Stat 145, Fri 12-Mar-2021 -- Fri 12-Mar-2021
Biostatistics
Spring 2021

Friday, March 12th 2021

Wk 6, Fr
Topic:: Randomization distributions
Read:: Lock5 4.4

Randomization distributions
- meant to approximate null distribution
- principles guiding construction
  1. Be consistent with the null hypothesis
     Most important indicator of this: that its mean is the null value
  2. Use data from the original sample
  3. Reflect the way the data were collected.

  missing here: any statement of "sample with (or w/out) replacement"

Randomization for a single mean
- null and alternative hypotheses

H_0: μ = some number (call it μ_0), H_a: $\mu \neq \mu_0$

- natural estimator: x-bar

- observed before that a bootstrap distribution
  has roughly the same spread as sampling distribution
  is centered in the wrong place:
    almost like sampling dist., translated left or right, equals bstrap dist
    see overlaid histograms for both in notes of Mar. 1

Idea for randomization: take a bootstrap dist., and translate it

Matched pairs vs. two independent samples

Two independent samples vs. matched pairs

Consider this research question: Is it better to fish a certain lake from shore, or from a boat?

Our response variable will be quantitative, the ratio of fishing hours to fish caught. Here is some data.

month	Apr.	May	June	July	Aug.	Sept.	Oct.
shore	3.3	3.6	3.9	3.2	3.0	1.8	1.6
boat	3.8	3.0	3.3	2.2	1.6	1.4	1.5

We have a binary categorical explanatory variable: "Where fishing from?", with values "shore" and "boat". We have a quantitative response variable. We have bootstrapped and tested hypotheses for the difference $\mu_1 - \mu_2$, but the methods I've discussed have presumed *independent samples*. The data collected to investigate the question do not represent independent samples. The responses in the different months are naturally related: when one goes up, the other seems more likely to go up, both being related to the population of fish in the lake during that month. This data is **matched pairs** data, and we should:

- use the months as *cases*, and produce for each case a single difference:

month	Apr.	May	June	July	Aug.	Sept.	Oct.
shore	3.3	3.6	3.9	3.2	3.0	1.8	1.6
boat	3.8	3.0	3.3	2.2	1.6	1.4	1.5
difference	-0.5	0.6	0.6	1.0	1.4	0.4	0.1

- Proceed as if in a "single mean" setting. A confidence interval would be for the purpose of estimating the mean difference μ_{diff}. An hypothesis test would focus on hypotheses:

$$\mathbf{H}_0\colon \mu_{\text{diff}} = 0 \qquad \text{vs.} \qquad \mathbf{H}_0\colon \mu_{\text{diff}} \neq 0.$$

  Either way, the slips of paper we would insert into a bag for bootstrapping or randomization would contain only the last set of numbers, the differences.

Practice: Does the data suggest independent samples, warranting analysis on the difference of two means $\mu_1 - \mu_2$, or is it matched pairs?

1. A study was conducted to investigate the effectiveness of hypnotism in reducing pain. Results for randomly selected subjects are shown in the table below. A lower score indicates less pain.

subject	A	B	C	D	E	F	G	H
before	6.6	6.5	9.0	10.3	11.3	8.1	6.3	11.6
after	6.8	2.4	7.4	8.5	8.1	6.1	3.4	2.0

2. To study the effects of a drug on blood pressure, patients had a base reading taken of their diastolic blood pressure. After 3 weeks on the medication, new readings of their diastolic blood pressures were taken.

3. A collection of statistics students is randomly assigned to two groups. One group is given a study regimen that includes listening to recordings of classical music by Mozart, while the other group must study in silence. The response variable is student scores on an exam.

Hypothesis test from

data frame: BodyTemp50

col/var (quantitative): BodyTemp

Hypotheses:

$H_0: \mu = 98.6$, $H_a: \mu \neq 98.6$

Studies involving two groups can be

- independent samples

Group 1 has a mean $\bar{x}_1$, sample

μ_1, population

Group similarly has $\bar{x}_2$, μ_2

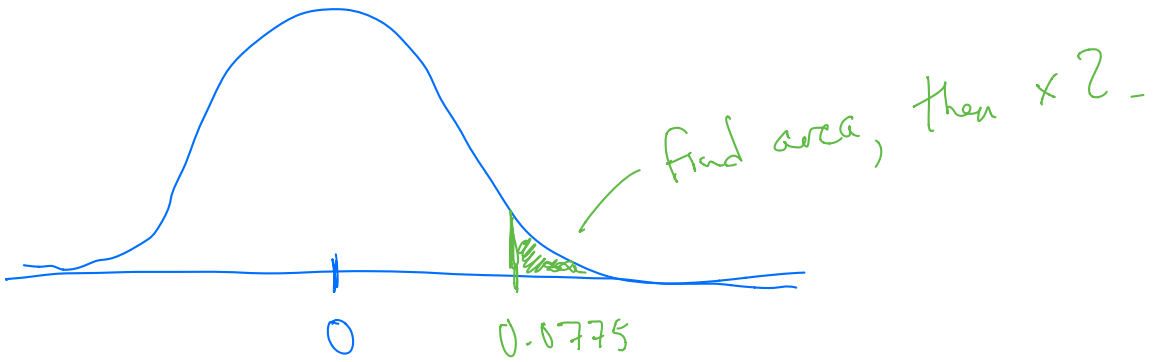
$H_0: \mu_1 - \mu_2 = 0$

Matched pairs

$H_0: \mu_{Diff} = 0$ $\quad H_a: \mu_{Diff} \neq 0$

Wetsuit data $\quad \bar{x}_{Diff} = 0.0775$ (observed)

To get P-value, construct a randomization dist



Demos of randomization distributions

Thomas Scofield

March 12, 2021

Randomization for a single mean

The population parameter in question is the mean μ. Natural hypotheses:

$$\mathbf{H}_0\colon \mu = \mu_0 \text{ (some number)} \qquad \text{vs.} \qquad \mathbf{H}_a\colon \mu \neq \mu_0.$$

In the context of `BodyTemp` values in the **BodyTemp50** data frame (Lock5withR package), one null hypothesis we can test:

$$\mathbf{H}_0\colon \mu = 98.6 \text{ (some number)} \qquad \text{vs.} \qquad \mathbf{H}_a\colon \mu \neq 98.6.$$

Our sample mean

```
xbar = mean(~BodyTemp, data=BodyTemp50)
xbar
```

```
## [1] 98.26
```

We could use the command

```
mean(~BodyTemp, data=resample(BodyTemp50))
```

to get a bootstrap statistic, but this would have a distribution centered on $\overline{x} = 98.26$, not on $\mu_0 = 98.6$. It becomes a randomization statistic after we move it:

```
mean(~BodyTemp, data=resample(BodyTemp50)) - xbar + 98.6
```

```
## [1] 98.59
```

Repeating this many times gives us a collection of randomization statistics/means:

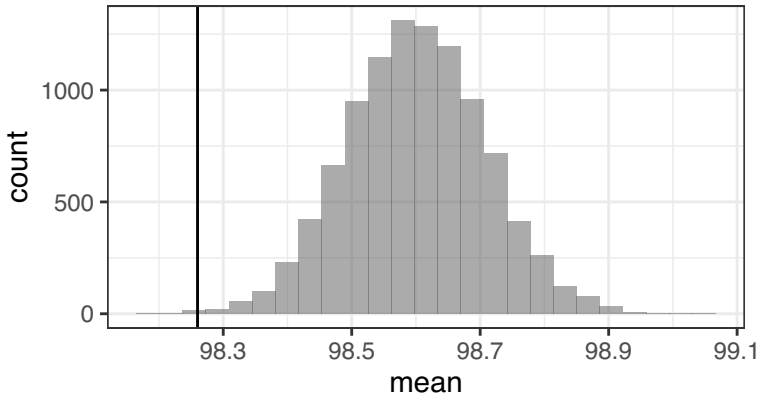
```
randomizedMeans <- do(10000) * mean(~BodyTemp, data=resample(BodyTemp50)) - xbar + 98.6
head(randomizedMeans)
```

```
##     mean
## 1 98.590
## 2 98.580
## 3 98.638
## 4 98.662
## 5 98.668
## 6 98.696
```

We can view the randomization distribution and verify it is centered in the correct place. Piping the result tto `gf_vline()` allows us to add a vertical line at the cutoff point corresponding to our original sample mean:

```
gf_histogram(~mean, data=randomizedMeans) %>% gf_vline(xintercept=xbar)
```

```
## Warning: geom_vline(): Ignoring `mapping` because `xintercept` was provided.
```



Instances as low as 98.26 are practically nonexistent, so we expect a small P-value. We compute an approximate P-value by counting those instances, after filtering down to just those randomization statistics in the left tail, by counting instances in the tail, dividing by 10000, then doubling the result.

```
2 * nrow( filter(randomizedMeans, mean <= 98.26) ) / 10000
```

```
## [1] 0.0014
```

An R interlude

Here are some incidental R commands and their results.

- In R it is sometimes helpful to have a list of numbers to work with. One way you can make a list is using the `c()` command.

```
x <- c(11, 38, 22, 14)
x
```

```
## [1] 11 38 22 14
```

In what follows, I will want to produce a random list of values all either $(+1)$ or (-1). To get a list of twelve such numbers:

```
posNegOne = c(-1, 1)
resample(posNegOne, size=12)
```

```
##  [1] -1  1 -1 -1  1  1 -1 -1 -1  1  1 -1
```

- If you multiply two lists of numbers and they have the same length, the result may be a little surprising.

```
list1 <- c(2, -3, 8)
list2 <- c(-3, -5, 2)
list1 * list2
```

```
## [1] -6 15 16
```

Put together, one can randomly change the sign of a list of numbers with a command like

```
originalList <- c(17, 31, -8, 22, 19)
randomSignChanger <- resample( c(-1,1), size=5 )
originalList * randomSignChanger
```

```
## [1]  17 -31   8  22 -19
```

Randomization for matched pairs (quantitative) data

As mentioned in class, when we have matched pairs data, every case contributes two values. In **Wetsuits**, the two values are `Wetsuit` and `NoWetsuit`. We want to subtract them and make a new column, `Difference`.

```
myData <- mutate(Wetsuits, Difference = Wetsuit - NoWetsuit)
myData
```

```
##    Wetsuit NoWetsuit Gender       Type    Sex Difference
## 1     1.57      1.49      F    swimmer Female       0.08
## 2     1.47      1.37      F triathlete Female       0.10
## 3     1.42      1.35      F    swimmer Female       0.07
## 4     1.35      1.27      F triathlete Female       0.08
## 5     1.22      1.12      M triathlete   Male       0.10
## 6     1.75      1.64      M    swimmer   Male       0.11
## 7     1.64      1.59      M    swimmer   Male       0.05
## 8     1.57      1.52      M triathlete   Male       0.05
## 9     1.56      1.50      M triathlete   Male       0.06
## 10    1.53      1.45      M triathlete   Male       0.08
## 11    1.49      1.44      M triathlete   Male       0.05
## 12    1.51      1.41      M triathlete   Male       0.10
```

Our hypotheses are

$$\mathbf{H}_0 : \mu_{\text{Diff}} = 0 \qquad \text{vs.} \qquad \mathbf{H}_a : \mu_{\text{Diff}} \neq 0.$$

The sample mean

```
xbarD <- mean(~Difference, data=myData)
xbarD
```

```
## [1] 0.0775
```

is $\overline{x}_{\text{Diff}} = 0.0775$.

In keeping with the null hypothesis (which suggest it is due to randomness, not bathing suits, that any difference exists between contributed values), we want to randomly change the sign of numbers in the `Difference` column before computing a randomization mean. One randomization statistic is obtained this way:

```
mean(~ (Difference * resample(c(-1,1), size=12)), data=myData)
```

```
## [1] -0.0175
```

We get many of them, then proceed as with other randomization distributions.

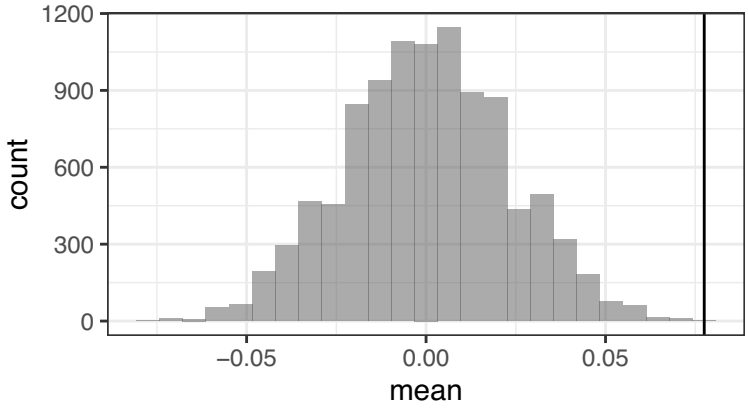
```
manyXbars <- do(10000) * mean(~(Difference*resample(c(-1,1), size=12)), data=myData)
head(manyXbars)
```

```
##            mean
## 1 -0.0291666667
## 2  0.0325000000
## 3  0.0025000000
## 4  0.0008333333
## 5  0.0058333333
## 6  0.0041666667
```

Here's a histogram of the randomization distribution, with another vertical line, this time out in the right tail, at the test statistic $\overline{x}_{\text{Diff}}$.

```
gf_histogram(~mean, data=manyXbars) %>% gf_vline(xintercept = xbarD)
```

```
## Warning: geom_vline(): Ignoring `mapping` because `xintercept` was provided.
```



We expect another very small P-value, and apply the same filtering method to obtain an approximation to it.

```
2 * nrow( filter(manyXbars, mean >= xbarD) ) / 10000
```

```
## [1] 4e-04
```