Statistics

The R Bootcamp
Twitter: @therbootcamp

September 2017

Statistics

In this tutorial we will cover

- How to calculate basic descriptive statistics
  - `mean()`, `median()`, `sd()`, ...
- How to conduct hypothesis tests and how to work with `htest` objects
  - `t.test()`, `cor.test()`, `aov()`, ...

Examples

```r
# mean weight
mean(ChickWeight$weight)

# Standard deviation of Time
sd(ChickWeight$Time)

# T-test comparing weights from Diets 1 and 2
t.test(formula = weight ~ Diet,
       data = Chickweight,
       subset = Diet %in% c(1, 2))

# Correlation test between weight and Time
cor.test(formula = ~ weight + Time,
         data = ChickWeight)
```

Two types of statistics: Descriptive and Inferential

Descriptive

- Also called *sample statistics*
- Used to describe general characteristics of a sample
- Descriptive statistics typically a single scaler value

Examples

Statistic	R Function
Mean	`mean(x)`
Median	`median(x)`
Mode	`mode(x)`
Standard Deviation	`sd(x)`

R implimentation

```
sd(c(5, 3, 6, 3, 2, 6))  # Standard deviation
```

```
## [1] 1.722
```

```
mean(ChickWeight$weight) # Mean weight
```

```
## [1] 121.8
```

```
median(ChickWeight$Time) # Mean Time
```

```
## [1] 10
```

Two types of statistics: Descriptive and Inferential

Inferential

- Used to make inferences about a larger population. Typically done in tandem with a *hypothesis test*

Examples

Hypothesis Test	R Function
T-test	`t.test()`
Correlation Test	`cor.test()`
Chi-Square Test	`chisq.test()`
ANOVA, Post-hoc	`aov(), TukeyHSD()`

- Hypothesis tests typically return lists of outputs (e.g.; p-value, test statistic)

R implimentation

```r
t.test(x = c(4, 3, 6, 5, 3, 2),
       mu = 0,
       alternative = "two.sided")
```

```
##
##     One Sample t-test
##
## data:  c(4, 3, 6, 5, 3, 2)
## t = 6.4, df = 5, p-value = 0.001
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  2.289 5.378
## sample estimates:
## mean of x
##     3.833
```

Inferential Statistics

Different tests, different arguments

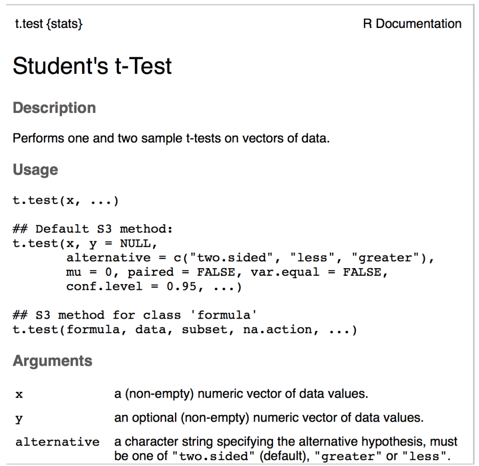
- A one-sample t-test requires just a vector, while an ANOVA requires more arguments.
- To see what arguments a test needs, consult the help menu (e.g.; `?t.test`)

Examples

Hypothesis Test	Help code
T-test	`?t.test()`
Correlation Test	`?cor.test()`
Chi-Square Test	`?chisq.test()`
ANOVA	`?aov()`

Always check help menus!

```
?t.test
```


t.test {stats} R Documentation

Student's t-Test

Description

Performs one and two sample t-tests on vectors of data.

Usage

```
t.test(x, ...)

## Default S3 method:
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)

## S3 method for class 'formula'
t.test(formula, data, subset, na.action, ...)
```

Arguments

`x`	a (non-empty) numeric vector of data values.
`y`	an optional (non-empty) numeric vector of data values.
`alternative`	a character string specifying the alternative hypothesis, must be one of `"two.sided"` (default), `"greater"` or `"less"`.

Inferential Statistics

Arguments to hypothesis tests

- Some arguments are manditory, and some are optional.
  - If you don't specify an optional argument, R will use a *default* value

Ex) Arguments to `t.test`

Argument	Description	Default
`x`, `formula`, `data`	Vector OR a formula and data	*Required*
`mu`	Null hypothesis	0
`alternative`	Alternative Hyp	`"two.sided"`

Specifying arguments to a hypothesis test

```
# 0: Won't work!
t.test()

# 1: Will work and use default arguments
t.test(x = ChickWeight$weight)

# 1b: Same as above
t.test(x = ChickWeight$weight,
       mu = 0,
       alternative = "two.sided")

# 2: Specified arguments
t.test(x = ChickWeight$weight,
       mu = 120,
       alternative = "greater")
```

Inferential Statistics

Formula

- Many tests allow you to include a `formula` argument

> `formula = y ~ a + b + ...`

Means…

> Model a dependent variable `y` as a function of `a` and `b` and `...`

- Formulas go together with dataframes `data` containing all variables in the formula, and optional `subset` arguments to specify which cases in `data` to include.

General structure of a hypothesis test and formula

```
my.test(formula = y ~ a + b, # Formula
        data = my.data,      # Dataframe
        ...                  # Additional
        )
```

- `y` is the dependent variable (e.g.; age), `a` and `b` are independent variables
- `data` is a dataframe containing the variables in `formula`; (`y`, `a`, `b`)
- `...` additional arguments specific to test

Inferential Statistics

Assigning hypothesis test objects

- Most hypothesis tests return an object of class `"htest"` which contain many values
- You can assign the results of a hypothesis test to an object, and then extract the info you want with the `$` operator:

Examples of what's in htest objects

Element	Result
`x$statistic`	A test statistic
`x$parameter`	Degrees of freedom
`x$p.value`	The p-value
`x$conf.int`	Confidence interval

What's in an htest object?

```
# One-sample t-test

weight.tt <- t.test(x = ChickWeight$weight,
                    mu = 120,
                    alternative = "two.sided")

class(weight.tt)
```

```
## [1] "htest"
```

```
# What's in the weight.tt object?
names(weight.tt)
```

```
## [1] "statistic"   "parameter"   "p.value"     "conf.i
## [8] "method"      "data.name"
```

Examples with ChickWeight Data

```
ChickWeight
```

```
##   weight Time Chick Diet
## 1     59    4    30    2
## 2     93    8    26    2
## 3     79    6    40    3
## 4    145   12    28    2
## 5     48    4     5    1
## 6    148   18    22    2
```

t-tests with `t.test()`

ChickWeight data

```
ChickWeight
```

```
##   weight Time Chick Diet
## 1     59    4    30    2
## 2     93    8    26    2
## 3     79    6    40    3
## 4    145   12    28    2
## 5     48    4     5    1
## 6    148   18    22    2
```

One sample t-test

> Is the mean weight of the chicks significantly different from 120?

```
t.test(x = ChickWeight$weight,        # Vector of values
       alternative = "two.sided",     # Two sided test
       mu = 120)                      # Null is 120
```

```
##
##     One Sample t-test
##
## data:  ChickWeight$weight
## t = 0.62, df = 580, p-value = 0.5
## alternative hypothesis: true mean is not equal to 120
## 95 percent confidence interval:
##  116.0 127.6
## sample estimates:
## mean of x
##     121.8
```

t-tests with `t.test()`

ChickWeight data

```
ChickWeight
```

```
##   weight Time Chick Diet
## 1     59    4    30    2
## 2     93    8    26    2
## 3     79    6    40    3
## 4    145   12    28    2
## 5     48    4     5    1
## 6    148   18    22    2
```

Two sample t-test

> Is the mean weight of the chicks on Diet 1 different from Diet 2?

```
t.test(formula = weight ~ Diet,     # Formula
       data = ChickWeight,          # Data in Chickweight
       subset = Diet %in% c(1, 2)) # Only Diets 1,2
```

```
## 
##     Welch Two Sample t-test
## 
## data:  weight by Diet
## t = -2.6, df = 200, p-value = 0.009
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -34.900  -5.042
## sample estimates:
## mean in group 1 mean in group 2
##           102.6           122.6
```

Correlation test with `cor.test()`

ChickWeight data

```
ChickWeight
```

```
##   weight Time Chick Diet
## 1     59    4    30    2
## 2     93    8    26    2
## 3     79    6    40    3
## 4    145   12    28    2
## 5     48    4     5    1
## 6    148   18    22    2
```

Correlation Test

> Is there a correlation between weight and Time?

- For `cor.test()`, formula looks like `formula = ~ a + b`

```
cor.test(formula = ~ weight + Time, # Formula
         data = ChickWeight)          # Data in Chickweigh
```

```
##
##     Pearson's product-moment correlation
##
## data:  weight and Time
## t = 37, df = 580, p-value <2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8109 0.8599
## sample estimates:
##    cor
## 0.8371
```

Chi-Square test with `chisq.test()`

ChickWeight data

```
ChickWeight
```

```
##   weight Time Chick Diet
## 1     59    4    30    2
## 2     93    8    26    2
## 3     79    6    40    3
## 4    145   12    28    2
## 5     48    4     5    1
## 6    148   18    22    2
```

Chi-Square test

> Are there more observations from one Diet than another?

- For `chisq.test()`, main argument should be a table of values created from the `table()` function:

```
chisq.test(x = table(ChickWeight$Diet))
```

```
##
##      Chi-squared test for given probabilities
##
## data:  table(ChickWeight$Diet)
## X-squared = 53, df = 3, p-value = 2e-11
```

ANOVA with `aov()`

ChickWeight data

```
ChickWeight
```

```
##   weight Time Chick Diet
## 1     59    4    30    2
## 2     93    8    26    2
## 3     79    6    40    3
## 4    145   12    28    2
## 5     48    4     5    1
## 6    148   18    22    2
```

ANOVA

> Is there difference in weights based on Diet?

- Applying `summary()` to an `aov` object prints a nice table.

```
E <- aov(formula = weight ~ Diet, # Formula
         data = ChickWeight)      # Data in Chickweight

summary(E)  # Sow a summary of the results
```

```
##              Df  Sum Sq Mean Sq F value  Pr(>F)
## Diet          3  155863   51954    10.8 6.4e-07 ***
## Residuals   574 2758693    4806
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Post-hoc tests with `TukeyHSD()`

> Which specific pairs of Diets differed?

Step 1: Create aov object

- Apply `TukeyHSD()` to an `aov` object to get post-hoc tests.

```
# Create an aov object called D

D <- aov(formula = weight ~ Diet,
         data = ChickWeight)
```

Step 2: Apply `TukeyHSD()` to object

```
TukeyHSD(D)  # Conduct post-hoc tests
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = weight ~ Diet, data = ChickWeight)
##
## $Diet
##       diff      lwr   upr  p adj
## 2-1 19.971  -0.2998 40.24 0.0552
## 3-1 40.305  20.0335 60.58 0.0000
## 4-1 32.617  12.2354 53.00 0.0003
## 3-2 20.333  -2.7268 43.39 0.1058
## 4-2 12.646 -10.5116 35.80 0.4954
## 4-3 -7.687 -30.8450 15.47 0.8278
```

Final notes

- When using a hypothesis test, always ask:

> What are the arguments?
>
> What format or class should the arguments be?

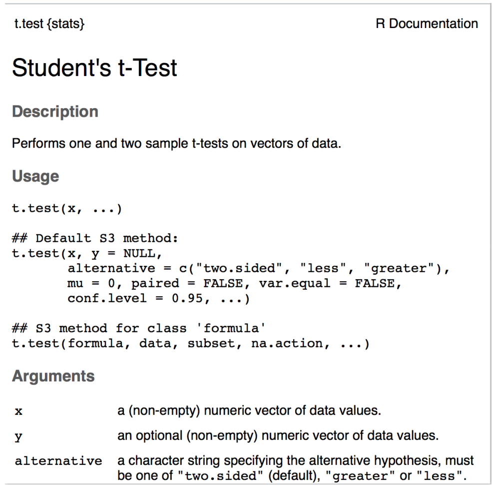
- When in doubt, always look at the help files and examples at the end.
- Save hypothesis tests as new objects, then apply `names()` to see what elements it contains, then extract what you want with `$`

```r
# Run test and save as test_A
test_A <- t.test(formula = weight ~ Diet,
                 data = ChickWeight,
                 subset = Diet %in% c(1, 2))

names(test_A)  # What is in the object?

test_A$statistic # Ah ok! Show me the test statistic
```

`?t.test`


t.test {stats} R Documentation

Student's t-Test

Description

Performs one and two sample t-tests on vectors of data.

Usage

```r
t.test(x, ...)

## Default S3 method:
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)

## S3 method for class 'formula'
t.test(formula, data, subset, na.action, ...)
```

Arguments

`x`	a (non-empty) numeric vector of data values.
`y`	an optional (non-empty) numeric vector of data values.
`alternative`	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less".

Questions?

Statistics Pratical

Link to Statistics practical