Lab 12: Pairwise comparisons

STAT218

The objective of this lab is to learn how to perform *post-hoc* inference in the analysis of variance context in R using emmeans. You’ll reproduce an end-to-end analysis and explore a few options along the way. Your goals are:

1. Review how to carry out a basic ANOVA
2. Learn how to compute point and interval estimates for group means
3. Learn how to construct intervals and carry out tests for pairwise comparisons
4. Compare multiple correction methods

## ANOVA with pairwise comparisons

We’ll use the longevity data from an experiment in which lifetimes in days of mice were measured after randomly allocating one of four levels of dietary intake restriction to each mouse. Load the necessary packages and dataset by executing the commands below.

# load packages  
library(tidyverse)  
library(emmeans)  
library(pander)  
  
# read in dataset and make variable names lowercase  
longevity <- read.csv('data/longevity.csv') |>   
 rename\_with(tolower)  
  
# convert grouping variable to a factor  
longevity$diet <- factor(longevity$diet,   
 levels = c('NP', 'N/N85',   
 'N/R50', 'N/R40'),  
 ordered = T)  
  
# preview  
head(longevity) |> pander()

| lifetime | diet |
| --- | --- |
| 35.5 | NP |
| 35.4 | NP |
| 34.9 | NP |
| 34.8 | NP |
| 33.8 | NP |
| 33.5 | NP |

You may notice that the grouping variable diet was converted to an ordered factor. This is a data type in R that encodes ordinal categorical variables. It’s not absolutely necessary to include the ordering, but it will make certain tasks more streamlined, such as ordering the groups in plots or making pairwise comparisons with the control level (NP).

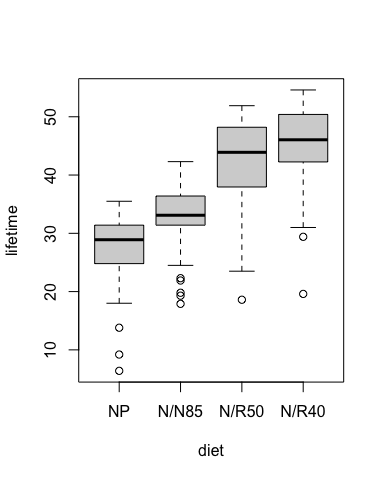
### Refresher: basic ANOVA

We’ll start by reviewing the steps of an analysis of variance:

1. Data visualization
2. Grouped summaries
3. Fitting an ANOVA model and testing for a difference in group means

These are outlined below.

# boxplot for visual group comparisons  
boxplot(lifetime ~ diet, data = longevity)



# grouped summaries  
longevity |>  
 group\_by(diet) |>   
 summarize(mean = mean(lifetime),  
 sd = sd(lifetime),  
 n = n(),  
 mean.se = sd/sqrt(n)) |>  
 pander()

| diet | mean | sd | n | mean.se |
| --- | --- | --- | --- | --- |
| NP | 27.4 | 6.134 | 49 | 0.8762 |
| N/N85 | 32.69 | 5.125 | 57 | 0.6789 |
| N/R50 | 42.3 | 7.768 | 71 | 0.9219 |
| N/R40 | 45.12 | 6.703 | 60 | 0.8654 |

# fit analysis of variance model  
fit <- aov(lifetime ~ diet, data = longevity)  
summary(fit) |> pander()

Analysis of Variance Model

|  | Df | Sum Sq | Mean Sq | F value | Pr(>F) |
| --- | --- | --- | --- | --- | --- |
| **diet** | 3 | 11426 | 3809 | 87.41 | 6.348e-38 |
| **Residuals** | 233 | 10152 | 43.57 | NA | NA |

The test indicates that the data provide strong evidence against the null hypothesis of no effect of diet on longevity. We now would like to know:

* estimated mean lifetimes for each diet
* which differences in mean lifetimes are significant
* estimated gains in lifetime for each level of dietary restriction relative to an unrestricted diet

We will use emmeans to perform these inferences.

### Model-based estimates

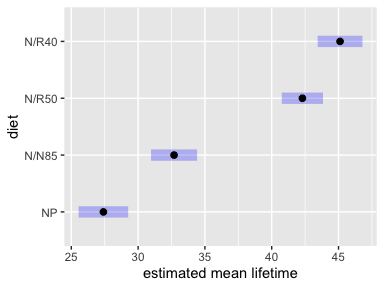
Estimates of the group means and the within-group standard deviation are computed when the model is fit. We can use these to compute confidence intervals for each group mean. Use the level = ... argument to change the confidence level.

# intervals for group means  
emmeans(fit, ~ diet) |>   
 confint(level = 0.95) |>  
 pander()

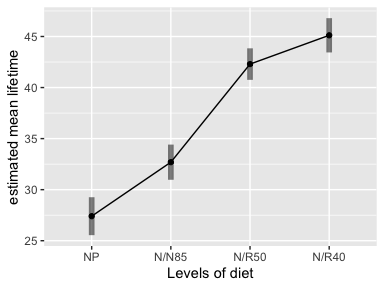
| diet | emmean | SE | df | lower.CL | upper.CL |
| --- | --- | --- | --- | --- | --- |
| NP | 27.4 | 0.943 | 233 | 25.54 | 29.26 |
| N/N85 | 32.69 | 0.8743 | 233 | 30.97 | 34.41 |
| N/R50 | 42.3 | 0.7834 | 233 | 40.75 | 43.84 |
| N/R40 | 45.12 | 0.8522 | 233 | 43.44 | 46.8 |

The emmeans package also has a plotting method available to visualize the ANOVA model estimates.

# basic plotting method  
emmeans(fit, ~ diet) |>   
 plot(level = 0.95, xlab = 'estimated mean lifetime')



# another option  
emmip(fit, ~ diet, CIs = T, level = 0.95, ylab = 'estimated mean lifetime')



|  |
| --- |
| Your turn |
| The Bonferroni correction is a way to adjust multiple inferences to control error accrual. In the context of intervals, this amounts to making them wider to achieve simultaneous coverage at a specific rate.  To implement the correction in a way that achieves simultaneous coverage for intervals, change the confidence level to where is the number of groups.  Adjust the table and plot above to achieve a simultaneous 95% coverage of all four means.  # intervals for group means emmeans(fit, ~ diet) |>   confint(level = 0.95) # change here  # plot emmeans(fit, ~ diet) |>   plot(level = 0.95, # change here  xlab = 'estimated mean lifetime') |

The Bonferroni correction can also be implemented by adding the argument adjust = 'bonferroni' to either confint() or plot(). If this approach is used, the level argument should specify the *simultaneous* coverage (not the individual coverage).

### Post-hoc inferences

#### Tests for means

While relatively uncommon, one could perform hypothesis tests for the group means on a post-hoc basis. For example, we might test the hypotheses:

This can be implemented using test() with the appropriate arguments, and corresponding one-sided intervals can be constructed with confint().

# test whether lifetimes exceed 30 days  
emmeans(fit, ~ diet) |>  
 test(null = 30, side = '>') |>  
 pander()

| diet | emmean | SE | df | null | t.ratio | p.value |
| --- | --- | --- | --- | --- | --- | --- |
| NP | 27.4 | 0.943 | 233 | 30 | -2.755 | 0.9968 |
| N/N85 | 32.69 | 0.8743 | 233 | 30 | 3.078 | 0.001166 |
| N/R50 | 42.3 | 0.7834 | 233 | 30 | 15.7 | 1.132e-38 |
| N/R40 | 45.12 | 0.8522 | 233 | 30 | 17.74 | 1.973e-45 |

# one-sided intervals  
emmeans(fit, ~ diet) |>  
 confint(level = 0.95, side = '>') |>  
 pander()

| diet | emmean | SE | df | lower.CL | upper.CL |
| --- | --- | --- | --- | --- | --- |
| NP | 27.4 | 0.943 | 233 | 25.84 | Inf |
| N/N85 | 32.69 | 0.8743 | 233 | 31.25 | Inf |
| N/R50 | 42.3 | 0.7834 | 233 | 41 | Inf |
| N/R40 | 45.12 | 0.8522 | 233 | 43.71 | Inf |

The inferences can also be adjusted for multiplicity using a Bonferroni correction by adding the argument adjust = 'bonferroni' as shown below.

# test whether lifetimes exceed 30 days  
emmeans(fit, ~ diet) |>  
 test(null = 30, side = '>', adjust = 'bonferroni')  
  
# one-sided intervals  
emmeans(fit, ~ diet) |>  
 confint(level = 0.95, side = '>', adjust = 'bonferroni')

|  |
| --- |
| Your turn |
| Test whether mean lifetime on each diet is less than 40 days. Adjust for multiple inferences using the Bonferroni correction  # modify this code to test whether mean lifetime is less than 40 emmeans(fit, ~ diet) |>  test(null = 30, side = '>', adjust = 'bonferroni') |

#### Pairwise comparisons

Pairwise comparisons are inferences for the pairwise differences between group means. They can take the form of tests, intervals, or both; regardless, multiple inference corrections should always be applied.

In emmeans, the pairs() function can be used in conjunction with test() or confint() to generate estimates and tests or intervals, respectively, for all pairwise differences. There is also a plotting method.

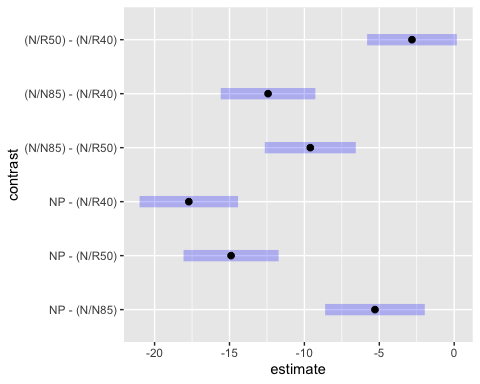
# pairwise comparisons: tests  
emmeans(fit, ~ diet) |>   
 pairs() |>  
 test() |>  
 pander()

| contrast | estimate | SE | df | t.ratio | p.value |
| --- | --- | --- | --- | --- | --- |
| NP - (N/N85) | -5.289 | 1.286 | 233 | -4.113 | 0.0003143 |
| NP - (N/R50) | -14.9 | 1.226 | 233 | -12.15 | 2.22e-13 |
| NP - (N/R40) | -17.71 | 1.271 | 233 | -13.94 | 2.207e-13 |
| (N/N85) - (N/R50) | -9.606 | 1.174 | 233 | -8.183 | 3.693e-13 |
| (N/N85) - (N/R40) | -12.43 | 1.221 | 233 | -10.18 | 2.573e-13 |
| (N/R50) - (N/R40) | -2.819 | 1.158 | 233 | -2.436 | 0.07326 |

# pairwise comparisons: intervals  
emmeans(fit, ~ diet) |>   
 pairs() |>  
 confint() |>  
 pander()

| contrast | estimate | SE | df | lower.CL | upper.CL |
| --- | --- | --- | --- | --- | --- |
| NP - (N/N85) | -5.289 | 1.286 | 233 | -8.617 | -1.962 |
| NP - (N/R50) | -14.9 | 1.226 | 233 | -18.07 | -11.72 |
| NP - (N/R40) | -17.71 | 1.271 | 233 | -21 | -14.43 |
| (N/N85) - (N/R50) | -9.606 | 1.174 | 233 | -12.64 | -6.568 |
| (N/N85) - (N/R40) | -12.43 | 1.221 | 233 | -15.58 | -9.266 |
| (N/R50) - (N/R40) | -2.819 | 1.158 | 233 | -5.815 | 0.1758 |

# plotting method  
emmeans(fit, ~ diet) |>  
 pairs() |>  
 plot()



|  |
| --- |
| Your turn |
| Experiment with the multiple correction methods by adding an adjust = ... argument with one of the following values:   * tukey is the default * bonferroni * scheffe * holm * hochberg   It’s easiest to see the effect by examining adjusted -values rather than interval endpoints. Focus on the last contrast between the two levels of restriction N/R50 - N/R40. Do the methods agree?  # try changing the multiple inference adjustment emmeans(fit, ~ diet) |>   pairs() |>  test(adjust = 'tukey') |

#### Comparisons involving a control

In the special case where one wishes to conduct multiple comparisons relative to a control, Dunnett’s method provides a better multiple inference correction.

Implementing these comparisons requires that the grouping variable be a factor whose first level is the control group. Provided this is the case, adding trt.vs.ctrl to the left-hand-side of the formula argument to emmeans() will generate the contrasts of interest.

# generate contrasts that compare with control  
emmeans(fit, trt.vs.ctrl ~ diet)

$emmeans  
 diet emmean SE df lower.CL upper.CL  
 NP 27.4 0.943 233 25.5 29.3  
 N/N85 32.7 0.874 233 31.0 34.4  
 N/R50 42.3 0.783 233 40.8 43.8  
 N/R40 45.1 0.852 233 43.4 46.8  
  
Confidence level used: 0.95   
  
$contrasts  
 contrast estimate SE df t.ratio p.value  
 (N/N85) - NP 5.29 1.29 233 4.113 0.0002  
 (N/R50) - NP 14.90 1.23 233 12.150 <.0001  
 (N/R40) - NP 17.71 1.27 233 13.938 <.0001  
  
P value adjustment: dunnettx method for 3 tests

|  |
| --- |
| Your turn |
| What if you wanted to test for diet restriction levels that caused an increase of *at least 10 days* relative to the unrestricted diet?  Consider the hypotheses:  While not especially common, one can test directional hypotheses for pairwise differences relative to non-zero null values. The test() function takes arguments null = ... to specify the null value for the test and side = ... to specify direction.   * side = '<' specifies a lower-sided test * side = '>' specifies an upper-sided test * side = '=' specifies a two-sided test   The default behavior in verbose form is given by the code below. Modify this to test the hypotheses mentioned at the outset.  # modify the default to test whether levels increase mean lifespan by at least 10 days emmeans(fit, trt.vs.ctrl ~ diet)$contrast |>  test(null = 0, side = '=') # modify here |

# Practice problem

The mussels data are standardized lengths of the anterior adductor muscle (AAM) of *Mytilus trossulus* mussels from five populations. Estimate mean AAM lengths for each population and test for differences between populations. If differences are determined to be significant, determine which populations differ significantly and provide interval estimates for the differences.

# read in data  
mussels <- read.csv('data/mussels.csv')  
head(mussels)  
  
# visualization via boxplots  
  
# grouped summaries, check SD's  
  
# anova fit and table  
  
# point and interval estimates of group means, with bonferroni correction  
  
# visualization of estimates  
  
# pairwise comparisons  
  
# visualization