Maximum Likelihood Estimation

Apoorva Lal

March 31, 2022

Stanford

Overview

1. Overview
2. Likelihood: An Intuition
3. Likelihood: A Recipe
4. Likelihood: An Example
5. MLE and Uncertainty

Motivating the Likelihood approach: modeling choices

- Social science is often about modeling choices made by strategic agents (countries, politicians, armies)
- Choices can be thought of as being grounded in utility maximisation (latent variable y^* defined net of costs)
- A military attempts a coup in year i ($y_i = 1$) if its utility $y_i^* > 0$. That is, $y_i = \mathbb{1}_{y_i^* > 0}$
- Model: $y_i^* = \beta + \varepsilon_i \quad , \varepsilon_i \sim \mathcal{N}(0, 1)$
- Simplest model: β is the average utility of a coup in that country; ε_is are idiosyncratic shocks. Can put covariates on y_i^*.
- **Cons**: ε_i distribution is paramount: model hinges entirely on noise distribution. Stark contrast to many estimators covered in 450b.
- **Pros**: Well understood properties, certain parametrisations are 'robust', used widely for classification
    - *Structural econometrics* uses this approach frequently because it allows one to perform 'counterfactual experiments' and quantify welfare implications (again, under the model)

Likelihood analysis

- $y_i = \mathbb{1}_{y_i^* * > 0}$ with $u_i = \beta + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, 1)$

$$\begin{aligned} P(y_i = 1) &= P(\beta + \epsilon_i > 0) \\ &= P(\epsilon_i > -\beta) \\ &= 1 - P(\epsilon_i \leq -\beta) \\ &= 1 - \Phi(-\beta) \\ &= \Phi(\beta). \end{aligned}$$

- Thus, $y_i \sim$ Bernoulli$(\Phi(\beta))$ with pmf

$$p_\beta(y_i) = \Phi(\beta)^{y_i}[1 - \Phi(\beta)]^{1-y_i}$$

The Likelihood

Likelihood: A General recipe

- Stipulate *family* of PDF/PMF that is said to have generated the data in question $f(y_i)$
- Write down likelihood function – the joint probability of observing all events under the assumed distribution

$$L(\theta|y_i) = f(y_i|\theta)$$
$$L(\theta|\mathbf{y}) = \prod f(\theta|y_i)$$

- Refactor so that we can take the logs more easily
- Take the logs so we have the log-likelihood function:

$$\ell(\theta|\mathbf{y}) = \log(L(\theta|\mathbf{y})) = \sum \log f(y_i, \theta)$$

- True values maximise log-likelihood $\widehat{\theta} = \arg\max_{\theta \in \Theta} \ell(\theta)$. Solution implicitly defined by first derivative

$$\frac{\partial \ell(\theta|\mathbf{y})}{\partial \theta} = 0$$

Properties

- **Consistency**

$$\lim_{N \to infty} \mathbf{Pr}\left(\left|\widehat{\theta} - \theta\right| > \epsilon\right) = 0$$

- **Asymptotic Normality**

$$\sqrt{n}(\widehat{\theta} - \theta) \xrightarrow{d} \mathcal{N}\left(0, -\mathbb{E}\left[\frac{\partial^2}{\partial\theta\partial\theta'}(Y_i, \theta)\right]^{-1}\right)$$

- Details, Proofs : Cameron and Trivedi, Microeconometrics, Ch 5

Setup

$$Y_i \sim \text{Bernoulli}(\theta)$$

$$\theta = \Lambda\left(X_i^T\beta\right) = \frac{\exp\left(X_i^T\beta\right)}{1+\exp\left(X_i^T\beta\right)} = \frac{1}{1+\exp\left(-X_i^T\beta\right)}$$

Likelihood:

$$f(Y \mid \theta) = \prod_{i=1}^{n} \Lambda\left(X_i^T\beta\right)^{y_i}\left(1-\Lambda\left(X_i^T\beta\right)\right)^{1-y_i}$$

Log likelihood:

$$\mathcal{L}(\beta) = \sum_{i=1}^{n} y_i \log\left(\Lambda\left(X_i^T\beta\right)\right) + \sum_{i=1}^{n}(1-y_i)\log\left(\Lambda\left(X_i^T\beta\right)\right)$$

Setup

$$Y_i \sim \text{Bernoulli}(\theta)$$
$$\theta = \Phi\left(X_i^T \beta\right)$$

Likelihood:

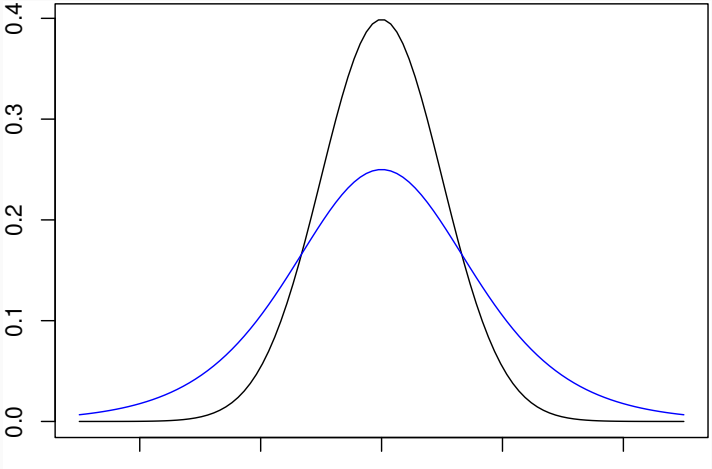
$$f(Y \mid \theta) = \prod_{i=1}^{n} \Phi\left(X_i^T \beta\right)^{y_i} \left(1 - \Phi\left(X_i^T \beta\right)\right)^{1-y_i}$$

Log likelihood:

$$\mathcal{L}(\beta) = \sum_{i=1}^{n} y_i \log\left(\Phi\left(X_i^T \beta\right)\right) + \sum_{i=1}^{n} (1 - y_i) \log\left(\Phi\left(X_i^T \beta\right)\right)$$

Model Dependence

- Perhaps surprisingly, our estimate of β can depend quite a bit on the distribution of utility shocks ϵ_i
- Suppose $P(\epsilon_i \leq x) = \Lambda(x) = 1/(1 + e^{-x})$, which is the cdf of the (standard) logistic distribution
- Standard logistic (blue) and standard normal (black) pdfs



Score Function - Probit

- Let $\ell_i(\beta) = \log p_\beta(y_i)$.
- The score function is

$$s_i(\beta) = \frac{\partial}{\partial \beta} \ell_i(\beta) \tag{1}$$

with $\bar{s}(\beta) = \frac{1}{N} \sum_{t=1}^{N} s_i(\beta) = \frac{\partial}{\partial \beta} \bar{\ell}(\beta)$.

- To maximize $\bar{\ell}(\beta)$, it usually suffices to find $\hat{\beta}$ such that $\bar{s}(\hat{\beta}) = 0$ (average score is zero).
- In our case,

$$\bar{s}(\beta) = \frac{\phi(\beta)[\bar{y} - \Phi(\beta)]}{\Phi(\beta)[1 - \Phi(\beta)]} \tag{2}$$

$$\bar{s}(\hat{\beta}) = 0 \implies \hat{\beta} = \Phi^{-1}(\bar{y}) \tag{3}$$

MLE with Logistic Errors

- Suppose $P(\epsilon_i \leq x) = \Lambda(x) = 1/(1 + e^{-x})$
- Then by similar arguments $y_i \sim$ Bernoulli$(\Lambda(\beta))$.
- Using the fact that $\frac{\partial}{\partial x}\Lambda(x) = \lambda(x) = \Lambda(x)[1 - \Lambda(x)]$,

$$p_\beta(y_i) = \Lambda(\beta)^{y_i}[1 - \Lambda(\beta)]^{1-y_i} \tag{4}$$

$$\ell_i(\beta) = y_i \log \Lambda(\beta) + (1 - y_i) \log[1 - \Lambda(\beta)] \tag{5}$$

$$s_i(\beta) = y_i - \Lambda(\beta) \tag{6}$$

- Thus, $\bar{s}(\beta) = \bar{y} - \Lambda(\beta)$, and we can see that $\hat{\beta} = \Lambda^{-1}(\bar{y}) \neq \Phi^{-1}(\bar{y})$
- e.g., `qnorm(0.1); qlogis(0.1)`
- We'll see later that this usually doesn't matter at all

Inference

- We get an estimate $\hat{\beta}$ from our sample, but how close is $\hat{\beta}$ to the true β in the population?
- Related: how do we form confidence intervals, or test hypotheses such as $H_0 : \beta = 0$.
- MLE is consistent: $\hat{\beta} \xrightarrow{p} \beta$ as $N \to \infty$
- Asymptotic normality:

$$\sqrt{N}(\hat{\beta} - \beta) \xrightarrow{d} \mathcal{N}(0, \mathcal{A}^{-1}) \tag{7}$$

where $\mathcal{A} = \mathbb{V}_{s(\beta)} [=] \mathbb{E}[s^2(\beta)] = -\mathbb{E}[\frac{\partial}{\partial \beta} s(\beta)]$.[†]

Standard Error

$$\sqrt{N}(\hat{\beta} - \beta) \xrightarrow{d} \mathcal{N}(0, \mathcal{A}^{-1}) \tag{8}$$

where $\mathcal{A} = \mathbb{V}\left[s(\beta)\right] = \mathbb{E}[s^2(\beta)] = -\mathbb{E}[\frac{\partial}{\partial \beta} s(\beta)]$.

- Let's use this to get a standard error for $\hat{\beta}$
- Under the logistic model, recall $s_i(\beta) = y_i - \Lambda(\beta)$
- Thus, $\frac{\partial}{\partial \beta} s_i(\beta) = -\lambda(\beta) \implies \mathcal{A} = \lambda(\beta)$
- Okay, but we don't know β, so we use the plug-in estimator $\hat{\mathcal{A}} = \lambda(\hat{\beta})$.
- Giving us a std. error for $\hat{\beta} = \Lambda^{-1}(\bar{y})$ of $1/\sqrt{N\lambda(\Lambda^{-1}(\bar{y}))}$

Maximum Likelihood with Multiple Parameters

- Most DGPs have multiple parameters, so maximising their $\log p_\theta(y_i)$ requires numerical methods
- However, the mathematics stays the same:

$$\ell_i(\theta) = \log p_\theta(y_i) \tag{9}$$

$$s_i(\theta) = \frac{\partial}{\partial \theta} \ell_i(\theta) = \nabla \ell_i(\theta) \tag{10}$$

- However, now the score function $s_i(\theta)$ is a k-vector
- $\mathcal{A} = \mathbb{V}\left[s(\theta)\right] = \mathbb{E}[s(\theta)s(\theta)^\top] = -\mathbb{E}[\nabla s(\theta)] = -\mathbb{E}[\nabla^2 \ell(\theta)]$ is a k-by-k matrix

Normal Distribution example

Likelihood: An Example

- **Motivation:** Suppose that we have N elections with two parties (A, B).
- A's vote share in the last five elections (`y_samp`) was:

```
mu = 50; sigma = 5
(y_samp <- rnorm(5, mu, sigma))
```

```
## [1] 56.85 47.18 51.82 53.16 52.02
```

- Can we describe the underlying data-generating process?
- What's our best guess?

Normal MLE Estimation

- **Let's assume:** A's vote share is drawn i.i.d. from a normal distribution with (unknown) mean μ and (unknown) variance σ^2.
- This gives us enough structure to proceed with MLE.
- If we **knew** mean and variance, we could calculate the probability of observing any value:

$$f(x) = \text{pdf}(\mathcal{N}(\mu, \sigma^2)) \equiv \phi(x)$$

- But we **don't**. So we have to make our best guess.

Normal MLE Estimation (cont'd)

- (Step 2). We ask: what is the likelihood of observing any set combination of parameters (μ, σ^2), **given the data that we observe?**

$$L(\mu, \sigma^2|\mathbf{y}) = \prod f(y_i|\mu, \sigma^2) \tag{11}$$

$$= \prod \frac{\exp(-\frac{(y_i-\mu)^2}{2\sigma^2})}{\sqrt{2\pi\sigma^2}} \tag{12}$$

- (Step 3). Refactoring.

$$L(\mu, \sigma^2|\mathbf{y}) = \frac{\exp(-\sum \frac{(y_i-\mu)^2}{2\sigma^2})}{(2\pi)^{n/2}\sigma^{2n/2}}$$

- (Step 4). Taking logs.

$$\ell(\mu, \sigma^2|\mathbf{y}) = -\sum \frac{(y_i - \mu)^2}{2\sigma^2} - \frac{n}{2}\log(\sigma^2) + C$$

- Now we can plug in any combination of candidate values for μ and σ^2 into this function and we get a score.
- We have a nicely defined function $\rightarrow$ now to maximise it!

Normal MLE Estimation (cont'd)

```
# likelihood
l_normal = \(y, μ, σ2){
  -sum((y - μ)^2) / (2 * σ2) - length(y) / 2 * log(σ2)
}
# for a given θ
l_normal(y_samp, 43, 2)
```

```
## [1] -119.7
```

Let's set up some more code to plot the likelihood for every combination of μ and σ^2.

```
viz_df = CJ(μ = seq(30, 70, by = 0.5), σ2 = seq(3, 60, by = 0.5))
viz_df[, likelihood := l_normal(y_samp, μ, σ2), .I]
viz_df[order(-likelihood)][1, ] %>% print()
```

```
##         μ    σ2 likelihood
##    <num> <num>      <num>
## 1:    52   9.5     -8.166
```

grid-search is extremely inefficient

- We can also find the parameter combination that optimises the likelihood algebraically.
- Recall from Today's lecture:

$$\mu^* = \frac{\sigma^2 y_i}{n} = \bar{y} \tag{13}$$

$$\sigma^{2*} = \frac{1}{n}\sum(y_i - \bar{y})^2 \tag{14}$$

Not all likelihoods have an analytic solution. Many likelihoods need to be maximised numerically.

numerical optimization algorithms

- Newton-Raphson
- Nelder-Mead
- BHHH

MLE and uncertainty (cont'd)

- What if I told you that the data were generated with:

$$\mu = 50, \sigma^2 = 25$$

- Our MLE estimate only takes the "sample" from the DGP.
- We can't make any assumptions about the parameters in the DGP: that's the thing we're trying to estimate using MLE!
- But because of the convergence in distribution, we can still infer how likely the observed MLE estimate is if we assume a true parameter θ_0.

MLE and uncertainty (cont'd)

- Let's create M samples with size n from our true DGP.
- For each of these samples, we calculate the MLE estimate and its variance.

```
get_mean_variance = \(n){
  y_samp = rnorm(n, 50, 5)
  ybar  = mean(y_samp); sig_hat = 1/n * sum((y_samp - ybar)^2)
  c(ybar, sig_hat)
}

samples = replicate(10000, get_mean_variance(100)) |> t()
rbind(apply(samples, 2, mean), apply(samples, 2, var))
```

```
##         [,1]  [,2]
## [1,] 50.0029 24.80
## [2,]  0.2519 12.48
```

What happens if we increase the sample size?

```
samples = replicate(10000, get_mean_variance(1000)) |> t()
rbind(apply(samples, 2, mean), apply(samples, 2, var))
```

```
##          [,1]   [,2]
## [1,] 50.00229 24.984
## [2,]  0.02485  1.256
```

MLE and uncertainty (cont'd)

Recall the asymptotic property of MLE estimators as $n \to \infty$:

$$p(\hat{\mu}, \hat{\sigma}^2) \xrightarrow{d} \text{MVN}\Big((\bar{y}, \frac{1}{n}\sum(y_i - \bar{y})^2), \begin{bmatrix} \frac{\sigma^2}{n} & 0 \\ 0 & \frac{2(\sigma^2)^2}{n} \end{bmatrix}\Big)$$

Since we know the true parameters...

- We have problems when $n = 5$
- Mean ($\mu = 50$) and Variance ($\sigma^2 = 25$) parameters are correctly estimated with $n = 100$
- "Sampling" uncertainty of these parameters falls with sample size
- Variance of sampling distribution converges to $25/100 = 0.25$ and $(2 * 25^2)/100 = 12.5$, respectively

Summary

- Generic recipe for a how to think about likelihood.
  - Decide on model
  - Write down likelihood f'n: how likely is θ given the observed data?
  - Refactor and take the logs
  - Maximise w.r.t. θ (take first derivative)
  - Derive second derivative / Hessian for variance
    - this last step is often non-trivial for more complicated likelihoods. if all else fails, use gradient-free optimizers like BFGS / Golden Section
- Applied to normal distribution (both with algebra and with code)
- Thinking about uncertainty and inference in the context of MLE