EMSE 6035: Marketing of Technology

Willingness to Pay & Market Simulation

John Paul Helveston, Ph.D.
Assistant Professor
Engineering Management & Systems Engineering
The George Washington University

Background: Estimate Utility Model Coefficients Using Maximum Likelihood Estimation

$$\tilde{u}_j = \boldsymbol{\beta}'\mathbf{x}_j + \tilde{\varepsilon}_j$$

$$= \boxed{\beta_1} x_{j1} + \boxed{\beta_2} x_{j2} + \ldots + \tilde{\varepsilon}_j$$

Weights that denote the *relative* value of attributes $x_{j1}, x_{j2}, \ldots$

Obtain estimates of $\widehat{\boldsymbol{\beta}}$ using maximum likelihood estimation:

Have units of "utility"

Est.	Std. Err.
$\hat{\beta}_1$	σ_1
$\hat{\beta}_2$	σ_2
$\vdots$	$\vdots$
$\hat{\beta}_m$	σ_m

Willingness to Pay (WTP)

Price

Non-price attributes

$$\tilde{u}_j = \alpha p_j + \boldsymbol{\beta}'\mathbf{x}_j + \tilde{\varepsilon}_j$$

Converts change in utility to change in $

WTP: $\boldsymbol{\omega} = -\dfrac{\widehat{\boldsymbol{\beta}}}{\hat{\alpha}}$

$$= -\left[\frac{\hat{\beta}_1}{\hat{\alpha}}, \frac{\hat{\beta}_2}{\hat{\alpha}}, \dots, \frac{\hat{\beta}_n}{\hat{\alpha}}\right]$$

Estimate using `logitr` library

Model Space	Model Specification	Parameters of interest	Units
Preference:	$\tilde{u}_j = \alpha p_j + \boldsymbol{\beta}'\mathbf{x}_j + \tilde{\varepsilon}_j$	$\alpha, \boldsymbol{\beta}$	"Utility"
WTP:	$\tilde{u}_j = \alpha(\boldsymbol{\omega}'\mathbf{x}_j + p_j) + \tilde{\varepsilon}_j$	$\boldsymbol{\omega}$	Currency ($)

Market Share Predictions

From the logit model, the market share for alternative j is:

$$\widehat{P}_j = \frac{e^{\hat{v}_j}}{\sum_{k=1}^{J} e^{\hat{v}_k}}, \quad \text{where } J \text{ is the number of alternatives}$$

$$\hat{v}_j = \widehat{\boldsymbol{\beta}}' \mathbf{x}_j = \hat{\beta}_1 x_{j1} + \hat{\beta}_2 x_{j2} + \ldots + \hat{\beta}_n x_{jn}$$

Market Share Predictions

$$\hat{P}_j = \frac{e^{\hat{v}_j}}{\sum_{k=1}^{J} e^{\hat{v}_k}}$$

To compute market shares:

1. Define the market, $\mathbf{x}$.

Example in *R*:

In *R*: Use the `matrix()` and `c()` functions to create X:

```
X = matrix(c(
    15, 20,  0,
    30, 100, 1,
    20, 40,  0),
    ncol=3, byrow=TRUE)
```

Each column is an attribute

There are *n* attributes

Each row is an alternative

There are *J* alternatives

Alt	Att_1	Att_2	…	Att_*n*
1	x_{11}	x_{12}	…	x_{1n}
2	x_{21}	x_{22}	…	x_{2n}
⋮	⋮	⋮	…	⋮
J	x_{J1}	x_{J2}	…	x_{Jn}

Market Share Predictions

$$\widehat{P}_j = \frac{e^{\hat{v}_j}}{\sum_{k=1}^{J} e^{\hat{v}_k}}$$

To compute market shares:

1. Define the market, $\mathbf{x}$.
2. Compute $\hat{v}_j$ for each alternative.

$$\widehat{\boldsymbol{\beta}} = [\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_n]$$

Alt	Att_1	Att_2	…	Att_*n*
1	x_{11}	x_{12}	…	x_{1n}
2	x_{21}	x_{22}	…	x_{2n}
$\vdots$	$\vdots$	$\vdots$	…	$\vdots$
J	x_{J1}	x_{J2}	…	x_{Jn}

$$\hat{v}_1 = \hat{\beta}_1 x_{11} + \hat{\beta}_2 x_{12} + \ldots + \hat{\beta}_n x_{1n}$$

$$\hat{v}_2 = \hat{\beta}_1 x_{21} + \hat{\beta}_2 x_{22} + \ldots + \hat{\beta}_n x_{2n}$$

$$\vdots$$

$$\hat{v}_j = \hat{\beta}_1 x_{j1} + \hat{\beta}_2 x_{j2} + \ldots + \hat{\beta}_n x_{jn}$$

Market Share Predictions

$$\hat{P}_j = \frac{e^{\hat{v}_j}}{\sum_{k=1}^{J} e^{\hat{v}_k}}$$

<u>To compute market shares</u>:

1. Define the market, $\mathbf{x}$.
2. Compute $\hat{v}_j$ for each alternative.

$$\hat{v} = \widehat{\boldsymbol{\beta}}'\mathbf{x}$$

$$= \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \dots & \vdots \\ x_{J1} & x_{J2} & \dots & x_{Jn} \end{bmatrix} \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_n \end{bmatrix}$$

$$= \begin{bmatrix} \hat{\beta}_1 x_{11} + \hat{\beta}_2 x_{12} + \dots + \hat{\beta}_n x_{1n} \\ \hat{\beta}_1 x_{21} + \hat{\beta}_2 x_{22} + \dots + \hat{\beta}_n x_{2n} \\ \vdots \\ \hat{\beta}_1 x_{J1} + \hat{\beta}_2 x_{J2} + \dots + \hat{\beta}_n x_{Jn} \end{bmatrix} = \begin{bmatrix} \hat{v}_1 \\ \hat{v}_2 \\ \vdots \\ \hat{v}_J \end{bmatrix}$$

In *R*: Use %*% for matrix multiplication:

```
v_j = X %*% beta
```

not

```
v_j = X * beta
```

https://en.wikipedia.org/wiki/Matrix_multiplication

Market Share Predictions

$$\hat{P}_j = \frac{e^{\hat{v}_j}}{\sum_{k=1}^{J} e^{\hat{v}_k}}$$

<u>To compute market shares</u>:

1. Define the market, $\mathbf{x}$.
2. Compute $\hat{v}_j$ for each alternative.
3. Compute $e^{\hat{v}_j}$ for each alternative.

In *R*: Use exp(x) for e^x:

```
exp_v_j = exp(v_j)
```

Obs. Utility		Exp. Obs. Utility
$\hat{v}_1$		$e^{\hat{v}_1}$
$\hat{v}_2$	➡	$e^{\hat{v}_2}$
$\vdots$		$\vdots$
$\hat{v}_J$		$e^{\hat{v}_J}$

Market Share Predictions

$$\widehat{P}_j = \frac{e^{\hat{v}_j}}{\sum_{k=1}^{J} e^{\hat{v}_k}}$$

<u>To compute market shares</u>:

1. Define the market, $\mathbf{x}$.
2. Compute $\hat{v}_j$ for each alternative.
3. Compute $e^{\hat{v}_j}$ for each alternative.
4. Compute the denominator of the $\widehat{P}_j$ fraction by summing all the $e^{\hat{v}_j}$ terms.

Obs. Utility
$\hat{v}_1$
$\hat{v}_2$
$\vdots$
$\hat{v}_J$

Exp. Obs. Utility
$e^{\hat{v}_1}$
$e^{\hat{v}_2}$
$\vdots$
$e^{\hat{v}_J}$

Sum of this = Denominator

In *R*: Use sum() :

```
denom = sum(exp_v_j)
```

Market Share Predictions

$$\hat{P}_j = \frac{e^{\hat{v}_j}}{\sum_{k=1}^{J} e^{\hat{v}_k}}$$

<u>To compute market shares</u>:

1. Define the market, $\mathbf{x}$.
2. Compute $\hat{v}_j$ for each alternative.
3. Compute $e^{\hat{v}_j}$ for each alternative.
4. Compute the denominator of the $\hat{P}_j$ fraction by summing all the $e^{\hat{v}_j}$ terms.
5. Compute $\hat{P}_j$ for each alternative by dividing each $e^{\hat{v}_j}$ by the denominator.

Exp. Obs. Utility
$e^{\hat{v}_1}$
$e^{\hat{v}_2}$
$\vdots$
$e^{\hat{v}_J}$

➡

$\hat{P}_j$
$e^{\hat{v}_1}$/denom
$e^{\hat{v}_2}$/denom
$\vdots$
$e^{\hat{v}_J}$/denom

Market Share Predictions

$$\widehat{P}_j = \frac{e^{\hat{v}_j}}{\sum_{k=1}^{J} e^{\hat{v}_k}}$$

<u>To compute market shares</u>:

1. Define the market, $\mathbf{x}$.
2. Compute $\hat{v}_j$ for each alternative.
3. Compute $e^{\hat{v}_j}$ for each alternative.
4. Compute the denominator of the $\widehat{P}_j$ fraction by summing all the $e^{\hat{v}_j}$ terms.
5. Compute $\widehat{P}_j$ for each alternative by dividing each $e^{\hat{v}_j}$ by the denominator.

Comes from estimated model

R code:

```
X = matrix(…)
beta = c(beta1, beta2, …)

v_j = X %*% beta

exp_v_j = exp(v_j)

denom = sum(exp_v_j)

P_j = exp_v_j / denom
```

Practice Question 1

Suppose we estimate the following utility model describing preferences for cars:

$$\tilde{u}_j = \alpha p_j + \beta_1 x_j^{\mathrm{mpg}} + \beta_2 x_j^{\mathrm{elec}} + \varepsilon_j$$

where the variables are:

p_j	Price in USD $1,000
x_j^{mpg}	Fuel economy in miles per gallon
x_j^{elec}	Variable that takes 1 if the car is an electric car and 0 otherwise

The estimated model produces the following coefficients:

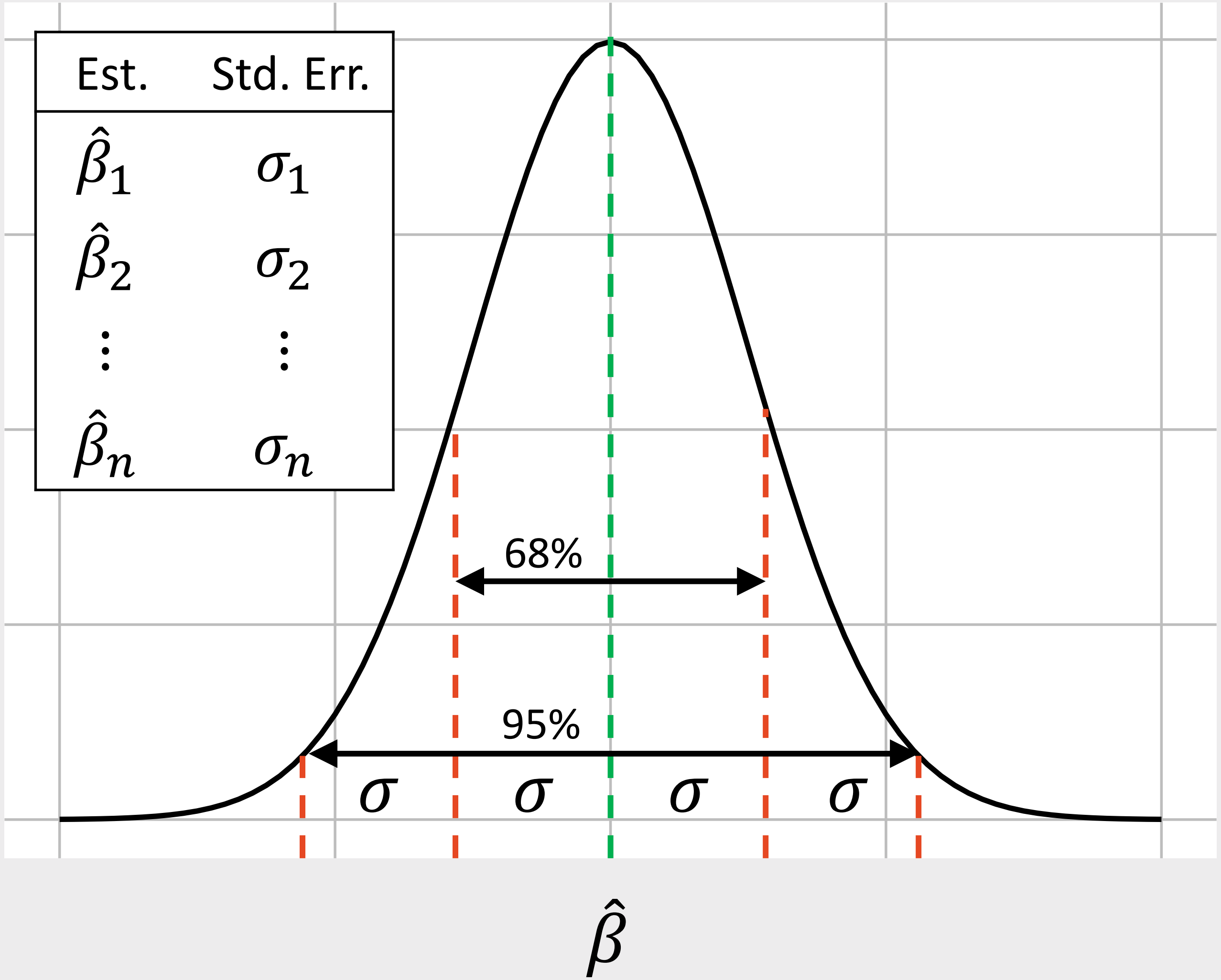
Parameter	Coef.
α	-0.7
β_1	0.1
β_2	-4.0

a) Use the estimated coefficients to compute the WTP for fuel economy and electric car vehicle type.
b) Use the estimated coefficients to compute market shares for the alternatives in the following market:

Alternative	price	mpg	elec.
1	15	20	0
2	30	100	1
3	20	40	0

Handling uncertainty with simulation

Use $\hat{\beta}$ and σ to generate samples of $N(\hat{\beta}, \sigma)$



Take sample draws of $\hat{\beta}$ to simulate uncertainty

Example in *R*:

```
> beta = 0.5
> sigma = 0.1
> draws = rnorm(10^5, beta, sigma)
>
> mean(draws)
[1] 0.4996797
> sd(draws)
[1] 0.1001574
>
> c(beta - 2*sigma, beta + 2*sigma)
[1] 0.3 0.7
> quantile(draws, c(0.025, 0.975))
     2.5%     97.5%
0.3044208 0.6964306
```

Sampling $\widehat{\boldsymbol{\beta}}$

$$\boldsymbol{\beta} \sim N\left(\widehat{\boldsymbol{\beta}}, \Sigma\right)$$

$$\begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_n \end{bmatrix} \qquad \underbrace{\begin{bmatrix} \sigma_{11}^2 & \cdots & \sigma_{1n}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{1n}^2 & \cdots & \sigma_{jn}^2 \end{bmatrix}}$$

$$-\Big[\underbrace{\nabla_{\boldsymbol{\beta}}^2 \ln(\mathcal{L})}_{\text{Hessian}}\Big]^{-1}$$

Example in *R*:

```
> library(MASS)
>
> beta = c(price = -0.7, mpg = 0.1, elec=-4.0)
> hessian = matrix(c(
+     -6000,   50,   60,
+        50, -700,   50,
+        60,   50, -300),
+     ncol=3, byrow=T)
> covariance = -1*(solve(hessian))
> draws  = mvrnorm(10^5, beta, covariance)
> head(draws)
          price        mpg      elec
[1,] -0.7184210 0.18428285 -3.951629
[2,] -0.6999711 0.16873388 -3.918036
[3,] -0.7192076 0.11657494 -3.971442
[4,] -0.6851790 0.10707172 -4.039762
[5,] -0.7048889 0.14175661 -4.050028
[6,] -0.6917784 0.09615243 -4.083626
```

Willingness to Pay Using Draws of $\widehat{\boldsymbol{\beta}}$

Mean WTP:

$$\boldsymbol{\omega} = -\frac{\widehat{\boldsymbol{\beta}}}{\hat{\alpha}}$$

$$= -\left[\frac{\hat{\beta}_1}{\hat{\alpha}}, \frac{\hat{\beta}_2}{\hat{\alpha}}, \dots, \frac{\hat{\beta}_n}{\hat{\alpha}}\right]$$

WTP with Uncertainty:

$$\begin{bmatrix} \boldsymbol{\omega}^1 \\ \boldsymbol{\omega}^2 \\ \vdots \\ \boldsymbol{\omega}^N \end{bmatrix} = -\begin{bmatrix} \hat{\beta}_1^1/\hat{\alpha}^1 & \hat{\beta}_2^1/\hat{\alpha}^1 & \dots & \hat{\beta}_n^1/\hat{\alpha}^1 \\ \hat{\beta}_1^2/\hat{\alpha}^2 & \hat{\beta}_2^2/\hat{\alpha}^2 & \dots & \hat{\beta}_n^2/\hat{\alpha}^2 \\ \vdots & \vdots & \vdots & \vdots \\ \hat{\beta}_1^N/\hat{\alpha}^N & \hat{\beta}_2^N/\hat{\alpha}^N & \dots & \hat{\beta}_n^N/\hat{\alpha}^N \end{bmatrix}$$

Example in *R*:

```
> wtp = -1*(draws[,2:3] / draws[,1])
> head(wtp)
           mpg      elec
[1,] 0.2565109 -5.500437
[2,] 0.2410584 -5.597425
[3,] 0.1620880 -5.521969
[4,] 0.1562682 -5.895921
[5,] 0.2011049 -5.745625
[6,] 0.1389931 -5.903084

> mpg_draws = wtp[,1]
> mean(mpg_draws)
[1] 0.1428379
> sd(mpg_draws)
[1] 0.05442786
> quantile(mpg_draws, c(0.025, 0.975))
      2.5%      97.5%
0.03603906 0.24957039
```

Market Shares Using Draws of $\widehat{\boldsymbol{\beta}}$

Mean Shares:

$$\widehat{P}_j = \frac{e^{\widehat{v}_j}}{\sum_{k=1}^{J} e^{\widehat{v}_k}}$$

Shares with Uncertainty:

$$\begin{bmatrix} P_j^1 \\ P_j^2 \\ \vdots \\ P_j^N \end{bmatrix} = \begin{bmatrix} e^{\hat{v}_j^1} \Big/ \sum e^{\hat{v}_k^1} \\ e^{\hat{v}_j^2} \Big/ \sum e^{\hat{v}_k^2} \\ \vdots \\ e^{\hat{v}_j^N} \Big/ \sum e^{\hat{v}_k^N} \end{bmatrix}$$

Example in *R*:

```
> shares = simulateMarketShares(model,
+            X, numDraws=10^4)
> shares
         mean         lower        upper
0.8784375517 0.8295211831 0.917781146
0.0008811825 0.0001743651 0.002643378
0.1206812658 0.0820445690 0.167899853
```

Code for `simulateMarketShares()` function:
https://github.com/jhelvy/mlogitCars

Practice Question 2

Suppose we estimate the following utility model describing preferences for cars:

$$\tilde{u}_j = \alpha p_j + \beta_1 x_j^{\text{mpg}} + \beta_2 x_j^{\text{elec}} + \varepsilon_j$$

The estimated model produces the following coefficients:

Parameter	Coef.
α	-0.7
β_1	0.1
β_2	-4.0

Hessian		
-6000	50	60
50	-700	50
60	50	-300

a) Generate 10,000 draws of the model coefficients using the estimated coefficients and hessian. Use the `mvrnorm()` function from the `MASS` library.
b) Use the draws to compute the mean WTP and 95% confidence intervals of WTP for fuel economy and electric car vehicle type.