Lecture 01b – Revision

ENVX2001 Applied Statistical Methods

Januar Harianto

Feb 2026

Refresher

Assumed knowledge

1. Samples, populations and statistical inference
2. Probability distributions
3. Parameter estimation
   - central tendency
   - spread or variability
4. Sampling distribution of the mean
   - Standard error
   - Confidence intervals
   - Central Limit Theorem

Why is this important?

- Understanding sampling **informs experimental design (Week 4 onward)**. *How many samples do we need and are our samples representative?*
- Recognising sample (*not sampling*) distributions helps us choose the right statistical model – e.g. *t*-test to compare two means that are normally distributed.
- Most statistical techniques use sample statistics for interpretation, e.g. the *t*-test can be explained using **confidence intervals**, and the ANOVA test can be interpreted in part using **means** and **standard errors**.

All of these concepts will make more sense as we go through the course, but if you do not try to understand them now, you **will** struggle.

Samples, populations and statistical inference

Populations and samples

Populations

- **All** the possible units and their associated observations of interest
- Scientists are often interested in making *inferences* about populations, but measuring every unit is impractical

Samples

- A collection of observations from any population is a sample, and the number of observations in it is the **sample size**
- We assume samples that we collect can be used to make inferences about the population
- **NEW**: Samples need to be *representative* of the population

Statistics vs parameters

- Characteristics of the **population** are called *parameters* (e.g. population mean or population regression slope)
- Characteristics of the **sample** are called *statistics* (e.g. sample mean or sample regression slope) – they are used to estimate the population parameters
- Statistics are what we use to help us understand the population
- Formal statistical methods can help us make inferences about the population based on the sample – statistical inference
- *Not all statistical techniques are inferential, but many are*

Sample data

Sample data are usually collected as **variables**, which are the characteristics we measure or record from each object.

Variables can be:

Categorical Variables

- **Nominal**: categories without a natural order (e.g. colors, names)
- **Ordinal**: categories with a natural order (e.g. ratings, rankings)

Numerical Variables

- **Continuous**: can take any value within a range (e.g. height, weight)
- **Discrete**: can take only specific values (e.g. counts, presence/absence)

YOU decide on what a variable represents

A numerical, continuous variable can be treated as a categorical variable if you decide to categorise it.

Examples

- **height (in cm)** – a numerical, **continuous** variable, can be treated as a categorical variable if you group it into categories (short, medium, tall)
- **age (in years)** – a numerical, **discrete** variable, can be treated as a continuous variable (if we allow for certain *issues*)
- **treatment (A, B, C)** – a categorical variable, can be treated as a numerical variable if we assign numbers to the treatments (1, 2, 3) and assume they are **ordered** e.g. effect of $1 < 2 < 3$ – *the basis of non-parametric tests*

Distribution of data

Types of probability distributions

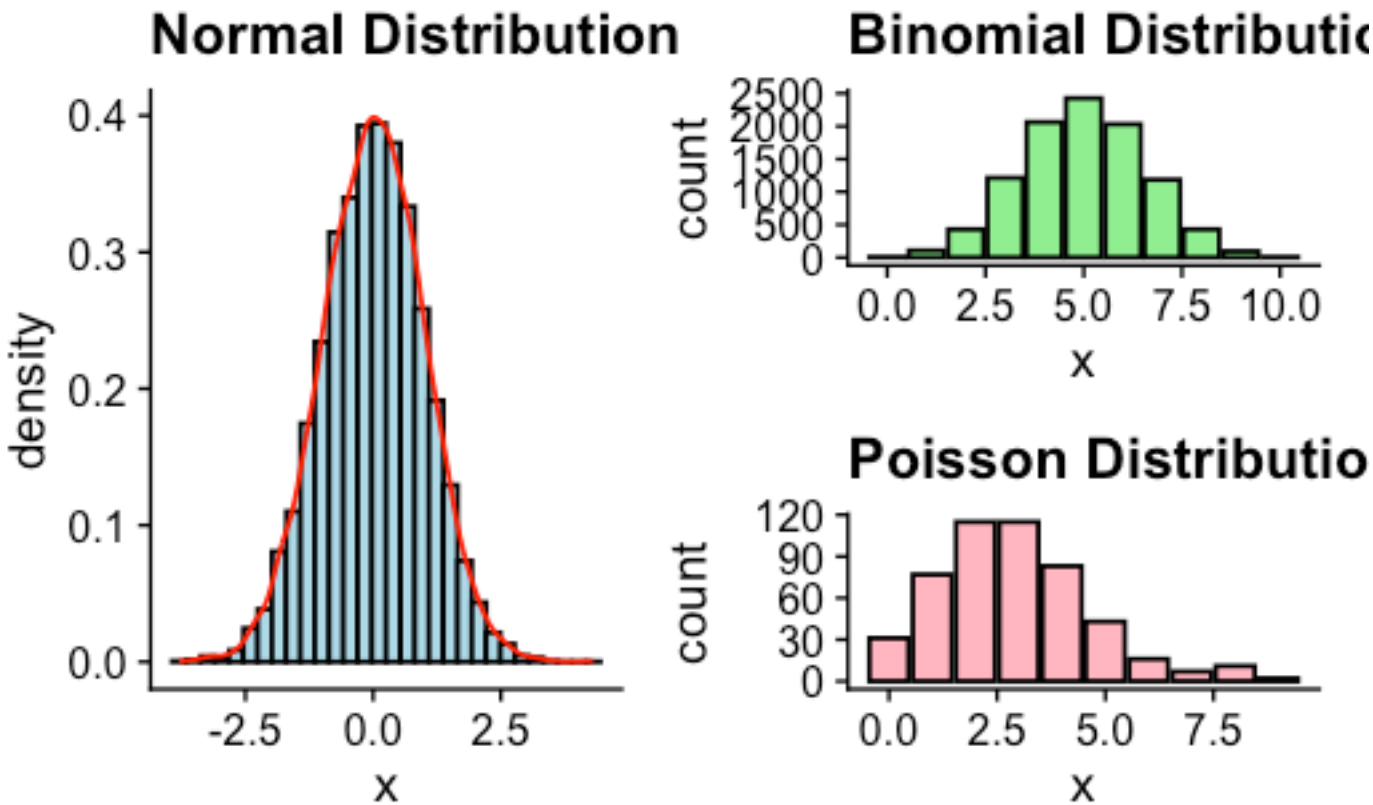
Populations can be described by probability distributions, and by now, you should be familiar with these distributions and their properties

- **Normal Distribution**: Bell-shaped curve, symmetric around the mean. **Data is continuous**
- **Binomial Distribution**: Models success/failure outcomes in a fixed number of trials. **Data is discrete**
- **Poisson Distribution**: Models count data when events occur at a constant rate. **Data is discrete**

Knowing the distribution of your data is important for choosing the right statistical model – although it is not always necessary.

```r
library(tidyverse)

set.seed(908)
normal_data ← data.frame(x = rnorm(10000, mean = 0, sd = 1))
binomial_data ← data.frame(x = rbinom(10000, size = 10, prob = 0.5))
poisson_data ← data.frame(x = rpois(500, lambda = 3))

# normal
p1 ← ggplot(normal_data, aes(x = x)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "lightblue", color = "black") +
  geom_density(color = "red") +
  ggtitle("Normal Distribution")

# binomial
p2 ← ggplot(binomial_data, aes(x = x)) +
  geom_bar(fill = "lightgreen", color = "black") +
  ggtitle("Binomial Distribution")

# poisson
p3 ← ggplot(poisson_data, aes(x = x)) +
```

```r
  geom_bar(fill = "lightpink", color = "black") +
  ggtitle("Poisson Distribution")

# Arrange plots
library(patchwork)
p1 | p2 / p3
```


Normal Distribution
density
0.4
0.3
0.2
0.1
0.0
-2.5
0.0
2.5
x
Binomial Distributio
count
2500
2000
1500
1000
500
0
0.0
2.5
5.0
7.5
10.0
x
Poisson Distributio
count
120
90
60
30
0
0.0
2.5
5.0
7.5
x

Parameter estimation

Measures of central tendency

Mean $\bar{x}$

- The arithmetic average of all values in a dataset
- Sum of all values divided by number of observations
- Sensitive to extreme values (outliers)

⚠ Formula

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

- where n is the number of observations
- x_i represents each individual value
- $\sum$ means we add up all values from $i = 1$ to n
- Example: for data {2,4,6,8}, $n = 4$ and $\bar{x} = \frac{2+4+6+8}{4} = 5$

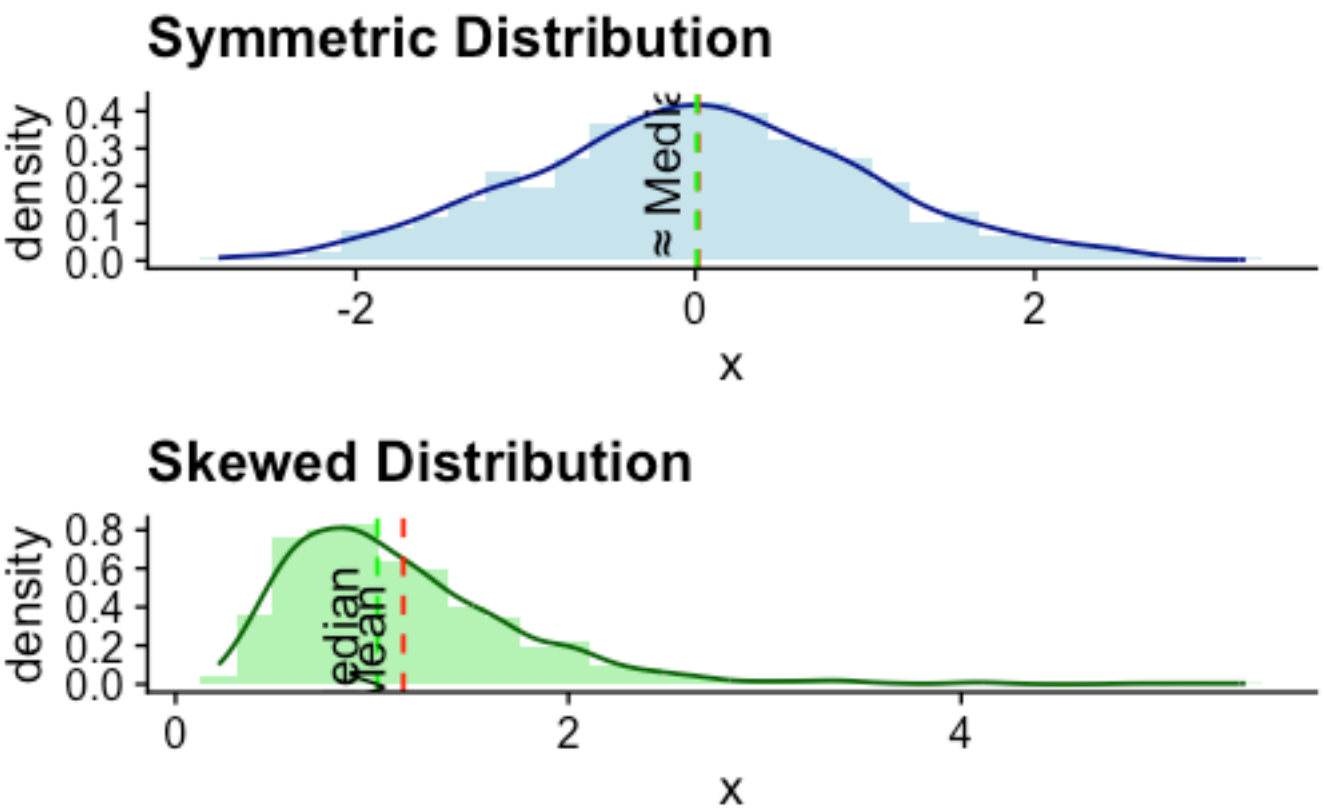
Median

- Middle value when data is ordered
- 50th percentile of the data
- More robust to outliers than mean
- For even n, average of two middle values

Mode

- Most frequently occurring value
- Can have multiple modes
- Only measure of central tendency for categorical data
- Not always meaningful for continuous data

How they compare



Depending on the distribution of the data, the mean and median can be different, and this can tell you something about the data.

Measures of dispersion

Variance s^2

- Measures how spread out the data is from the mean
- Calculated as average squared deviations from the mean
- Squared units make it harder to interpret
- Sensitive to outliers (squares large deviations)

Measures of dispersion

⚠ Formula

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

- where n is the number of observations
- x_i represents each individual value
- $\bar{x}$ is the sample mean
- For data {2,4,6,8}:
  - mean = 5
  - differences = (-3,-1,1,3), squares = (9,1,1,9), sum = 20
  - Therefore, $s^2 = \frac{20}{3} \approx 6.67$

Measures of dispersion

Standard Deviation s

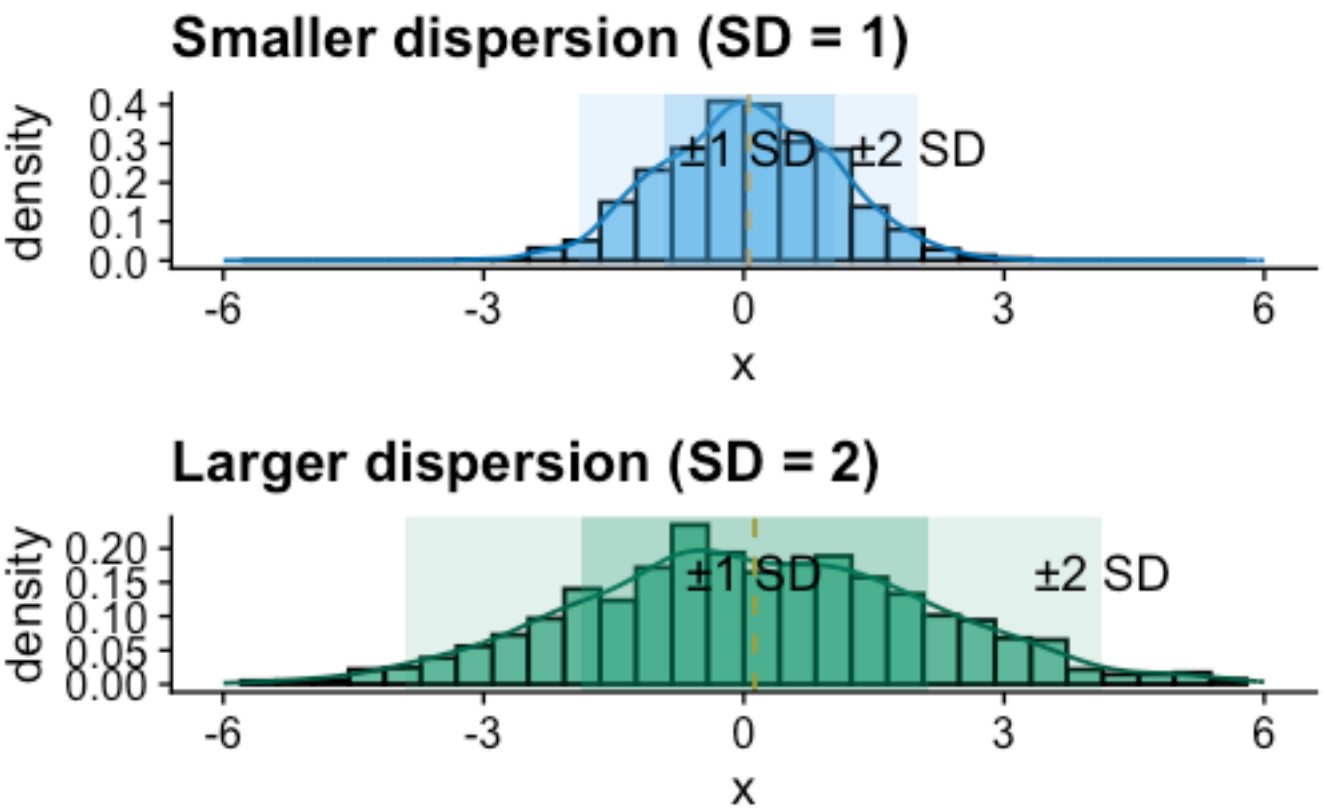
- Square root of variance
- Same units as original data
- More interpretable than variance
- Empirical rule for normal distributions:
  - ≈68% of data within ±1 SD
  - ≈95% of data within ±2 SD
  - ≈99.7% of data within ±3 SD

⚠ Formula

$$s = \sqrt{s^2}$$

- Simply the square root of variance
- For our example data {2,4,6,8}:
  - $s = \sqrt{6.67} \approx 2.58$
- Interpretation: On average, values deviate about 2.58 units from the mean

Visualising standard deviation



Population parameters vs sample statistics

For those of you interested:

Mean

Population Parameter	Sample Statistic
$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$	$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$

Variance

Population Parameter	Sample Statistic
$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N} (x_i - \mu)^2$	$s^2 = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \bar{x})^2$

Standard Deviation

Population Parameter	Sample Statistic
$\sigma = \sqrt{\sigma^2}$	$s = \sqrt{s^2}$

Note

Notice the use of $n - 1$ in sample variance and standard deviation

Why n-1?

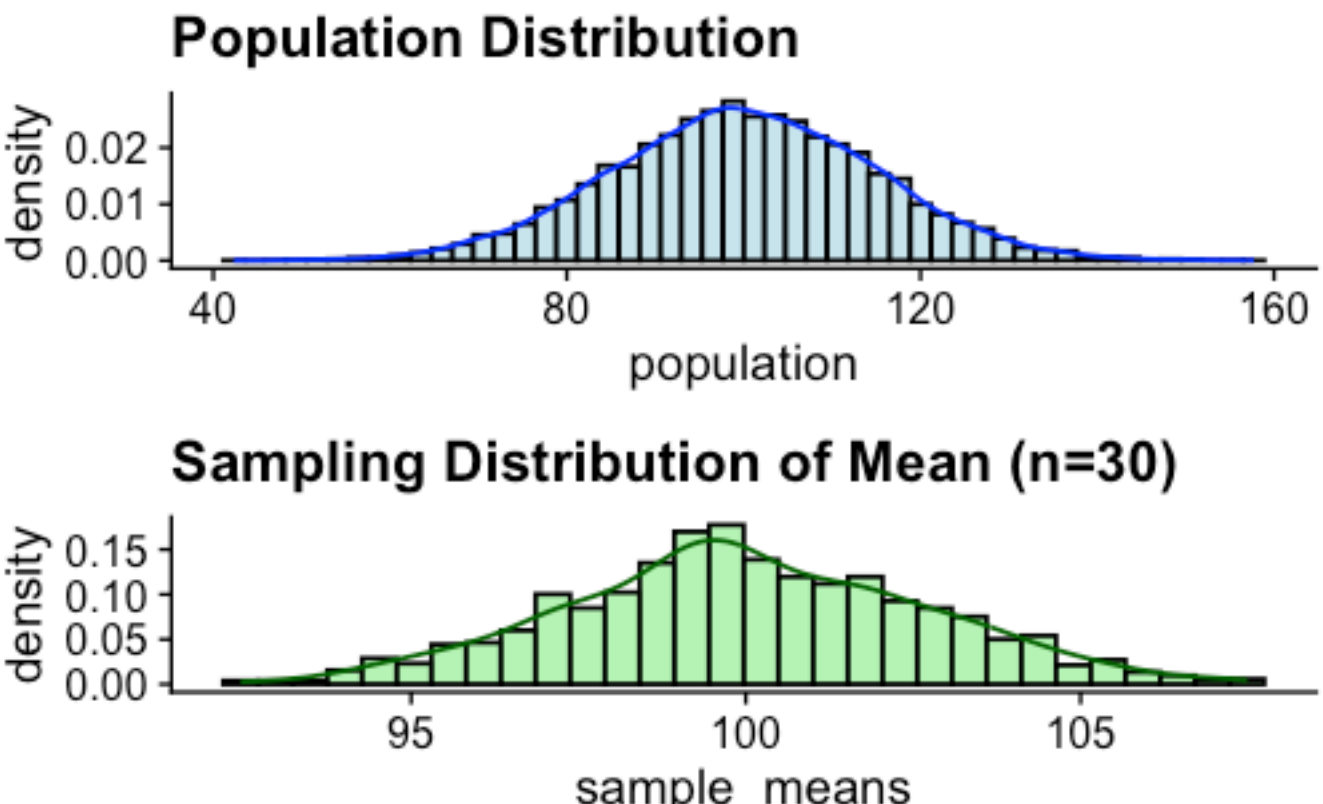
- When calculating sample variance, we use $n - 1$ instead of n in the denominator
- This is called "Bessel's correction"
- Why? Because we lose one "degree of freedom" when we estimate the mean:
  1. If you know the sample mean ($\bar{x}$)
  2. And you know all but one value in your sample
  3. The last value is *constrained* - it must make the mean equal $\bar{x}$

Sampling distributions and CLT

What is a sampling distribution?

- Distribution of a statistic (e.g., mean) calculated from repeated samples
- Shows how sample statistics vary from sample to sample
- Important for understanding sampling variability and making inferences

Sampling distribution of the mean



Central Limit Theorem

> I know of scarcely anything so apt to impress the imagination as the wonderful form of **cosmic order** expressed by the Central Limit Theorem. The law would have been personified by the Greeks and deified, if they had known of it."

– Sir Francis Galton, 1889, Natural Inheritance

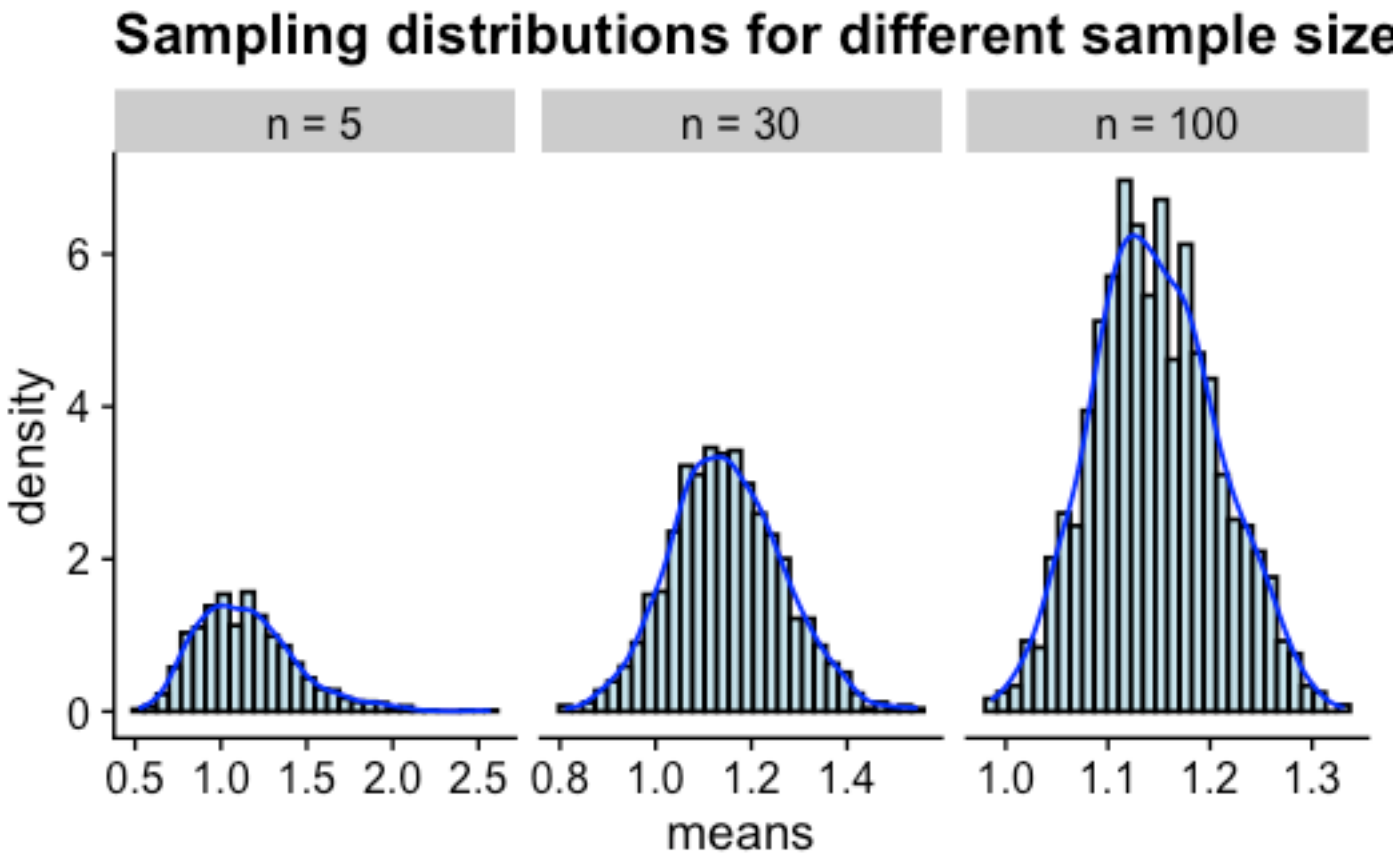
The Central Limit Theorem (CLT) states that for sufficiently large samples:

1. The sampling distribution of the mean follows a normal distribution
2. The mean of the sampling distribution equals the population mean
3. The standard deviation of the sampling distribution (standard error) = $\frac{\sigma}{\sqrt{n}}$

CLT in action

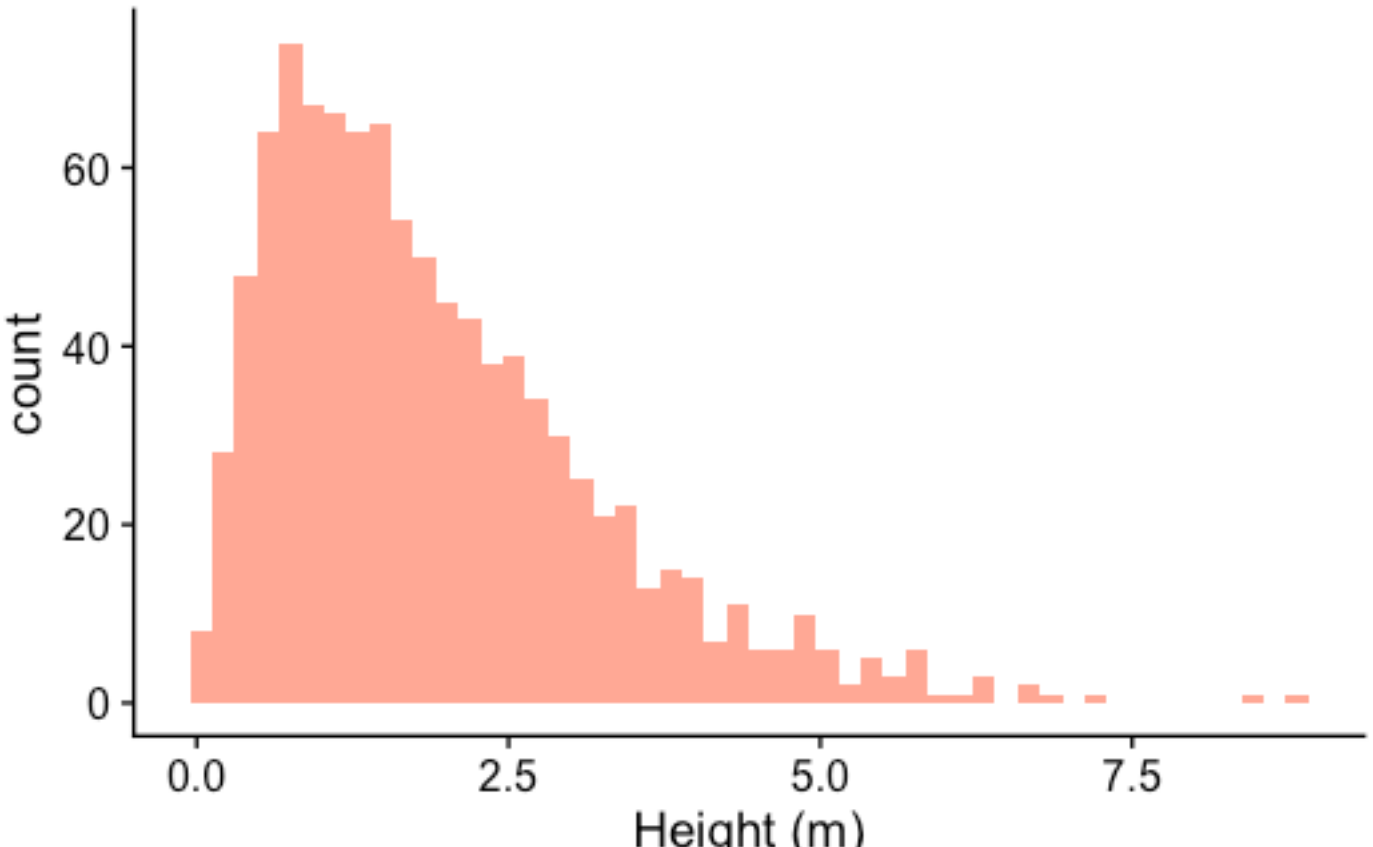
```
# Create a skewed population
set.seed(456)
skewed_pop ← exp(rnorm(10000, mean = 0, sd = 0.5))

# Sample means for different sample sizes (ordered small to large)
sample_sizes ← c(5, 30, 100)
sample_labels ← factor(paste("n =", sample_sizes),
  levels = paste("n =", sample_sizes)
) # preserve order
sample_dist_data ← lapply(sample_sizes, function(n) {
  means ← replicate(1000, mean(sample(skewed_pop, size = n)))
  data.frame(means = means, size = factor(paste("n =", n), levels = levels(sample_labels)))
})
sample_dist_df ← do.call(rbind, sample_dist_data)

# Plot
ggplot() +
  geom_histogram(aes(x = means, y = ..density..),
    data = sample_dist_df,
```

```
  bins = 30, fill = "lightblue", color = "black", alpha = 0.7
) +
geom_density(aes(x = means), data = sample_dist_df, color = "blue") +
facet_wrap(~size, scales = "free_x") +
ggtitle("Sampling distributions for different sample sizes")
```


Sampling distributions for different sample size
n = 5
n = 30
n = 100
density
0
2
4
6
0.5
1.0
1.5
2.0
2.5
0.8
1.0
1.2
1.4
1.0
1.1
1.2
1.3
means

Example

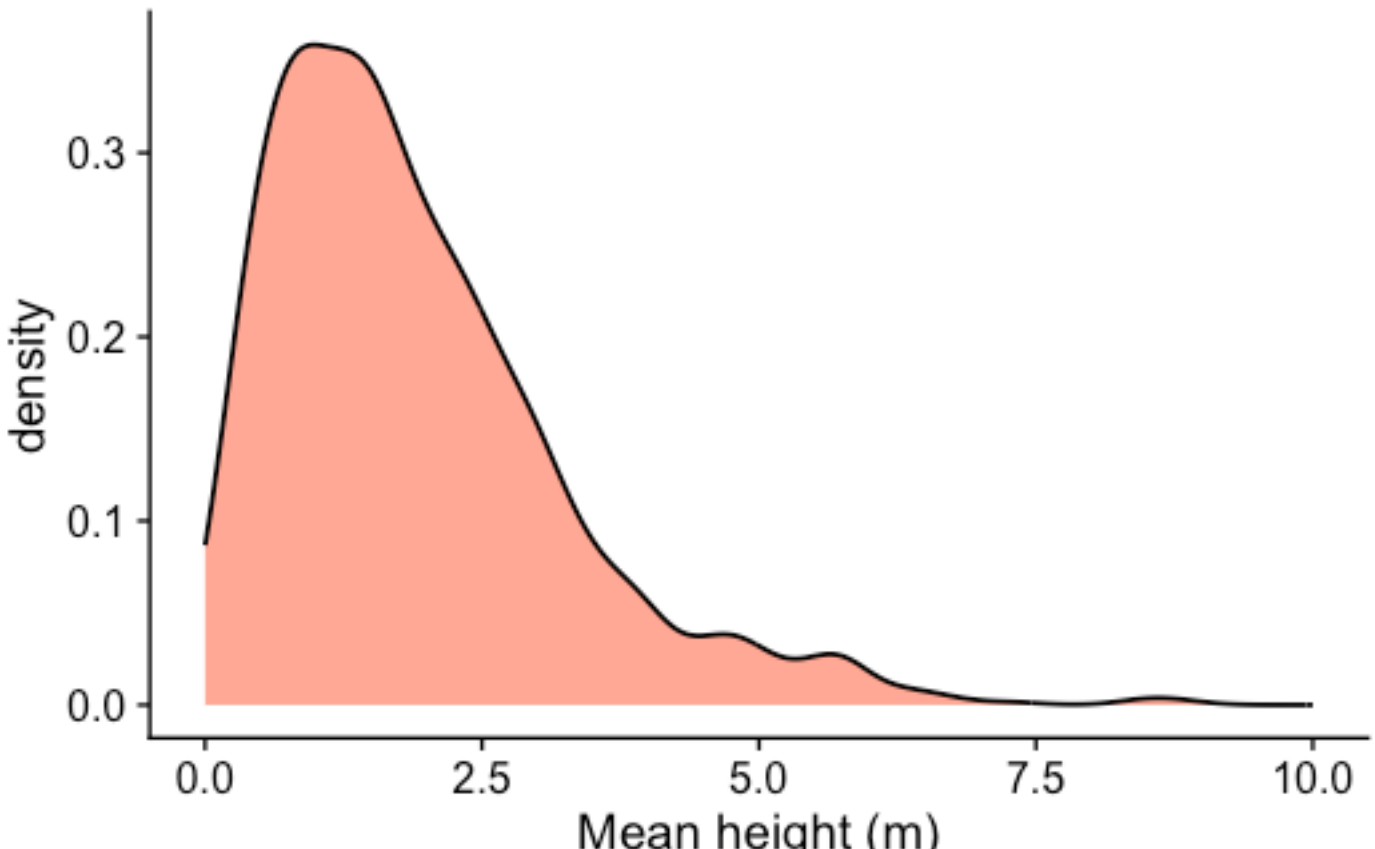
```
set.seed(239)
# Generate a skewed distribution
skewed ← tibble(
  x = rgamma(1000, shape = 2, scale = 1)
)

# plot in ggplot2
ggplot(data = skewed, aes(x = x)) +
  geom_histogram(
    fill = "orangered",
    alpha = 0.5, bins = 50
  ) +
  xlab("Height (m)")
```



- Skewed population distribution for tree heights.
- We want to estimate the mean height of the trees in the forest.

1 sample (no summary statistic)

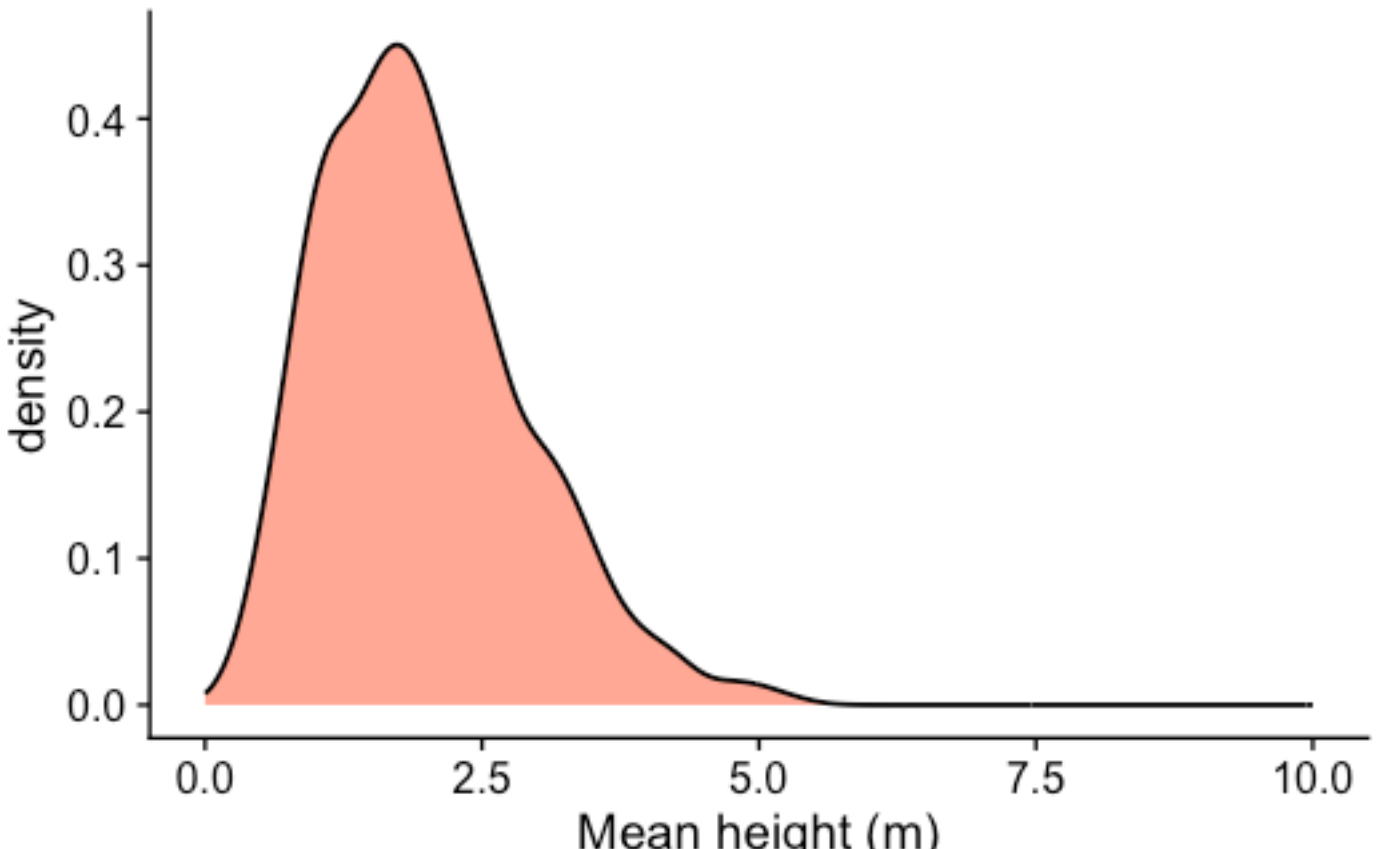
```
skewed |>
  infer::rep_sample_n(
    size = 1,
    reps = 1000
  ) |>
  group_by(replicate) |>
  summarise(xbar = mean(x)) |>
  ggplot(aes(x = xbar)) +
  geom_density(
    fill = "orangered",
    alpha = 0.5, bins = 50
  ) +
  xlim(0, 10) +
  xlab("Mean height (m)")
```



With only one sample, we are not really seeing a sampling distribution – we are just replicating the same population distribution. A sampling distribution emerges when we take multiple samples and calculate their means.

2 samples

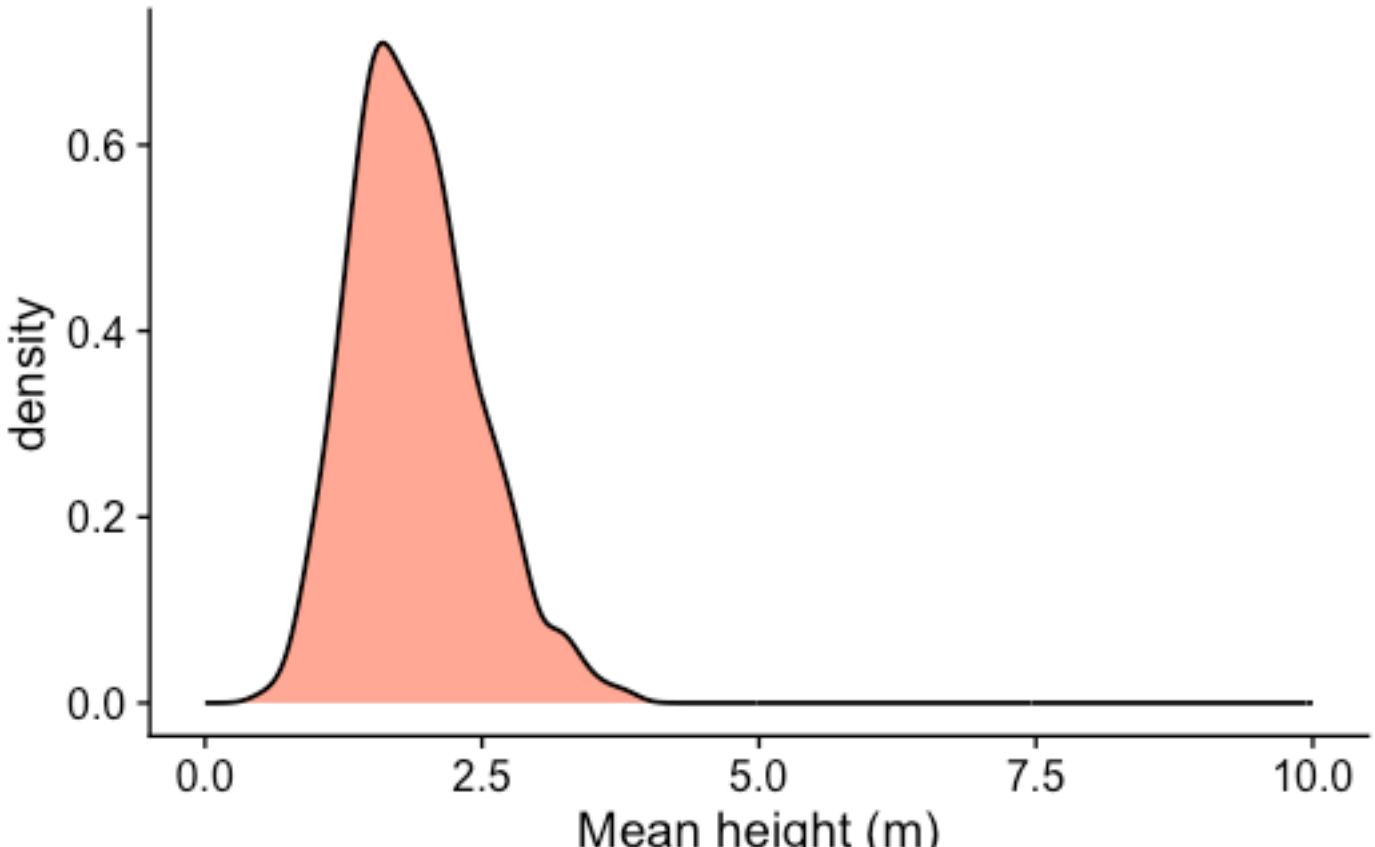
```
skewed |>
  infer::rep_sample_n(
    size = 2,
    reps = 1000
  ) |>
  group_by(replicate) |>
  summarise(xbar = mean(x)) |>
  ggplot(aes(x = xbar)) +
  geom_density(
    fill = "orangered",
    alpha = 0.5, bins = 50
  ) +
  xlim(0, 10) +
  xlab("Mean height (m)")
```



- We sample 2 trees and calculate the mean height, and repeat this 1000 times.
- The distribution of sample means is starting to look more like a normal distribution.

5 samples

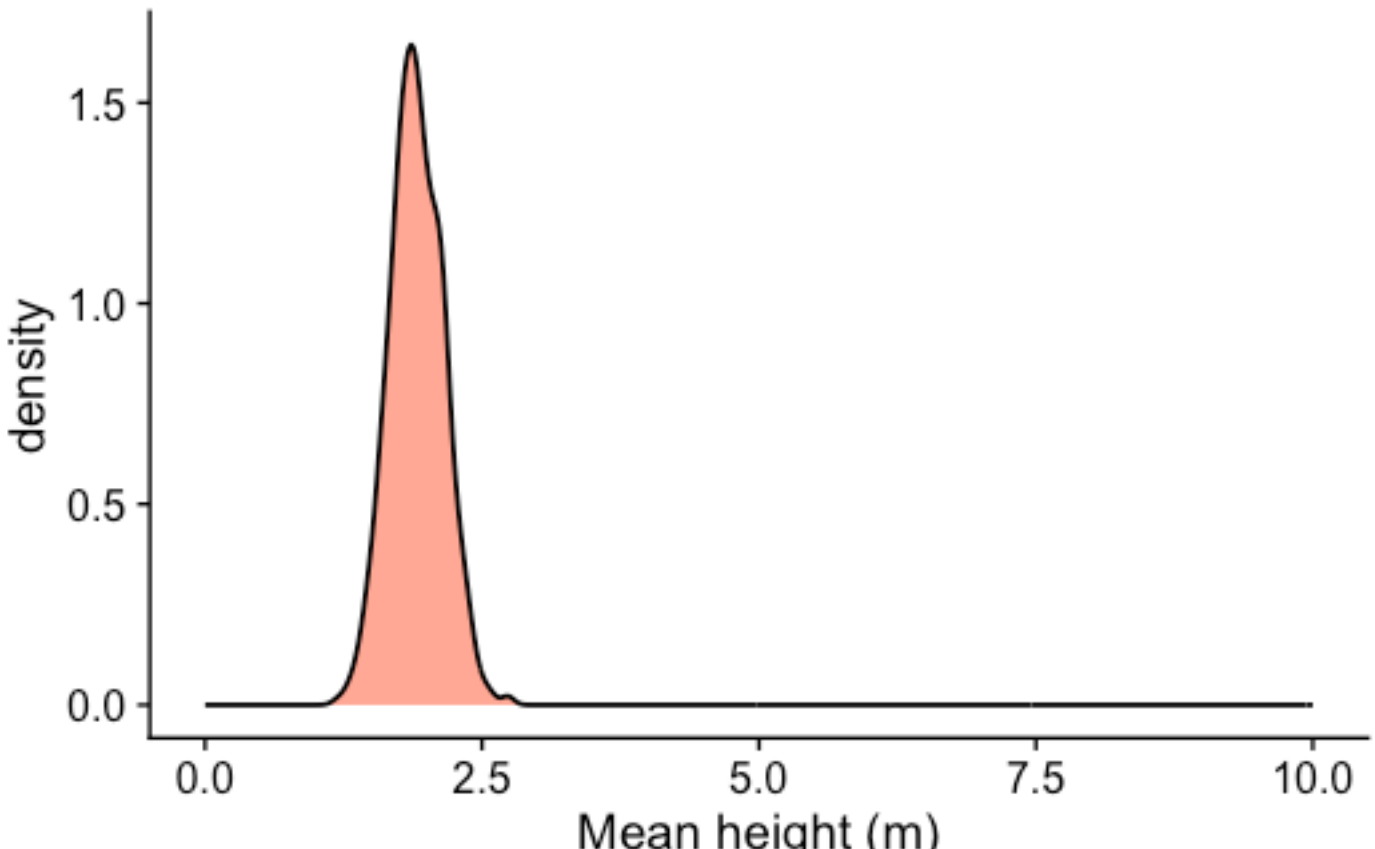
```
skewed |>
  infer::rep_sample_n(
    size = 5,
    reps = 1000
  ) |>
  group_by(replicate) |>
  summarise(xbar = mean(x)) |>
  ggplot(aes(x = xbar)) +
  geom_density(
    fill = "orangered",
    alpha = 0.5, bins = 50
  ) +
  xlim(0, 10) +
  xlab("Mean height (m)")
```



- Five random samples per calculated mean, repeated 1000 times.
- The distribution is becoming more normal, and the spread is decreasing: estimate is getting more **precise**.

30 samples

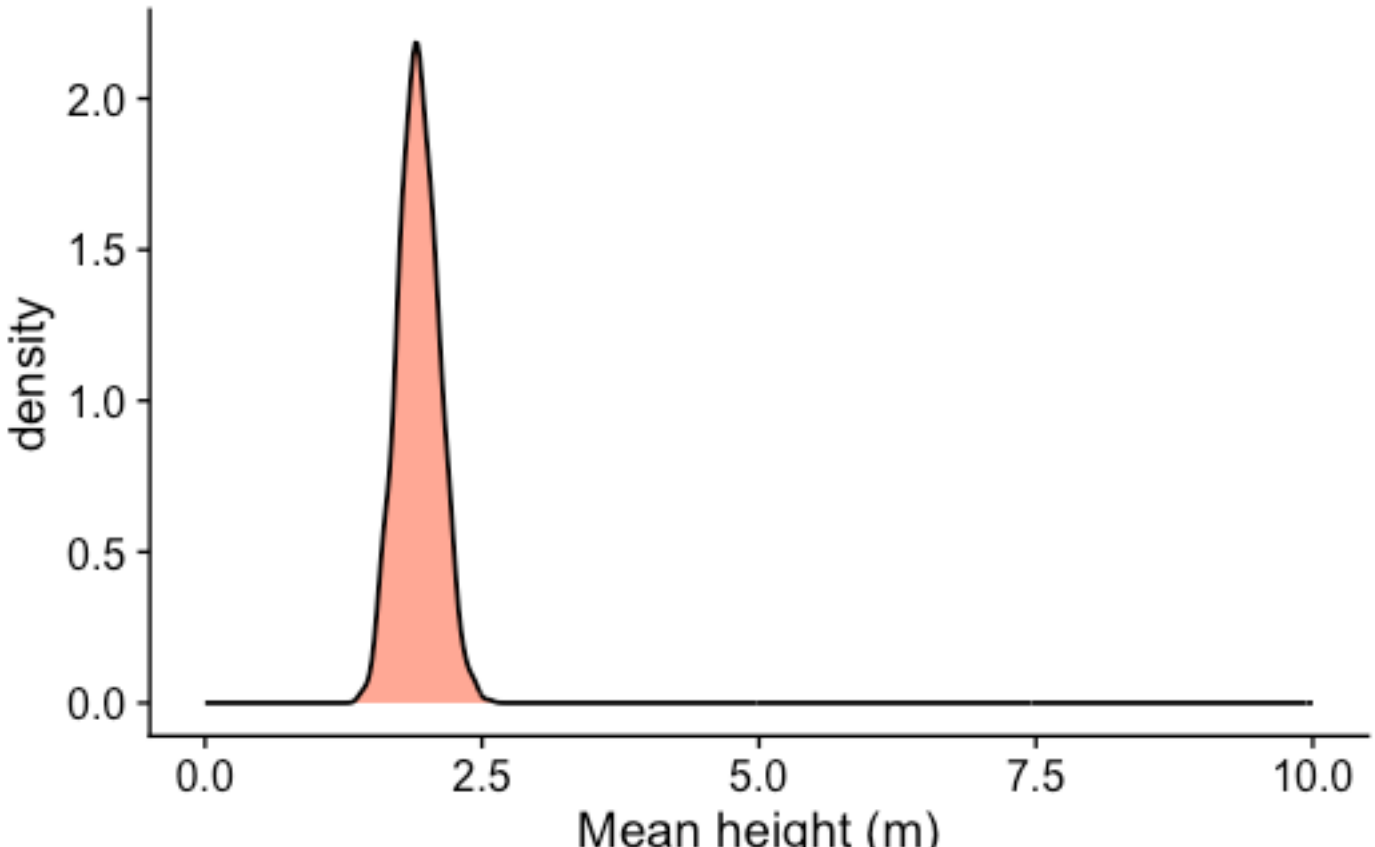
```
skewed |>
  infer::rep_sample_n(
    size = 30,
    reps = 1000
  ) |>
  group_by(replicate) |>
  summarise(xbar = mean(x)) |>
  ggplot(aes(x = xbar)) +
  geom_density(
    fill = "orangered",
    alpha = 0.5, bins = 50
  ) +
  xlim(0, 10) +
  xlab("Mean height (m)")
```



- Thirty random samples per calculated mean, repeated 1000 times.
- The distribution of sample means is very close to a normal distribution.

50 samples

```
skewed |>
  infer::rep_sample_n(
    size = 50,
    reps = 1000
  ) |>
  group_by(replicate) |>
  summarise(xbar = mean(x)) |>
  ggplot(aes(x = xbar)) +
  geom_density(
    fill = "orangered",
    alpha = 0.5, bins = 50
  ) +
  xlim(0, 10) +
  xlab("Mean height (m)")
```

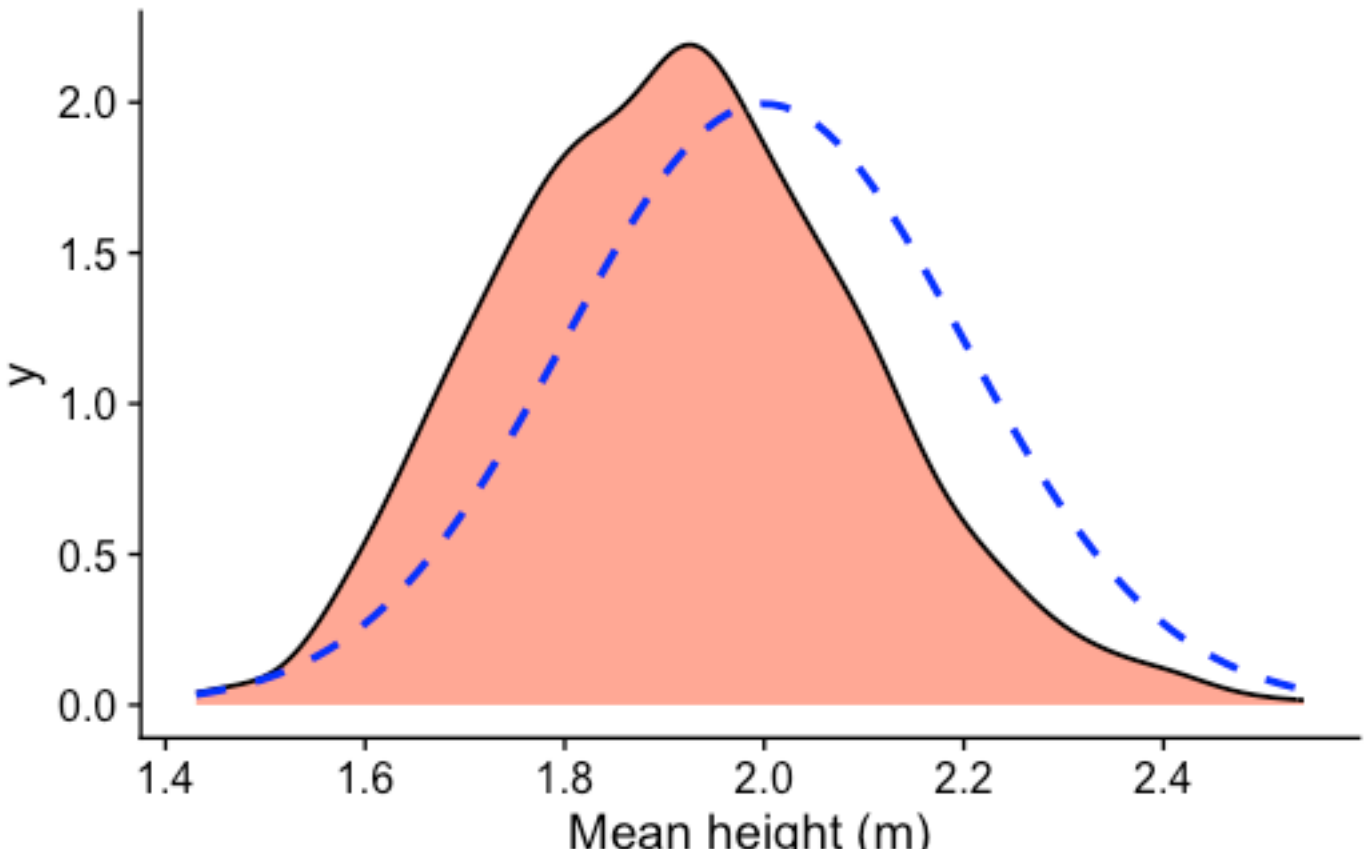


- Fifty random samples per calculated mean, repeated 1000 times.
- **How many samples is enough?**

Are 50 samples “normal” enough?

```
skewed |>
  infer::rep_sample_n(
    size = 50,
    reps = 1000
  ) |>
  group_by(replicate) |>
  summarise(xbar = mean(x)) |>
  ggplot(aes(x = xbar)) +
  geom_density(
    fill = "orangered",
    alpha = 0.5, bins = 50
  ) +
  stat_function(
    fun = dnorm,
    args = list(
      mean = 2, # population mean for gamma(2,1)
      sd = sqrt(2) / sqrt(50) # theoretical SE for gamma(2,1)
    ),
    linewidth = 1,
```

```
    color = "blue",
    linetype = "dashed"
  ) +
  xlab("Mean height (m)")
```

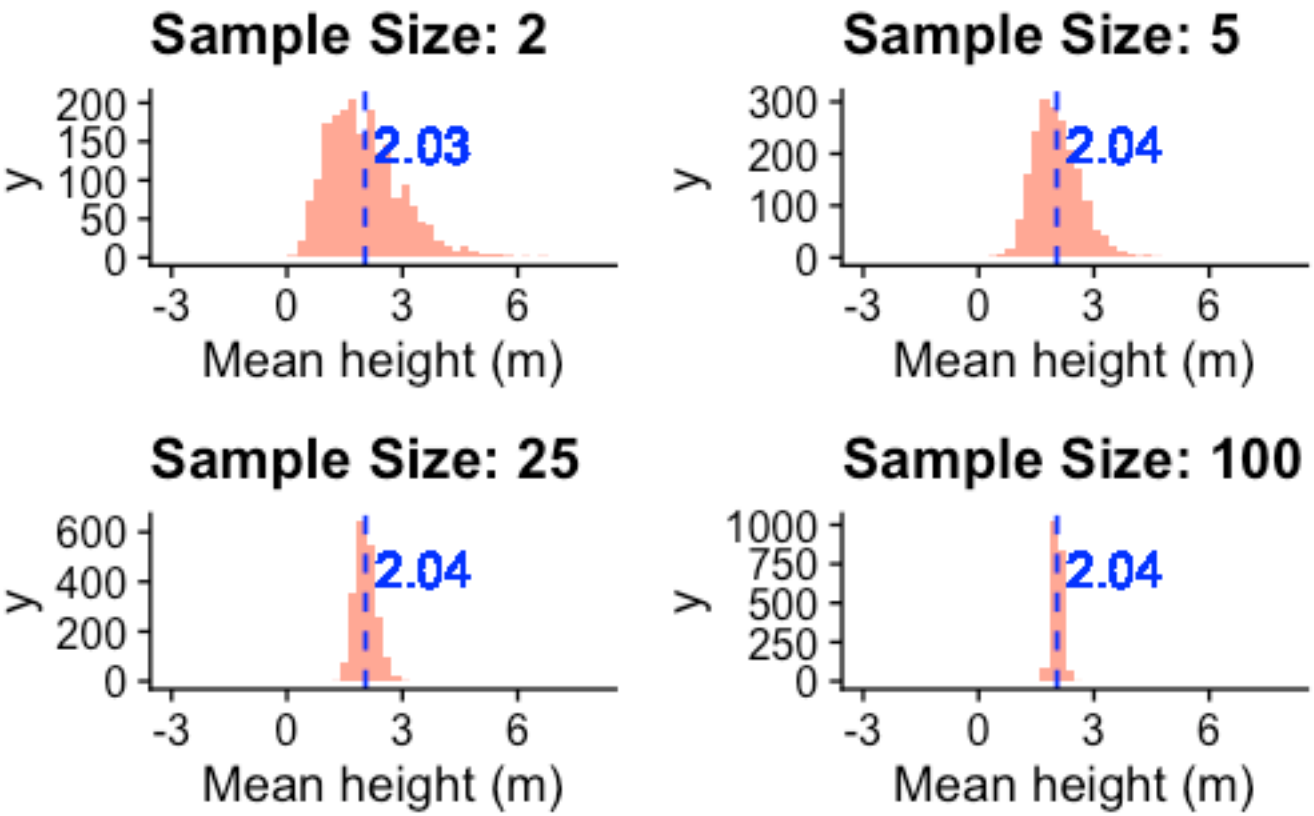


- Fifty random samples per calculated mean, repeated 1000 times.
- **How many samples is enough?**

Effect of sample size

```
library(tidymodels)
library(patchwork)
set.seed(642)

heights ← tibble(heights = rnorm(1000, 1.99, 1))
popmean ← mean(heights$heights)
sample_sizes ← c(2, 5, 25, 100)
n ← length(sample_sizes)

heights ← tibble(heights = rgamma(1000, shape = 2, scale = 1))
sample_sizes ← c(2, 5, 25, 100)
n ← length(sample_sizes)

plots ← lapply(sample_sizes, function(size) {
  df ← heights |>
    rep_sample_n(size = size, reps = 2000) |>
    group_by(replicate) |>
    summarise(xbar = mean(heights))
```

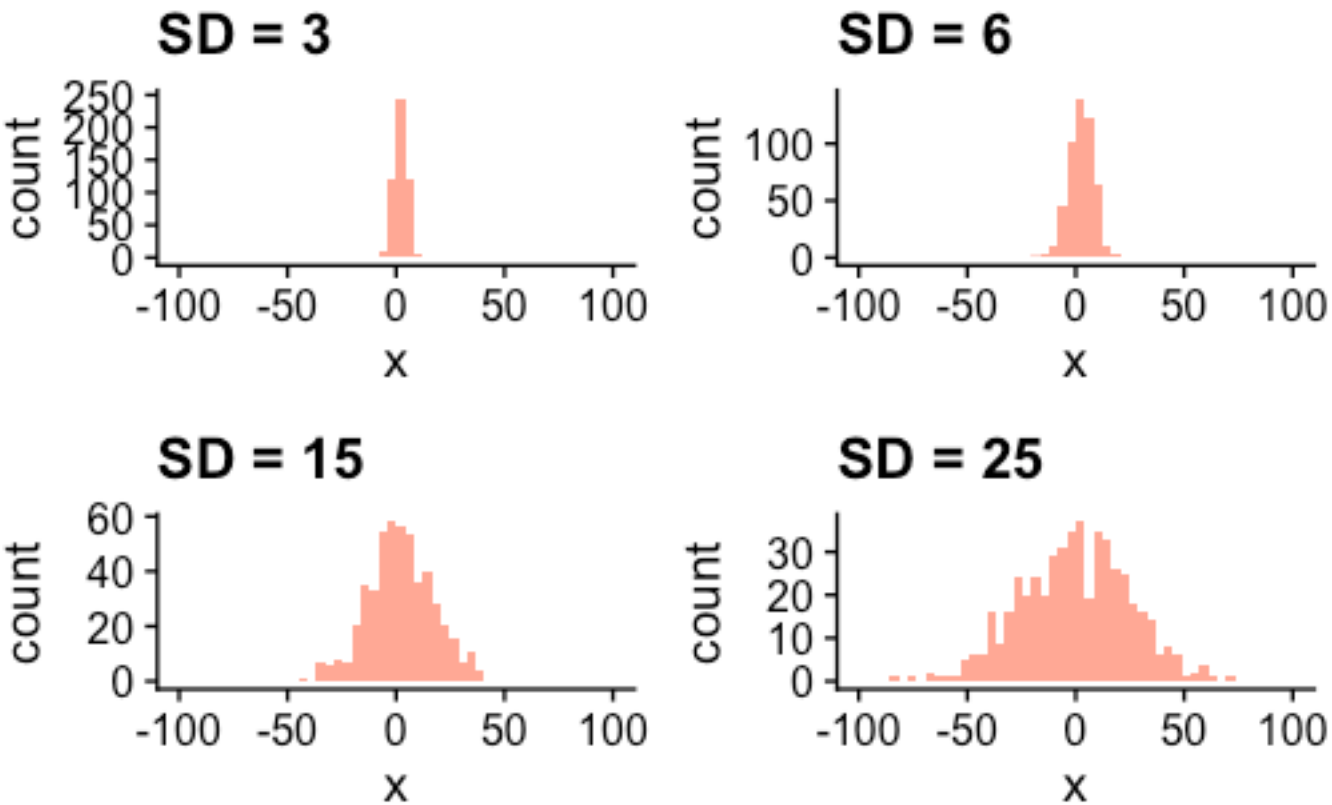
```r
  mean_xbar ← mean(df$xbar)

  ggplot(df, aes(x = xbar)) +
    geom_histogram(fill = "orangered", alpha = 0.5, bins = 50) +
    geom_vline(aes(xintercept = mean_xbar), color = "blue", linetype = "dashed") +
    geom_text(aes(x = mean_xbar, label = sprintf("%.2f", mean_xbar), y = Inf), hjust = -0.1, vjust
= 2, color = "blue") +
    ggtitle(paste0("Sample Size: ", size)) +
    xlab("Mean height (m)") +
    xlim(-3, 8)
})
wrap_plots(plots)
```



Increased sample size leads to a more accurate estimate of the population mean, reflected by the **narrower distribution** of the sample mean, which is captured by the **standard error**.

Effect of variability

```r
set.seed(1221)

# Define a function to generate ggplot objects
generate_plot ← function(sd) {
  data ← rnorm(500, 1.99, sd)
  p ← ggplot(data = tibble(x = data), aes(x = x)) +
    geom_histogram(fill = "orangered", alpha = 0.5, bins = 50) +
    ggtitle(paste("SD =", sd)) +
    xlim(-100, 100)
  return(p)
}

# Apply the function to a list of standard deviations
sds ← c(3, 6, 15, 25)
plots ← lapply(sds, generate_plot)

# Wrap the plots
wrap_plots(plots)
```



Increased variability (i.e. wide range of tree heights) leads to a wider distribution of the sample mean (i.e. less precision), which is *also* reflected by the **standard error**.

CLT drives statistical inference

Because of how predictable the CLT applies to sample means, we can use this to make reasonably accurate inferences about the population mean, even if we do *not* know the population distribution.

- A sampling distribution of the mean *will* be normally distributed for sufficiently large samples – how large is "sufficient" depends on the population distribution
- The mean of the sampling distribution trends towards the population mean with increasing sample size
- To determine how well the sample mean estimates the population mean, we use the standard error of the mean – basically a standard deviation of the sampling distribution

Standard error and confidence intervals

Standard Error of the Mean

- Measures the precision of a sample mean
- Describes variation in sample means – around the true population mean
- Decreases as sample size increases, because we become more “confident” in our estimate

⚠ Formula

$$SE_{\bar{x}} = \frac{s}{\sqrt{n}}$$

- where s is the sample standard deviation
- n is the sample size

When to report SD or SE

Standard Deviation (SD)

- Describes variability in your **data**
- **Stays constant regardless of sample size**

Standard Error (SE)

- Describes precision of your **mean estimate**
- **Decreases with larger sample size ($SE = \frac{SD}{\sqrt{n}}$)**

When reporting statistics:

- Use **mean ± SE** to show precision of your estimate
- Use **mean ± SD** to show spread of your raw data
- SE can appear deceptively small with large sample sizes – **always report sample size**!

Confidence intervals

What is a confidence interval?

- Range of values likely to contain the true population parameter
- Level of confidence (usually 95%) indicates reliability
- Wider intervals = less precise estimates

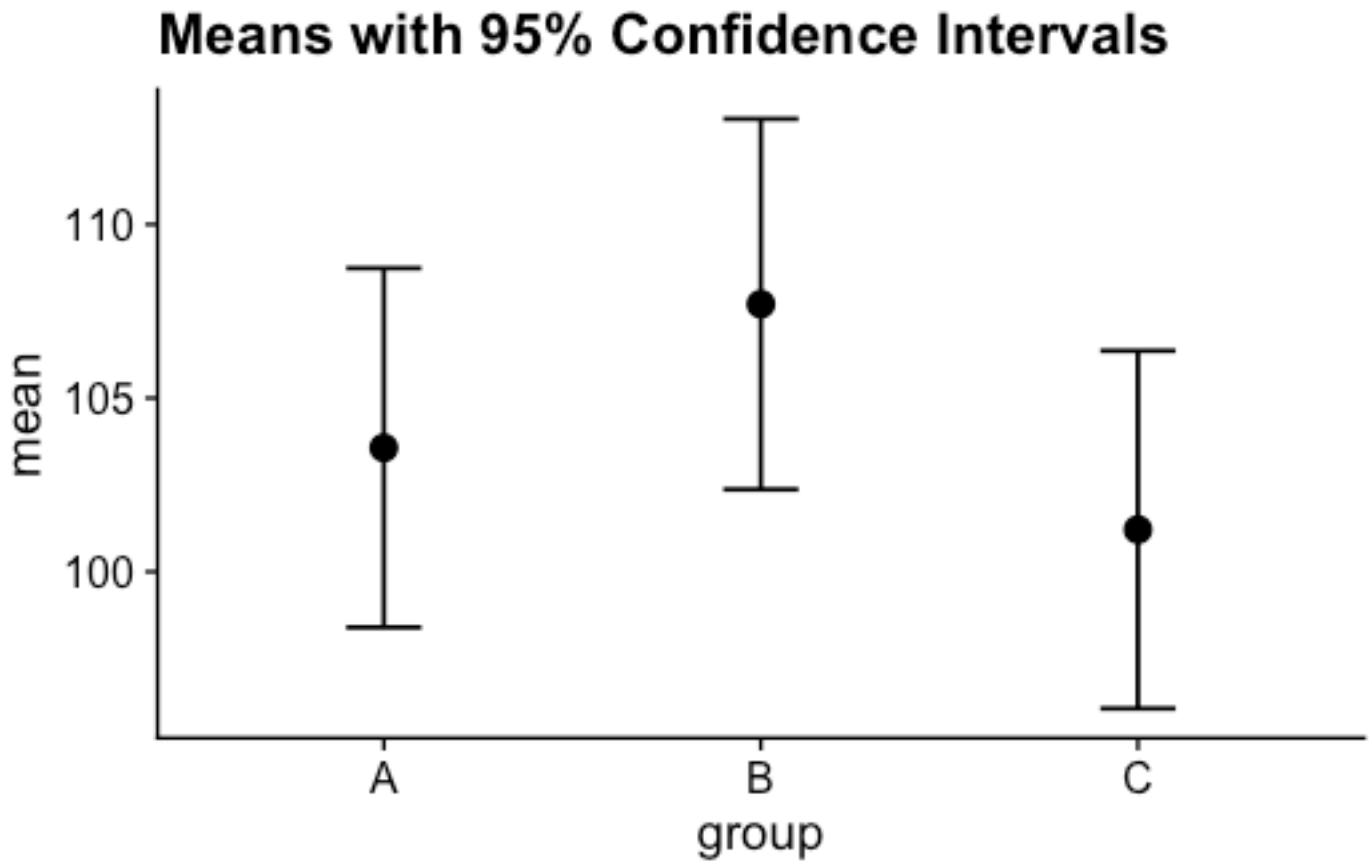
⚠ Formula for 95% CI

$$\bar{x} \pm (t_{n-1} \times SE_{\bar{x}})$$

Visualising confidence intervals

```
# Generate sample data
set.seed(253)
sample_data ← data.frame(
  group = rep(c("A", "B", "C"), each = 30),
  value = c(
    rnorm(30, 100, 15),
    rnorm(30, 110, 15),
    rnorm(30, 105, 15)
  )
)

# Calculate means and CIs
ci_data ← sample_data %>%
  group_by(group) %>%
  summarise(
    mean = mean(value),
    se = sd(value) / sqrt(n()),
    ci_lower = mean - qt(0.975, n() - 1) * se,
    ci_upper = mean + qt(0.975, n() - 1) * se
```

```r
  )

# Plot
ggplot(ci_data, aes(x = group, y = mean)) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = ci_lower, ymax = ci_upper), width = 0.2) +
  ggtitle("Means with 95% Confidence Intervals")
```



We will learn more about confidence intervals in the next lecture.

Thanks for listening! Questions?