Week 9: *Uncertainty*

EMSE 6035: Marketing Analytics for Design Decisions

John Paul Helveston

October 27, 2021

Quick correction from last week

Observations - Height of students (inches):

```
#>  [1] 65 69 66 67 68 72 68 69 63 70
```

a) Let's say we know that the height of students, $\tilde{x}$, in a classroom follows a normal distribution. A professor obtains the above height measurements students in her classroom. What is the log-likelihood that $\tilde{x} \sim \mathcal{N}(68, 4)$? In other words, compute $\ln \mathcal{L}(\mu = 68, \sigma = 4)$.

b) Compute the log-likelihood function using the same standard deviation $(\sigma = 4)$ but with the following different values for the mean, $\mu : 66, 67, 68, 69, 70$. How do the results compare? Which value for μ produces the highest log-likelihood?

Computing the *likelihood*

Load the data

```
x <- c(65, 69, 66, 67, 68, 72, 68, 69, 63,
70)
```

Compute the value of f(x) for each x

```
f_x <- dnorm(x, 68, 4)
```

Likelihood is the product of values in `f_x`

```
prod(f_x)
```

```
#> [1] 1.447528e-11
```

Computing the *log-likelihood*

Take the log of the likelihood

```
log(prod(f_x))
```

```
#> [1] -24.95858
```

The way we typically compute the log-likelihood is by summing up the log of the values in `f_x`

```
sum(log(f_x))
```

```
#> [1] -24.95858
```

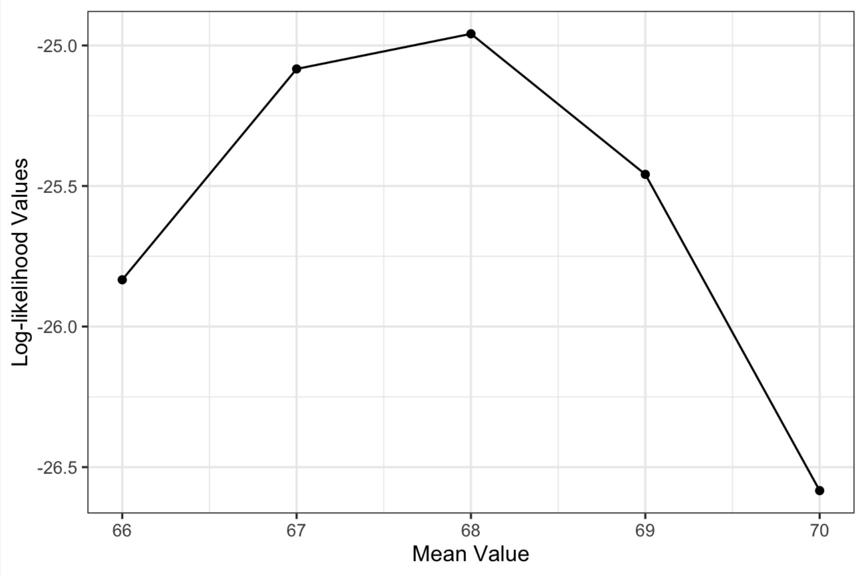
```r
library(tidyverse)

# Create a vectors of values for the mean
means <- c(66, 67, 68, 69, 70)

# Compute the likelihood using different
values for the mean:
L1 <- sum(log(dnorm(x, means[1], 4)))
L2 <- sum(log(dnorm(x, means[2], 4)))
L3 <- sum(log(dnorm(x, means[3], 4)))
L4 <- sum(log(dnorm(x, means[4], 4)))
L5 <- sum(log(dnorm(x, means[5], 4)))
logLiks <- c(L1, L2, L3, L4, L5)

# Plot the result:
df <- data.frame(means, logLiks)

df %>%
  ggplot(aes(x = means, y = logLiks)) +
  geom_line() +
  geom_point() +
  theme_bw() +
  labs(
      x = "Mean Value",
      y = "Log-likelihood Values"
  )
```



Week 9: *Uncertainty*

1. Computing uncertainty

2. Reshaping data

BREAK

3. Cleaning pilot data

4. Estimating pilot data models

Week 9: *Uncertainty*

1. Computing uncertainty

2. Reshaping data

BREAK

3. Cleaning pilot data

4. Estimating pilot data models

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 $x_{j1}, x_{j2}, \ldots$

Estimate β_1 , β_2 , ... , by minimizing the negative log-likelihood function:

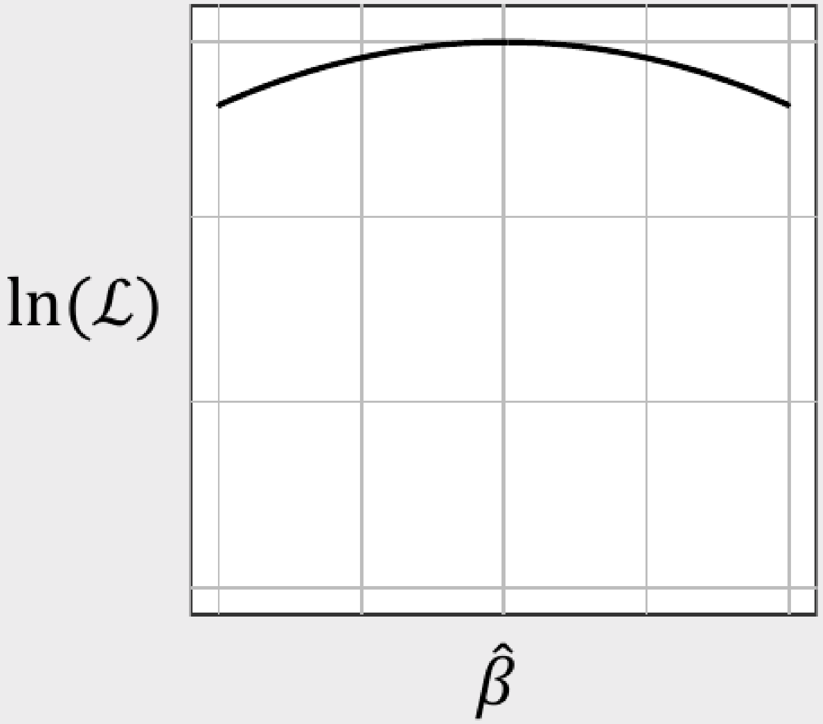
minimize $- \ln(\mathcal{L}) = -\sum_{j=1}^{J} y_j \ln[P_j(\boldsymbol{\beta}|\mathbf{x})]$

with respect to $\boldsymbol{\beta}$

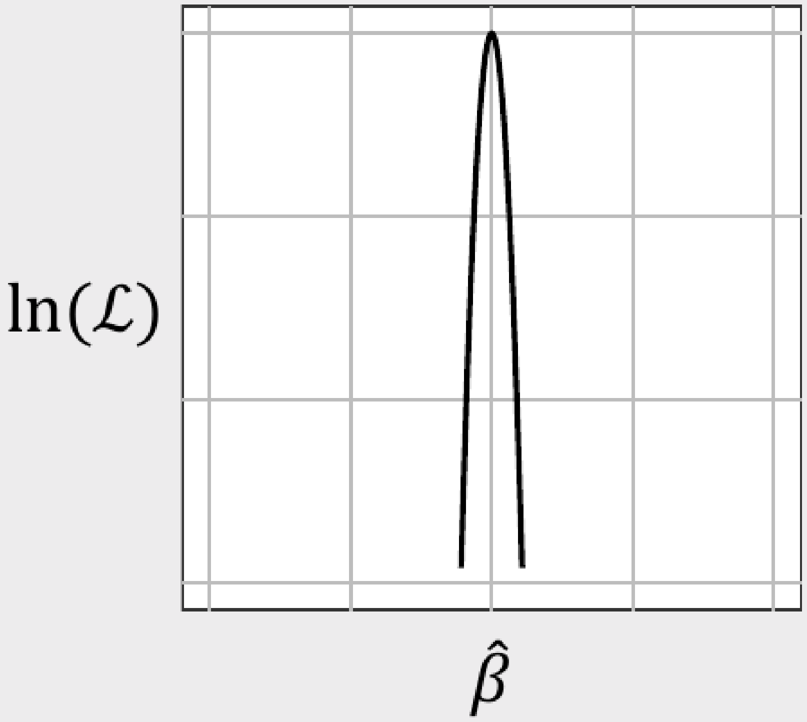
$y_j = 1$ if alternative j was chosen
$y_j = 0$ if alternative j was not chosen

The certainty of $\widehat{\boldsymbol{\beta}}$ is inversely related to the curvature of the log-likelihood function

Greater variance in $\ln(\mathcal{L})$,
Less certainty in $\hat{\beta}$



Less variance in $\ln(\mathcal{L})$,
Greater certainty in $\hat{\beta}$



The *curvature* of the log-likelihood function is related to the hessian

$$\sum\nolimits_{\boldsymbol{\beta}} = -\left[\overbrace{\nabla^2_{\boldsymbol{\beta}} \ln(\mathcal{L})}^{\text{Hessian}}\right]^{-1}$$

Covariance of $\widehat{\boldsymbol{\beta}}$

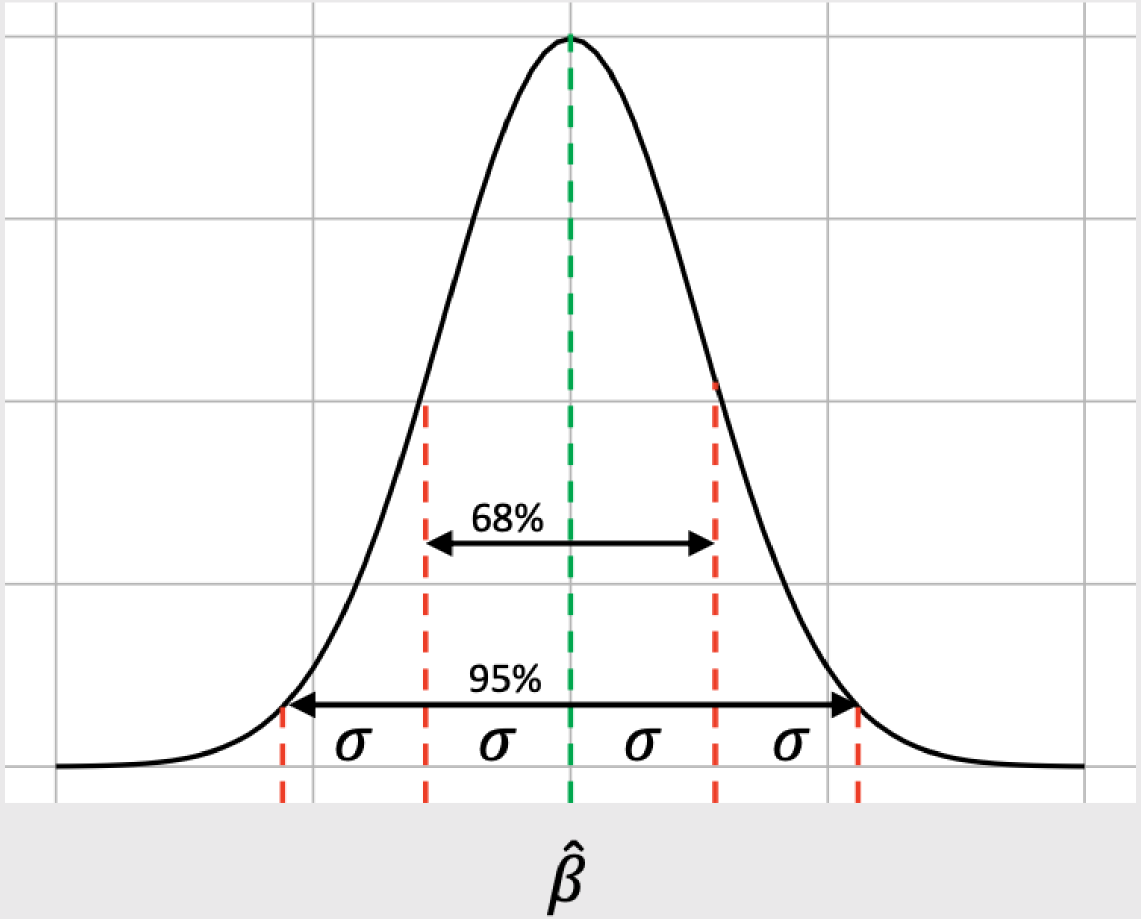
The *curvature* of the log-likelihood function is related to the hessian

$$\sum\nolimits_{\boldsymbol{\beta}} = -\,\overbrace{\left[\nabla^2_{\boldsymbol{\beta}} \ln(\mathcal{L})\right]}^{\text{Hessian}}{}^{-1} = \begin{bmatrix} \sigma^2_{11} & \cdots & \sigma^2_{m1} \\ \vdots & \ddots & \vdots \\ \sigma^2_{1n} & \cdots & \sigma^2_{mn} \end{bmatrix}$$

Covariance of $\widehat{\boldsymbol{\beta}}$

Usually report parameter uncertainty ("standard errors") with σ values

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



A 95% confidence interval is approximately $[\hat{\beta} - 2\sigma, \hat{\beta} + 2\sigma]$

Practice Question 1

Suppose we estimate a model and get the following results:

$$\hat{\beta} = \begin{bmatrix} -0.4 \\ 0.5 \end{bmatrix}$$

$$\nabla^2_\beta \ln(\mathcal{L}) = \begin{bmatrix} -6000 & 60 \\ 60 & -700 \end{bmatrix}$$

a) Use the hessian to compute the standard errors for $\hat{\beta}$

b) Use the standard errors to compute a 95% confidence interval around $\hat{\beta}$

Simulating uncertainty

We can use the coefficients and hessian from a model to obtain draws that reflect parameter uncertainty

```r
beta <- c(-0.7, 0.1, -4.0)

hessian <- matrix(c(
    -6000,   50,   60,
       50, -700,   50,
       60,   50, -300),
    ncol = 3, byrow = TRUE)
```

```r
covariance <- -1*solve(hessian)
draws <- MASS::mvrnorm(10^5, beta,
covariance)

head(draws)
```

```
#>             [,1]      [,2]      [,3]
#> [1,] -0.6946433 0.1206494 -3.973694
#> [2,] -0.7128098 0.1381762 -3.975379
#> [3,] -0.6941685 0.1334979 -4.002586
#> [4,] -0.7166425 0.1122484 -4.079662
#> [5,] -0.6983785 0.1447645 -4.033314
#> [6,] -0.7060643 0.1088229 -3.999648
```

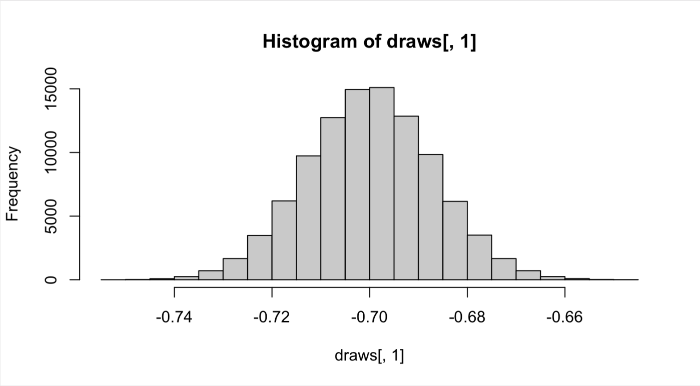
Simulating uncertainty

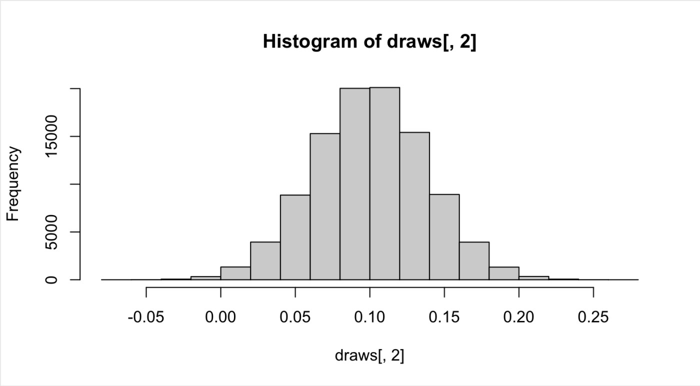
We can use the coefficients and hessian from a model to obtain draws that reflect parameter uncertainty

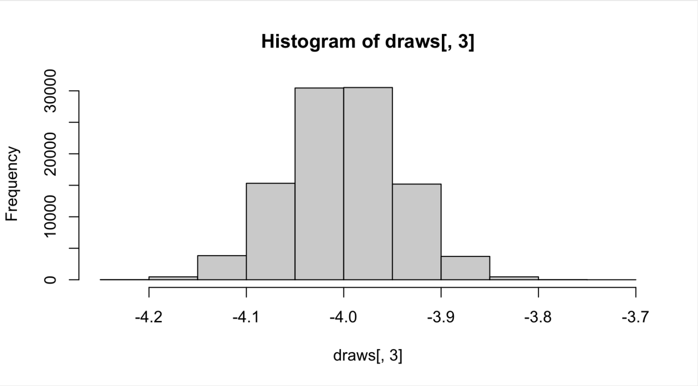
```r
hist(draws[, 1])
```

```r
hist(draws[, 2])
```

```r
hist(draws[, 3])
```







Practice Question 2

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

$$u_j = \alpha p_j + \beta_1 x_j^{mpg} + \beta_2 x_j^{elec} + \varepsilon_j$$

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 and 95% confidence intervals of each parameter estimate.

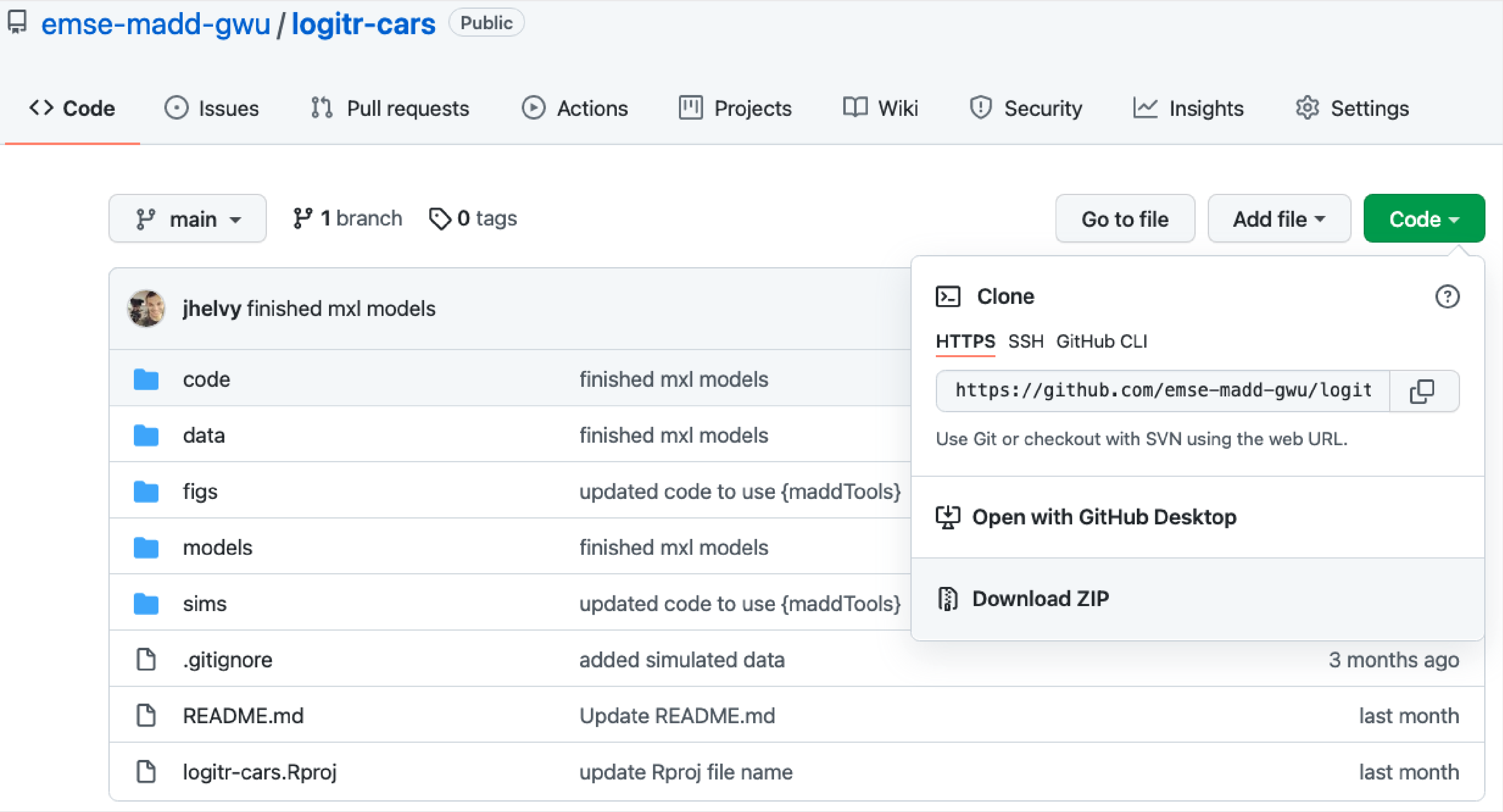
The estimated model produces the following results:

Parameter	Coefficient
α	-0.7
β_1	0.1
β_2	-0.4

Hessian:

$$\begin{bmatrix} -6000 & 50 & 60 \\ 50 & -700 & 50 \\ 60 & 50 & -300 \end{bmatrix}$$

Download the logitr-cars repo from GitHub



Computing and visualizing uncertainty

1. Open `logitr-cars`
2. Open `code/5.1-uncertainty.R`

Week 9: *Uncertainty*

1. Computing uncertainty
2. Reshaping data

BREAK

3. Cleaning pilot data
4. Estimating pilot data models

Names, Values, and Observations

- Variable **Name**: The name of something you can measure
- Variable **Value**: One instance of a measured variable
- **Observation**: A set of associated measurements across multiple variables

```
head(fed_spend_long)
```

```
#> # A tibble: 6 × 3
#>   department  year rd_budget_mil
#>   <chr>      <dbl>         <dbl>
#> 1 DOD         1976         35696
#> 2 NASA        1976         12513
#> 3 DOE         1976         10882
#> 4 HHS         1976          9226
#> 5 NIH         1976          8025
#> 6 NSF         1976          2372
```

"Long" format data

- Each **variable** has its own **column**
- Each **observation** has its own **row**

"Long" format data

- Each **variable** has its own **column**
- Each **observation** has its own **row**

```
#> # A tibble: 6 × 3
#>   department  year rd_budget_mil
#>   <chr>      <dbl>         <dbl>
#> 1 DOD         1976         35696
#> 2 NASA        1976         12513
#> 3 DOE         1976         10882
#> 4 HHS         1976          9226
#> 5 NIH         1976          8025
#> 6 NSF         1976          2372
```

"Long" format

```
#> # A tibble: 6 × 3
#>   department  year rd_budget_mil
#>   <chr>      <dbl>         <dbl>
#> 1 DOD         1976         35696
#> 2 NASA        1976         12513
#> 3 DOE         1976         10882
#> 4 HHS         1976          9226
#> 5 NIH         1976          8025
#> 6 NSF         1976          2372
```

"Wide" format

```
#> # A tibble: 6 × 8
#>    year   DHS   DOC   DOD   DOE   DOT   EPA   HHS
#>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1  1976     0   819 35696 10882  1142   968  9226
#> 2  1977     0   837 37967 13741  1095   966  9507
#> 3  1978     0   871 37022 15663  1156  1175 10533
#> 4  1979     0   952 37174 15612  1004  1102 10127
#> 5  1980     0   945 37005 15226  1048   903 10045
#> 6  1981     0   829 41737 14798   978   901  9644
```

"Long" format: variable names describe the values below them

"Long" format

```
#> # A tibble: 6 × 3
#>   department  year rd_budget_mil
#>   <chr>      <dbl>         <dbl>
#> 1 DOD         1976         35696
#> 2 NASA        1976         12513
#> 3 DOE         1976         10882
#> 4 HHS         1976          9226
#> 5 NIH         1976          8025
#> 6 NSF         1976          2372
```

"Wide" format

```
#> # A tibble: 6 × 8
#>    year   DHS   DOC   DOD   DOE   DOT   EPA   HHS
#>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1  1976     0   819 35696 10882  1142   968  9226
#> 2  1977     0   837 37967 13741  1095   966  9507
#> 3  1978     0   871 37022 15663  1156  1175 10533
#> 4  1979     0   952 37174 15612  1004  1102 10127
#> 5  1980     0   945 37005 15226  1048   903 10045
#> 6  1981     0   829 41737 14798   978   901  9644
```

Quick practice 1: "long" or "wide" format?

Description: Tuberculosis cases in various countries

```
#> # A tibble: 6 × 4
#>   country      year  cases population
#>   <chr>       <dbl>  <dbl>      <dbl>
#> 1 Afghanistan  1999    745   19987071
#> 2 Afghanistan  2000   2666   20595360
#> 3 Brazil       1999  37737  172006362
#> 4 Brazil       2000  80488  174504898
#> 5 China        1999 212258 1272915272
#> 6 China        2000 213766 1280428583
```

Quick practice 2: "long" or "wide" format?

Description: Word counts by character type in "Lord of the Rings" trilogy

```
#> # A tibble: 9 × 4
#>   Film                       Race   Female  Male
#>   <chr>                      <chr>   <dbl> <dbl>
#> 1 The Fellowship Of The Ring Elf      1229   971
#> 2 The Fellowship Of The Ring Hobbit     14  3644
#> 3 The Fellowship Of The Ring Man         0  1995
#> 4 The Return Of The King     Elf       183   510
#> 5 The Return Of The King     Hobbit      2  2673
#> 6 The Return Of The King     Man       268  2459
#> 7 The Two Towers             Elf       331   513
#> 8 The Two Towers             Hobbit      0  2463
#> 9 The Two Towers             Man       401  3589
```

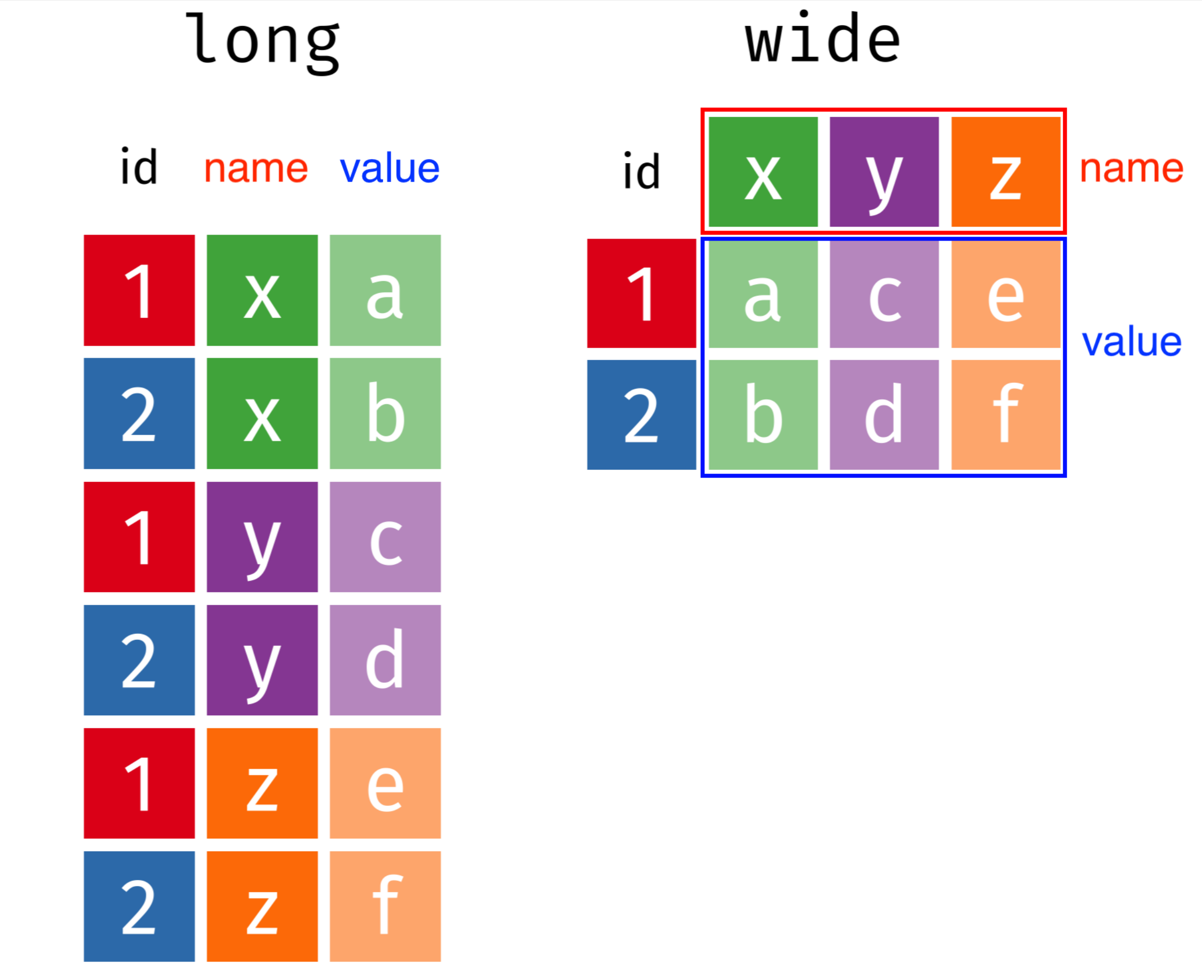
Quick practice 3: "long" or "wide" format?

Description: Word counts by character type in "Lord of the Rings" trilogy

```
#> # A tibble: 18 × 4
#>    Film                       Race   Gender Word_Count
#>    <chr>                      <chr>  <chr>       <dbl>
#>  1 The Fellowship Of The Ring Elf    Female       1229
#>  2 The Fellowship Of The Ring Elf    Male          971
#>  3 The Fellowship Of The Ring Hobbit Female         14
#>  4 The Fellowship Of The Ring Hobbit Male         3644
#>  5 The Fellowship Of The Ring Man    Female          0
#>  6 The Fellowship Of The Ring Man    Male         1995
#>  7 The Return Of The King     Elf    Female        183
#>  8 The Return Of The King     Elf    Male          510
#>  9 The Return Of The King     Hobbit Female          2
#> 10 The Return Of The King     Hobbit Male         2673
#> 11 The Return Of The King     Man    Female        268
#> 12 The Return Of The King     Man    Male         2459
#> 13 The Two Towers             Elf    Female        331
#> 14 The Two Towers             Elf    Male          513
#> 15 The Two Towers             Hobbit Female          0
#> 16 The Two Towers             Hobbit Male         2463
#> 17 The Two Towers             Man    Female        401
#> 18 The Two Towers             Man    Male         3589
```

Reshaping data with `pivot_longer()` and `pivot_wider()`

From "long" to "wide" with `pivot_wider()`



From "long" to "wide" with `pivot_wider()`

```r
head(fed_spend_long)
```

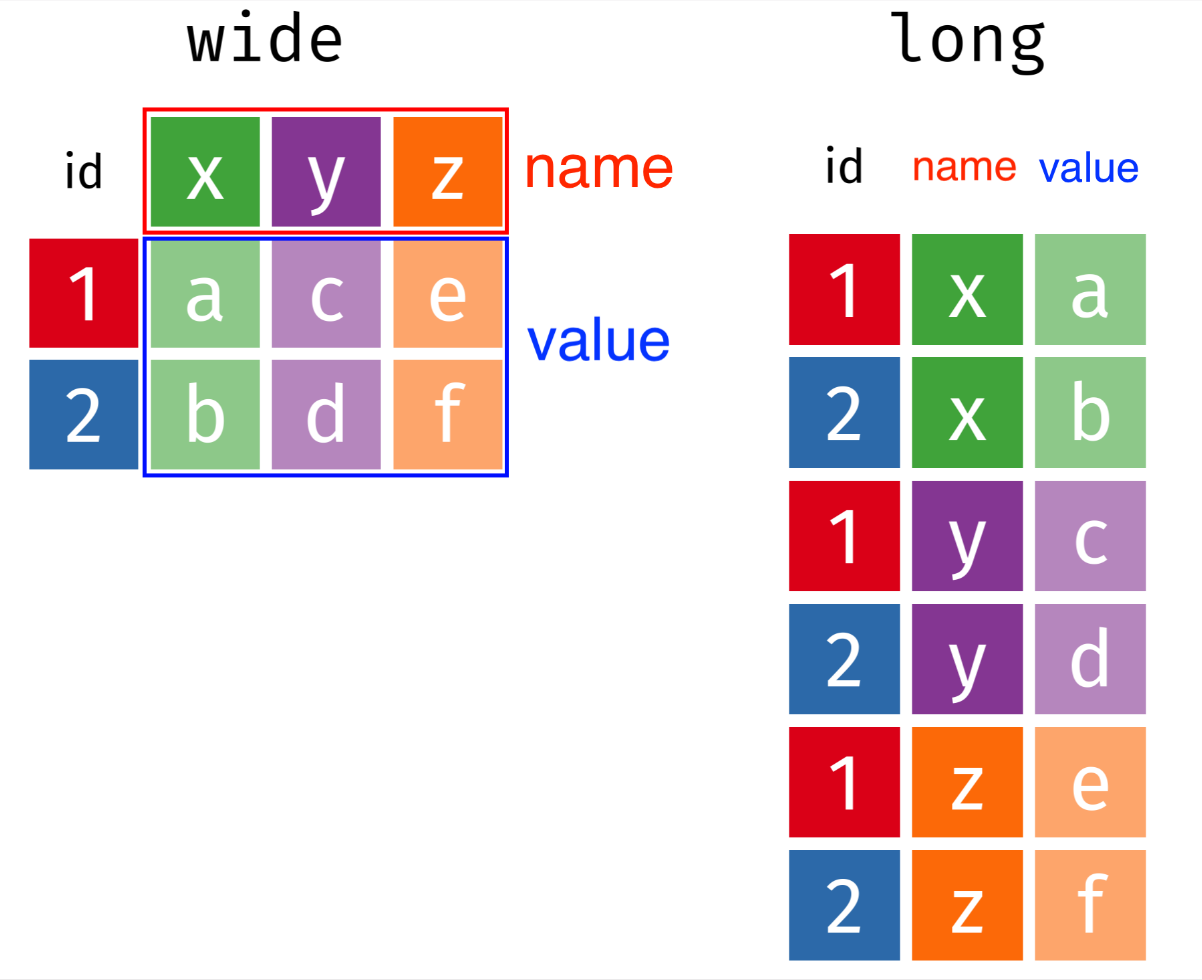
```
#> # A tibble: 6 × 3
#>   department  year rd_budget_mil
#>   <chr>      <dbl>         <dbl>
#> 1 DOD         1976         35696
#> 2 NASA        1976         12513
#> 3 DOE         1976         10882
#> 4 HHS         1976          9226
#> 5 NIH         1976          8025
#> 6 NSF         1976          2372
```

```r
fed_spend_wide <- fed_spend_long %>%
    pivot_wider(
        names_from = department,
        values_from = rd_budget_mil)

head(fed_spend_wide)
```

```
#> # A tibble: 6 × 7
#>    year   DHS   DOC   DOD   DOE   DOT   EPA
#>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1  1976     0   819 35696 10882  1142   968
#> 2  1977     0   837 37967 13741  1095   966
#> 3  1978     0   871 37022 15663  1156  1175
#> 4  1979     0   952 37174 15612  1004  1102
#> 5  1980     0   945 37005 15226  1048   903
#> 6  1981     0   829 41737 14798   978   901
```

From "wide" to "long" with `pivot_longer()`



From "wide" to "long" with `pivot_longer()`

```r
names(fed_spend_wide)
```

```
#>  [1] "year"     "DHS"      "DOC"      "DOD"      "DOE"      "DOT"      "EPA"
#>  "HHS"      "Interior" "NASA"     "NIH"      "NSF"      "Other"    "USDA"     "VA"
```

```r
fed_spend_long <- fed_spend_wide %>%
    pivot_longer(
        cols = DHS:VA,
        names_to = "department",
        values_to = "rd_budget_mil")

head(fed_spend_long)
```

```
#> # A tibble: 6 × 3
#>    year department rd_budget_mil
#>   <dbl> <chr>              <dbl>
#> 1  1976 DHS                    0
#> 2  1976 DOC                  819
#> 3  1976 DOD                35696
#> 4  1976 DOE                10882
#> 5  1976 DOT                 1142
#> 6  1976 EPA                  968
```

Can also set `cols` by selecting which columns *not* to use

```r
names(fed_spend_wide)
```

```
#>  [1] "year"     "DHS"      "DOC"
"DOD"      "DOE"      "DOT"      "EPA"
"HHS"      "Interior" "NASA"     "NIH"
"NSF"      "Other"    "USDA"     "VA"
```

```r
fed_spend_long <- fed_spend_wide %>%
    pivot_longer(
        cols = -year,
        names_to = "department",
        values_to = "rd_budget_mil")

head(fed_spend_long)
```

```
#> # A tibble: 6 × 3
#>    year department rd_budget_mil
#>   <dbl> <chr>              <dbl>
#> 1  1976 DHS                    0
#> 2  1976 DOC                  819
#> 3  1976 DOD                35696
#> 4  1976 DOE                10882
#> 5  1976 DOT                 1142
#> 6  1976 EPA                  968
```

Your turn: Long <--> Wide

20:00

Open the `practice.Rmd` file.

Under "In Class Question 1", write code to read in the following two files:

- `pv_cells.csv`: Data on solar photovoltaic cell production by country
- `milk_production.csv`: Data on milk production by state

Now modify the format of each:

- If the data are in "wide" format, convert it to "long" with `pivot_longer()`
- If the data are in "long" format, convert it to "wide" with `pivot_wider()`

Break

05:00

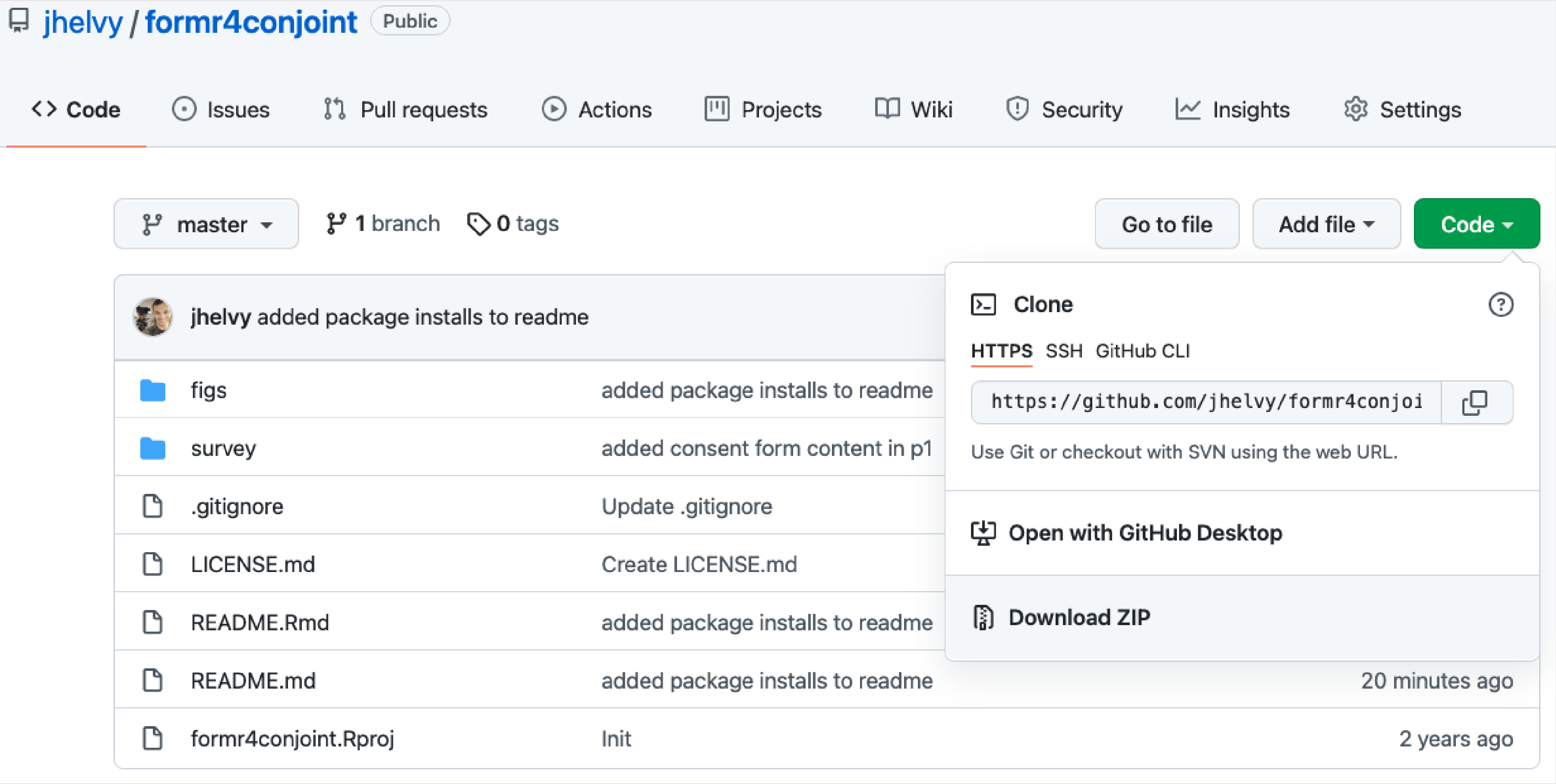
Week 9: *Uncertainty*

1. Computing uncertainty
2. Reshaping data

BREAK

3. Cleaning pilot data
4. Estimating pilot data models

Download the formr4conjoint repo from GitHub



Cleaning formr survey data

1. Open `formr4conjoint.Rproj`
2. Open `code/data_cleaning.R`

Your Turn

20:00

As a team, pick up where you left off last week and create a `choiceData` data frame in a "long" format

Week 9: *Uncertainty*

1. Computing uncertainty

2. Reshaping data

BREAK

3. Cleaning pilot data

4. Estimating pilot data models

Estimating pilot data models

1. Open `formr4conjoint.Rproj`
2. Open `code/modeling.R`

Your Turn

As a team:

1. Use your `choiceData` data frame to estimate preliminary choice models.
2. Interpret your model coefficients with uncertainty.