14 Sep. 2020 @ ZOOM

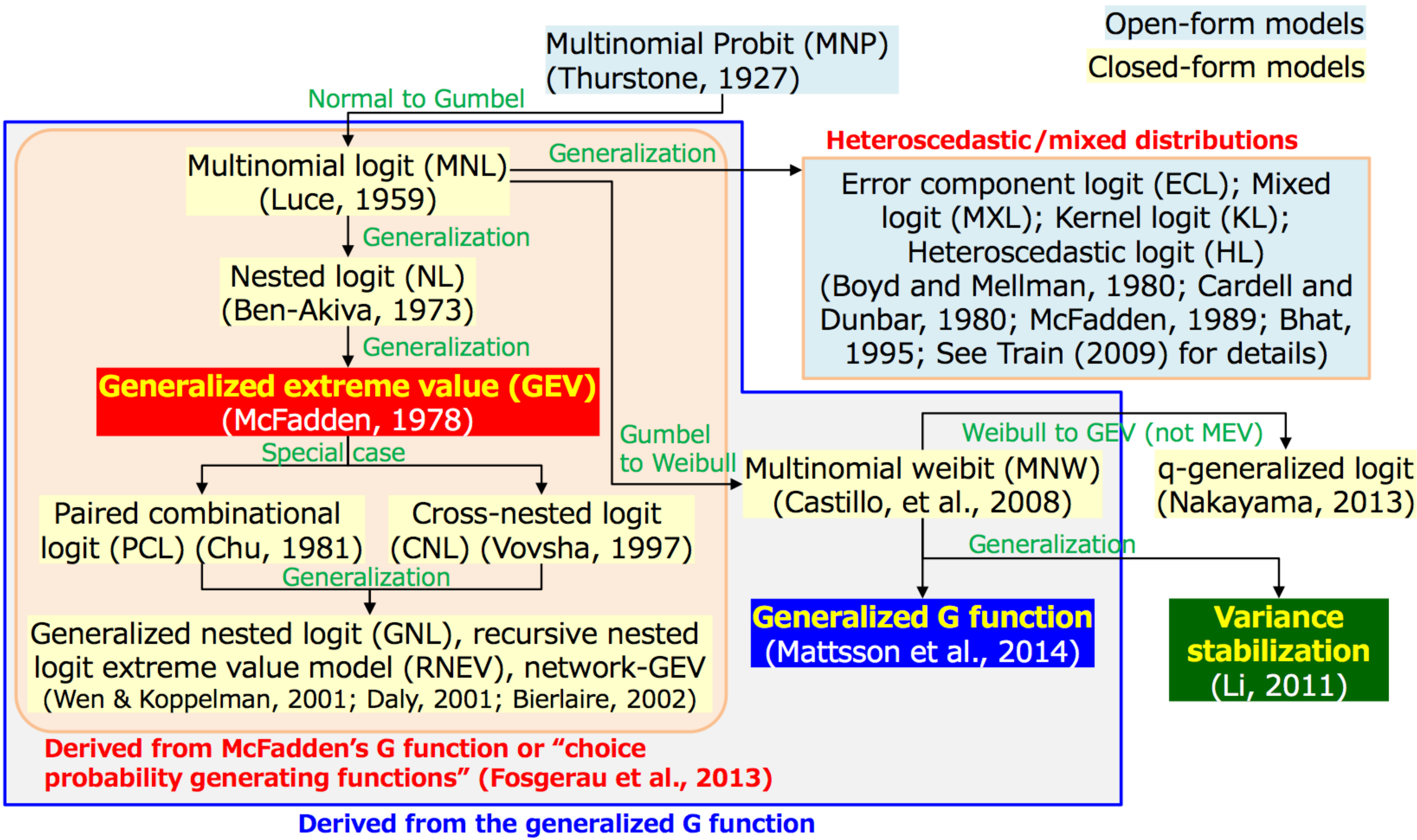
Machine Learning and Advanced Estimation Method

Tokyo University of Science

Hideki YAGINUMA

yaginuma@rs.tus.ac.jp

1. Overview of DCM



By Chikaraishi(Hiroshima Univ)

1. Closed-form vs Open-form

GEV model (Closed-form)

Multinomial Logit (MNL)

$$P(i)=\frac{\exp(\mu V_i)}{\sum_{j\in C}\exp(\mu V_j)}$$

- Luce(1959), McFadden(1974)
- Not consider correlation of choice alternatives (IIA)
- Easy and fast estimation
- High operability
  (easy evaluation for new additional choice alternative ⇒ benefit of IIA)

Non-GEV model (Open-form)

Multinomial Probit (MNP)

$$P(i)=\int_{\varepsilon_1=-\infty}^{\varepsilon_i+V_i-\varepsilon_1}\cdots\int_{\varepsilon_i=-\infty}^{\infty}\cdots\int_{\varepsilon_J=-\infty}^{\varepsilon_i+V_i-\varepsilon_J}\phi(\varepsilon)d\varepsilon_J\cdots d\varepsilon_1$$

$$\phi(\varepsilon)=\frac{1}{\left(\sqrt{2\pi}\right)^{J-1}|\Sigma|^{1/2}}\exp\left(-\frac{1}{2}\varepsilon\Sigma^{-1}\varepsilon'\right)$$

- Thurstone(1927)
- Consider correlation of choice alternatives based on Variance-Covariance matrix
- Difficult estimation
  (need calculation of multi-dimensional interrelation depend on N of alternatives')

Non-GEV model has high power of expression, however parameter estimation cost is high.

Mixed Loigt (Train 2000)

High flexible model structure by two error term.

Utility function

$$U_i = V_i + \eta_i + \nu_i$$

ν dist.: assume any G function

- IID Gamble (Logit Kernel) ⇒ MNL
- any G function (GEV Kernel) ⇒ NL, PCL, CNL, GNL…

η dist.: basically assume "*Normal dist.*"

In the case of normal distribution takes a non-realistic value, it can assume a variety of probability distribution (triangular distribution, cutting normal distribution, lognormal distribution, Rayleigh distribution, etc.).

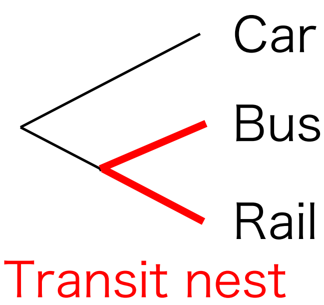
- Error Component: approximate to any GEV model
- Random Coefficient: Consider the heterogeneity

1. Error Component Model

Approximation of Nested Logit (NL)

Describe the nest (covariance) using structured $\boldsymbol{\eta}$.

Ex: model choice



Normal ⇒ nest

$$U_{car} = \beta \mathbf{X}_{car} + \qquad + \nu_{car}$$
$$U_{bus} = \beta \mathbf{X}_{bus} + \sigma_{transit}\eta_{transit} + \nu_{bus}$$
$$U_{rail} = \beta \mathbf{X}_{rail} + \sigma_{transit}\eta_{transit} + \nu_{rail}$$

IID Gamble ⇒ Logit

Choice prob. (open-form)

$$P_{rail} = \int_{\eta_{transit}} \frac{e^{V_{rail}+\sigma_{transit}\eta_{transit}}}{e^{V_{car}} + e^{V_{bus}+\sigma_{transit}\eta_{transit}} + e^{V_{rail}+\sigma_{transit}\eta_{transit}}} f\left(\eta_{transit}\right) d\eta_{transit}$$

$$\eta_{transit} \approx N(0,1)$$

Choice prob. (Simulated)

$$P_{rail} = \frac{1}{N}\sum_{N} \frac{e^{V_{rail}+\sigma_{transit}\eta^{N}_{tranist}}}{e^{V_{car}} + e^{V_{bus}+\sigma_{transit}\eta^{N}_{tranist}} + e^{V_{rail}+\sigma_{transit}\eta^{N}_{tranist}}}$$

1. Random Coefficient Model

Approximation of unobservable heterogeneity

Assume the heterogeneity of parameter

⇒In the case of parameter following Normal dist., we estimate the dist.'s hyper-parameter (mean and variance).

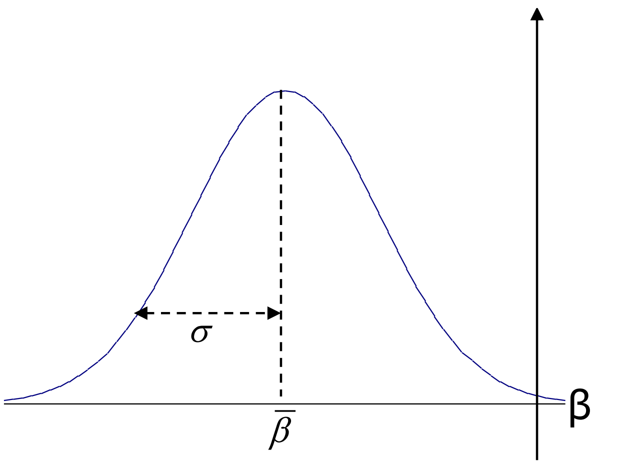
$$U_{car,n} = \beta_n T_{car,n} + \nu_{car,n}$$

$$\beta_n \approx N\left(\bar{\beta}, \sigma^2\right)$$

$$U_{car,n} = \bar{\beta} T_{car,n} + \sigma \eta_n T_{car,n}$$

$$U_{bus,n} = \bar{\beta} T_{bus,n} + \sigma \eta_n T_{bus,n}$$

$$U_{rail,n} = \bar{\beta} T_{rail,n} + \sigma \eta_n T_{rail,n}$$



$\eta_n \approx N(0,1)$ $\bar{\beta}, \sigma : unknown\ parameter$

Hyper-parameter can describe using observable variables

$$\bar{\beta}_n = \gamma_0 + \gamma_1 income_n$$

β depend on observable income variable

1. Difficulty of Estimation

Why estimation methods is important ?

- ✓ Advance GEV model (CNL, GNL, n-GEV…）has many parameter.
  ⇒ Convergence becomes unstable (Hessian passed away…)
- ✓ non-GEV model requires multiple integral calculations .
  ⇒ ML estimation cannot be used
- ✓ Stricture of utility function (non-liner, complex distribution)
- ✓ Dynamic choice behavior (Dynamic Programing:DP)
- ✓ Interaction between decision-maker (Endogeneity）
- ✓ high dimensional data ⇒ N of Sample < N of Variable

The analyst needs to select an appropriate estimation method corresponding to the model !

1. Overview of Estimation

To solve Integral

Maximum Likelihood estimation

To Consider Dynamics Interaction

Bayesian estimation

General method

Structural estimation
- ✓ Heckman's 2-step
- ✓ Pseudo Likelihood
- ✓ nested Fixed Point (NFXP)
- ✓ Nestef Pseudo Likelihood …
- ✓ MPEC

Numerical method
- ✓ Newton-Cotes rules
- ✓ Simpson's rule

Low-dimensional

Analytical approximation
- ✓ Series approximation
- ✓ Clark approximation
- ✓ MACML

Random approximation
- ✓ Maximum Simulated Likelihood (Monte Carlo integral, GHK…)
- ✓ MCMC (Gibbs, MH, HM…)

High-dimensional

Machine Learning (ML)

Neural network, (reverse) Reinforcement learning, Sparse modeling, Gaussian Process…

⇒ Several methodologies are useful for DCM parameter estimation !

2. Bayesian Estimation

- Model parameter estimation based on Bayes theory

Posterior Dist. Likelihood Priori Dist.

$$\pi(\theta \mid D) \propto f(D \mid \theta) \cdot \pi(\theta)$$

θ: Parameter dist.
D: Data

Ex: Estimate the average value of $\boldsymbol{\theta}$

- Likelihood: Binominal distribution
- Priori distribution: Exponential distribution.

Likelihood × Priori Dist. = Posterior Dist. (Dist. of average $\boldsymbol{\theta}$)

$$nCr\theta^{r}(1-\theta)^{n-r} \times \lambda e^{-\lambda\theta} = \frac{\int_0^1 \theta \cdot \theta^{r}(1-\theta)^{n-r}\lambda e^{-\lambda\theta} d\theta}{\int_0^1 \theta^{r}(1-\theta)^{n-r}\lambda e^{-\lambda\theta} d\theta}$$

Analytical formula is too complex !

2. Parameter estimation

To estimate the model parameter based on Bayes statistic, should be considered method of approximation of multi-dimensional integrals.

❖Conjugate distribution methods

Analytical approximations using property of conjugate dist..

- Model: change (= approximate well-known distribution)
- Calculation cost: Low

❖Markov chain Monte Carlo(MCMC) methods

Random approximations using computational technique.

- Model: not change (= flexible distribution is available)
- Calculation cost: High

2. Conjugate Distribution

- If the posterior distributions are in the "***same family***" as the prior distribution, the prior and posterior are then called **conjugate distributions**, and the prior is called a conjugate prior for the likelihood function.
- A conjugate prior is an algebraic convenience giving a *closed-form expression* for the posterior. Otherwise a difficult numerical integration may be necessary.

Example of conjugate distribution (Discrete distribution)

Likelihood	Model Parameter	Prior Dist.	Prior parameter	Posterior Dist.
Binomial	p (probability)	Beta	α, β	Beta
Poisson	λ (rate)	Gamma	κ, θ	Gamma
Categorical	p, k (N of categories)	Dirichlet	α	Dirichlet
Multinomial	p, k (N of categories)	Dirichlet	α	Dirichlet

2. MCMC

- Markov chain Monte Carlo (MCMC) methods are a class of algorithms for sampling from a probability distribution based on constructing a Markov chain that has the desired distribution as its equilibrium distribution.

  - *Gibbs sampling*: Requires all the conditional distributions of the target distribution to be sampled exactly. It is popular partly because it does not require any 'tuning'.

  - *Metropolis–Hastings algorithm*: Generates a random walk using a proposal density and a method for probabilistic rejecting some of the proposed moves.

  - *Other MCMC methods*: Slice sampling, Multiple-try Metropolis, Reversible-jump, Hybrid Monte Carlo, Hamiltonian Monte Carlo

2. Gibbs Sampling Algorithm

❖Gibbs sampling is that given a multivariate distribution it is simpler to sample from a conditional distribution than to marginalize by integrating over a joint distribution.

STEP0: Set initial values

- Iterator $\boldsymbol{i} = 0$
- Maximum iteration number
- Period of "burn-in"
- Initial value vector

$$X^{(0)} = \left(x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)}\right)$$

Period of to stabilize calculation

STEP1: Sampling

- Iterator $\boldsymbol{i} := \boldsymbol{i} + 1$
- sample each variable $x_j^{(i)}$ from the conditional distribution

$$p\left(x_j \mid x_1^{(i)}, \ldots, x_{j-1}^{(i)}, x_{j+1}^{(i-1)}, \ldots, x_n^{(i-1)}\right)$$

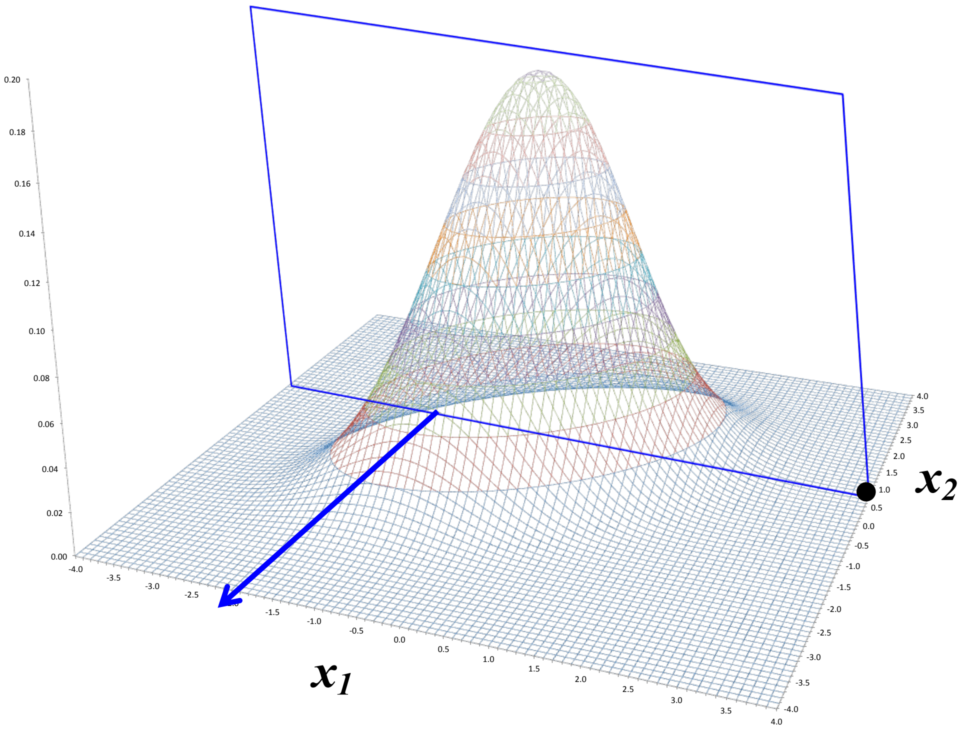
sample each variable from the distribution of that variable conditioned on all other variables, making use of the most recent values and updating the variable with its new value as soon as it has been sampled.

STEP2: Repeat

- Repeat STEP1 until reach to max iteration
- If finish, cut the data include period of "burn-in"

2. Example of Gibbs Sampling

Ex: Sampling from Bivariate standard normal distribution



STEP0: Set initial values

$$X^{(0)} = \left(x_1^{(0)}, x_2^{(0)}\right)$$

STEP1: Sampling

$$x_1^{(i)} \sim N\left(\rho x_2^{(i-1)}, \sqrt{1-\rho^2}\right)$$

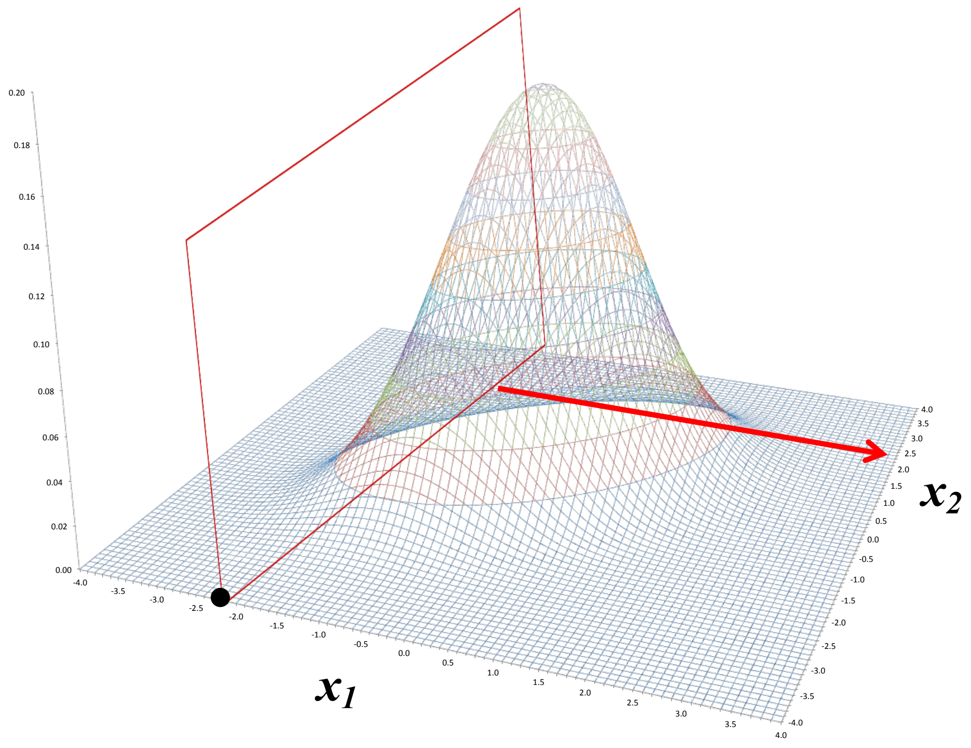
$$x_2^{(i)} \sim N\left(\rho x_1^{(i)}, \sqrt{1-\rho^2}\right)$$

*Random number based on Bivariate normal distribution

$$x \sim N\left(\mu_x, \sqrt{\sigma_x^2}\right) \qquad y \sim N\left(\mu_y + \rho \frac{\sigma_y^2}{\sigma_x^2}\left(x-\mu_x\right), \sqrt{\left(1-\rho^2\right)\sigma_x^2}\right)$$

2. Example of Gibbs Sampling

Ex: Sampling from Bivariate standard normal distribution



STEP0: Set initial values

$$X^{(0)} = \left(x_1^{(0)}, x_2^{(0)}\right)$$

STEP1: Sampling

$$x_1^{(i)} \sim N\left(\rho x_2^{(i-1)}, \sqrt{1-\rho^2}\right)$$

$$x_2^{(i)} \sim N\left(\rho x_1^{(i)}, \sqrt{1-\rho^2}\right)$$

*Random number based on Bivariate normal distribution

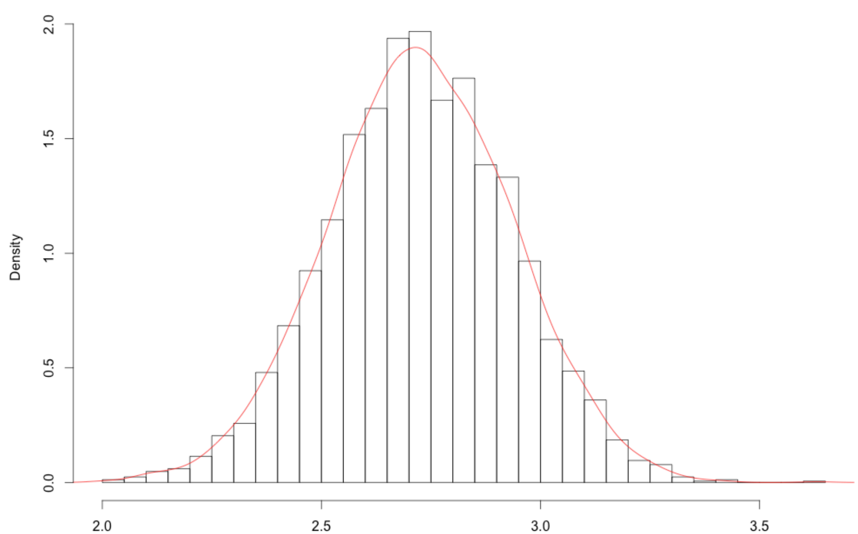
$$x \sim N\left(\mu_x, \sqrt{\sigma_x^2}\right) \qquad y \sim N\left(\mu_y + \rho\frac{\sigma_y^2}{\sigma_x^2}\left(x-\mu_x\right), \sqrt{\left(1-\rho^2\right)\sigma_x^2}\right)$$

2. Parameter Estimation

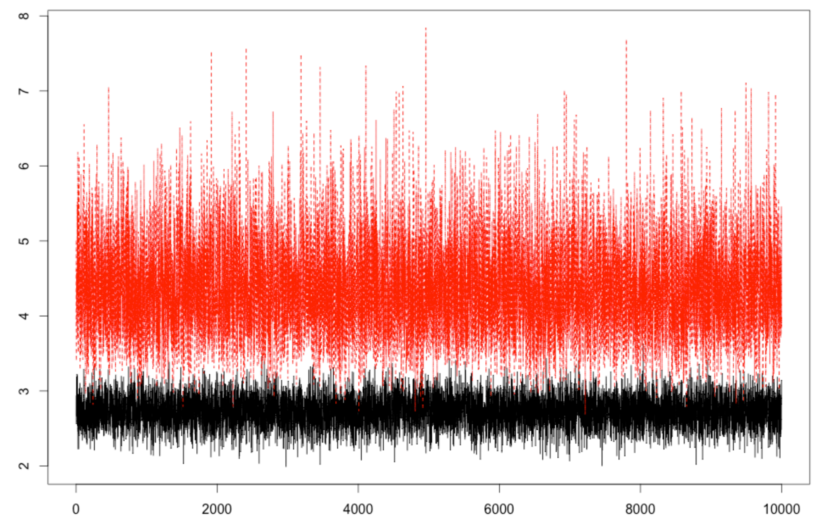
Example Data

- Data: Artificial N(mean=3, SD=2)
- Estimate arguments (mean and Sigma) in Likelihood assumed Normal dist.
- Prior and Posterior use conjugate dist.
  ⇒ mean: Normal dist. Sigma: Gamma dist.

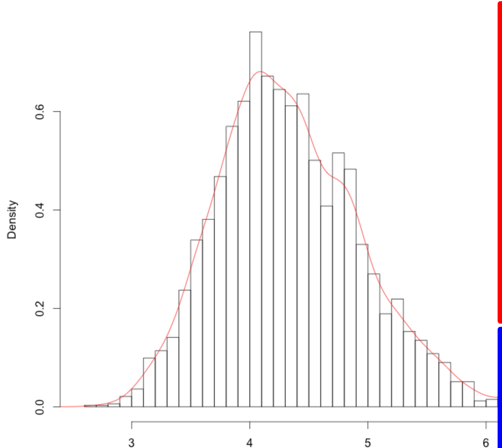
$$\mu \sim N\left(\mu_0, \kappa_0^2\right), \quad \sigma^2 \sim Gamma\left(\frac{\nu_0}{2}, \frac{2}{s_0}\right)$$



Sample path



Estimation results




```
> bmean
[1] 3.146730 4.266865
> bsd
[1] 0.2056611 0.6032165
> QB
          [,1]     [,2]
2.5%  2.736511 3.256377
50%   3.145574 4.213542
97.5% 3.553047 5.586441
> mX
[1] 3.150161
> var(X)
[1] 4.17942
>
```

3. MACML (1)

Bhat, C.R.: The maximum approximate composite marginal likelihood (MACML) estimation of multinomial probit-based unordered response choice models, *Transportation Research Part B: Methodological*, Vol.45, No.7, pp.923-939, 2011.

❖ Propose a simple and fast method for estimating parameters of open-from models (c.f. MNP, MXL)

❖ MACML estimation consists of two techniques

- Analytic approximation method for MVNCD
- Parameter estimation by CML

❖ Compared with the normal estimation method (MSL), the calculation time is about 38 times faster (66.09 → 1.96), and the bias of the estimated value is 7.3 points lower (9.8% → 2.5%).

※1 MVNCD: Multi-Variate standard Normal Cumulative Distribution
※2CML: Composite Marginal Likelihood

3. MACML (2)

Analytic approximation method for MVNCD

⇒ Approximate a multivariate normal dist. by a product of univariate dist.

【Setting1: decomposition of distribution】

$$\Pr(\boldsymbol{W} < \boldsymbol{w}) = \Pr(W_1 < w_1, W_2 < w_2, W_3 < w_3, \ldots, W_I < w_I).$$

W: multivariate normal dist.

Decompose joint probability into product of distributions as follows

$$\Pr(\boldsymbol{W} < \boldsymbol{w}) = \underline{\Pr(W_1 < w_1, W_2 < w_2)} \times \prod_{i=3}^{I} \underline{\Pr(W_i < w_i | W_1 < w_1, W_2 < w_2, W_3 < w_3, \ldots, W_{i-1} < w_{i-1})}.$$

Bivariate marginal distribution   Univariate conditional distribution（I>3）

【Setting 2 : covariance matrix by indicator **I**】

$$\tilde{\mathbf{I}}_i = \begin{cases} 1 & W_i < w_i \\ 0 & otherwise \end{cases} \qquad E(\tilde{I}_i) = \Phi(w_i)$$

Evaluate the expected value of I by univariate cumulative normal dist. Φ

$$Cov(\tilde{I}_i, \tilde{I}_i) = Var(\tilde{I}_i) = \Phi(w_i) - \Phi^2(w_i) = \Phi(w_i)[1 - \Phi(w_i)],$$

$$\mathrm{Cov}(\tilde{I}_i, \tilde{I}_j) = E(\tilde{I}_i\tilde{I}_j) - E(\tilde{I}_i)E(\tilde{I}_j) = \Phi_2(w_i, w_j, \rho_{ij}) - \Phi(w_i)\Phi(w_j), i \neq j$$

Integrate Setting1 and 2

$$\Pr(W_i < w_i | W_1 < w_1, W_2 < w_2, W_3 < w_3, \ldots, W_{i-1} < w_{i-1}) = E(\tilde{I}_i | \tilde{I}_1 = 1, \tilde{I}_2 = 1, \tilde{I}_3 = 1, \ldots, \tilde{I}_{i-1} = 1).$$

3. MACML (3)

【assume liner regression model】

$$\Pr(W_i < w_i | W_1 < w_1, W_2 < w_2, W_3 < w_3, \ldots, W_{i-1} < w_{i-1}) = E(\tilde{I}_i | \tilde{I}_1 = 1, \tilde{I}_2 = 1, \tilde{I}_3 = 1, \ldots, \tilde{I}_{i-1} = 1).$$

$$\tilde{I}_i - E(\tilde{I}_i) = \boldsymbol{\alpha}' [\tilde{\boldsymbol{I}}_{<i} - E(\tilde{\boldsymbol{I}}_{<i})] + \tilde{\eta},$$

error term

※ $\tilde{\boldsymbol{I}}_{<i} = (\tilde{I}_1, \tilde{I}_2, \ldots \tilde{I}_{i-1})$

$$\hat{\boldsymbol{\alpha}} = \boldsymbol{\Omega}_{<i}^{-1} \cdot \boldsymbol{\Omega}_{i,<i},$$ parameter

$$\boldsymbol{\Omega}_{<i} = \mathrm{Cov}(\boldsymbol{I}_{<i}, \boldsymbol{I}_{<i}) = \begin{bmatrix} \mathrm{Cov}(\tilde{I}_1, \tilde{I}_1) & \mathrm{Cov}(\tilde{I}_1, \tilde{I}_2) & \mathrm{Cov}(\tilde{I}_1, \tilde{I}_3) & \cdots & \mathrm{Cov}(\tilde{I}_1, \tilde{I}_{i-1}) \\ \mathrm{Cov}(\tilde{I}_2, \tilde{I}_1) & \mathrm{Cov}(\tilde{I}_2, \tilde{I}_2) & \mathrm{Cov}(\tilde{I}_2, \tilde{I}_3) & \cdots & \mathrm{Cov}(\tilde{I}_2, \tilde{I}_{i-1}) \\ \mathrm{Cov}(\tilde{I}_3, \tilde{I}_1) & \mathrm{Cov}(I_3, I_2) & \mathrm{Cov}(\tilde{I}_3, \tilde{I}_3) & \cdots & \mathrm{Cov}(\tilde{I}_3, \tilde{I}_{i-1}) \\ \vdots & & & & \\ \mathrm{Cov}(\tilde{I}_{i-1}, \tilde{I}_1) & \mathrm{Cov}(\tilde{I}_{i-1}, \tilde{I}_2) & \mathrm{Cov}(\tilde{I}_{i-1}, \tilde{I}_3) & \cdots & \mathrm{Cov}(\tilde{I}_{i-1}, \tilde{I}_{i-1}) \end{bmatrix}, \quad \boldsymbol{\Omega}_{i,<i} = \mathrm{Cov}(\boldsymbol{I}_{<i}, \boldsymbol{I}_i) = \begin{bmatrix} \mathrm{Cov}(\tilde{I}_i, \tilde{I}_1) \\ \mathrm{Cov}(\tilde{I}_i, \tilde{I}_2) \\ \mathrm{Cov}(\tilde{I}_i, \tilde{I}_3) \\ \vdots \\ \mathrm{Cov}(\tilde{I}_i, \tilde{I}_{i-1}) \end{bmatrix}.$$

【approximation by univariate dist.】

$$\Pr(W_i < w_i | W_1 < w_1, W_2 < w_2, \ldots, W_{i-1} < w_{i-1}) \approx \Phi(w_i) + (\boldsymbol{\Omega}_{<i}^{-1} \cdot \boldsymbol{\Omega}_{i,<i})'(1 - \Phi(w_1), 1 - \Phi(w_2) \ldots 1 - \Phi(w_{i-1}))'$$

Multivariate normal distribution expressed as univariate normal distribution with N of alternatives -1

⇒ Calculation cost is greatly reduced!

3. Numerical Test

Model

Cross-section random coefficients model (Mixed MNP)

$$U_{qi} = \boldsymbol{\beta}_q' \boldsymbol{x}_{qi} + \varepsilon_{qi} \qquad \boldsymbol{\beta}_q \sim MVN(\boldsymbol{b}, \boldsymbol{\Omega}),$$

$$L_q = \int_{\beta=-\infty}^{\infty} \left\{ \int_{\lambda=-\infty}^{\infty} \left(\prod_{i \neq m} \left[\Phi \left\{ \left[-\sqrt{2} (\boldsymbol{\beta}' \boldsymbol{z}_{qim}) \right] + \lambda \right\} \right] \right) \phi(\lambda) d\lambda \right\} f(\boldsymbol{\beta} | \boldsymbol{b}, \boldsymbol{\Omega}) \boldsymbol{d\beta},$$

where $\boldsymbol{z}_{qim} = \boldsymbol{x}_{qi} - \boldsymbol{x}_{qm}$,

q: individual
i : alternatives
ε: Error: IID Gumbel

True valule

$$\boldsymbol{b} = (1.5, -1, 2, 1, -2) \qquad \boldsymbol{\Omega} = \begin{bmatrix} 1 & -0.50 & 0.25 & 0.75 & 0 \\ -0.50 & 1 & 0.25 & -0.50 & 0 \\ 0.25 & 0.25 & 1 & 0.33 & 0 \\ 0.75 & -0.50 & 0.33 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Create experimental data using random numbers of virtual data for 5000 people

3. Numerical Test (vs MSL)

Cross-sectional random coefficients model

- Estimate the lower triangle of the variance-covariance matrix
- time： About 33 times faster on average and stable
- bias： About 2.1 points lower on average than MSL method

$$\tilde{\Omega} = \begin{bmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ l_{41} & l_{42} & l_{43} & l_{44} & \\ l_{51} & l_{52} & l_{53} & l_{54} & l_{55} \end{bmatrix}$$

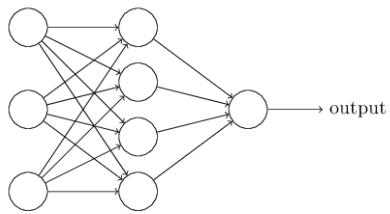
Table 1b
Evaluation of the ability to recover true parameters for the cross−sectional non-diagonal case.

Parameter	True value	MSL method					MACML method				
		Parameter estimates		Standard error estimates			Parameter estimates		Standard error estimates		
		Mean estimate	Absolute percentage bias (%)	Asymptotic standard error	Simulation standard error	Simulation adjusted asymptotic standard error	Mean estimate	Absolute percentage bias (%)	Asymptotic standard error	Approximation standard error	Approximation adjusted asymptotic standard error
Mean values of the β vector											
b1	1.500	1.374	8.4	0.133	0.049	0.142	1.443	3.8	0.147	0.022	0.148
b2	−1.000	−0.912	8.8	0.093	0.037	0.100	−0.959	4.1	0.102	0.014	0.103
b3	2.000	1.830	8.5	0.174	0.068	0.187	1.923	3.8	0.191	0.029	0.193
b4	1.000	0.914	8.6	0.092	0.032	0.097	0.958	4.2	0.101	0.014	0.102
b5	−2.000	−1.849	7.6	0.176	0.068	0.189	−1.941	3.0	0.194	0.028	0.196
Cholesky parameters characterizing the covariance matrix of the β vector											
l11	1.000	0.909	9.1	0.112	0.040	0.119	0.959	4.1	0.119	0.017	0.120
l12	−0.500	−0.463	7.3	0.085	0.029	0.090	−0.472	5.6	0.085	0.010	0.085
l13	0.250	0.231	7.5	0.089	0.036	0.096	0.233	6.7	0.087	0.009	0.088
l14	0.750	0.689	8.2	0.092	0.028	0.097	0.707	5.7	0.095	0.013	0.096
l15	0.000	0.006	0.6	0.086	0.040	0.095	0.015	1.5	0.088	0.008	0.089
l22	0.866	0.756	12.7	0.109	0.043	0.117	0.809	6.5	0.116	0.017	0.117
l23	0.433	0.431	0.5	0.105	0.050	0.117	0.436	0.6	0.100	0.012	0.101
l24	−0.144	−0.149	3.6	0.101	0.041	0.109	−0.170	17.8	0.093	0.010	0.094
l25	0.000	−0.021	2.1	0.101	0.055	0.115	−0.019	1.9	0.098	0.010	0.099
l33	0.866	0.750	13.4	0.130	0.073	0.149	0.812	6.3	0.131	0.019	0.132
l34	0.237	0.242	2.0	0.112	0.055	0.125	0.259	9.3	0.106	0.011	0.106
l35	0.000	−0.031	3.1	0.120	0.081	0.145	−0.029	2.9	0.116	0.011	0.117
l44	0.601	0.464	22.9	0.126	0.085	0.152	0.531	11.6	0.125	0.015	0.126
l45	0.000	−0.053	5.3	0.168	0.134	0.214	−0.053	5.3	0.171	0.017	0.172
l55	1.000	0.885	11.5	0.125	0.089	0.153	0.956	4.4	0.136	0.018	0.137
Overall mean value across parameters		–	7.6	0.116	0.057	0.130	–	5.5	0.120	0.015	0.121
Mean time		174.32					5.19				
Std. dev. of time		28.13					0.84				
% of Runs converged		100					100				

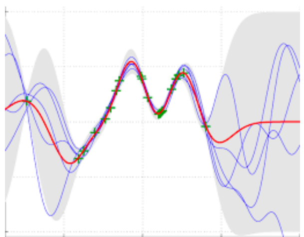
4. Hints of DCM + ML

Estimation methods in the field of machine learning can be applied to DCM estimation.

Neural network (Back propagation)

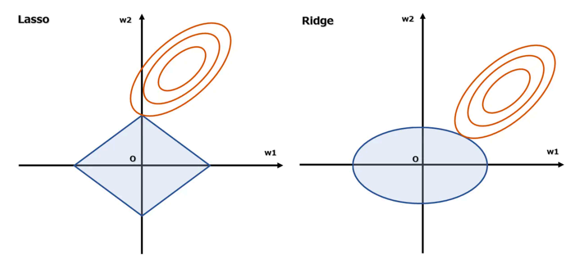


Gaussian Process



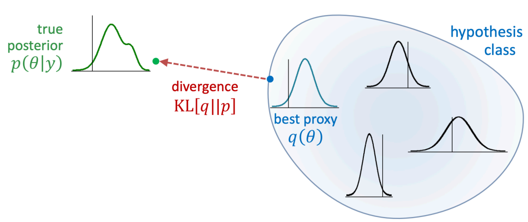
Estimation considering complex nonlinear structures

Sparse modeling



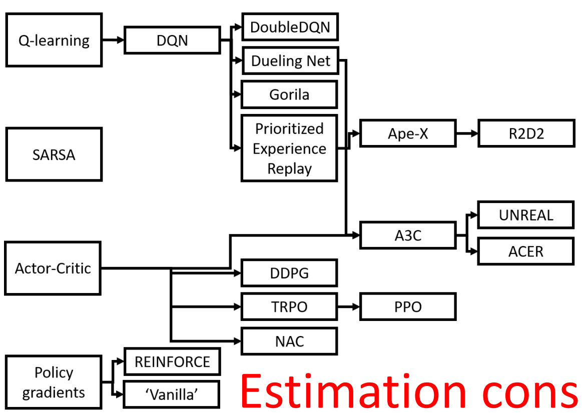
Estimation considering parameter dimension reduction

Variational Bayesian



Estimation considering complex probability distribution

Reinforcement learning



Estimation considering complex and dynamic choice

4. Conclusions

- Necessary to select an appropriate estimation method according to the structure and complexity of the model.

- Bayesian estimation is an estimation method that can support various models. In addition, parameters can be expressed as distributions (not point estimation).

- The computational cost is a important issue, and it is effective to use analytical approximation (MACML).

- Machine learning theory can be an effective method for estimating model parameters.