Planned comparisons using contrasts in R

January 31, 2022

1 Planned comparisons using contrasts in R

[1]:
```
# Load needed packages
library(tidyverse)
library(ggplot2)
```

```
  Attaching packages                                tidyverse
1.3.0

  ggplot2 3.3.0       purrr   0.3.4
  tibble  3.0.1       dplyr   0.8.5
  tidyr   1.0.2       stringr 1.4.0
  readr   1.3.1       forcats 0.5.0

  Conflicts
tidyverse_conflicts()
  dplyr::filter() masks stats::filter()
  dplyr::lag()    masks stats::lag()
```

This module uses data simulated to match means and SDs of data analyzed in A. Parenti, L. Guerrini, P. Masella, S. Spinelli, L. Calamai, P. Spugnoli (2014). "Comparison of Espresso Coffee Brewing Techniques," *Journal of Food Engineering*, Vol. 121, pp. 112-117.

The goal of this experiment in Parenti et al. is to "evaluate and compare the differences in terms of quality between espresso coffee made using three different extraction procedures." Quality, in this exercise, is measured in terms of the "foam index". Foam, according to the authors of this study, is a "distinctive feature of espresso coffee, as it is absent in other coffee brews and is required for consumer acceptance."

More compactly, here are the variables in our experimental data set:

1. foamIndx: The foam index is defined as the ratio between the foam and liquid volume (vol vol $^{-1}$%) measured 30 seconds after extraction

2. method: Three methods were studied: a. Method 1=Bar Machine(BM); b. Method 2=Hyper-Espresso Method (HIP); and c. Method 3= I-Espresso System (IT).

Suppose that, in our espresso study, we pre-planned to study the hypothesis test:

$$H_0 : \mu_2 = \frac{1}{2}(\mu_1 + \mu_3)$$
$$H_1 : \mu_2 > \frac{1}{2}(\mu_1 + \mu_3)$$

[2]:
```
# Load the data
esp = read.csv("espresso1.txt", sep="\t")

esp$method = as_factor(esp$method)
esp$method = recode(esp$method, "1" = "Bar Machine", "2" = "Hyper-Espresso␣
↪Method", "3" = "I-Espresso System")

summary(esp)

lmod = lm(foamIndx ~ method, data = esp)
summary(lmod)
```

```
    foamIndx                         method
 Min.   :21.02   Bar Machine           :9
 1st Qu.:35.66   Hyper-Espresso Method:9
 Median :38.52   I-Espresso System     :9
 Mean   :44.47
 3rd Qu.:55.23
 Max.   :73.19


Call:
lm(formula = foamIndx ~ method, data = esp)

Residuals:
   Min     1Q Median     3Q    Max
-14.62  -6.60   0.41   5.73  16.49

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                   32.400      2.819  11.492 3.04e-11 ***
methodHyper-Espresso Method   28.900      3.987   7.248 1.73e-07 ***
methodI-Espresso System        7.300      3.987   1.831   0.0796 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.458 on 24 degrees of freedom
Multiple R-squared:  0.7031,Adjusted R-squared:  0.6783
F-statistic: 28.41 on 2 and 24 DF,  p-value: 4.699e-07
```

Note that:

1. The test was pre-planned.
2. The full F-test is significant (p-value: 4.7×10^{-7})
3. & 4. There is only one contrast, so we've met the conditions of orthogonality.

Let's first "hard code" our contrast hypothesis test. Then, we'll use an R package to perform the test automatically.

[5]:
```
c = c(-0.5, 1, -0.5) #constants in the contrast specified above

b = coef(lmod); #ANOVA regression model coefficients
n =  length(resid(lmod)) #total number of espressos brewed
n_method = with(esp, c(length(foamIndx[method == "Bar Machine"]),
            length(foamIndx[method == "Hyper-Espresso Method"]),
            length(foamIndx[method == "I-Espresso System"])))
            #vector with number of espressos brewed by each method
J = length(unique(esp$method)); J

ybar = c(b[1], b[1] + b[2],b[1] + b[3]); #vector of sample means of foam index␣
 ↪for each method
ybar[2] - 0.5*(ybar[1] + ybar[3])
rss = sum(resid(lmod)^2); #residual sum of squares
sighat = sqrt(rss/(n-J)); #estimate of sigma^2 hat
gammahat = ybar[2] - 0.5*(ybar[1] + ybar[3]);
cat("The estimate of the contrast is is", as.numeric(gammahat))
se = (sqrt(sighat^2*(1/(4*n_method[1]) + 1/(n_method[2]) + 1/(4*n_method[3]))))␣
 ↪#standard error of gamma hat
z = gammahat/se # test statistic
pval = 1-pnorm(z) #p-value for upper tailed test

cat(". The test statistic is ", as.numeric(z), ". The p value for the test is",␣
 ↪pval,".")
```

3

(Intercept): 25.25

```
The estimate of the contrast is is 25.25. The test statistic is  7.312531 . The
p value for the test is 1.311173e-13 .
```

Luckily, R has a function that will compute contrasts for us. This isn't a substitution for understanding contrasts, but it is helpful in saving time! The `glht()` function will allow us to conduct "general linear hypotheses and multiple comparisons for parametric models, including [ANOVA] generalized linear models, linear mixed effects models, and survival models."

[6]:
```
#install.packages("multcomp") #if we need to install the multcomp package
library(multcomp) #load the multicomp package
```

```
contrast = glht(lmod, linfct = mcp(method = c(-0.5, 1, -0.5))) #glht =
↪generalized linear hypothesis test
summary(contrast)
```

```
 Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: User-defined Contrasts


Fit: lm(formula = foamIndx ~ method, data = esp)

Linear Hypotheses:
       Estimate Std. Error t value Pr(>|t|)
1 == 0   25.250      3.453   7.313 1.49e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- single-step method)
```

[ ]:

```

```