Class 32

DATA1220-55, Fall 2024

Sarah E. Grabinski

2024-11-22

Correlation vs Causation

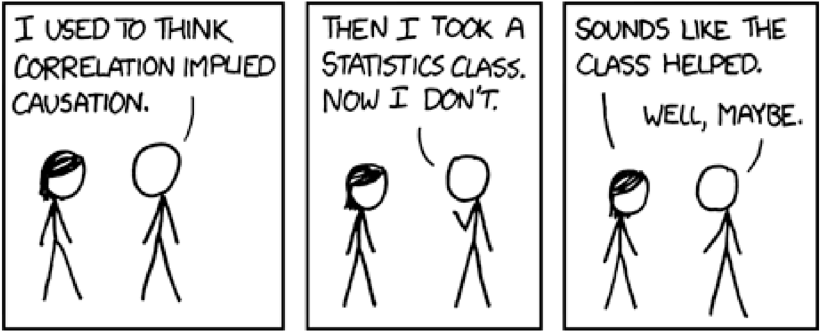


Figure 1: Source: XKCD

Tidyverse

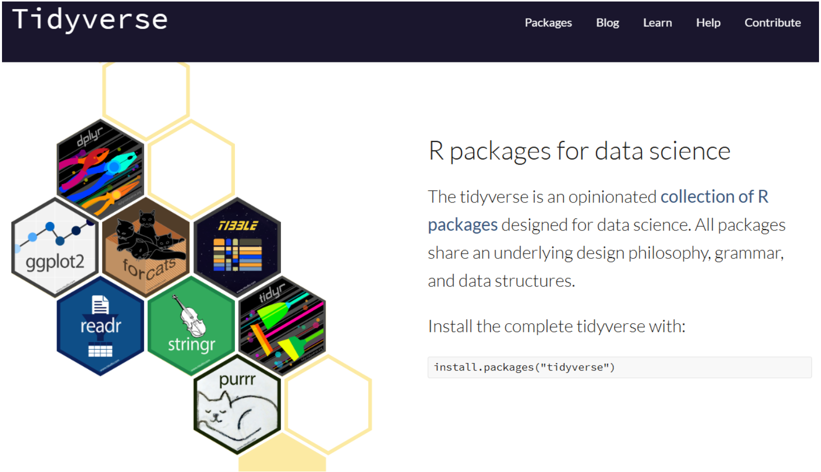


Figure 2: The goal of the ‘tidyverse‘ is to provide tools that support “human-centered” data analysis.

Tidyverse

- `readr`: functions for importing data
- `dplyr`: data cleaning and manipulation
- `ggplot2`: data visualization
- `tibble`: better formats for dataframes
- `forcats`: tools for working with factors
- `stringr`: tools for working with text strings
- `purrr`: tools for functions and vectors
- `tidyr`: tools for reshaping data

Other Useful Packages

- `janitor`: functions like `drop_na()`, `clean_names()`, and `tabyl()`
- `naniar`: very useful for managing missing data
- `kableExtra`: attractive formatting for tables
- `gtsummary`: summary statistic tables with attractive formatting
- `patchwork`: combine ggplot2 figures
- `Hmisc`: statistical analysis tools like `describe()`
- `mosaic`: statistical analysis tools like `favstats()`

Infer Package

- Functions for “tidy” statistical analysis
- Specify statistical models, calculate statistics
- Infer sampling distributions, test hypotheses
- Uses theoretical or permutation based null distributions

Tests in the Infer Package

- 1- or 2-sample proportion- or Z-tests
- 1- or 2-sample t-tests for means
- Chi-squared test of independence for categorical variables
- ANOVA test of independence for numeric variables
- Correlations and simple linear regression

Primary Functions

- `specify()`: set response variable (and explanatory, if needed)
- `calculate()`: calculate statistics
- `observe()`: combines `specify()` and `calculate()`
- `assume()`: sets a null distribution
- `hypothesize()`: sets a null hypothesis
- `get_ci()`: calculate a confidence interval from given distribution
- `visualize()`, `shade_p_value()`: visualize observed statistics vs null hypotheses
- `get_p_value()`: get p-value for observed statistic under null hypothesis

Packages for Today

We will be working with the `gss` dataset from the `infer` package.

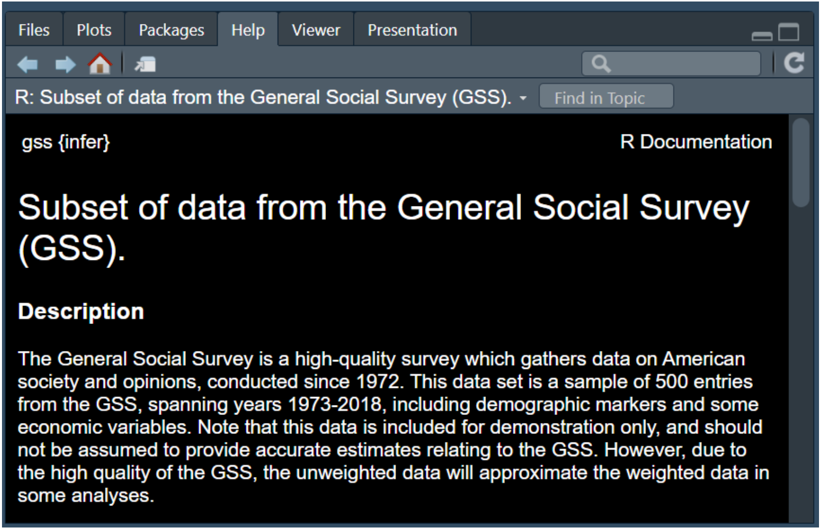
```
library(infer) # statistical functions
library(tidyverse) # always load last in list
```

What's in the data?

Running a question mark before a dataset, function, or package name will do a search in RStudio for help pages on that topic.

```
?gss
```

What's in the data?



gss {infer} R Documentation

Subset of data from the General Social Survey (GSS).

Description

The General Social Survey is a high-quality survey which gathers data on American society and opinions, conducted since 1972. This data set is a sample of 500 entries from the GSS, spanning years 1973-2018, including demographic markers and some economic variables. Note that this data is included for demonstration only, and should not be assumed to provide accurate estimates relating to the GSS. However, due to the high quality of the GSS, the unweighted data will approximate the weighted data in some analyses.

The Data

```
# from the dplyr package
glimpse(gss)
```

```
Rows: 500
Columns: 11
$ year    <dbl> 2014, 1994, 1998, 1996, 1994, 1996, 1990, 2
$ age     <dbl> 36, 34, 24, 42, 31, 32, 48, 36, 30, 33, 21,
$ sex     <fct> male, female, male, male, male, female, fem
$ college <fct> degree, no degree, degree, no degree, degre
$ partyid <fct> ind, rep, ind, ind, rep, rep, dem, ind, rep
$ hompop  <dbl> 3, 4, 1, 4, 2, 4, 2, 1, 5, 2, 4, 3, 4, 4, 2
$ hours   <dbl> 50, 31, 40, 40, 40, 53, 32, 20, 40, 40, 23,
$ income  <ord> $25000 or more, $20000 - 24999, $25000 or m
$ class   <fct> middle class, working class, working class,
$ finrela <fct> below average, below average, below average
$ weight  <dbl> 0.8960034, 1.0825000, 0.5501000, 1.0864000,
```

The Data

```
# from base R
str(gss)
```

```
tibble [500 x 11] (S3: tbl_df/tbl/data.frame)
 $ year    : num [1:500] 2014 1994 1998 1996 1994 ...
 $ age     : num [1:500] 36 34 24 42 31 32 48 36 30 33 ...
 $ sex     : Factor w/ 2 levels "male","female": 1 2 1 1 1 2
 $ college: Factor w/ 2 levels "no degree","degree": 2 1 2
 $ partyid: Factor w/ 5 levels "dem","ind","rep",..: 2 3 2
 $ hompop : num [1:500] 3 4 1 4 2 4 2 1 5 2 ...
 $ hours  : num [1:500] 50 31 40 40 40 53 32 20 40 40 ...
 $ income : Ord.factor w/ 12 levels "lt $1000"<"$1000 to 29
 $ class  : Factor w/ 6 levels "lower class",..: 3 2 2 2 3
 $ finrela: Factor w/ 6 levels "far below average",..: 2 2
 $ weight : num [1:500] 0.896 1.083 0.55 1.086 1.083 ...
```

Codebook

- `age` - age at time of survey
- `sex` - respondent's sex
- `college` - whether subject has a degree
- `partyid` - political affiliation
- `hours` - number of hours worked last week
- `finrela` - opinion of family income

1-Sample Proportion

What proportion of the subjects were female?

```
phat_female <- gss |>
  observe(response = sex,
          success = 'female',
          stat = 'prop')

phat_female
```

```
Response: sex (factor)
# A tibble: 1 x 1
   stat
  <dbl>
1 0.474
```

Infer sampling distribution

Generate a theoretical sampling distribution for the proportion of female respondents

```
dist_female <- gss |>
  specify(response = sex,
          success = 'female') |>
  assume(distribution = "z")

dist_female
```

```
A Z distribution.
```

Confidence Interval

95% confidence interval for the proportion of female respondents

```
ci_female <- get_ci(dist_female,
       point_estimate = phat_female)

ci_female
```

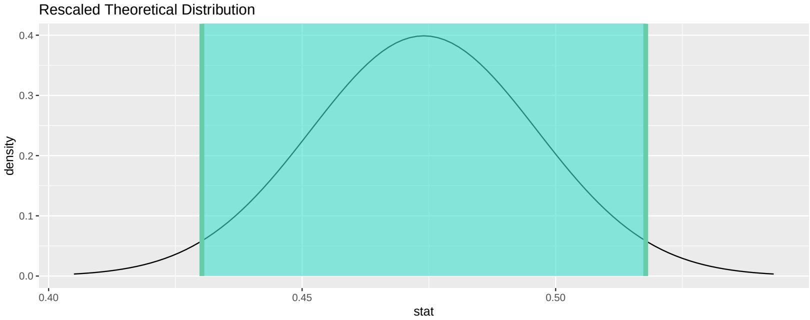
```
# A tibble: 1 x 2
  lower_ci upper_ci
     <dbl>    <dbl>
1    0.430    0.518
```

Visualize

95% confidence interval against the theoretical sampling distribution

```
visualize(dist_female) +  # NOTE THE + for ggplot2
  shade_confidence_interval(endpoints = ci_female)
```

Visualize



Rescaled Theoretical Distribution
density
0.4
0.3
0.2
0.1
0.0
0.40
0.45
0.50
stat

Hypothesis Test

Let's test if a majority of respondents were women. Set the null distribution.

```
null_female <- gss |>
  specify(response = sex,
          success = 'female') |>
  hypothesize(null = 'point',
              p = 0.5) |>
  assume('z')

null_female
```

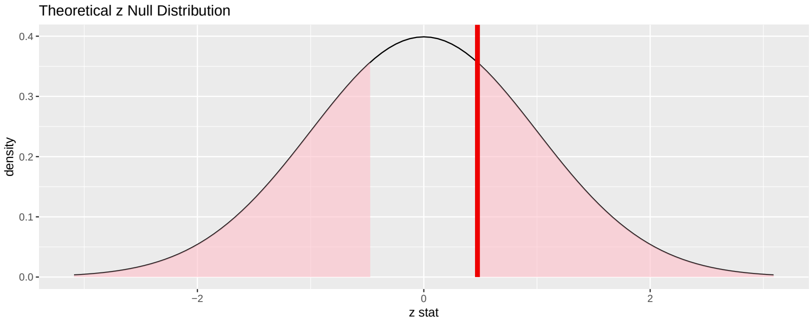
```
A Z distribution.
```

Hypothesis Test

Let's visualize our observed data against the null distribution.

```
visualize(null_female) +
  shade_p_value(obs_stat = phat_female,
                direction = 'two-sided')
```

Hypothesis Test



Hypothesis Test

Let's find the p-value for our observed data under the null hypothesis.

```
null_female |>
  get_p_value(obs_stat = phat_female,
              direction = 'two-sided')
```

```
# A tibble: 1 x 1
  p_value
    <dbl>
1   0.635
```

2-Sample T-Test

Do college graduates work the same number of hours as non-college graduates?

$$H_0: \mu_{\text{degree}} - \mu_{\text{no degree}} = 0$$

$$H_0: \mu_{\text{degree}} - \mu_{\text{no degree}} \neq 0$$

Sample Statistics

```
xbar_diff <- gss |>
  observe(hours ~ college,
          stat = 'diff in means',
          order = c('degree',
                    'no degree'))

xbar_diff
```

```
Response: hours (numeric)
Explanatory: college (factor)
# A tibble: 1 x 1
   stat
  <dbl>
1  1.54
```

Infer the sampling distribution

```
dist_diff <- gss |>
  specify(hours ~ college) |>
  assume(distribution = "t")

dist_diff
```

```
A T distribution with 366 degrees of freedom.
```

Confidence interval for difference

```
ci_diff <- get_ci(dist_diff,
       point_estimate = xbar_diff)

ci_diff
```

```
# A tibble: 1 x 2
  lower_ci upper_ci
     <dbl>    <dbl>
1    -1.16     4.24
```

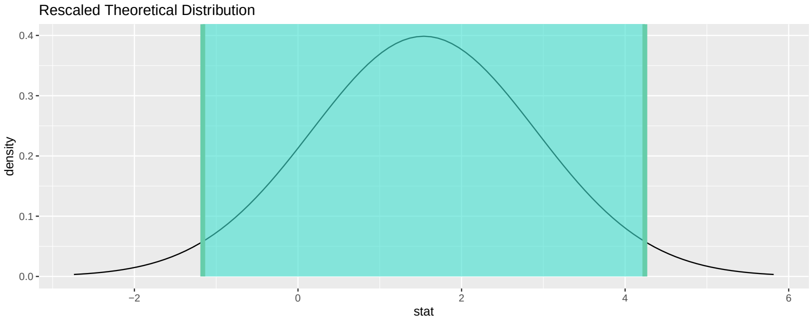
Visualize

95% confidence interval against the theoretical sampling distribution

```
visualize(dist_diff) +  # NOTE THE + for ggplot2
  shade_confidence_interval(endpoints = ci_diff)
```

Visualize

95% confidence interval against the theoretical sampling distribution



Hypothesis Test

```
null_diff <- gss |>
  specify(hours ~ college) |>
  hypothesize(null = 'independence') |>
  assume('t')

null_diff
```

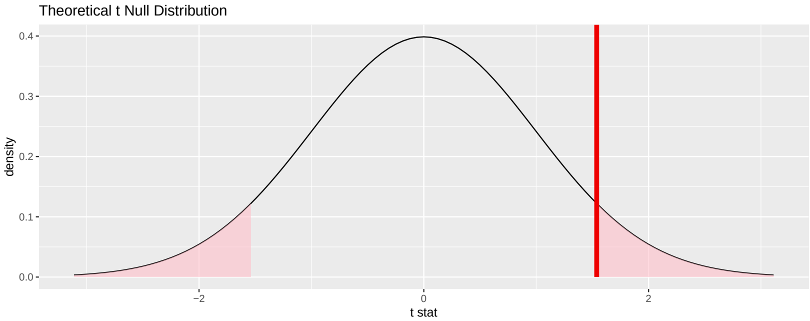
```
A T distribution with 366 degrees of freedom.
```

Hypothesis Test

Let's visualize our observed data against the null distribution.

```
visualize(null_diff) +
  shade_p_value(obs_stat = xbar_diff,
                direction = 'two-sided')
```

Hypothesis Test



Hypothesis Test

Let's find the p-value for our observed data under the null hypothesis.

```
null_diff |>
  get_p_value(obs_stat = xbar_diff,
              direction = 'two-sided')
```

```
# A tibble: 1 x 1
  p_value
    <dbl>
1   0.125
```

Practice

- Open RStudio and import the `LungCapData.xls` from the quiz
- Construct a 95% confidence interval for the proportion of US citizens that smoke
- Conduct a 2-sample t-test for a difference in lung capacity between smokers and non-smokers.
- `infer` package coding examples: https://infer.netlify.app/articles/observed_stat_examples