

Contrasts and ANOVA

Neil J. Hatfield

3/15/2021

In this tutorial, we are going to explore setting up and testing Contrasts in the context of ANOVA models. So that we have some data to work with, we will make use of Table 5.2/Example 5.5 from page 91 of Oehlert, **Free Amino Acids in Cheese**. Take a moment to create this data frame for yourself. (Check your code against what appears in the Code Appendix.)

I will leave both the checking of base requirements as well as assumptions for both the parametric and nonparametric shortcuts to you. However, Table 1 provides the modern ANOVA table for our cheese data. (Reminder: don't forget to tell R to use the Sum to Zero constraint: `options(contrasts = c("contr.sum", "contr.poly"))`.)

Table 1: ANOVA Table for Free Amino Acids in Cheese Study

Source	SS	df	MS	F	p-value	Omega Sq.	Eta Sq.	Epsilon Sq.
strain	5.6279	3	1.8760	11.9318	0.0183	0.8039	0.8995	0.8241
Residuals	0.6289	4	0.1572					

Setting Up Contrasts

There are two parts to setting up your contrast in R: making sure you know the correct order of your factor levels and then saving the weights.

Check the Order

Checking the order of your factor levels is a quick application of the `levels` function:

```
# Checking the order of factor levels
levels(cheese$strain)
```

```
## [1] "A"      "A & B" "B"      "None"
```

The output tells us that our contrasts will need to be in the order (A, AB, B, None).

Save the Weights

Now that we know the order of the factor levels, we can create contrasts by saving their weights as a vector. For example,

```
# Save the contrasts weights as vectors
c1 <- c(1/3, 1/3, 1/3, -1)
c2 <- c(1/2, -1, 1/2, 0)
```

The first contrast, `c1`, pools the cheeses that received either Strain A or B together and compares them against those which received the base set of cultures ("None"). The second contrast, `c2`, compares the pooling

of Strain A and Strain B against the combination of both strains. By saving the contrasts, they are now available for our use.

Checking Contrasts

Before we leave this section, remember that for you contrast to be valid, the weights must add to 0. You can quickly check in R by using the `sum` function:

```
# Check that the weights add to zero
sum(c1)
```

```
## [1] -5.551115e-17
```

```
sum(c2)
```

```
## [1] 0
```

While the `c1` spit out a non-zero value, that value is essentially zero and happens to be the result of computer arithmetic. You will have to build your own intuition as to whether you're seeing something non-zero or the side-effects of computer arithmetic.

Checking Orthogonality

You can use R to calculate whether or not two contrasts are orthogonal. However, I'm going to leave deciding how to use R to do this to each of you. Think of this as an opportunity to push your understanding of coding in R. Remember, that you're working with the formula

$$\sum_{i=1}^g \frac{w_i \cdot w_i^*}{n_i}$$

Testing Contrasts

Once you have set up your contrasts, you have two routes to take for testing them: base packages or using the `DescTools` package.

Base Package Approach

Perhaps the easiest approach is to use the base packages of R. However, I must quickly caution that you **do not** get any Type I Error Rate control with this method. Thus, if you go down this route, you will need to manually apply your selected method. There is a four-step process that you'll need to do:

- 1) You need to take the contrasts that you saved and bind them to your factor.
 - Use `contrasts(dataFrame$factor)` then
 - Assign (`<-`) your contrast to this call.
 - If you have more than one contrast, wrap them all in `cbind`.
- 2) (Re-)Run the ANOVA F Test via the `aov` or `lm` calls.
 - Note: I typically run one `aov` call for checking residuals and then run a **new** call for applying my contrasts.
- 3) Save the classical ANOVA table to an object using the `summary.aov` call.
 - You will not be able to use the `parameters` package nor the regular `summary` call and get the contrasts to appear.
- 4) Make a professional looking table.

See the example code below.

```
# Base Packages Approach to ANOVA Contrasts
## Step 1--Bind contrasts to the factor
contrasts(cheese$strain) <- cbind(c1, c2)
```

```

## Step 2--Run the ANOVA omnibus
cheeseModelC <- aov(
  formula = acids ~ strain,
  data = cheese
)

## Step 3--Save the output table
conOut <- summary.aov(
  object = cheeseModelC,
  split = list( # You can give meaning to your constrats here
    strain = list(
      "A, A&B, B vs. Null" = 1,
      "A, B vs. A&B" = 2
    )
  )
)

## Step 4--Create a Professional looking table
knitr::kable(
  x = conOut[[1]], # Grab the summary.aov output
  digits = 4,
  col.names = c(
    "DF", "SS", "MS", "F", "p-value"),
  caption = "ANOVA Table for Free Amino Acids in Cheese Study",
  booktabs = TRUE,
  align = rep("c", 5)
) %>%
kableExtra::kable_styling(
  font_size = 12,
  latex_options = c("HOLD_position")
)

```

Table 2: ANOVA Table for Free Amino Acids in Cheese Study

	DF	SS	MS	F	p-value
strain	3	5.6279	1.8760	11.9318	0.0183
strain: A, A&B, B vs. Null	1	2.0434	2.0434	12.9969	0.0227
strain: A, B vs. A&B	1	2.8188	2.8188	17.9287	0.0133
Residuals	4	0.6289	0.1572		

Notice how the two contrasts are now listed as the second and third rows of our table. Keep in mind that their p -values are **not adjusted**. You will need to manually calculate \mathcal{E}_I^* using your chosen method to figure out what to use for each test's individual unusualness threshold.

DescTools Package

The `DescTools` package includes the `ScheffeTest` function that will allow you test your contrasts with the Scheffé method.

Unlike the base package approach, you will not get an ANOVA table, but rather, you'll get a table like what you would see when doing pairwise post hoc analysis. The process here has three steps:

- 1) Fit your ANOVA model just as you typically would.
 - You do not need to bind your contrasts to your factor beforehand.
- 2) Save the output of the `DescTools::ScheffeTest` to an object.
- 3) Make a professional looking table.

Examine the following example code:

```
# DescTools Scheffe Test Approach
## Step 1 -- Fit the ANOVA model as usual
cheeseModel <- aov(
  formula = acids ~ strain,
  data = cheese,
  na.action = "na.omit"
)

## Step 2--Save output of Scheffe Test
scheffeCheese <- DescTools::ScheffeTest(
  x = cheeseModel,
  contrasts = cbind(c1, c2),
  conf.level = 0.9, # 1 -- Your Overall Type I Error Rate
)

## Step 3--Make a Professional looking table
knitr::kable(
  x = scheffeCheese[[1]], # Grab the output
  digits = 4,
  col.names = c(
    "Difference", "Lower Bound", "Upper Bound", "p-value"),
  caption = "Scheffe Test Results",
  booktabs = TRUE,
  align = rep("c", 4)
) %>%
kableExtra::kable_styling(
  font_size = 12,
  latex_options = c("HOLD_position")
)
```

Table 3: Scheffe Test Results

	Difference	Lower Bound	Upper Bound	p-value
A,A & B,B-None	1.1672	0.0192	2.3151	0.0953
A,B-A & B	-1.4540	-2.6716	-0.2364	0.0585

We will not worry about effect sizes with contrasts (shocking, I know).

Code Appendix

```
# Setting Document Options
knitr::opts_chunk$set(
  echo = FALSE,
  warning = FALSE,
  message = FALSE,
  fig.align = "center"
)

packages <- c("tidyverse", "knitr", "kableExtra",
             "parameters", "DescTools")
lapply(packages, library, character.only = TRUE)

options(knitr.kable.NA = "")
options(contrasts = c("contr.sum", "contr.poly"))

source("https://raw.githubusercontent.com/neilhatfield/STAT461/master/rScripts/ANOVATools.R")
# Create the Cheese data frame
cheese <- data.frame(
  strain = as.factor(
    c("None", "None", "A", "A",
      "B", "B", "A & B", "A & B")
  ),
  acids = c(
    4.195, 4.175, 4.125, 4.735,
    4.865, 5.745, 6.155, 6.488
  )
)

# Create a modern ANOVA table to get started.
cheeseModel <- aov(
  formula = acids ~ strain,
  data = cheese,
  na.action = "na.omit"
)

parameters::model_parameters(
  model = cheeseModel,
  omega_squared = "raw",
  eta_squared = "raw",
  epsilon_squared = "raw"
) %>%
knitr::kable(
  digits = 4,
  col.names = c(
    "Source", "SS", "df", "MS", "F", "p-value",
    "Omega Sq.", "Eta Sq.", "Epsilon Sq."
  ),
  caption = "ANOVA Table for Free Amino Acids in Cheese Study",
  booktabs = TRUE,
  align = c("l", rep("c", 8))
) %>%
kableExtra::kable_styling(
  font_size = 10,
```

```

    latex_options = c("scale_down", "HOLD_position")
  )

# Checking the order of factor levels
levels(cheese$strain)

# Save the contrasts weights as vectors
c1 <- c(1/3, 1/3, 1/3, -1)
c2 <- c(1/2, -1, 1/2, 0)

# Check that the weights add to zero
sum(c1)
sum(c2)

# Base Packages Approach to ANOVA Contrasts
## Step 1--Bind contrasts to the factor
contrasts(cheese$strain) <- cbind(c1, c2)

## Step 2--Run the ANOVA omnibus
cheeseModelC <- aov(
  formula = acids ~ strain,
  data = cheese
)

## Step 3--Save the output table
conOut <- summary.aov(
  object = cheeseModelC,
  split = list( # You can give meaning to your constrats here
    strain = list(
      "A, A&B, B vs. Null" = 1,
      "A, B vs. A&B" = 2
    )
  )
)

## Step 4--Create a Professional looking table
knitr::kable(
  x = conOut[[1]], # Grab the summary.aov output
  digits = 4,
  col.names = c(
    "DF", "SS", "MS", "F", "p-value"),
  caption = "ANOVA Table for Free Amino Acids in Cheese Study",
  booktabs = TRUE,
  align = rep("c", 5)
) %>%
kableExtra::kable_styling(
  font_size = 12,
  latex_options = c("HOLD_position")
)

# DescTools Scheffe Test Approach
## Step 1 -- Fit the ANOVA model as usual
cheeseModel <- aov(

```

```

formula = acids ~ strain,
data = cheese,
na.action = "na.omit"
)

## Step 2--Save output of Scheffe Test
scheffeCheese <- DescTools::ScheffeTest(
  x = cheeseModel,
  contrasts = cbind(c1, c2),
  conf.level = 0.9, # 1 -- Your Overall Type I Error Rate
)

## Step 3--Make a Professional looking table
knitr::kable(
  x = scheffeCheese[[1]], # Grab the output
  digits = 4,
  col.names = c(
    "Difference", "Lower Bound", "Upper Bound", "p-value"),
  caption = "Scheffe Test Results",
  booktabs = TRUE,
  align = rep("c", 4)
) %>%
kableExtra::kable_styling(
  font_size = 12,
  latex_options = c("HOLD_position")
)

```