

Overview

Review of ggplot

Review of material covered in the class so far

If there is time/interest

• Bonus features of ggplot: special geoms, animation, interactive graphics

Announcements

Midterm exam is on Thursday

- Bring a pen and a pencil
- One page (2 sides) with code and equations only!
 - You will turn in this page of notes with your exam (put your name on it)
 - You can write down conditions for hypothesis tests. Have SE formulas, etc.

Office hours this week

- Stephan and Nathan are doing another review session tonight
- No TA office hours since no homework

Review of the grammar of graphics and ggplot



The grammar of graphics

Leland Wilkinson noticed similarities between many graphs and tried to generate a 'grammar' that could be used to express a graph

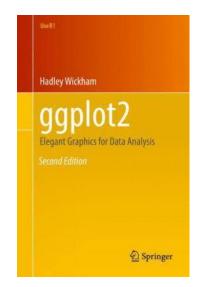
• i.e., a list elements that can be combined together to create a graph

Statistics and Computing

Leland Wilkinson

The Grammar of Graphics
Second Edition

Hadley Wickham implemented these ideas in R in the ggplot2 package



Graphs are composed of...

A Frame: Coordinate system on which data is placed

• ggplot() +

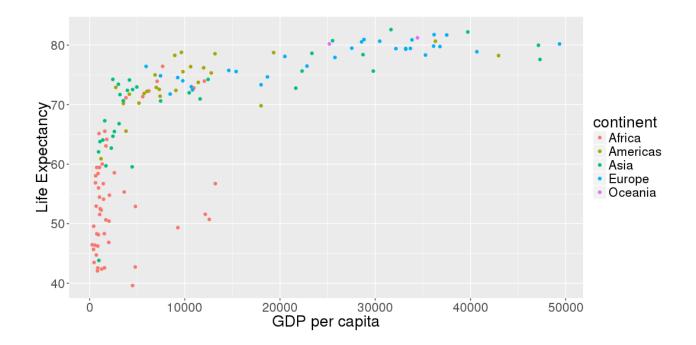
Glyphs: basic graphic unit representing cases or statistics

- Data is **mapped** onto these aesthetics such as: shape, color, size, etc. and/or aesthetics can be set to a fixed value
 - geom_point(aes(x = gdpPercap, y = lifeExp, color = continent))
 geom_point(aes(x = gdpPercap, y = lifeExp), color = "red")

Scales and guides: shows how to interpret axes and other properties of the glyphs

scale x continuous(trans = "log10")

scale color brewer(type = "qua", palette = 2)



Plots can also contain...

Facets: allows for multiple side-by-side graphs based on a categorical variable

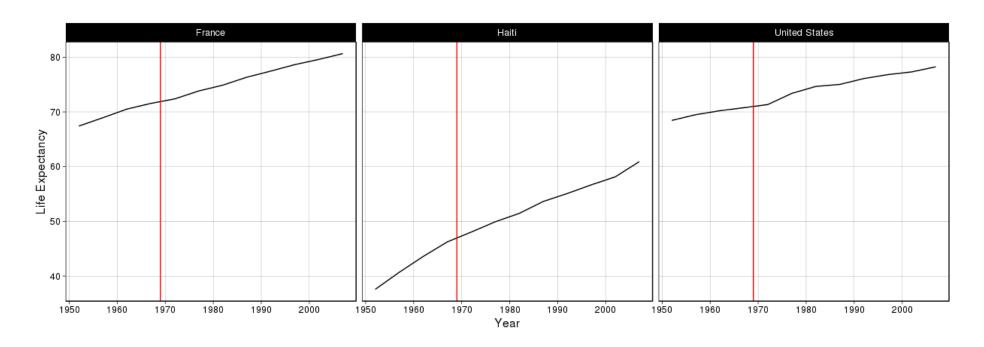
facet_wrap(~country)

Layers: allows for more than one types of data to be mapped onto the same figure

geom_vline(xintercept = 1969, col = "red")

Theme: contains finer points of display (e.g., font size, background color, etc.)

theme_wsj()



Questions?

ggplot2 cheat sheet

Data visualization with ggplot2:: CHEAT SHEET

Basics

ggplot2 is based on the grammar of graphics, the idea that you can build every graph from the same components: a data set, a coordinate system, and geoms-visual marks that represent data points.



To display values, map variables in the data to visual properties of the geom (aesthetics) like size, color, and x and y locations.



Complete the template below to build a graph.

ggplot (data = <DATA>) + <GEOM FUNCTION> (mapping = aes) <MAPPINGS> stat = <STAT>, position = <POSITION>) + <COORDINATE FUNCTION> + <FACET FUNCTION> + <SCALE FUNCTION> +

<THEME FUNCTION> ggplot(data = mpg, aes(x = cty, y = hwy)) Begins a plot that you finish by adding layers to. Add one geom function per layer.

last_plot() Returns the last plot.

ggsave("plot.png", width = 5, height = 5) Saves last plot as 5' x 5' file named "plot.png" in working directory. Matches file type to file extension.

Aes Common aesthetic values.

color and fill - string ("red", "#RRGGBB")

linetype - integer or string (0 = "blank", 1 = "solid", 2 = "dashed", 3 = "dotted", 4 = "dotdash", 5 = "longdash", 6 = "twodash")

lineend - string ("round", "butt", or "square") linejoin - string ("round", "mitre", or "bevel")

size - integer (line width in mm) 0 1 2 3 4 5 6 7 8 9 00 11 12 □○△+×◇▽宮*◆●苅田 shape - integer/shape name or 13 14 15 16 17 18 19 20 21 22 23 24 25 a single character ("a") ⊠⊠□○△○○○□◆△▽

Geoms Use a geom function to represent data points, use the geom's aesthetic properties to represent variables. Each function returns a layer.

GRAPHICAL PRIMITIVES

a <- ggplot(economics, aes(date, unemploy)) b <- ggplot(seals, aes(x = long, y = lat))

> a + geom_blank() and a + expand_limits() Ensure limits include values across all plots.

b + geom curve(aes(yend = lat + 1. xend = long + 1), curvature = 1) - x, xend, y, yend, alpha, angle, color, curvature, linetype, size

a + geom_path(lineend = "butt", linejoin = "round", linemitre = 1) x, y, alpha, color, group, linetype, size

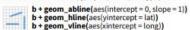
a + geom_polygon(aes(alpha = 50)) - x, y, alpha, color, fill, group, subgroup, linetype, size b + geom_rect(aes(xmin = long, ymin = lat,

xmax = long + 1, ymax = lat + 1)) - xmax, xmin, ymax, ymin, alpha, color, fill, linetype, size

a + geom_ribbon(aes(ymin = unemploy - 900, ymax = unemploy + 900)) - x, ymax, ymin, alpha, color, fill, group, linetype, size

LINE SEGMENTS

common aesthetics: x, y, alpha, color, linetype, size



b + geom_segment(aes(vend = lat + 1, xend = long + 1)) b + geom_spoke(aes(angle = 1:1155, radius = 1))

ONE VARIABLE continuous

c <- ggplot(mpg, aes(hwy)); c2 <- ggplot(mpg)



c + geom_area(stat = "bin") x, y, alpha, color, fill, linetype, size



c + geom_dotplot() x, y, alpha, color, fill



c + geom_freqpoly() x, y, alpha, color, group, linetype, size



c2 + geom_qq(aes(sample = hwy)) x, y, alpha, color, fill, linetype, size, weight

d <- ggplot(mpg, aes(fl))



d + geom_bar() x, alpha, color, fill, linetype, size, weight

TWO VARIABLES

both continuous e <- ggplot(mpg, aes(cty, hwy))



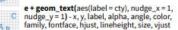
e + geom_label(aes(label = cty), nudge_x = 1, nudge_y = 1) - x, y, label, alpha, angle, color. family, fontface, hjust, lineheight, size, vjust



e + geom_quantile() x, y, alpha, color, group, linetype, size, weight



e + geom smooth(method = lm) x, y, alpha, color, fill, group, linetype, size, weight



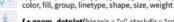
one discrete, one continuous

f <- ggplot(mpg, aes(class, hwy))



f + geom col() x, y, alpha, color, fill, group, linetype, size

x, y, lower, middle, upper, ymax, ymin, alpha,



f + geom_dotplot(binaxis = "y", stackdir = "center") x, y, alpha, color, fill, group



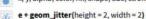
f + geom_violin(scale = "area") x, y, alpha, color, fill, group, linetype, size, weight

both discrete

g <- ggplot(diamonds, aes(cut, color))



g + geom_count() x, y, alpha, color, fill, shape, size, stroke

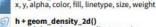


x, y, alpha, color, fill, shape, size

I + geom_contour_filled(aes(fill = z))

continuous bivariate distribution h <- ggplot(diamonds, aes(carat, price))





x, y, alpha, color, group, linetype, size

ggplot.



h + geom_hex() x, y, alpha, color, fill, size

continuous function

i <- ggplot(economics, aes(date, unemploy))



i + geom area()

x, y, alpha, color, fill, linetype, size



/ i + geom line() x, y, alpha, color, group, linetype, size

i + geom_step(direction = "hv")

x, y, alpha, color, group, linetype, size

visualizing error

df <- data.frame(grp = c("A", "B"), fit = 4:5, se = 1:2) j <- ggplot(df, aes(grp, fit, ymin = fit - se, ymax = fit + se))



ymin, alpha, color, fill, group, linetype, size

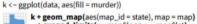




x, ymin, ymax, alpha, color, group, linetype, size



data <- data.frame(murder = USArrests\$Murder, state = tolower(rownames(USArrests))) map <- map data("state")



+ expand_limits(x = map\$long, y = map\$lat) map_id, alpha, color, fill, linetype, size

THREE VARIABLES

seals\$z <- with(seals, sqrt(delta_long^2 + delta_lat^2)); I <- ggplot(seals, aes(long, lat))



l + geom_contour(aes(z = z)) x, y, z, alpha, color, group, linetype, size, weight

x, y, alpha, color, fill, group, linetype, size, subgroup



l + geom_raster(aes(fill = z), hjust = 0.5, viust = 0.5, interpolate = FALSE) x, y, alpha, fill



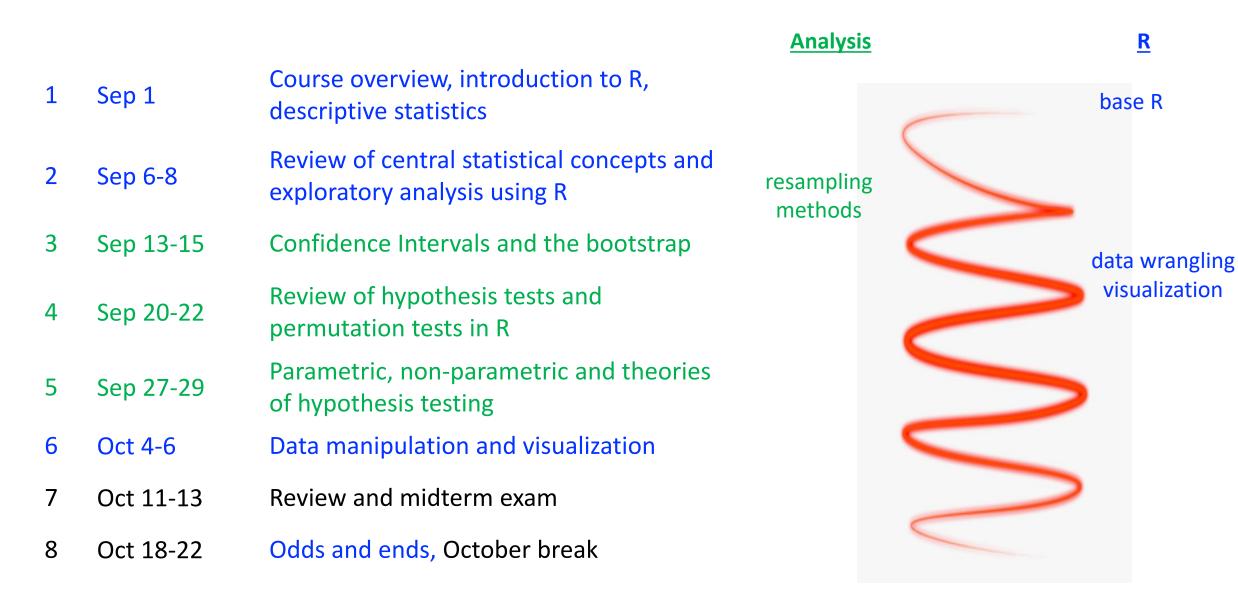
l + geom_tile(aes(fill = z)) x, y, alpha, color, fill, linetype, size, width



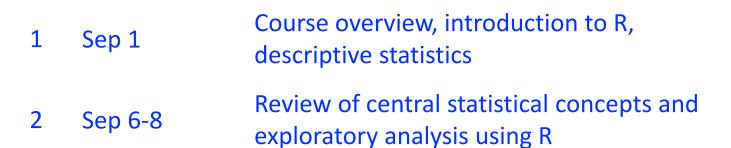


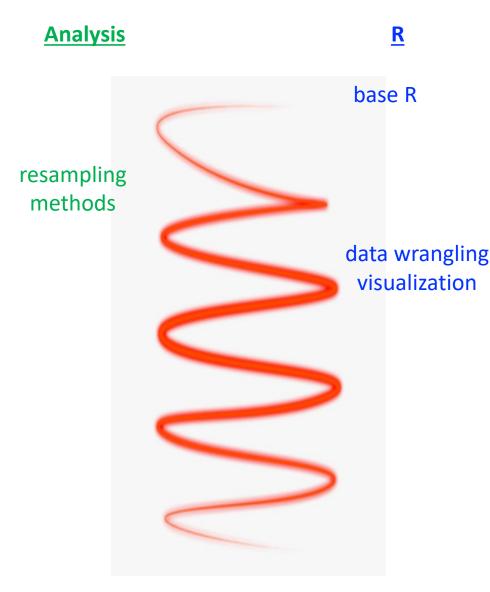


What we have covered so far...



What we have covered so far...





Parameters and statistics commonly used symbols



	Population parameter (Plato)	Sample statistic (shadow)
Mean	μ	χ
Standard deviation	σ	S
Proportion	π	ρ̂
Correlation	ρ	r
Regression slope	β	b

Base R

Basics of R

```
> my_vec <- c(5, 28, 19)
```

> inds_less_than_10 <- 10

How to plot data in base R

- > drinks_table <- table(profiles\$drinks)
- > barplot(drinks_table)
- > pie(drinks_table)
- > hist(profiles\$height)

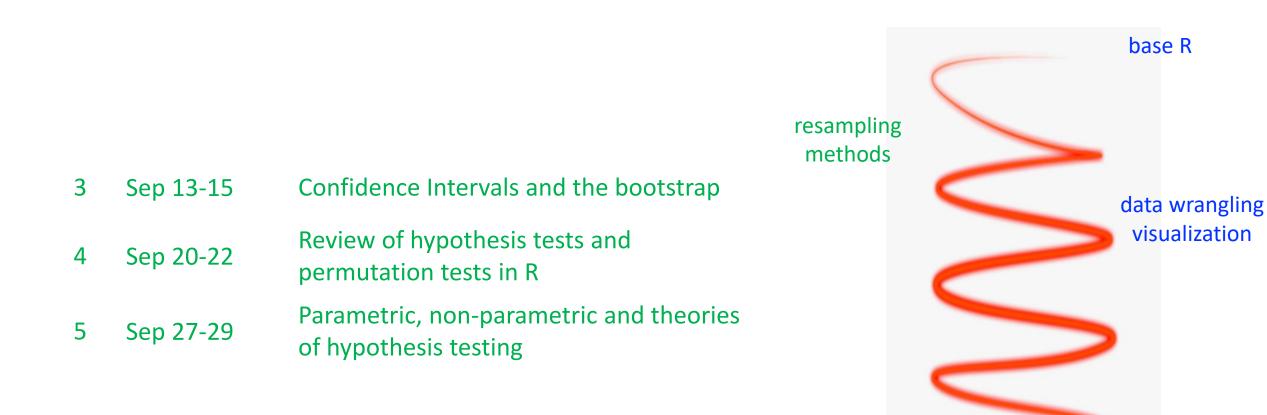
```
For loops

my_results <- NULL

for (i in 1:100) {

    my_results[i] <- i^2
}
```

What we have covered so far...



Analysis

<u>R</u>

Probability and confidence intervals

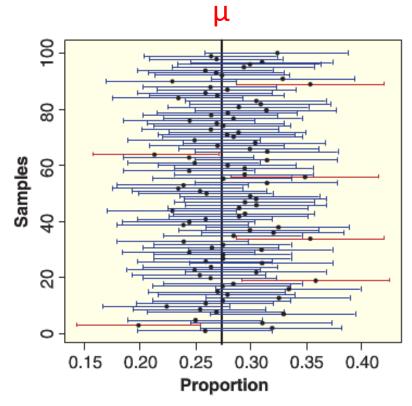
Probability functions; e.g., rnorm, pnorm, dnorm, qnorm

Confidence intervals:

$$Cl_{95} = stat \pm 2 \cdot SE$$

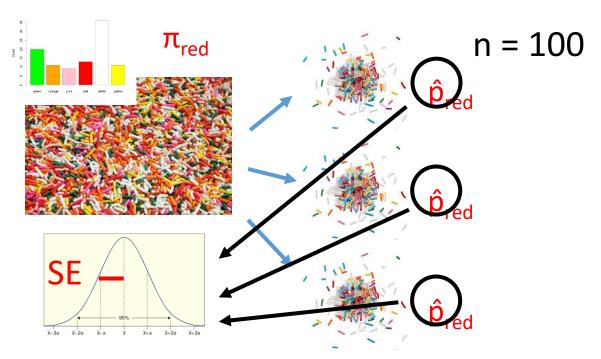






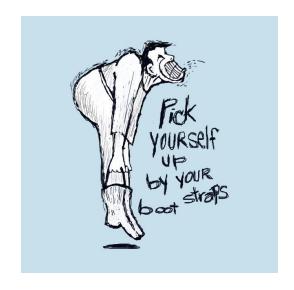
Sampling and bootstrap distributions

Sampling distribution



Sampling distribution!

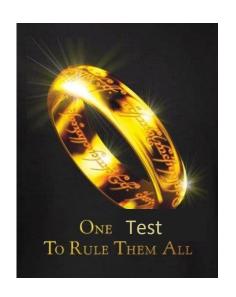
Bootstrap distribution

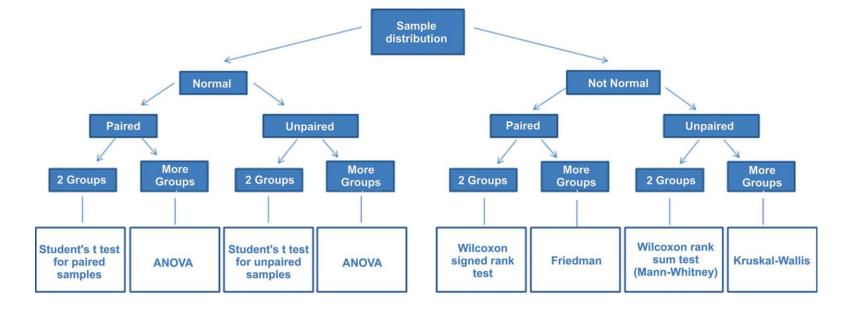


Sample with replacement from our original sample to mimic a sampling distribution

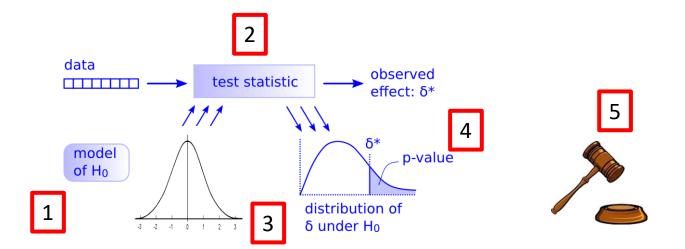
$$Cl_{95} = stat \pm 2 \cdot SE^*$$

Hypothesis tests

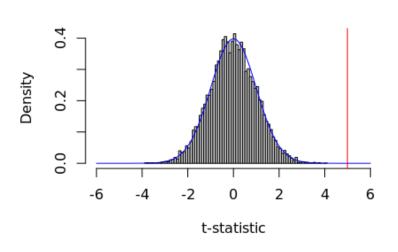




Just need to follow 5 steps!



Null distribution



Randomization/permutation tests

Create a null distribution through computational simulations/shuffling

• rbinom(), sample(), etc.

 H_0 : $\pi = 0.5$

 H_{Δ} : $\pi > 0.5$

$$H_0: \mu_T - \mu_C = 0$$

 H_A : $\mu_T - \mu_C > 0$

$$H_0: \mu_i = \mu_j ... = ... \mu_k$$

 $H_A: \mu_i \neq \mu_i$ for some i, j







Data	1 Sample	2 Samples	> 2 Samples
	H_0 : π = p_0 H_A : π ≠ p_0	$H_0: \pi_1 = \pi_2$ $H_A: \pi_1 \neq \pi_2$	H_0 : $\pi_1 = p_1$, $\pi_2 = p_2$,, $\pi_k = p_k$ H_A : At least one p_i is different than specified
Categorical	Flip "coins"	Flip "coins"	Flip coins
data	rflip_count()	rflip_count()	rflip_count()
	H_0 : $\mu = v_0$ H_A : $\mu \neq v_0$	$H_0: \mu_1 = \mu_2$ $H_A: \mu_1 \neq \mu_2$	H_0 : $\mu_1 = \mu_2 = = \mu_k$ H_A : At least one μ_i is different
Quantitative data	<u>resample</u>	Shuffle data	Shuffle data
	sample(, replace = TRUE)	shuffle()	shuffle()

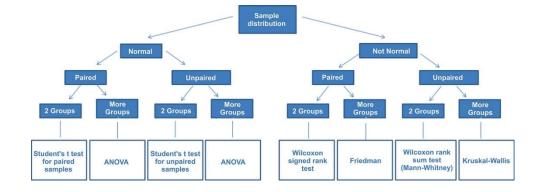
Parametric tests

Use mathematical density functions for the null distribution

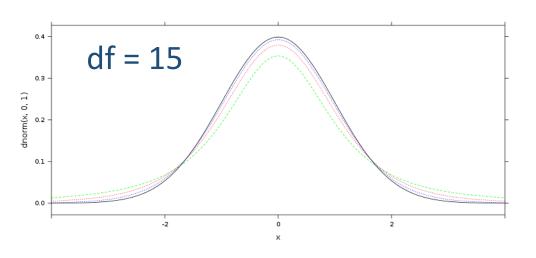
$$H_0: \mu_T - \mu_C = 0$$

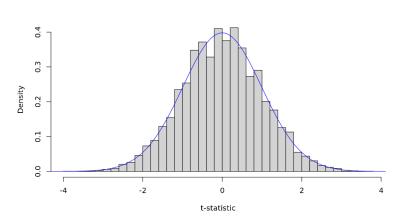
 $H_A: \mu_T - \mu_C > 0$

$$t = \frac{\bar{x}_t - \bar{x}_c}{\sqrt{\frac{s_t^2}{n_t} + \frac{s_c^2}{n_c}}}$$









Data	1 Sample	2 Samples	> 2 Samples
	H_0 : $\pi = p_0$ H_A : $\pi \neq p_0$	$H_0: \pi_1 = \pi_2$ $H_A: \pi_1 \neq \pi_2$	H_0 : $\pi_1 = p_1$, $\pi_2 = p_2$,, $\pi_k = p_k$ H_A : At least one p_i is different than specified
Categorical data	<u>z-test</u>	z-test or a chi-square	<u>chi-square test</u>
	$z = \frac{\hat{p} - p_0}{\sqrt{\frac{p_0(1 - p_0)}{n}}}$	$z = \frac{\hat{p_1} - \hat{p_2}}{\sqrt{\frac{\hat{p_1}(1-\hat{p_1})}{n_1} + \frac{\hat{p_2}(1-\hat{p_2})}{n_2}}}$	$\chi^2 = \sum_{i=1}^k \frac{(Observed_i - Expected_i)^2}{Expected_i}$
	H_0 : $\mu = v_0$ H_A : $\mu \neq v_0$	$H_0: \mu_1 = \mu_2$ $H_A: \mu_1 \neq \mu_2$	H_0 : $\mu_1 = \mu_2 = = \mu_k$ H_A : At least one μ_i is different
Quantitative data	One sample t-test	Two sample t-test	Analysis of Variance
	$t = \frac{\bar{x} - v_0}{s / \sqrt{n}}$	$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$	$F = \frac{\frac{1}{K-1} \sum_{i=1}^{K} n_i (\bar{x}_i - \bar{x}_{tot})^2}{\frac{1}{N-K} \sum_{i=1}^{K} \sum_{j=1}^{n_i} (x_{ij} - \bar{x}_i)^2}$
			$df_1 = k, df_2 = n - k$

Data	1 Sample	2 Samples
Categorical Data	$SE = \sqrt{rac{\pi(1-\pi)}{n}}$	$SE = \sqrt{\frac{\pi_1(1-\pi_1)}{n_1} + \frac{\pi_2(1-\pi_2)}{n_2}}$
	$\hat{p} \pm z^* \cdot \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$	$\hat{p_1} - \hat{p_2} \pm z^* \sqrt{\frac{\hat{p_1}(1-\hat{p_1})}{n_1} + \frac{\hat{p_2}(1-\hat{p_2})}{n_2}}$
Quantitative Data	$SE = \frac{s}{\sqrt{n}}$ $x \pm t^* \frac{s}{\sqrt{n}}$	$SE = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$ $(\overline{x_1} - \overline{x_2}) \pm t^* \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$
	V T	V 12

Theories of hypothesis testing



Fisher (1890-1962)

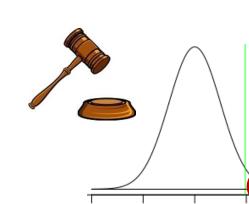


Neyman (1894-1981)



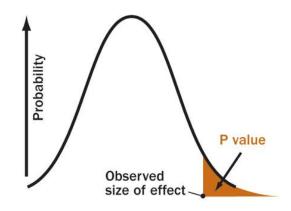
Pearson (1895-1980)

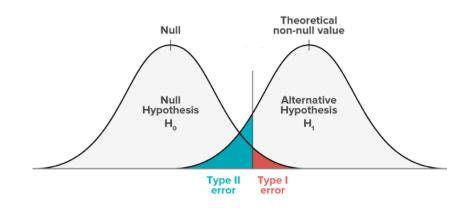
p-value a strength of evidence



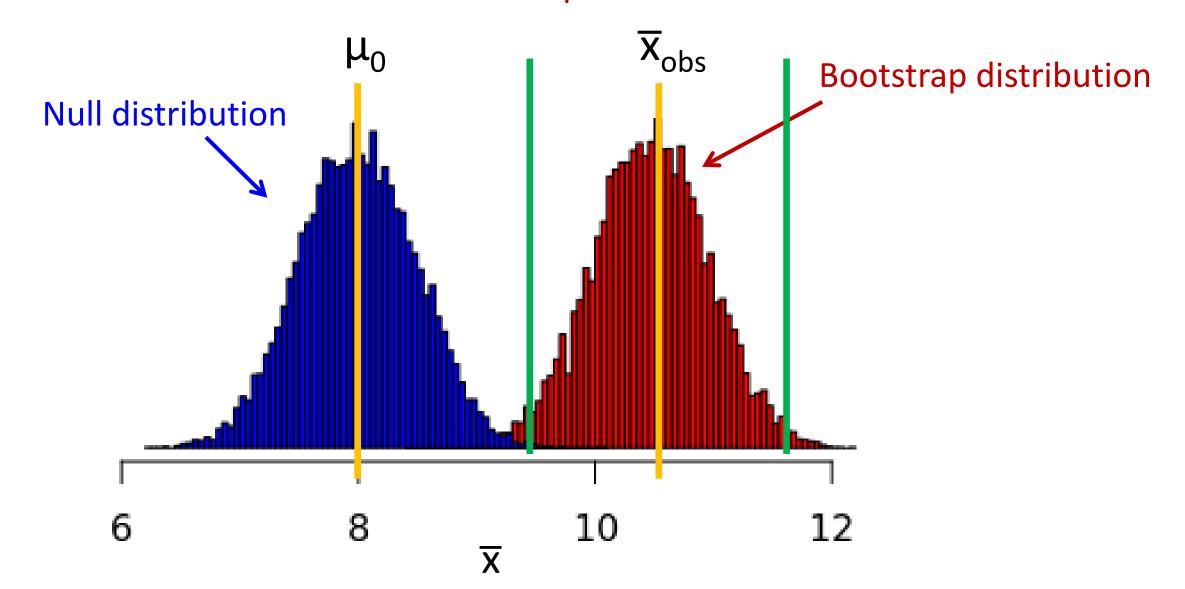
Use p-value to make a decision







Relationship between null and bootstrap distributions



Data manipulation with dplyr

dplyr is a package that has a set of verbs for transformations data

- All these function take a data frame and other arguments and return a data frame
- 1. filter()
- 2. select()
- 3. mutate()
- 4. arrange()
- 5. summarize()
- 6. group_by()

```
age body_type diet

22 a little extra strictly anything

35 average mostly other

38 thin anything

23 thin vegetarian

29 athletic NA

29 average mostly anything
```

```
film results <- movies |>
   filter(title_type == "Feature Film") |>
   select(critics score, audience score, genre) |>
   mutate(audience prefers =
         audience score - critics score) |>
   group_by(genre) |>
    summarize(mean audience prefers =
          mean(audience prefers)) |>
     arrange(desc(mean audience prefers))
head(film results)
```

Grammar of graphics with ggplot

A Frame: Coordinate system on which data is placed

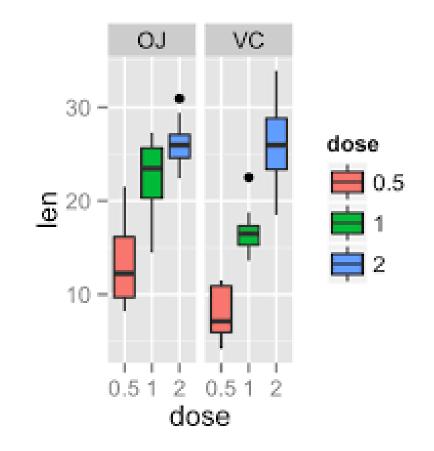
Glyphs: basic graphic unit representing cases or statistics

Scales and guides: shows how to interpret axes and other properties of the glyphs

Facets: allows for multiple side-by-side graphs based on a categorical variable

Layers: allows for more than one types of data to be mapped onto the same figure

Theme: contains finer points of display (e.g., font size, background color, etc.)



Questions

