpha_i - \bar{\alpha})p_{j'}}|\bar{\alpha}, \bar{\beta}, \Sigma\right]$$

- α_i, β_i は確率変数で、これは多変量正規分布 $N((\bar{\alpha}, \bar{\beta})', \Sigma)$

  から取り出せると仮定する。

BLPアルゴリズム

Random Coefficient (BLP)

- 間接効用関数

$$u_{ijt} = x_{jt}\beta_i + \alpha_i p_{jt} + \xi_{jt} + \epsilon_{ijt}$$

$$\begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} = \underbrace{\begin{pmatrix} \hat{\hat{\alpha}} \\ \hat{\hat{\beta}} \end{pmatrix}}_{\theta_1} + \underbrace{\Pi D_i + \Sigma_{\nu_i}}_{\theta_2=(\Pi,\Sigma_{\nu_i})}$$

- 間接効用関数を2つの部分に分ける。

$$u_{ijt} = \underbrace{\delta(x_{jt}, p_{jt}, \xi_{jt}; \theta_1)}_{Mean} + \underbrace{\mu(x_{jt}, p_{jt}, D_i, \nu_i; \theta_2)}_{HH Deviations from Mean} + \epsilon_{ijt}$$

- $\delta_{jt} = x_{jt}\beta + \alpha p_{jt} + \xi_{jt}$ と、$\mu_{ijt} = (p_{jt} \ x_{jt})(\Pi D_i + \Sigma_{\nu_i})$
- 解析的にδ_{jt}を計算することはできない。

BLP Algorithm

1. 消費者のセットに対してν_i（と必要ならD_i）をドローする。Homogeneous Logit からδ_{jt}の初期値を得る。

2. シェアを予測する。
   - 所与のθ_2に対して HH Deviations from Mean を計算する。
     $\mu(x_{jt}, p_{jt}, D_i, \nu_i; \theta_2)$
   - 所与の平均効用δ_tとθ_2に対して予測シェアを計算する。
     $\tilde{\sigma}_j(\delta_t, x_t, p_t; \theta_2)$

3. Contraction Mapping : 所与の非線形パラメータθ_2に対して、
$s_{jt} = \sigma_j(\delta_t, x_t, p_t; \theta_2)$を満たすような$\delta_t$を探索する。
4. δ_t から解析的に線形パラメータを推定する。
GMMの目的関数$Q(\theta_2)$を求める。
5. $Q(\theta_2)$ を θ_2 に関して最小化する。
θ_2 の試行ごとに、各Step2-4 をネストする。

Step1: Simulating HH draws

- ν_i とD_i はそれぞれの分布 $F^*_\nu(\nu), F^*_D(D)$ から発生させる。

$$\begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} = \underbrace{\begin{pmatrix} \hat{\bar{\alpha}} \\ \hat{\bar{\beta}} \end{pmatrix}}_{\theta_1} + \underbrace{\Pi D_i + \Sigma_{\nu_i}}_{\theta_2 = (\Pi, \Sigma_{\nu_i})}$$

- Σ はHomogeneous Logitの分散行列のCholesky分解。

  Π はHomogeneous Logitの係数行列。

Step2: Computing market shares

- ロジット、入れ子ロジットのときは、解析的に解ける。

- ランダム係数の場合は、シミュレーションで積分計算。

$$\sigma_{jt} \approx \frac{1}{NS} \sum_{i=1}^{NS} \frac{exp(\delta_{jt} + (p_{jt} \quad x_{jt})(\Pi D_i + \nu_i)}{\sum_{k=0}^{J} exp(\delta_{kt} + (p_{kt} \quad x_{kt})(\Pi D_i + \nu_i)}$$

Step3: Contraction Mapping

- ロジットと入れ子ロジットでは、平均効用deltaを解析的に求められる。

$$ln(s_{jt}) - ln(s_{0t}) = \delta_{jt} - \delta_{0t} = X_j'\beta - \alpha p_j + \xi_j + \epsilon_{ij}$$

$$ln(s_{jt}) - ln(s_{0t}) = \delta_{jt} - \delta_{0t} = X_j'\beta - \alpha p_j + \sigma ln(s_{jt}/s_{Gt}) + \xi_j + \epsilon_{ij}$$

- ランダム係数では、Contraction Mapping が必要。収束するまで($\delta_t^{h+1} = \delta_t^h < Tolerance$)繰り返し計算をする。

$$\delta_t^{h+1} = \delta_t^h + ln(s_t) - ln(\sigma_t^h(\delta_t^h, x_t, p_t, F_{NS}; \theta_2), h = 1, ..., H$$

Step4: Estimate parameters in two steps

A. 所与の $\delta_t(\theta_2)$ に対して、θ_1 を推定しξ_{jt}を計算する。

$$\theta_1 = (X'ZWZ'X)^{-1}(X'ZWZ'\delta)$$

$$\xi_{jt}(\theta_2) = \delta_{jt} - x_{jt}\beta - \alpha p_{jt}$$

B. GMM目的関数を求める。$\xi(\theta)'ZWZ'\xi(\theta)$

ただしWはGMM weight matrixで、

$$(E((\xi(\theta)'Z)'(Z'\xi(\theta))))^{-1}$$

Wはθに依存。初期値W=I として繰り返し更新していく。

実行上は、Homogeneous Logit の結果を使えばよい。

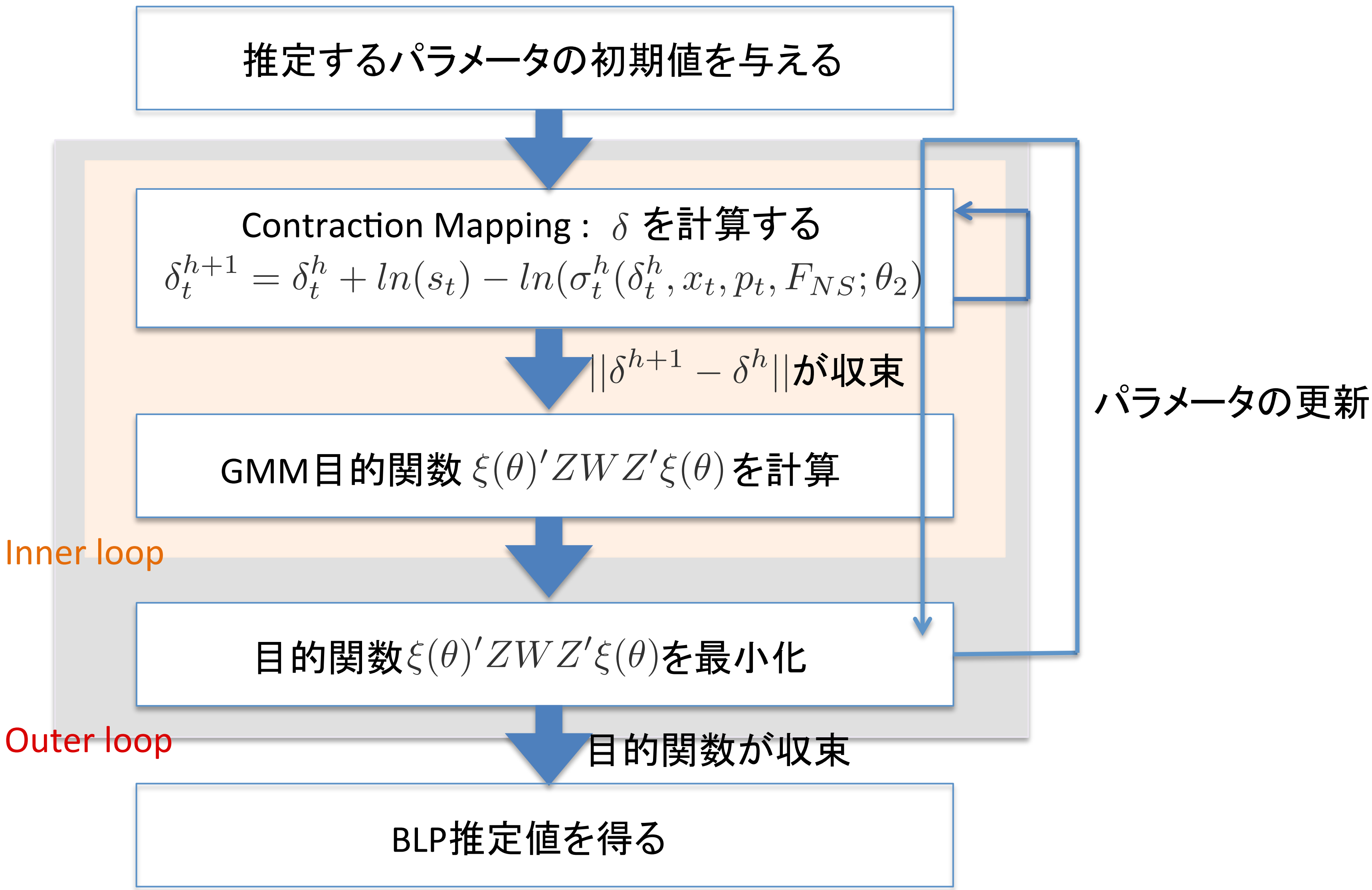
Step5: Optimizing over θ_2

- Step4 で得られた GMM目的関数を最小化する。

$$\min_{\theta_2} \xi(\theta)' Z W Z' \xi(\theta)$$

- 最適化ツールは関数を繰り返し呼び出すので、

  Step2 – Step4 を関数内で実行すればネストされる。


推定するパラメータの初期値を与える
Contraction Mapping： δ を計算する
$\delta_t^{h+1} = \delta_t^h + ln(s_t) - ln(\sigma_t^h(\delta_t^h, x_t, p_t, F_{NS}; \theta_2))$
$||\delta^{h+1} - \delta^h||$が収束
GMM目的関数 $\xi(\theta)'ZWZ'\xi(\theta)$ を計算
Inner loop
目的関数$\xi(\theta)'ZWZ'\xi(\theta)$を最小化
パラメータの更新
Outer loop
目的関数が収束
BLP推定値を得る

nevofiles

	mean	sigma	income	income^2	age	child
constant						
	-2.6797	0.3195	4.1266	0	0.2311	0
	0.2249	0.1171	0.9766	0	0.3442	0
price						
	-30.5981	2.3351	16.4214	-0.8937	0	2.9625
	7.1174	0.8487	156.0640	8.0705	0	3.5366
sugar						
	0.1197	0.0158	-0.2319	0	0.0576	0
	0.2249	0.0093	0.0355	0	0.0235	0
mushy						
	-0.2941	0.2335	1.4380	0	-0.8770	0
	0.0130	0.1427	0.4847	0	0.4988	0

```
***********************************************************************

GMM objective:  23.4556
MD R-squared:  0.4976
MD weighted R-squared:  0.062216
run time (minutes):  0.48567
```