Class 8: Introduction to Maximum Likelihood Estimation

MFE 402

Dan Yavorsky

Last Class

- Introduced Rubin's Causal Model
- Explained the fundamental problem of causal inference
- Discussed select situations in which we can make causal statements
  - Under true randomization
  - Under as-if random (thanks to covariate adjustments)
  - Under natural experiments (Diff in Diff, RDD)

Topics for Today

- Define the likelihood function
- Introduce maximum likelihood estimation
- Examples with one-parameter models
  - Exponential
  - Normal
  - Binomial
- Examples with multi-parameter models

The Likelihood Function

Parametric Statistical Models

A major class of statistical inference concerns Maximum Likelihood (ML) estimation of parametric models.

Parametric models are statistical models that are *complete* probability functions.

Compare:

- The general regression model is **non-parametric**
  - $Y_i = f(X_i) + e_i$ with light assumptions on e_i
- The linear regression model is **semi-parametric**
  - $Y_i = X_i'\beta + e$ with $\mathbb{E}[e_i|X_i] = 0$
- The normal linear regression model is **parametric**
  - $Y_i = X_i'\beta + e_i$ with $e_i \sim N(0, \sigma^2)$ can be written as $Y_i \sim N(\mu_i, \sigma^2)$ with $\mu_i = X_i'\beta$

Inverse Probability, Generally

We typically think of probability as uncertainty around the to-be realized values of variables, given a set of assumptions about the world:

$$\mathbb{P}(y|M) \equiv \mathbb{P}(\text{data}|\text{model})$$

where y is observed or hypothetical data and M summarizes all features of a statistical model.

To **do** statistical inference is to learn about M: $\mathbb{P}(M|y)$. For example, given some data:

- what is the estimated slope coefficient β_1?
- is there homogeneity ($\sigma^2(X) = \sigma^2$)?

Typically learning about **all** of M is too difficult and possibly not of interest, but we can sub-divide M into the components we wish to learn about (θ) and the rest (M^*). Our goal is then to evaluate:

$$\mathbb{P}(\theta|y, M^*) \equiv \mathbb{P}(\theta|y)$$

Given y

Our goal is to evaluate $\mathbb{P}(\theta|y)$.

Notice that this expression is **conditional on y.** The data are fixed. They're sitting in a file on your computer.

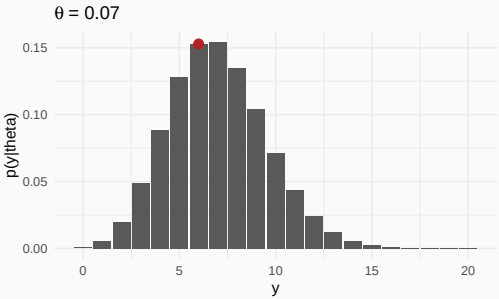
The likelihood method of inference asks how likely various models are, given the data we've observed (i.e., what is $\mathbb{P}(\theta)$ for various θ).

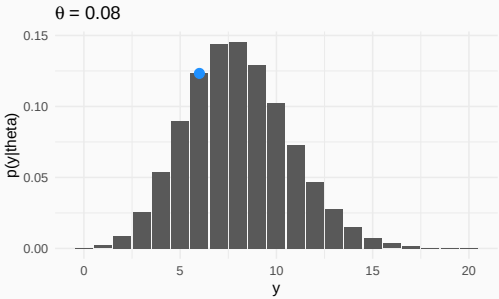
In other words, the likelihood describes the extent to which the sample provides support for any particular model (i.e., set of parameter values). Higher likelihoods for a particular value of θ correspond to more support for that value of θ.

Example

Suppose we take a binary view on employment and consider that MFE students can either get a job in Algorithmic Trading (with probability θ) or in some other field (with probability 1 - θ).

Consider two possible models of the world: $\theta_1 = 0.07$ and $\theta_2 = 0.08$:





A sample of 100 past students was taken ($n = 100$) and 6 students got jobs in Algorithmic Trading ($y = 6$). Which value of θ is more likely?

Example

Y is a binomial random variable: $f(y|\theta, n) = \binom{n}{y}\theta^y(1 - \theta)^{n-y}$

```
L1 <- dbinom(x=6, prob=0.07, size=100)
L2 <- dbinom(x=6, prob=0.08, size=100)

c(L1=L1, L2=L2, ratio=L1/L2)
```

```
       L1        L2     ratio
0.1528554 0.1232795 1.2399087
```

Thus, $\theta = 0.07$ has 1.123 times the support of $\theta = 0.08$, given our sample.

Notice that terms not involving θ cancel from the ratio of probabilities:

$$\frac{f(y|\theta_1, n)}{f(y|\theta_2, n)} = \frac{\binom{n}{y}\theta_1^y(1 - \theta_1)^{n-y}}{\binom{n}{y}\theta_2^y(1 - \theta_2)^{n-y}} = \frac{\theta_1^y(1 - \theta_1)^{n-y}}{\theta_2^y(1 - \theta_2)^{n-y}}$$

Inverse Probability, Specifically

Bayes' Rule allows us to write:

$$\mathbb{P}(\theta|y) = \frac{\mathbb{P}(y|\theta)\mathbb{P}(\theta)}{\mathbb{P}(y)}$$

We will take a "position of indifference" and treat $\mathbb{P}(\theta)$ as a constant, representing a uniform distribution over the parameter space.

Notice also that $\mathbb{P}(y)$ is a constant, since it is a function of the data, which is fixed.

$\Longrightarrow$ Therefore, the likelihood that a given model produced the data we observe is proportional to the conditional probability of the data given the model.

$$\mathbb{P}(\theta|y) = \mathbb{P}(y|\theta)k(y) \propto \mathbb{P}(y|\theta)$$

The (Joint) Likelihood Function

Consider a random variable Y and a scalar parameter θ.

Our model for a random sample assumes that Y_i for $i = 1, \ldots, n$ are i.i.d. with known density function $f(y|\theta)$ that depends on the unknown parameter θ.

The joint density of the data is

$$f(y_1, \ldots, y_n|\theta) = f(y_1|\theta)f(y_2|\theta)\cdots f(y_n|\theta) = \prod_{i=1}^{n} f(y_i|\theta)$$

The likelihood function **is** the joint density of the data, ***viewed as a function of the parameter*** θ:

$$L_n(\theta) = f(Y_1, \ldots, Y_n|\theta) = \prod_{i=1}^{n} f(Y_i|\theta)$$

Whereas a density function $f(y|\theta)$ shows which values of Y are most likely to occur given a specific value of θ, the likelihood function $L_n(\theta)$ shows the values of θ most likely to have generated the observations $\{Y_i\}_{i=1}^{n}$.

The Log-Likelihood Function

Since $f(y_i|\theta)$ is a density, $f(y_i|\theta) \in [0, 1]$ and $L_n(\theta)$ can be numerically small for large n

Therefore, it is often more convenient to work with the log-likelihood function:

$$\ell_n(\theta) = \log L_n(\theta) = \sum_{i=1}^{n} \log f(Y_i|\theta)$$

Two things to note:

- log is a monotonically increasing function, so the maximums occur at the same value of θ, i.e., $\arg\max_\theta L_n(\theta) = \arg\max_\theta \ell_n(\theta)$
- A convenient algebraic simplification that occurs regularly with log-likelihoods is $\sum_{i=1}^{n} Y_i = n\bar{Y}$

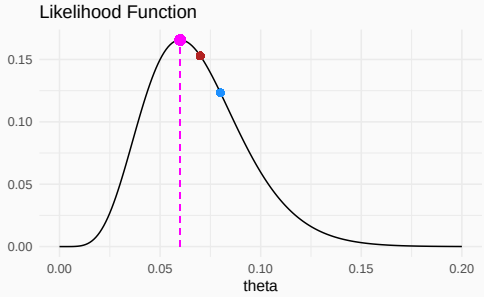
The Maximum Likelihood Estimator (MLE)

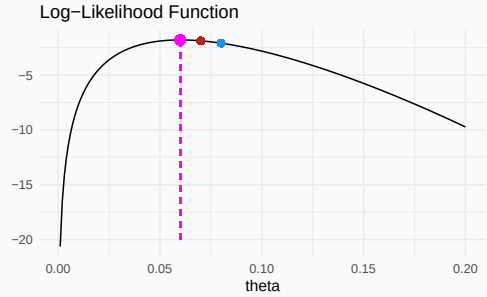
The maximum likelihood estimator (MLE) $\hat{\theta}_{\text{MLE}}$ of θ is the value that maximizes $L_n(\theta)$ and $\ell_n(\theta)$:

$$\hat{\theta}_{\text{MLE}} = \underset{\theta \in \Theta}{\text{argmax}}\, L_n(\theta) = \underset{\theta \in \Theta}{\text{argmax}}\, \ell_n(\theta)$$

$\hat{\theta}_{\text{MLE}}$ is the value of θ most-likely to have generated the observed data.

Example





Recall that 6 out of 100 MFE students in our sample went into Algorithmic Trading.
$\hat{\theta}_{\text{MLE}} = 6/100 = 0.06$ as you might expect.

MLE Recipe

To find the MLE, use the following steps:

1. Construct $f(y|\theta)$ as a function of y and θ
2. Take the logarithm $\log f(y|\theta)$
3. Evaluate $y = Y_i$ and sum over i: $\ell_n(\theta) = \sum_{i=1}^{n} \log f(Y_i|\theta)$
4. Maximize: $\hat{\theta} = \arg\max \ell_n(\theta)$
   - If possible, solve the first-order condition to find the maximum
   - Check the second-order condition to verify that it is a maximum
   - If solving the FOC is not possible, use numerical methods to maximize $\ell_n(\theta)$

General Comments

1. Notice that the entire likelihood function is a summary estimator of θ
   - Some criticize use of $\hat{\theta}$ in place of the full likelihood function due to the loss of information
   - This criticism is technically correct
   - With few parameters, one may heed this criticism
   - In high dimensional space, however, ML estimation is a reasonable and practical alternative
2. Likelihoods of distributions that belong to the exponential family are well-behaved, but in general, this need not be true
   - There can be "ridges" that challenge certain optimization routines or yield a set of maxima
   - There can be local maxima
   - The likelihood can be degenerate, approaching ∞ for some values of θ such that the MLE does not exist
3. MLEs are not necessarily unbiased estimators

Example: $Y \sim \textbf{Expon}(\lambda)$

MLE from $L_n(\lambda)$ for $Y \sim \text{Expon}(\lambda)$

Suppose $f(y|\lambda) = \lambda \exp\{-\lambda y\}$

The likelihood function is:

$$L_n(\lambda) = \prod_{i=1}^{n} \lambda \exp\{-\lambda Y_i\} = \lambda^n \exp\left\{-\lambda \sum_{i=1}^{n} Y_i\right\} = \lambda^n \exp\{-n\lambda \bar{Y}\}$$

Then:

$$\frac{d}{d\lambda} L_n(\lambda) = n\lambda^{n-1} \exp\{-n\lambda \bar{Y}\} - n\lambda^n \bar{Y} \exp\{-n\lambda \bar{Y}\} = 0$$

$$\Rightarrow \qquad n\lambda^{n-1} \exp\{-n\lambda \bar{Y}\} = n\lambda^n \bar{Y} \exp\{-n\lambda \bar{Y}\}$$

Cancel terms to find $\hat{\lambda}_{\text{MLE}} = 1/\bar{Y}$

MLE from $\ell_n(\lambda)$ for $Y \sim$ Expon(λ)

Suppose $f(y|\lambda) = \lambda \exp\{-\lambda y\}$ so that $\log f(y|\lambda) = \log(\lambda) - \lambda y$

The log-likelihood function is:

$$\ell_n(\lambda) = \sum_{i=1}^{n} (\log(\lambda) - \lambda Y_i) = n \log(\lambda) - n\lambda \bar{Y}$$

Then:

$$\frac{d}{d\lambda}\ell_n(\lambda) = \frac{n}{\lambda} - n\bar{Y} = 0$$

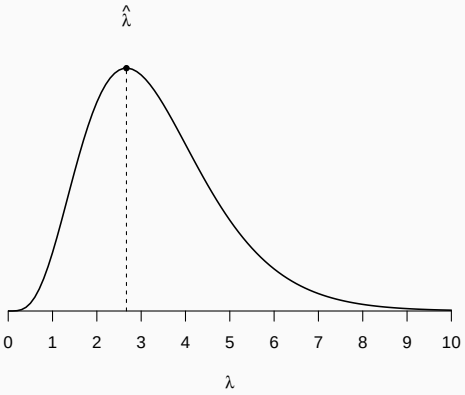
$$\Rightarrow \qquad \frac{1}{\lambda} = \bar{Y}$$

And so $\hat{\lambda}_{\text{MLE}} = 1/\bar{Y}$ as expected

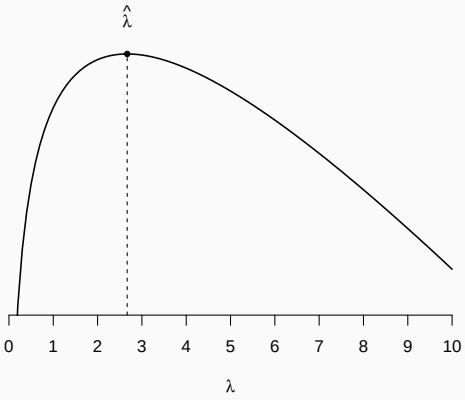
Likelihood and Log-Likelihood Plots for $Y \sim \text{Expon}(\lambda)$

Suppose $n = 4$ and $1/\bar{Y} = 2.67$, then:

$L_n(\lambda)$



$\ell_n(\lambda)$



Code for $Y \sim \text{Expon}(\lambda)$

$$L_n(\lambda) = \lambda^n \exp\{-n\lambda \bar{Y}\}$$

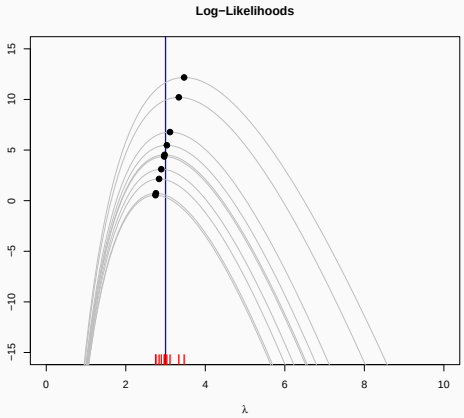
```
y <- rexp(4, rate=3)
s <- seq(from=0, to=10, by=0.1)

lik <- function(y, lam) {
    n <- length(y)
    lam^n * exp(-n*lam*mean(y))
}

plot(s, lik(y, s), type="l")
```

$$\ell_n(\lambda) = n \log(\lambda) - n\lambda \bar{Y}$$

```
y <- rexp(4, rate=3)
s <- seq(from=0, to=10, by=0.1)

loglik <- function(y, lam) {
    n <- length(y)
    n*log(lam) - n*lam*mean(y)
}

plot(s, loglik(y, s), type="l")
```

Likelihood Function in Repeated Samples

```r
# log-likelihood for y distributed exponential
loglik <- function(y, lam) {
    length(y) * (log(lam) - lam*mean(y))
}

# plot 10 likelihoods, each sample has n=50
true_lam <- 3
for(i in 1:10) {
    n <- 50
    y <- rexp(n, true_lam)
    s <- seq(from=0, to=10, by=0.1)
    lines(x=s, y=loglik(y, s), col="gray")
}
```



Example:

$Y \sim N(\mu, \sigma^2)$ **with** σ^2 **known**

MLE for $Y \sim N(\mu, \sigma^2)$ with σ^2 known

$$\text{Suppose} \qquad f(y|\mu) = (2\pi\sigma^2)^{-1/2} \exp\{-(y-\mu)^2/2\sigma^2\}$$
$$\Rightarrow \quad \log f(y|\mu) = -\log(2\pi\sigma^2)/2 - (y-\mu)^2/2\sigma^2$$

The log-likelihood function is:

$$\ell_n(\mu) = \sum_{i=1}^{n} \left(-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(Y_i-\mu)^2}{2\sigma^2} \right) = -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(Y_i-\mu)^2$$

Then:

$$\frac{d}{d\mu}\ell_n(\mu) = \frac{1}{\sigma^2}\sum_{i=1}^{n}(Y_i-\mu) = \frac{1}{\sigma^2}(n\bar{Y} - n\mu) = 0$$

And so $\hat{\mu}_{\text{MLE}} = \bar{Y}$

Example:

$Y \sim N(\mu, \sigma^2)$ **with** μ **known**

MLE for $Y \sim N(\mu, \sigma^2)$ with μ known

$$\text{Suppose} \qquad f(y|\sigma^2) = (2\pi\sigma^2)^{-1/2} \exp\{-(y-\mu)^2/2\sigma^2\}$$
$$\Rightarrow \quad \log f(y|\sigma^2) = -\log(2\pi\sigma^2)/2 - (y-\mu)^2/2\sigma^2$$

The log-likelihood function is:

$$\ell_n(\sigma^2) = \sum_{i=1}^{n} \left(-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(Y_i-\mu)^2}{2\sigma^2} \right) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(Y_i-\mu)^2$$

Then:

$$\frac{d}{d\sigma^2}\ell_n(\sigma^2) = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4}\sum_{i=1}^{n}(Y_i-\mu)^2 = 0$$

And so $\hat{\sigma}^2_{\text{MLE}} = \frac{1}{n}\sum_{i=1}^{n}(Y_i-\mu)^2$

Example: $Y \sim N(\mu, \sigma^2)$

MLE for $Y \sim N(\mu, \sigma^2)$

The log-likelihood function is:

$$\ell_n(\mu, \sigma^2) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(Y_i - \mu)^2$$

Then:

$$\frac{\partial}{\partial \mu}\ell_n(\mu, \sigma^2) = \frac{1}{\sigma^2}(n\bar{Y} - n\mu) = 0$$

$$\frac{\partial}{\partial \sigma^2}\ell_n(\mu, \sigma^2) = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4}\sum_{i=1}^{n}(Y_i - \mu)^2 = 0$$

And so:

$$\hat{\mu}_{\text{MLE}} = \bar{Y}$$

$$\hat{\sigma}^2_{\text{MLE}} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{\mu}_{\text{MLE}})^2$$

Example:

$Y_i \sim N(\mu_i, \sigma^2)$ **with** $\mu_i = X_i'\beta$

MLE for $Y_i \sim N(\mu_i, \sigma^2)$ with $\mu_i = X_i'\beta$

The log-likelihood function is:

$$\ell_n(\beta, \sigma^2) = \sum_{i=1}^{n} \log f(Y_i | Y_i, \beta, \sigma^2) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\underbrace{\sum_{i=1}^{n}(Y_i - X_i'\beta)^2}_{(Y-X\beta)'(Y-X\beta)}$$

Then:

$$\frac{\partial}{\partial\beta}\ell_n(\beta, \sigma^2) = \frac{1}{\sigma^2}(X'y - X'X\beta) = 0$$
$$\frac{\partial}{\partial\sigma^2}\ell_n(\beta, \sigma^2) = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4}(Y - X\beta)'(Y - X\beta) = 0$$

And so:

$$\hat{\beta}_{\text{MLE}} = (X'X)^{-1}X'y = \hat{\beta}_{\text{OLS}}$$
$$\hat{\sigma}^2_{\text{MLE}} = \frac{1}{n}\hat{e}'\hat{e} = \hat{\sigma}^2$$

Example: $Y \sim \textbf{Bern}(\pi)$

MLE for $Y \sim \text{Bern}(\pi)$

Suppose $f(y|\pi) = \pi^y \times (1 - \pi)^{1-y}$ so that $\log f(y|\pi) = y \log(\pi) + (1 - y) \log(1 - \pi)$

The log-likelihood function is:

$$\begin{aligned}\ell_n(\pi) &= \sum_{i=1}^{n} Y_i \log(\pi) + (1 - Y_i) \log(1 - \pi) \\ &= n\bar{Y} \log(\pi) + n(1 - \bar{Y}) \log(1 - \pi)\end{aligned}$$

Then:

$$\frac{d}{d\pi}\ell_n(\pi) = \frac{n\bar{Y}}{\pi} - \frac{n(1 - \bar{Y})}{1 - \pi} = 0 \quad \Rightarrow \quad \hat{\pi}_{\text{MLE}} = \bar{Y}$$

Example:

$Y_i \sim$ **Bern**(π_i) **with** $\pi_i = g^{-1}(X_i'\beta)$

Generalized Linear Models

Suppose we allow the probability of "success" π_i in our Bernoulli model to depend on covariates X_i.

Generalized linear models (GLMs) propose a "link" function $g(\cdot)$ such that $g(\mathbb{E}[Y_i|X_i]) = X_i'\beta$

We will typically consider the inverse-link function, i.e., $\mathbb{E}[Y_i|X_i] = g^{-1}(X_i'\beta)$

For the Bernoulli model, we want $\pi_i \in [0,1]$, but $X_i'\beta \in \mathbb{R}$.

One popular link function gives rise to the Logit Regression Model:

- Logit link: $g(\pi_i) = \text{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = X_i'\beta$
- Inverse Logit = Logistic: $\pi_i = \frac{1}{1+\exp\{-X_i'\beta\}}$

Another popular link function gives rise to the Probit Regression Model:

- Probit link: $g(\pi) = \Phi^{-1}(\pi_i) = X_i'\beta$ where $\Phi(\cdot)$ is the standard normal CDF
- Inverse Probit = Normal CDF: $\pi_i = \Phi(X_i'\beta)$

MLE for $Y_i \sim \text{Bern}(\pi_i)$ with $\pi_i = g^{-1}(X_i'\beta)$

Suppose $f(y|\pi_i) = \pi_i^y \times (1 - \pi_i)^{1-y}$ with $\pi_i = g^{-1}(X_i'\beta)$
where $g^{-1}(\cdot)$ is the logistic (ie, inverse-logit) function.

Then $\log f(y|\beta) = y \log\left(g^{-1}(X'\beta)\right) + (1 - y) \log\left(1 - g^{-1}(X'\beta)\right)$

The log-likelihood function is:

$$\ell_n(\pi) = \sum_{i=1}^{n} [Y_i \log(\pi_i) + (1 - Y_i) \log(1 - \pi_i)] \quad \text{with} \quad \pi_i = \frac{1}{1 + \exp\{-X_i'\beta\}}$$

You can make some progress on the algebra, but it's easiest to just throw the log-likelihood function into an optimizer.

Code for Logit and Probit Log-Likelihoods

Logit

```
# by hand
ll1 <- function(beta, X, y) {
    pi_i <- 1 / (1 + exp(-1 * X %*% beta))
    ll <- sum(y*log(pi_i) + (1-y)*log(1-pi_i))
    return(ll)
}

out_logit <- optim(par=rep(0,k), fn=ll1,
                   X=X, y=y,
                   control=list(fnscale=-1))

out_logit$par

# built-in
glm(y ~ x1 + x2, data=df,
    family=binomial(link='logit'))
```

Probit

```
# by hand
ll2 <- function(beta, X, y) {
    pi_i <- pnorm(X %*% beta)
    ll <- sum(y*log(pi_i) + (1-y)*log(1-pi_i))
    return(ll)
}

out_probit <- optim(par=rep(0,k), fn=ll2,
                    X=X, y=y,
                    control=list(fnscale=-1))

out_probit$par

# built-in
glm(y ~ x1 + x2, data=df,
    family=binomial(link='probit'))
```

Consumer Search Example

A Model of Sequential Consumer Search

Suppose consumer i gets utility u from purchasing vehicle j:

$$u_{ij} = \underbrace{x_j'\beta + \eta_{ij}}_{\delta_{ij}} + \varepsilon_{ij}$$

- x_j are observable vehicle characteristics
- $\eta_{ij} \sim \mathrm{N}(0, 1)$ are preferences known only to the consumer
- $\varepsilon_{ij} \sim \mathrm{N}(0, \sigma)$ are consumer-product specific match values
- Consumer knows $F_\varepsilon(\varepsilon)$, but must search to discover ε_{ij}
- Search costs $c_{ij} = \exp\{\gamma_0 + d_{ij}\gamma_1\}$ where d_{ij} is distance
- Perfect recall, costless revisits, no outside option

Reservation Utilities

Weitzman (1979) characterizes optimal behavior via "reservation utilities"

The marginal benefit from an additional search:

$$B_{ij}(u_i^*) = \int_{u_i^*}^{\infty} (u_{ij} - u_i^*)\, f_{u_{ij}}(u_{ij})\, du_{ij}$$

Define a reservation utility (z_{ij}) by equating marginal benefit and marginal cost:

$$c_{ij} = B_{ij}(z_{ij}) \quad \Longrightarrow \quad c_{ij} = \left[1 - \Phi(\zeta_{ij})\right] \times \left[\frac{\phi(\zeta_{ij})}{1 - \Phi(\zeta_{ij})} - \zeta_{ij}\right] \times \sigma$$

Where $\zeta_{ij} = \left(z_{ij} - \delta_{ij}\right) / \sigma$

Rational (Optimal) Behavior

Weitzman (1979) shows that a consumer acting optimally will search alternatives in order of descending reservation utilities, continuing until the maximum realized utility of the searched alternatives is higher than the reservation utility of the next-to-be-searched alternative

For a consumer who makes K searches,

- **Continuation:** $z_k \geq \max_{h<k} \{u_h\}$ for $k = 2, \ldots, K$
- **Selection:** $z_1 \geq z_2 \geq \ldots \geq z_K \geq \max_{h>K} \{z_h\}$
- **Stopping:** $\max_{k \leq K} \{u_k\} \geq \max_{h>K} \{z_h\}$
- **Choice:** $u_{j^*} = \arg\max_{k \leq K} \{u_k\}$

Example

	Year-Make-Model	Dealership	z_{ij}	u_{ij}
1	2016 Honda Odyssey	First TX	10	6.2
2	2016 Toyota Sienna	Round Rock Toyota	9	6.7
3	2016 Honda Odyssey	Round Rock Honda	8	7.5
4	2016 Honda Odyssey	Howdy Honda	7	–
5	2016 Nissan Quest	Clay Cooley	6	–
6	2016 Nissan Quest	Round Rock Nissan	5	–
7	2016 Toyota Sienna	Toyota South Austin	4	–

Likelihood

Individual Likelihood

$$L_i(\beta,\gamma) = \int \mathbb{1}\Big[z_{ij} \geq \max_{h<j}\{u_h\} \text{ for } j = 2,\ldots,K_i \bigcap$$
$$z_{ij} = \arg\max_{k>j}\{z_{ik}\} \text{ for } j = 1,\ldots,K_i \bigcap$$
$$\max_{h\leq K_i}\{u_{ih}\} \geq \max_{k>K_i}\{z_{ik}\} \bigcap$$
$$u_{ij_i^*} = \arg\max_{h\leq K_i}\{u_{ih}\}\Big]\, dF(\eta,\varepsilon)$$

Total Likelihood

$$L(\beta,\gamma) = \prod_{i=1}^{N} L_i(\beta,\gamma)$$

Estimation via KSF MLE

We approximate the integral with an average, and we logit-transform the probabilities in order to smooth them:

For example, the continuation rule becomes $\nu_{1,j} = z_{ij} - \max_{h \leq j} \{u_{ih}\}$ for $j = 2, \ldots, K_i$

Calculate L_i for one set of draws (q) for consumer i:

$$\tilde{L}_i^q = \left(1 + \sum_{j=2}^{K_i} e^{-\lambda \nu_{1,j}} + \sum_{j=1}^{K_i} e^{-\lambda \nu_{2,j}} + e^{-\lambda \nu_3} + e^{-\lambda \nu_4}\right)^{-1}$$

Average over draws and consumers:

$$\log\left(\tilde{L}\right) = \sum_{i=1}^{N} \log\left(\frac{1}{Q} \sum_{q=1}^{Q} \tilde{L}_i^q\right)$$

Results

	Model (i)		Model (ii)		Model (iii)	
	Coef.	Std. Err.	Coef.	Std. Err.	Coef.	Std. Err.
Preference Parameters						
MSRP (in $10,000)	−0.182**	0.018	−0.186**	0.019	−0.202**	0.020
Chevrolet	−0.807**	0.035	−0.787**	0.034	−0.803**	0.038
Ford	−0.602**	0.037	−0.579**	0.039	−0.578**	0.040
Nissan	−0.068**	0.038	−0.063	0.040	−0.073	0.039
Toyota	0.047	0.031	0.072**	0.030	0.097**	0.033
Large Vehicle x Chevrolet	0.611**	0.059	0.611**	0.060	0.670**	0.066
Large Vehicle x Ford	0.886**	0.056	0.823**	0.060	1.020**	0.064
Large Vehicle x Toyota	0.028	0.053	0.011	0.054	0.046	0.058
Horsepower (in 100)	−0.473**	0.030	−0.501**	0.031	−0.518**	0.034
Engine Size	0.132**	0.019	0.146**	0.018	0.151**	0.021
MPG	0.042**	0.005	0.042**	0.006	0.044**	0.006
MPG x Large Vehicle	0.005	0.003	0.006	0.003	0.004	0.004
Log Search Cost Parameters						
Intercept	0.361**	0.018	0.273**	0.043	2.250**	0.040
Urban			0.203**	0.048	−0.055**	0.015
Distance (in Miles)			0.017**	0.001	0.006**	0.000
Urban x Distance			0.002**	0.001	0.003**	0.000
Match-Value Standard Deviation						
Sigma	1.00		1.00		8.16**	0.310
Number of Consumers	6,511		6,511		6,511	
Number of Products	175,840		175,840		175,840	
Log-Likelihood	−20,765		−18,362		−17,652	
BIC	41,645		36,872		35,453	

Note: Asterisks indicate statistical significance at the 95% confidence level. The base brand is Honda. Optimization via BFGS with relative tolerance convergence criterion set to 1e-6. Simulated likelihood using Q=1,000 independent random draws of the random utility error and the match-value distribution. Search costs are parameterized as $c_{ij} = \exp\{\gamma_0 + \gamma_1 \text{Urban} + \gamma_2 \text{Distance} + \gamma_3 \text{Urban} \times \text{Distance}\}$.

Next Time

- Properties of ML Estimators
  - Invariance
  - Consistency
  - Asymptotic Normality
  - Efficiency
- Logit Example