ExpEcon Methods: MLE, Finite Mixture Models, & Model Selection

ECON 8877
P.J. Healy
First version thanks to Hyoeun Park

Updated 2023-11-17

Contents

Maximum Likelihood Estimation

Likelihood function

- y: random variable, θ: set of parameters
- $f(\boldsymbol{y}|\theta)$:pdf, identifies the DGP
- The joint density of n i.i.d. observations from this process

$$f(y_1 \ldots y_n|\boldsymbol{\theta}) = \prod_{i=1}^{n} f(y_i|\boldsymbol{\theta})$$

Maximum Likelihood Estimation

Likelihood function

- y: random variable, θ: set of parameters
- $f(\boldsymbol{y}|\theta)$:pdf, identifies the DGP
- The joint density of n i.i.d. observations from this process

$$f(y_1 \dots y_n|\boldsymbol{\theta}) = \prod_{i=1}^{n} f(y_i|\boldsymbol{\theta}) = L(\boldsymbol{\theta}|\boldsymbol{y})$$

- $L(\boldsymbol{\theta}|\boldsymbol{y})$: function of the unknown parameter vector, $\boldsymbol{\theta}$, given observed data $\boldsymbol{y}$

Maximum Likelihood Estimation

Example

- 2-player, 3×3 game
- S_i set of strategies, $s_i \in S_i$
- $\sigma_i(s_i)$: i's probability of playing s_i in mixed strategy $\sigma_i(\cdot)$
- $u_i(\sigma_1, \sigma_2) = \sum_{(s_1,s_2)} \sigma_1(s_1)\sigma_2(s_2)u_i(s_1, s_2)$
- $BR_i(\sigma_j) = argmax_{s_i} u_i(s_i, \sigma_i)$
- logistic response

$$LR_i(\sigma_j|\lambda_i, S_i')(s_i) = \begin{cases} \frac{exp(\lambda u_i(s_i,\sigma_j))}{\sum_{s_i' \in S_i'} exp(\lambda u_i(s_i',\sigma_j))} \text{ if } s_i \in S_i' \\ 0 \text{ if } s_i \notin S_i' \end{cases}$$

Maximum Likelihood Estimation

Example
Level-k with logistic trembles

- Observed data: a played strategy
- parameter:
  ϵ (prob. tremble), λ (precision parameter), k (hierarchy level)
- pdf(pmf) identifies DGP..

Maximum Likelihood Estimation

Example
Level-k with logistic trembles

- Observed data: a played strategy
- parameter:
  ϵ (prob. tremble), λ (precision parameter), k (hierarchy level)
- pdf(pmf) identifies DGP.. Thus, pdf = probability of playing s_i

Maximum Likelihood Estimation

Example
Level-k with logistic trembles

- Observed data: a played strategy
- parameter:
  ϵ (prob. tremble), λ (precision parameter), k (hierarchy level)
- pdf(pmf) identifies DGP.. Thus, pdf = probability of playing s_i
- Assume for logistic trembles, tremble over all possible strategies
- Let $\sigma_i^{LK}(\cdot|k) = BR_i(\sigma_j^{LK}(\cdot|k-1))$, and assume unique best response

$$f^{LK}(s_i|\epsilon, \lambda, k) = \mathbf{1}_{BR_i^k(s_i)} \cdot (1-\epsilon) + \epsilon \cdot LR_i(\sigma_j^{LK}(\cdot|(k-1))|\lambda, S_i)(s_i)$$

Maximum Likelihood Estimation

Example

QRE

- Observed data: a played strategy
- parameter:λ (precision parameter)
- pdf(pmf): probability of playing s_i under QRE

$$f^{QRE}(s_i|\lambda) = LR_i(\sigma_j^{QRE}(\cdot|\lambda)|\lambda, S_i)(s_i)$$

For a set of game $\mathbf{G} = \{1, 2, 3 \ldots, G\}$, let s_i^g be a strategy that i plays in game g. Then,

$$f^{LK}(\boldsymbol{s}|\epsilon, \lambda, k) = \prod_{g \in \mathbf{G}} f^{LK}(s_i^g|\epsilon, \lambda, k) = L^{LK}(\epsilon, \lambda, k|\boldsymbol{s})$$

and

$$f^{QRE}(\boldsymbol{s}|\lambda) = \prod_{g \in \mathbf{G}} f^{QRE}(s_i^g|\lambda) = L^{QRE}(\lambda|\boldsymbol{s})$$

Maximum Likelihood Estimation

- Maximum Likelihood Estimation:
  Finding a parameter(s) that maximizes a likelihood function
- Selects the parameter values that make the observed data most likely.
- Usually, take log to the likelihood function for convenience

Maximum Likelihood Estimation

Example

Game 1	T	M	B
T	25	30	100
M	40	45	65
B	31	0	40

Game 2	T	M	B
T	30	50	100
M	40	45	10
B	35	60	0

Game 3	T	M	B
T	10	100	40
M	0	70	50
B	20	50	60

- Suppose that a subject plays M, T, B
- Let $\lambda \in \{0.01, 0.05, 1\}$
- $log(L^{QRE}(0.01|(M,T,B))) = log(0.3628) + log(0.3914) + log(0.3311)$
- $log(L^{QRE}(0.05|(M,T,B))) = log(0.5391) + log(0.5355) + log(0.3518)$
- $log(L^{QRE}(1|(M,T,B))) = log(1) + log(0.7024) + log(0.9999)$
- Thus, in this example $\hat{\lambda} = 1$

Finite Mixture Model

- So far, only one model for one likelihood function

Finite Mixture Model

- So far, only one model for one likelihood function
- Is it a valid approach?

Finite Mixture Model

- So far, only one model for one likelihood function
- Is it a valid approach?
    - For example, Georganas et al., (2015) show that a cognitive hierarchy is not persistent across classes of games
    - Suggests that estimating with *one* hierarchy cannot be valid.

Finite Mixture Model

- So far, only one model for one likelihood function
- Is it a valid approach?
    - For example, Georganas et al., (2015) show that a cognitive hierarchy is not persistent across classes of games
    - Suggests that estimating with *one* hierarchy cannot be valid.
    - Another example at the population level, people might have different risk preferences, etc.,

Finite Mixture Model

“Mix"models

- $m = 1, 2, \ldots, M$ denotes model
- $f(\boldsymbol{y}|\psi) = \sum_{m=1}^{M} \pi_m f_m(\boldsymbol{y}|\boldsymbol{\theta_m})$, where $\psi = (\{\boldsymbol{\theta_m}\}_{m=1}^{M}, \pi_1, \pi_2, \ldots, \pi_m)$
- Usually, $f_m(\boldsymbol{y}, \boldsymbol{\theta_m})$ (called component density) are taken to belong to the same parametric family.
    - There are special cases where component densities are taken to be different (nonstandard mixture)
- posterior probability that data is drawn from model m, given observed data $\boldsymbol{y}$ is $\pi_m \cdot \frac{f_m(\boldsymbol{y}|\boldsymbol{\theta}_m)}{f(\boldsymbol{y}|\psi)}$
- A parametric family of densities is primitive. Each component has distinct values
- Using MLE to fitting mixture distributions π (Most commonly used way)

Model Selection

Going back to Level-*k* and QRE Example

- Suppose that an experimenter wants to compare which model better explains data
- They can horse-race models
- Using MLE?
  - For level-*k*, find the ML estimates $(\hat{\epsilon}, \hat{\lambda}, \hat{k})$
    plug in those values to the level-*k* model's likelihood function
  - For QRE, find the ML estimate $\hat{\lambda}$
    plug in $\hat{\lambda}$ to the QRE's likelihood function
  - Compare the likelihood values of two models and pick the model that gives the higher value

Model Selection

What is the problem with ML approach?

- level-k has three parameters, while QRE has only one parameter
- level-k has more "flexibility"
- Consider a weird model with ∞ numbers of parameters
  - ∞ flexibility
  - Can explain any behavior in the data
  - Then, this model "wins"just because it has more flexibility, not because it is true DGP.
- Need for fixing the problem of over-fitting due to large # of parameters

Model Selection

How to penalize over-fitting due to large numbers of parameters?

- AIC (Akaike information criterion)
- BIC (Bayesian information criterion)
- Cross-Validation

Model Selection: AIC, BIC

- AIC $=2k\ln(n) - 2\ln(\widehat{L})$
- BIC$=k\ln(n) - 2\ln(\widehat{L})$

where

- $\widehat{L}$=the maximized value of the likelihood function of the model
  - From observed data, get ML estimates, and plug into the ML function
- n= number of observed data point
  - In our example, the number of games subjects played
- k = number of parameters
  - In level-k model, k=3
    In QRE, $k = 1$

Model Selection: AIC, BIC

- AIC $=2k\ln(n) - 2\ln(\widehat{L})$
- BIC$=k\ln(n) - 2\ln(\widehat{L})$

- The preferred model is the one with the minimum AIC/BIC value
- The second term gives benefits to the model with goodness-of-fit
- The first term gives a penalty to the number of parameters
- Those can be only used for linear-models
- Can be used only when $n >> k$

Model Selection: Cross-Validation

- k-fold cross-validations
- Divide data into k sub-samples
  - For example, 12 data points, 4 sub-samples that include 3 data points each
- $k-1$ sub-samples = training data
  - Fit the data to a model (MLE, MSE ...)
- one sub-sample = testing data
  - Using the fitted parameters from training data, test the model i.e., Plug in the estimated parameter to the goodness-of-fit function used for training (MLE, MSE, ...)
- Repeat this for K times
- Extreme case of Ko fold cross-validation is leave-one-out cross-validation that $K = n$, where n= number of data points

Model Selection: Cross-Validation

How does Cross-Validation penalize the number of parameters?

Consider the following example..

- Suppose that subjects played four games
- For three games, a subject's choices coincide with the level-k level-1's predictions
- Then $\hat{\epsilon} = 0$ and log-likelihood function value is 0.
- Suppose that the subject did not play level-1's predicted strategy.
- Then for the testing data (fourth game), $\hat{\epsilon} = 0$ results in $-\infty$
- For QRE, less likely to over-fit (since it has only one parameter). Less likely to have $-\infty$ for testing data

Model Selection: Cross-Validation

- Cross-Validation penalizes the number of parameters internally.
- Over-fitting due to a higher number of parameters penalizes deviation from the prediction a lot in the testing data
- No restriction for models being tested; does not have to be linear

Model Selection

- SO, is Cross-Validation a perfect solution for model selection?

Model Selection

- SO, is Cross-Validation a perfect solution for model selection?
- NO!

Model Selection

- SO, is Cross-Validation a perfect solution for model selection?
- NO!
- The penalization can be “tooßevere

Model Selection

- SO, is Cross-Validation a perfect solution for model selection?
- NO!
- The penalization can be “tooßevere
- See Healy & Park (2023) for suggestions :)