Simple Statistical Data Analysis

Mark Andrews

April 5, 2017

Introduction

With extremely rare exceptions, any statistical method that you might like to use for data analysis is probably already available in R. Here, we will start with the basics.

T-tests

```
set.seed(101)
x <- rnorm(10)
y <- rnorm(12)

t.test(x, y)
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = 1.7025, df = 18.512, p-value = 0.1054
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1310695  1.2631469
## sample estimates:
##  mean of x  mean of y
##  0.2450401 -0.3209986
```

If we want to assume that the variance of the two groups are equal, then specify this as follows:

```
M <- t.test(x, y, var.equal=T)
```

From this, the object M contains all the properties of the t-test results. For example,

```
M$statistic # The test statistic
```

```
##        t
## 1.629947
```

```
M$parameter # The degrees of freedom
```

```
## df
## 20
```

```
M$p.value # The p-value
```

```
## [1] 0.1187618
```

A paired sample t-test

```
set.seed(102)
N <- 10
x <- rnorm(N)
y <- rnorm(N)
(M <- t.test(x, y, paired = T))
```

```
##
##  Paired t-test
##
## data:  x and y
## t = 1.8357, df = 9, p-value = 0.0996
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2190858  2.1049655
## sample estimates:
## mean of the differences
##               0.9429398
```

A one sample t-test

```
set.seed(103)
x <- rnorm(10)
(M <- t.test(x))
```

```
##
##  One Sample t-test
##
## data:  x
## t = -1.1976, df = 9, p-value = 0.2616
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -1.0131208  0.3117197
## sample estimates:
##  mean of x
## -0.3507005
```

Some non-parametric tests

Non-parametric counterparts of the independent samples and the paired samples t-tests are the Mann-Whitney U test and the Wilcoxon signed ranks tests.

This is the Mann Whitney U test:

```
set.seed(101)

x <- rnorm(10)
y <- rnorm(12)

wilcox.test(x, y) # Yes, I know it is not called Mann Whitney
```

```
##
##  Wilcoxon rank sum test
##
## data:  x and y
## W = 84, p-value = 0.1229
## alternative hypothesis: true location shift is not equal to 0
```

This is the Wilcoxon signed ranks test:

```r
set.seed(102)
N <- 10
x <- rnorm(N)
y <- rnorm(N)
wilcox.test(x, y, paired = TRUE)
```

```
##
##  Wilcoxon signed rank test
##
## data:  x and y
## V = 43, p-value = 0.1309
## alternative hypothesis: true location shift is not equal to 0
```

Pearson's χ^2 test

For this, we will use the *Titanic*[1] data set.

```r
data("Titanic") # load it up
```

This is a four dimensional table of frequencies:

```r
dimnames(Titanic)
```

```
## $Class
## [1] "1st"  "2nd"  "3rd"  "Crew"
##
## $Sex
## [1] "Male"   "Female"
##
## $Age
## [1] "Child" "Adult"
##
## $Survived
## [1] "No"  "Yes"
```

We'll concatenate by 'Sex' and 'Survived' to make a 2 by 2 table to use as our observed frequencies:

```r
(observed <- apply(Titanic, c('Sex', 'Survived'), sum))
```

```
##         Survived
## Sex        No Yes
##   Male   1364 367
##   Female  126 344
```

To do the χ^2 test, it is simply

[1] *Titanic* is a ship named after a famous movie from the 1990's.

```r
(M <- chisq.test(observed))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  observed
## X-squared = 454.5, df = 1, p-value < 2.2e-16
```

As before, we can access properties of the test, e.g.

```r
M$expected
```

```
##         Survived
## Sex              No      Yes
##   Male   1171.8264 559.1736
##   Female  318.1736 151.8264
```

Correlations

```r
set.seed(104)
N <- 20

x <- rnorm(N)
y <- rnorm(N)
```

To do a good old Pearson's product moment correlation:

```r
cor.test(x, y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  x and y
## t = 0.30043, df = 18, p-value = 0.7673
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3838846  0.4976024
## sample estimates:
##       cor
## 0.0706355
```

And a good old Spearman's ρ:

```r
cor.test(x, y, method='spearman')
```

```
##
##  Spearman's rank correlation rho
##
## data:  x and y
## S = 1152, p-value = 0.5725
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.1338346
```