

# Week 6: Intro to inference

SOC6302 Winter 2023

## 1 By the end of this lab you should know

## 2 Call the packages we need

We'll need the tidyverse package to create tibbles and do some graphs.

```
library(tidyverse)
```

## 3 Simulating success/failures and calculating proportions

We're going to simulate data asking 1000 people whether or not they like ice-cream, like in the lecture. For this simulation, we want to generate responses (yes and no) based on the probability of a “Yes” being 0.7. That means the probability of a “No” is 0.3 (why?). To do this, we use the `sample` function again (like with flipping a coin), but we add in a `prob` argument which defines our probabilities:

```
possible_outcomes <- c("Y", "N")
samp <- sample(possible_outcomes, 1000, replace = TRUE, prob = c(0.7, 0.3))
samp[1:10] # look at first ten
```

```
[1] "Y" "Y" "Y" "N" "Y" "Y" "Y" "Y" "Y" "Y"
```

### 3.1 Calculate proportion of people who like ice-cream

To calculate the proportion of people who like ice-cream based on this sample, we need to sum up the “Y”s and divide this by the total number of people in the sample:

```
prop <- sum(samp=="Y")/length(samp)
prop
```

```
[1] 0.704
```

## 4 Calculating standard errors and 95% confidence interval

The standard error of the proportion is:

```
se <- sqrt(prop*(1-prop)/100)
se
```

```
[1] 0.0456491
```

### 4.1 Exercise

- What's the probability that the proportion of people who like ice cream will be greater than 0.85?

We can calculate the lower and upper bounds of the 95% confidence interval:

```
lower <- prop - 1.96*se
upper <- prop + 1.96*se
lower; upper
```

```
[1] 0.6145278
```

```
[1] 0.7934722
```

## 5 Sampling distribution

Now we're going to take 100 samples of 1000 people, ask them whether they like ice-cream, and calculate the proportion of people who do like ice-cream. To do this, similarly to last week, we're going to use a for loop:

```
possible_outcomes <- c("Y", "N")
res <- tibble(sample = 1:100, prop = NA) # define an empty dataframe to store the results
set.seed(876)
for (i in 1:100) {
  samp <- sample(possible_outcomes, 1000, replace = TRUE, prob = c(0.7, 0.3))
  res$prop[i] <- sum(samp=="Y")/length(samp)
}
```

## 5.1 Exercises

- Plot the estimated proportions as a histogram
- Change the number of samples to be 500 and plot the results again. What do you notice?

## 6 *t*-distributions and sample means

As in the lecture, I sampled 19 students and asked them how much ice-cream they had eaten in the past week. The sample mean was 400g and the sample standard deviation was 20g.

### 6.1 Calculating 95% confidence interval

To calculate a 95% confidence interval, we need to first find the critical *t*-value:

```
df <- 19-1
t_star <- qt(0.025, df)
t_star
```

```
[1] -2.100922
```

We can then calculate the confidence interval as:

```
n <- 19
x_bar <- 400
se <- 20/sqrt(n)
df <- n - 1
t_star <- qt(0.025, df = 18)
x_bar - t_star*se
```

```
[1] 409.6397
```

```
x_bar + t_star*se
```

```
[1] 390.3603
```

## 6.2 Exercise

- Using the data above, calculate a 99% confidence interval for ice cream consumed.

## 6.3 Hypothesis testing

As in the lecture, imagine I know that the average amount consumed in 2019 was 380 grams. We want to determine using the data collected above whether students are consuming more or less ice cream, versus the other possibility of no change.

So we want to test:

- $H_0$ : no change in consumption ( $\bar{x} = \mu$ )
- $H_A$ : change in consumption ( $\bar{x} \neq \mu$ )

We need to calculate the T-score:

```
T_score <- (x_bar - 380)/se  
T_score
```

```
[1] 4.358899
```

Then calculate the p-value:

```
pt(-1*T_score, df = 18)*2
```

```
[1] 0.0003783575
```

This is significant at a significance level of  $\alpha = 0.05$ , so we reject the null and believe instead that ice cream consumption has changed.

## 6.4 Exercises

- Look back at the 95% confidence interval for  $\bar{x}$ . Did it include 380g? How does this relate to the hypothesis test
- What if instead consumption in 2019 was 395. Repeat the hypothesis test and interpret the results.