Lecture 17

Statistical inference

Julian Reif
Fall 2025

RStudio setup for this lecture

- Log into RStudio on your Amazon EC2 instance
  - Use AMI `FIN550-RStudio` with IAM role `BigDataEC2Role`

```
# This is a Unix command. Enter via RStudio Terminal
aws s3 cp --recursive s3://bigdata-fin550-reif/lecture-17 ~/fin550/lecture-17
```

Estimating the unknown

- Suppose you measure the average height of a **random sample** of 10 US women
  - $\bar{X} =$ 64 inches (163 centimeters)
  - Because the sample is random, this estimate is unbiased

- How confident should you be that average height of **all** US women is 64 inches?

- Does your answer change if you measure the height of 100 women? Or 1,000 women?

Statistical inference

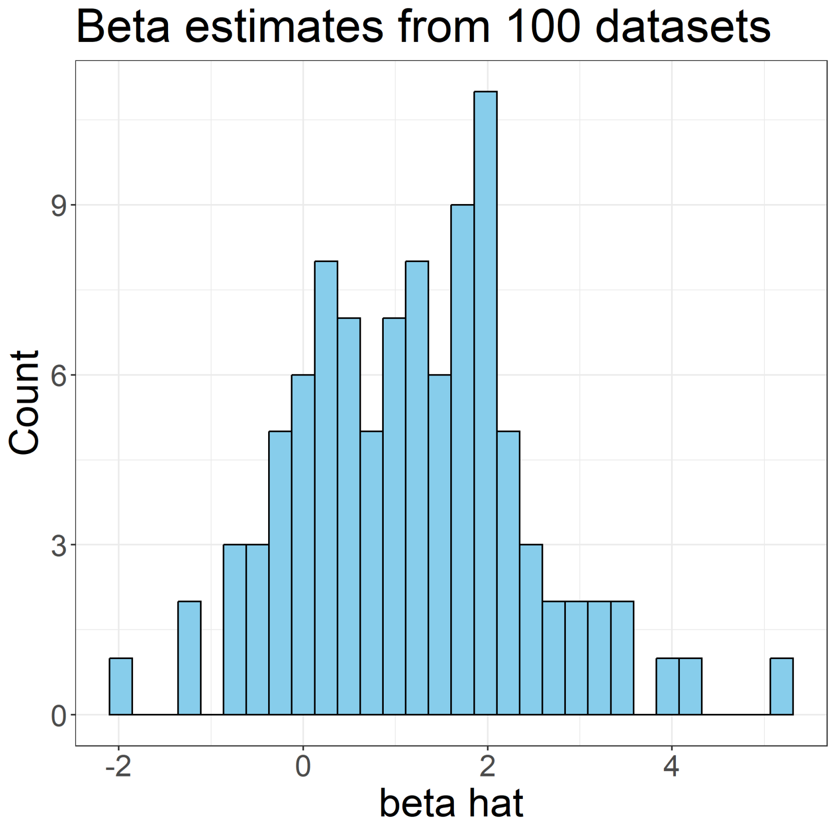
- Statistical inference is the process of describing the uncertainty in our estimate

- Uncertainty generally comes from two sources:
  1. Model uncertainty
  2. Sampling uncertainty

- We will focus on sampling uncertainty and ignore model uncertainty
  - That is not great, but it is standard practice

Sampling uncertainty (error)

- Sampling error arises because data have a random component (noise)
- Large sampling error causes imprecision (different samples produce different estimates)



Estimands, estimators, and estimates

- **Estimand**: the parameter you are trying to estimate

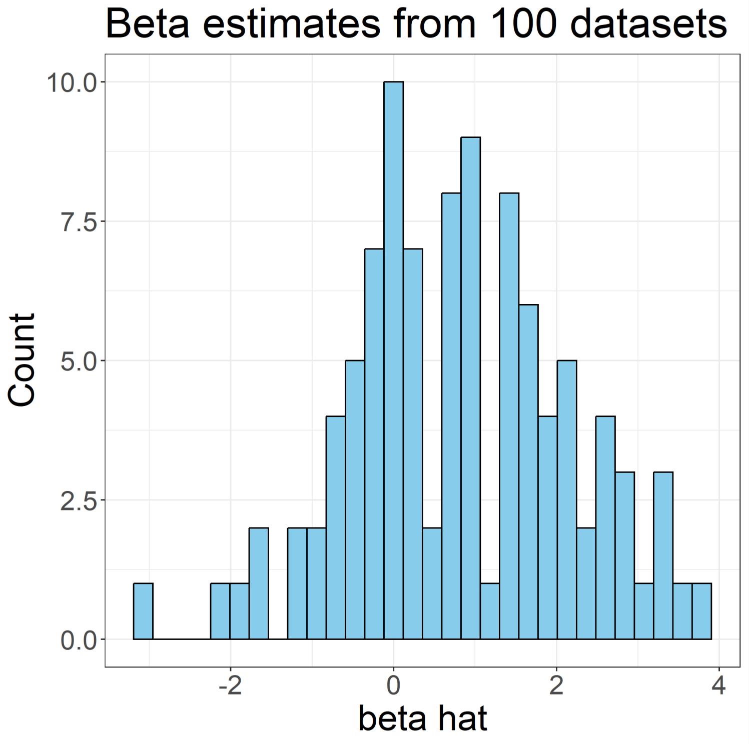
- **Estimator**: a procedure that uses observed data to estimate the estimand

- **Estimate**: the value calculated by the estimator
  - Sampling error causes the estimate to differ from the estimand

Important estimator properties

- **Consistency**: does estimate converge to the estimand as the sample size grows large?

- **Precision**: for a given sample size, how close is the estimate to the quantity of interest?
  - Precision is often quantified by characterizing the distribution of the estimate



Law of Large Numbers (LLN)

- Let μ be the mean of a variable X in a population
- Draw a **random sample** of N observations $x_1, x_2, \ldots, x_N$
- LLN states that the sample average is a **consistent estimator** of the population mean:

$$\lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} x_i = \mu$$

- In other words, sample mean is good approximation of population mean if N is large

Central Limit Theorem (CLT)

- If N is "large", then sampling distribution of the estimate follows the normal distribution
  - See previous figure for an example
  - Note: this is a stronger result than LLN

- The distribution is centered on the mean, and the variance decreases with N

- What is "large"? Depends on the data and the estimator, but rule of thumb is $N > 30$

LLN and CLT are fundamental to statistical inference

- LLN and CLT let us use observed data (x_i) to draw conclusions about larger population

- They form the basis for standard errors, confidence intervals, and p-values

- They can be applied to estimators such as linear regression

Estimator example: sample mean

- Suppose we want to estimate the average height of women in the United States
  - True height is μ, with variance σ^2

- Estimate this mean by calculating average height of N random women
  - Our **estimand** is μ and our **estimate** is $\bar{x}$

- Properties of this **estimator**:
  - Consistent: $\bar{x}$ approaches μ as N becomes large
  - Distribution: $\bar{x}$ is distributed normally with mean μ and variance σ^2/N

Estimator example: ordinary least squares (OLS)

- Suppose we want to estimate the slope parameter of the following linear equation:

$$Y = \alpha + \beta X + \epsilon$$

  where $VAR[\epsilon] = \sigma^2$

- The **estimand** is the parameter β, and the **estimate** is $\hat{\beta}$

- The **estimator** is OLS (linear regression)

- Properties of this estimator:
  - Consistent: $\hat{\beta}$ approaches β as N becomes large
  - Distribution: $\hat{\beta}$ is distributed normally with mean β and variance σ^2/N

Three common ways to quantify precision

1. **Standard errors** describe the standard deviation $(\sqrt{VAR})$ of an estimate

2. **Confidence intervals** describe a range of values that likely contain the true value
    - Over many samples, 95% confidence interval contains the true value 95% of the time
    - 95% confidence interval is about ± 2 standard errors

3. **p-values** describe the probability that the estimate would arise, if true value was 0

Hypothesis testing

- Hypothesis testing consists of two steps:
  1. Make a **null hypothesis** about a parameter (e.g., $\beta = 0$)
  2. Use data and an estimator to test the null hypothesis

- Sampling error means we can never reject the null hypothesis with 100% certainty

- Instead, ask: does hypothesized value lie inside a given (95%) confidence interval?
  - If yes, the result is **statistically insignificant** at the 5% level
  - If no, the result is **statistically significant** at the 5% level

These precision measures are all related

- Suppose the null hypothesis is $\beta = 0$

- We estimate $\hat{\beta} = 2$, with a 95% confidence interval $[-1, 5]$

- The following statements are equivalent:
  1. The estimate, $\hat{\beta}$, is not statistically significant at the 95% confidence level
  2. The 95% confidence interval for $\hat{\beta}$ includes 0
  3. The p-value for $\hat{\beta}$ exceeds $0.05 = 1 - 0.95$

- Conversely, a statistically significant estimate has $p < 0.05$ and a 95% confidence interval that excludes 0

Statistical inference in R

Estimating a linear regression model with OLS

Consider the following data generating process:

$$Y = \alpha + \beta x + \epsilon$$

where

- $x \in [0, 1]$
- α and β are fixed at some value
- ϵ is a mean-zero random error

Function to create a simulated dataset

```r
library(tidyverse)
library(ggplot2)
library(broom)

# Function to create dataset with N observations
data_sample <- function(N = 100, alpha = 0, beta = 1, sd_e = 4) {
  data <- tibble(
    x = runif(N, 0, 1),
    y = alpha + beta*x + rnorm(N, mean=0, sd=sd_e) )
  return(data)
}
```

Create simulated dataset with N=100 observations

```
set.seed(1) # Set a seed because the error term is random
df100 <- data_sample(N=100)

nrow(df100)
head(df100)
```

```
# [1] 100
# # A tibble: 6 × 2
#       x     y
#   <dbl> <dbl>
# 1 0.266  1.86
# 2 0.372 -2.08
# 3 0.573  1.94
# 4 0.908 -3.61
# 5 0.202  5.93
# 6 0.898  8.82
```

Try it: estimate beta using OLS

```r
# Estimate y = alpha + beta*x + e
lm1 <-

# What is the standard error of beta? p-val and 95% confidence interval?
# Is the estimate statistically significant at 95% confidence level?
tidy(lm1, conf.int = T, conf.level = 0.95)
```

Estimate beta using OLS

```r
# Estimate y = alpha + beta*x + e
lm1 <- lm(y ~ x, data = df100)

# What is the standard error of beta? p-val and 95% confidence interval?
# Is the estimate statistically significant at 95% confidence level?
tidy(lm1, conf.int = T, conf.level = 0.95)
```

```
# # A tibble: 2 × 7
#   term        estimate std.error statistic p.value conf.low conf.high
#   <chr>          <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
# 1 (Intercept)   -0.717     0.823    -0.871   0.386   -2.35      0.916
# 2 x              2.25      1.41      1.59    0.115   -0.556     5.06
```

Approximate the 95% confidence interval

```r
beta <- lm1$coefficients["x"]
stderr <- sqrt(diag(vcov(lm1)))["x"]

# 95% confidence interval is approximately beta +/- 2 stderrs
beta - 2*stderr
beta + 2*stderr
```

```
#          x
# -0.5782947
#        x
# 5.077039
```

What happens to estimates when sample size increases?

```r
set.seed(1)
df1000 <- data_sample(N=1000)

# Did the estimate of beta change?
# What happened to std error, p-val, and confidence interval?
lm2 <- lm(y ~ x, data = df1000)
tidy(lm2, conf.int = T, conf.level = 0.95)
```

```
# # A tibble: 2 × 7
#   term        estimate std.error statistic p.value conf.low conf.high
#   <chr>          <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
# 1 (Intercept)   -0.140     0.262    -0.536  0.592    -0.654     0.373
# 2 x              1.08      0.454     2.39   0.0170    0.195     1.98
```

Write a function to automate the estimation of beta

```r
# Create a random dataset, and return the estimate of beta
estimate_beta <- function(N = 100) {
  df <- data_sample(N)
  lm <- lm(y ~ x, data = df)
  return(lm$coefficient["x"])
}

set.seed(1)
estimate_beta()
```

```
#        x
# 2.249372
```

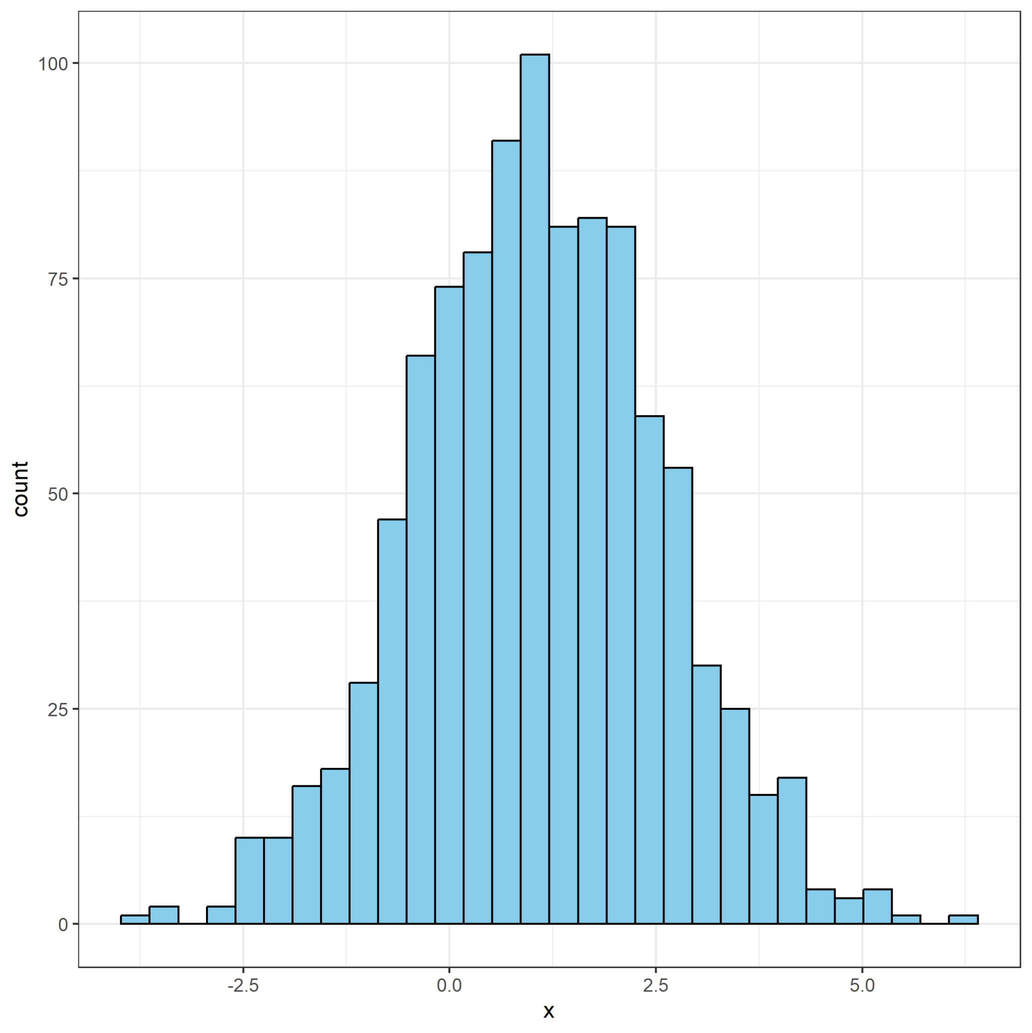
Create 1000 datasets and estimate the betas

```r
N <- 100
ndatasets <- 1000

# Recall: lapply() is like a for loop
set.seed(1)
betas <- lapply(1:ndatasets, function(i) estimate_beta(N)) %>%
  bind_rows()

nrow(betas)
```

```
# [1] 1000
```

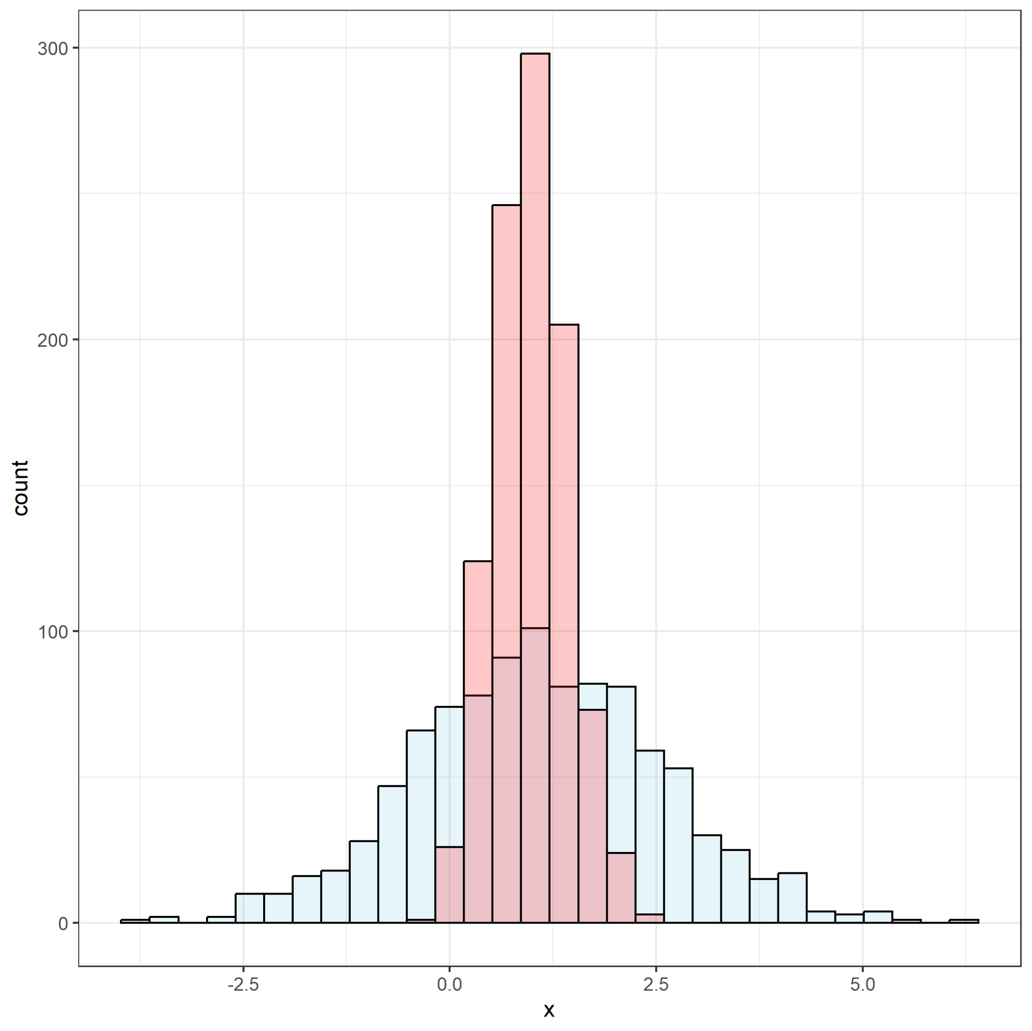
Create 1000 datasets and estimate the betas

```r
ggplot(betas, aes(x=x)) +
  theme_bw() +
  geom_histogram(color="black",
                 fill="skyblue")
```



What happens if we increase the sample size?

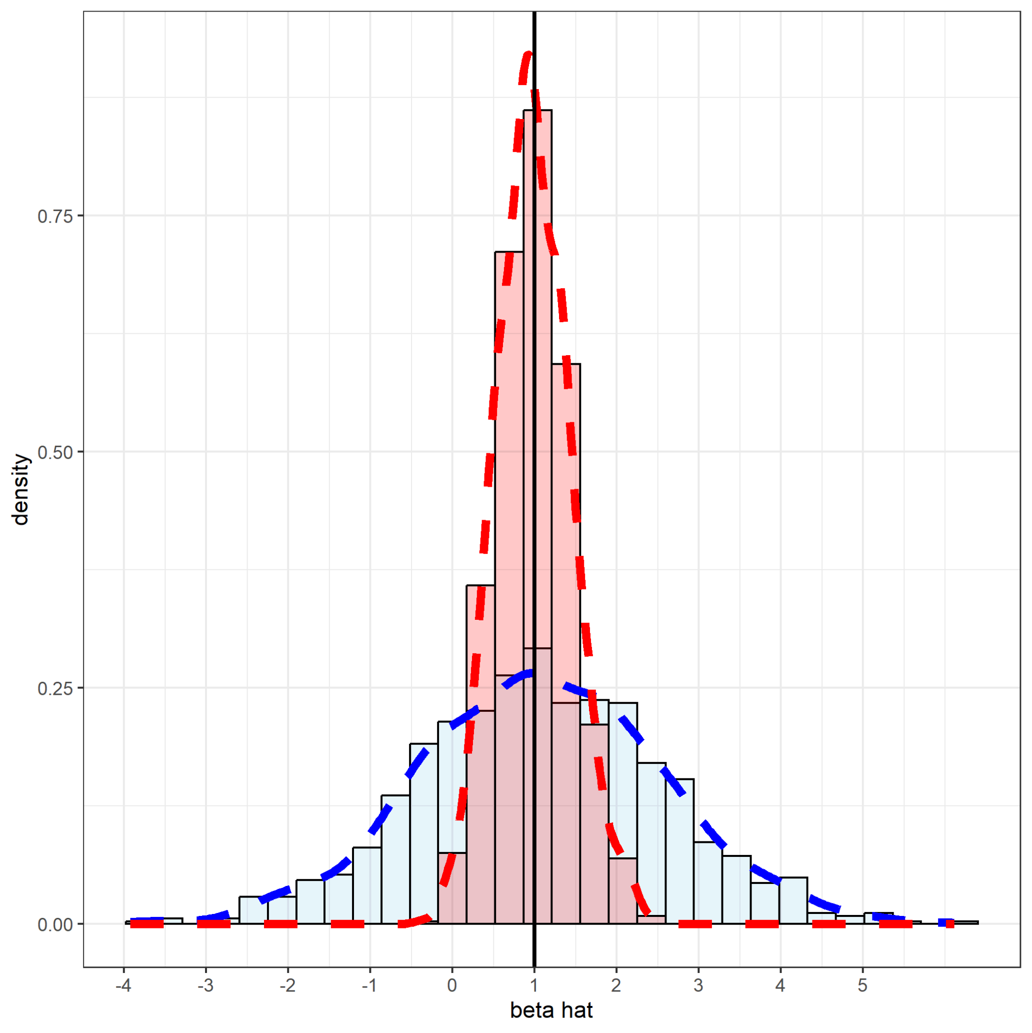
```r
set.seed(1)
N <- 1000
betas$x2 <-lapply(1:ndatasets,
                function(i) estimate_beta(N)) %>%
       bind_rows()

ggplot(betas) +
  theme_bw() +
  geom_histogram(aes(x=x), alpha=0.2,
                 color="black",
                 fill="skyblue") +
  geom_histogram(aes(x=x2$x), alpha=0.2,
                 color="black",
                 fill="red")
```



These results reflect the Central Limit Theorem

- Consistency

- Normal distribution
  - Variance decreases with N



Summary

- Statistical inference lets us describe the precision of our estimate

- When CLT applies, estimates are normally distributed with a variance decreasing in N

- Precision is quantified using standard errors, confidence intervals, and p-values

- Problem Set 1 has been assigned: due **Friday, November 21** at 11:59pm
- Final Project will be posted today: due **Thursday, December 4** at 11:59pm
  - You should start working on this now!

- For both assignments, you may work in groups of up to 3 people