Stat 145, Thu 11-Mar-2021 -- Thu 11-Mar-2021

Biostatistics

Spring 2021

Thursday, March 11th 2021

Due:: PS06 due at 11 pm

Thursday, March 11th 2021

Wk 6, Th

Topic:: Conclusions of hypothesis tests

Read:: Lock5 4.3

$H_0: \mu_1 - \mu_2 = 0$

$H_a: \mu_1 - \mu_2 \boxed{\neq} 0$

$>$

$<$

$H_0: \mu = 24$

$H_a: \mu < 24$

Review P-value

- meaning
- always between 0 and 1
- calculated as a proportion of values in
  - two tail = twice what it is in one tail, use if alt hyp is "not equal"
  - one tail, use if alt hyp is "greater than" or "less than"
- smaller it is, the stronger the evidence in favor of alternative hypothesis
- Acceptable and unacceptable language

New concepts:

- significance level
  - provides a framework for a formal decision rule
  - denoted as alpha
  - common choices
  - serves as threshold for "reasonable doubt"

- Type I and Type II error
  - meaning
  - Probability of committing Type I error is tied to alpha

When P-value is small (statistically significant)

Can reject H_0 in favor of the alternative hyp.

(acceptable)

But, when P-value is not small enough (not statistically sign.)

unacceptable: We accept the null hypothesis,

acceptable: ① We fail to reject null hypothesis

② The evidence is consistent with a true null hyp.

Significance level α

common choices: $\alpha = 0.1$

$\alpha = 0.05$

$\alpha = 0.01$

Formally decide to reject H_0 when P-value is $< \alpha$.

Test of hypotheses

your conclusion based on data

		reject H_0	fail to reject H_0
reality	H_0 true	Type I error	OK
	H_0 false	OK	Type II error

By setting α, we have determined the probability of a Type I error.

Some scenarios and types of hypotheses we might propose in them

1. **Univariate binary categorical data**

   Hypotheses: $\mathbf{H}_0: \ p = p_0$ (some number) vs. $\mathbf{H}_a: \ p \neq p_0$ (or 1-sided version)

   Common test statistics: X, the *count* of successes; $\widehat{p} = X/n$, the proportion of successes

   Useful R commands:

   For generating a single randomization statistic: `rflip()`,

   ```
   rflip(50, prob=0.3)      # simulates 50 draws (called H or T) when p=0.3
   ```

   For generating a randomization distribution:

   ```
   manyRes <- do(5000) * rflip(50, prob=0.3)     # 5000 instances of a randomization stat
   ```

   For determining a P-value (after the command above):

   ```
   2*nrow( filter( manyRes, columnName >= testStatistic) ) / 5000   # tailor this to needs
   ```

   cutoff value = $\hat{p}$ (orig. sample proportion)

2. **Quantitative response variable, two groups, independent samples**

   Hypotheses: $\mathbf{H}_0: \ \mu_1 - \mu_2 = 0$ vs. $\mathbf{H}_a: \ \mu_1 - \mu_2 \neq 0$ (or 1-sided version)

   Only reasonable test statistic: $\bar{x}_1 - \bar{x}_2$

   The layout of the data should be similar to that in **CaffeineTaps**, with one column/variable (explanatory) indicating the group for cases, and another indicating the (quantitative) response.

   ```
   head(sample(CaffeineTaps)[,1:2])

      Taps      Group
   16  248 NoCaffeine
   8   248   Caffeine
   5   248   Caffeine
   20  242 NoCaffeine
   6   250   Caffeine
   18  244 NoCaffeine
   ```

   Useful R commands:

   For generating the test statistic in original sample: `diffmean()`

   ```
   diffmean(responseVar ~ grpVariable, data=dataFrame)
   ```

   For generating a single randomization statistic: `shuffle()`

   ```
   diffmean(responseVar ~ shuffle(grpVariable), data=dataFrame)
   ```

   For generating a randomization distribution:

```
manyDiffs <- do(5000)*diffmean(responseVar~shuffle(grpVariable),data=dataFrame)
```

For determining a P-value (after the command above):

```
2*nrow( filter( manyDiffs, diffmean >= testStat ) ) / 5000   # tailor to needs
```

3. **Bivariate quantitative data, scatter plot displays linear pattern**

   Hypotheses (two options):

   $\mathbf{H}_0: \ \rho = 0$ vs. $\mathbf{H}_a: \ \rho \neq 0$ (or 1-sided version)
   $\mathbf{H}_0: \ \beta = 0$ vs. $\mathbf{H}_a: \ \beta \neq 0$ (or 1-sided version)

   test statistics: r (for correlation hypotheses), b (for slope hypotheses)

   Useful R commands (directed at correlation case):

   For generating the test statistic in original sample:

   ```
   cor(responseVar ~ explanatoryVar, data=dataFrame)
   ```

   For generating a single randomization statistic: `shuffle()`

   ```
   cor(responseVar ~ shuffle(explanatoryVar), data=dataFrame)
   ```

   For generating a randomization distribution:

   ```
   manyCors <- do(5000)*cor(responseVar ~ shuffle(explanatoryVar), data=dataFrame)
   ```

   For determining a P-value (after the command above):

   ```
   2*nrow( filter( manyCors, diffmean >= testStat ) ) / 5000   # tailor to needs
   ```

4. **Univariate quantitative data**

   Hypotheses: $\mathbf{H}_0: \ \mu = \mu_0$ (some number) vs. $\mathbf{H}_a: \ \mu \neq \mu_0$ (or 1-sided version)

   test statistics: $\overline{x}$

   Useful R commands (directed at correlation case):

   For generating the test statistic in original sample:

   ```
   original_xbar <- mean(variableInFocus, data=dataFrame)
   ```

   For generating a single randomization statistic: `resample()`

   ```
   mu0 <- valueFromNullHypothesis
   mean(variableInFocus, data=resample(dataFrame)) + mu0 - original_xbar
   ```

   For generating a randomization distribution:

   ```
   manyRecenteredMeans <- do(5000)*mean(variableInFocus, data=resample(dataFrame)) +
       mu0 - original_xbar
   ```

   For determining a P-value (after the command above):

```
2*nrow( filter( manyRecenteredMeans, mean >= original_xbar ) ) / 5000   # tailor to needs
```

5. **Quantitative response variable, two groups, paired samples**

   Hypotheses: $\mathbf{H}_0: \ \mu_{\text{Diff}} = 0$ vs. $\mathbf{H}_a: \ \mu_{\text{Diff}} \neq 0$ (or 1-sided version)

   Test statistic: $\bar{x}_{\text{Diff}}$

   The layout of the data should be similar to that in **Wetsuits**, where there are two response columns (quantitative) per case, one column representing one sample, and the other representing a second under (presumably) different conditions.

   ```
   head(sample(Wetsuits))

      Wetsuit NoWetsuit Gender       Type orig.id
   5     1.22      1.12      M triathlete       5
   4     1.35      1.27      F triathlete       4
   10    1.53      1.45      M triathlete      10
   2     1.47      1.37      F triathlete       2
   1     1.57      1.49      F    swimmer       1
   8     1.57      1.52      M triathlete       8
   ```

   Useful R commands:

   For generating the test statistic from original sample: `diffmean()`

   ```
   myData <- mutate(dataFrame, diffs = qVar1 - qVar2)
   mean(~diffs, data=myData)
   ```

   For generating a single randomization statistic: `shuffle()`

   ```
   mean(~diffs, data=resample(myData))
   ```

   For generating a randomization distribution:

   ```
   manyDiffs <- do(5000) * mean(~diffs, data=resample(myData))
   ```

   For determining a P-value (after the command above):

   ```
   2*nrow( filter( manyDiffs, mean >= testStat ) ) / 5000   # tailor to needs
   ```

6. **Bivariate binary categorical variables**, where one variable acts as a group identifier (explanatory), and the other is the response

   Hypotheses: $\mathbf{H}_0: \ p_1 - p_2 = 0$ vs. $\mathbf{H}_a: \ p_1 - p_2 \neq 0$ (or 1-sided version)

   Test statistic: $\widehat{p_1} - \widehat{p_2}$

   This is often the most complicated case to carry out in R, as you may only be given summary data, and have to make your own raw data set first. In the example below, I use the context of Examples 4.28–4.29 on p. 268. The data for that example comes from Table 4.9, but to make both variables binary we ignore all the "Desipramine" cases. You may well decide to use StatKey, and avoid RStudio, for these scenarios.

   Useful R commands:

For preparing a data frame: `data.frame()`, `rbind()`

```
cokeAddiction <- rbind(
  do(18) * data.frame(group = "lithium", relapse = "yes", stringsAsFactors = TRUE),
  do(6) * data.frame(group = "lithium", relapse = "no", stringsAsFactors = TRUE),
  do(20) * data.frame(group = "placebo", relapse = "yes", stringsAsFactors = TRUE),
  do(4) * data.frame(group = "placebo", relapse = "no", stringsAsFactors = TRUE)
)
head(sample(cokeAddiction))
```

```
     group relapse .row .index orig.id
45 placebo      no    1      1      45
33 placebo     yes    1      9      33
6  lithium     yes    1      6       6
42 placebo     yes    1     18      42
38 placebo     yes    1     14      38
20 lithium      no    1      2      20
```

For generating the test statistic from original sample: `diffprop()`

```
diffprop(relapse ~ group, data=cokeAddiction)
```

For generating a single randomization statistic: `shuffle()`

```
diffprop(relapse ~ shuffle(group), data=cokeAddiction)
```

For generating a randomization distribution:

```
manyDiffs <- do(5000) * diffprop(relapse ~ shuffle(group), data=cokeAddiction)
```

For determining a *P*-value (after the command above):

```
2*nrow( filter( manyDiffs, diffprop >= testStat ) ) / 5000   # tailor to needs
```