ANOVA

Stat 145

Sandwiches and Ants

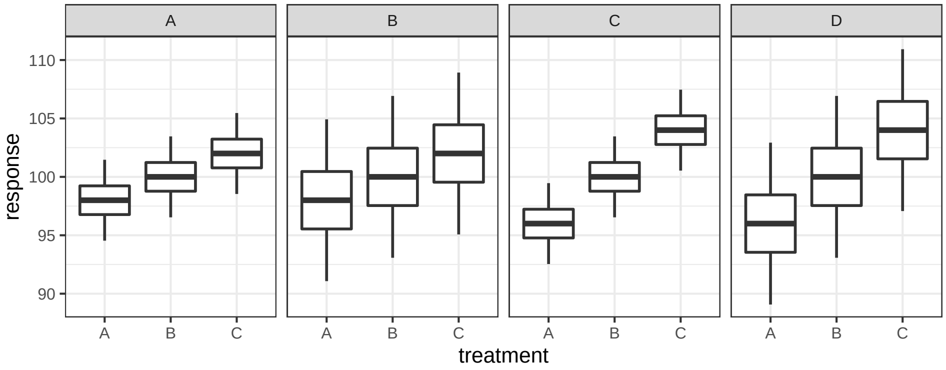
As young students, Dominic Kelly and his friends enjoyed watching ants gather on pieces of sandwiches. Later, as a university student, Dominic decided to study this with a more formal experiment. He chose three types of sandwich fillings (vegemite, peanut butter, and ham & pickles). To conduct the experiment he randomly chose a sandwich, broke off a piece, and left it on the ground near an ant hill. After several minutes he placed a jar over the sandwich bit and counted the number of ants. He repeated the process several times, allowing time for ants to return to the hill after each trial.

Dominic wanted to know whether ants are more attracted to some types of sandwiches than they are to others. The data are in `SandwichAnts` in the `Lock5withR` package.

1. Create a plot that shows how the number of ants is related to the type of filling. What sorts of plots might be good for this? When might you prefer each?
2. Compute the mean and standard deviation for the number of ants that went to each of the three filling types.
3. Based on your graphical and numerical summaries, do you think filling type matters, or do you think it was just random chance that brought more ants to some fillings than to others? Explain. What features of your summaries are you considering?
4. If we were to do a formal hypothesis test for this situation, what would the null hypothesis be?

The ANOVA Test Statistic

The trickiest part of this situation is coming up with a good test statistic. Consider the following plots



5. Assume that the sample sizes are the same for each of the four panels. Which plot gives the strongest evidence that the mean responses differs among treatment groups? Which gives the weakest evidence? Why?
6. If the sample sizes are not the same, how would your answer change?
7. What sorts of ingredients do you think the test statistic will include?

A small data set

Let's consider an even smaller data set (too small really, but useful for seeing how the method works.) In practice, we will never do the calculations below by hand, instead we will let R do them for us. But it is useful to see all the details one time.

```r
Small <- read.csv("https://rpruim.github.io/s145/data/small-anova.csv")
Small %>% pander()
```

group	response
A	4
A	5
A	6
B	7
B	8
B	8
B	5
C	10
C	8
C	7
C	9

8. Create a table with ten rows and 8 columns that has the group and response for this data in the first two columns. (You can use the table above as a starting point if your handwriting is small, but you might prefer to do this on the board or on a separate sheet of paper.)

9. Compute the mean of the entire data set and add a column that has that number copied over and over from top to bottom. Label it "grand mean".

10. Create a column labeled "group mean". Fill it with the appropriate group means.

11. Create a column labeled "(group mean - grand mean)" and fill in the values.

12. Create a column labeled "(group mean - grand mean)^2" and fill in the values. (Square the values of the previous column to get this.)

13. Create a column labeled "(response - group mean)" and fill in the values.

14. Create a column labeled "(response - group mean)^2" and fill in the values. (Square the values of the previous column to get this.)

15. Add up the values in the column labeled (group mean - grand mean)^2. Call this SSM. SS stands for sum of squares. M stands for model.
    This measures the variability **between groups**. If it is large, the group means are quite different. If it is small the group means are very similar. (Note: this is sometimes denoted as SSG, G for groups.)

16. Add up the values in the column labeled (response - group mean)^2. Call this SSE. SS stands for sum of squares. E stands for "error". This measures the variability **within groups**. If it is large, there is a lot of variability within the individual groups. If it is small there is little variability within groups. (This is sometimes called SSR, R for residuals. Recall that residual = observed = predicted. In this case observed is the response and our prediction is the group mean.)

17. Finally we need to adjust for the number of groups and the sample sizes. Compute $MSM = SSM/DFM$ and $MSE = SSE/DFE$ where

    - DFM is one less than the number of groups.
    - DFE is (number of observations) - (number of groups). (This is also the sum of one less than the number in each group.)

18. What do you get when you add $DFM + DFE$? This is sometimes called DFT, the total degrees for freedom.
19. Our test statistic is $F = \dfrac{MSM}{MSE}$. Compute F.
20. What makes F large? What makes F small? Will we reject the null hypothesis when F is large, when F is small, or both?

Automating this

That's a pretty tedious calculation. Fortunately, R will do the whole thing for us! Here's how:

```r
small_model <- lm(response ~ group, data = Small)
anova(small_model)
```

```
## Analysis of Variance Table
##
## Response: response
##           Df Sum Sq Mean Sq F value  Pr(>F)
## group      2     21  10.500  6.4615 0.02137
## Residuals  8     13   1.625
```

You should recognize DFM, DFE, SSM, SSE, MSM, MSE, and F from your calculations. (If things don't look right, see if you can find your error.) It even computes a p-value for us.

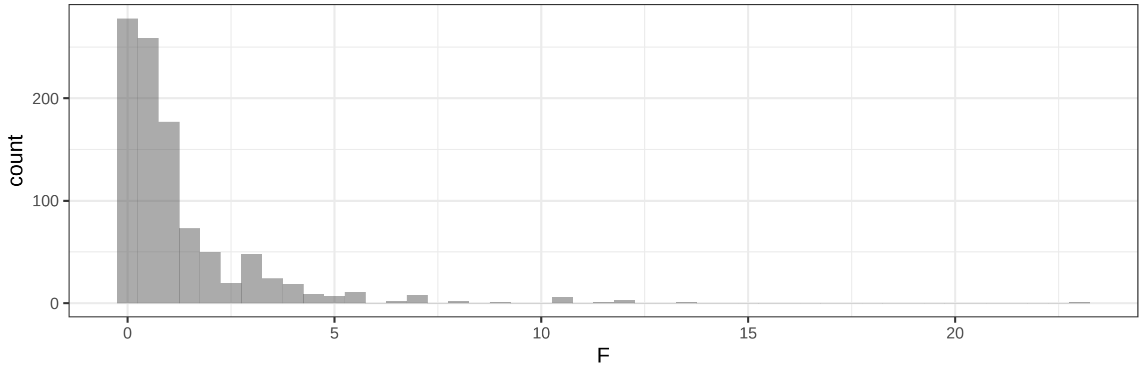
ANOVA P-value

21. For comparison, create a randomization distribution for F. Here's a hint: What do you need to change in the code below?

```r
do(3) * lm(response ~ group, data = Small)
```

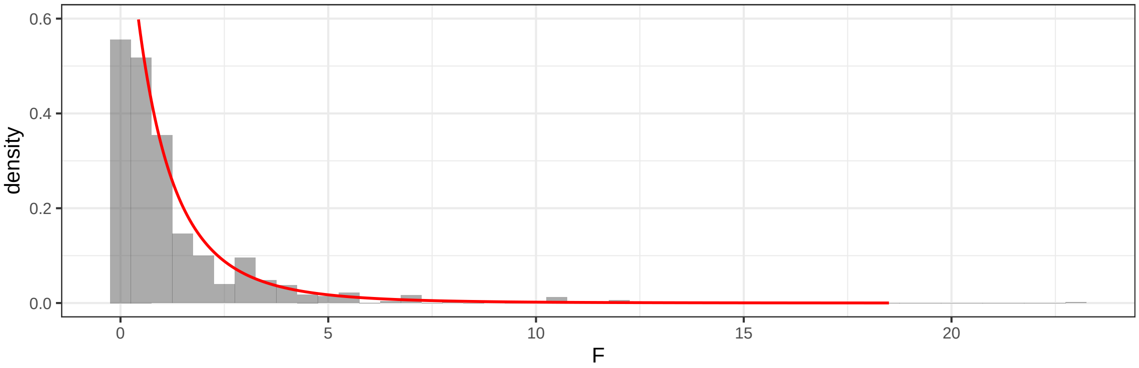
```
##   Intercept groupB groupC    sigma r.squared        F numdf dendf .row .index
## 1         5      2    3.5 1.274755 0.6176471 6.461538     2     8    1      1
## 2         5      2    3.5 1.274755 0.6176471 6.461538     2     8    1      2
## 3         5      2    3.5 1.274755 0.6176471 6.461538     2     8    1      3
```

22. Use your randomization distribution to compute a p-value. You should get a randomization distribution something like this:



In many situations (see the next section), this distribution will follow a shape called an F distribution (the one used above when R computed a p-value for us). The F distributions have two parameters: `df1` = DFM and `df2` = DFE. Let's see how our histogram compares to an F(2, 8) distribution:

```
gf_dhistogram(~ F, data = Small_Rand, binwidth = 0.5) %>%
  gf_dist("f", df1 = 2, df2 = 8, color = "red")  %>%
  gf_lims(y = c(0, 0.6))
```



When is the F-distribution approximation good enough?

The F approximation is perfect if

- each group is normally distributed in the population
- the standard deviations are the same in each group.

In practice, this is never exactly true, but it is often close enough to true. Our rule of thumb will be to use the F distribution approximation when

- sample sizes are not too small (at least 30 in each group) or we are quite sure that the populations are approximately normal, and
- the standard deviations in the groups are fairly similar (largest is no more than double the smallest).
    - This matters more if the sample sizes are different. If the sample sizes are very similar, the standard deviations can be more different.

If these conditions are not met, we will favor the simulation method (or other procedures).

Still To Do

- We need some procedures for checking whether the conditions for ANOVA are met.
- If the p-value is small, we would like to know which means differ and by how much.

We'll save these important parts of ANOVA for another day.

More Pracice

Sandwiches and Ants again

23. Compute a p-value for the sandwiches and ants data. Do it two ways: once by creating a randomization distribution and once letting `lm()` and `anova()` do all the work. How do the results compare?

FAMuSS

The Functional polymorphisms Associated with human Muscle Size and Strength study (FAMuSS) measured a variety of demographic, phenotypic, and genetic characteristics for about 1,300 participants. Data from the study have been used in a number of subsequent studies, including one examining the relationship between muscle strength and genotype at a specific location (`actn3.r577x`) on the ACTN3 gene, sometimes called "the sports gene". One measure of muscle strength was the percent change in non-dominant arm strength (`ndrm.ch`) before and after a training program.

```
library(oibiostat)
data(famuss)
head(famuss) %>% pander()
```

ndrm.ch	drm.ch	sex	age	race	height	weight	actn3.r577x	bmi
40	40	Female	27	Caucasian	65	199	CC	33.11
25	0	Male	36	Caucasian	71.7	189	CT	25.84
40	0	Female	24	Caucasian	65	134	CT	22.3
125	0	Female	40	Caucasian	68	171	CT	26
40	20	Female	32	Caucasian	61	118	CC	22.29
75	0	Female	24	Hispanic	62.2	120	CT	21.8

23. Use `df_stats()` to create a summary that lists the means, standard deviations, and sample sizes for the three groups.
24. Make an appropriate plot for these data.
25. What is the null hypothesis for the ANOVA test in this situation?
26. Based on your plot, how large do you expect the p-value will be for this test?
27. Use `lm()` and `anova()` to compute the p-value. Interpret the result.
28. What are the degrees of freedom for this test? Why?
29. Use `do()` and `lm()` and `shuffle()` to create a randomization distribution and compute the p-value from that. How does this p-value compare to your previous p-value?

Blisters

31. Use this data set to see whether three different treatments for blisters have different mean healing times. Again, do it both ways and compare the results. Does the treatment method seem to matter? If so, which treatments seem better/worse? (Don't forget to plot the original data.)

    ```
    Blisters <- read.csv("https://rpruim.github.io/s145/data/blisters.csv")
    head(Blisters) %>% pander()
    ```

days	treatment	group
5	A	1
6	A	1
6	A	1
7	A	1
7	A	1
8	A	1

Checking the conditions for ANOVA

The mathematical model for the F distribution used in ANOVA comes from a situation where

1. Each group is normally distributed.
2. Each group has the same standard deviation (σ).
3. Each observation is independent of the others.

In practice, we need to assess whether these are "close enough to true". We can assess #2 by computing the standard deviations for each group and comparing them. They won't be exactly the same, but we would like the largest to be no more than twice as large as the smallest, unless the sample sizes are very similar, in which case the differences in standard deviations matter less.

Whether #3 holds depends on the design our study and how the data are collected.

The easiest way to assess #1, especially when the sample sizes are small, is to look at all the groups together. We can do that if we look at the residuals:

$$\text{residual} = \text{observed} - \text{predicted} = \text{observed} - \text{groupmean}$$

The distribution of the residuals should be approximately normal with a mean of 0, no matter what group we are looking at. If each group has the same standard deviation σ, then they will all be normal with mean 0 and standard deviation σ. We can inspect the distribution of residuals using a histogram, but there is an even better plot called a normal-quantile plot. The normal quantile plot is a scatter plot that relates the observed data (sample) values to the values they would have in an idealized (theoretical) normal distribution. This special scatter plot will be approximately linear when the distribution is normal. This is nice because our eyes are better at judging whether a shape is a line than whether a shape has a particular type of curve. The downside of this is that we are easily distracted by any departure from the line and need to get calibrated a bit before it is easy to interpret these plots.

And as always, shape is very hard to determine from a small sample and matters less when we have a large sample.

Example: Sandwich Ants

Let's illustrate how to assess these conditions using the sandwich ants example.

In the study, the researchers made sure to allow sufficient time between sandwiches so that the ants could get back to their ant hill and "start over" with the new sandwich. If they had not done that, then the number of ants coming to one sandwich might have an impact on the number of ants coming to the next. For example, if a lot of ants come to a sandwich and then we immediately place new sandwich, then there will be more ants nearby to find it. But if few ants come to one sandwich, there would be fewer nearby to find the next. That would introduce a positive correlation between one observation and the next.

Checking the standard deviations is as simple as computing the sample standard deviations and looking at their values.

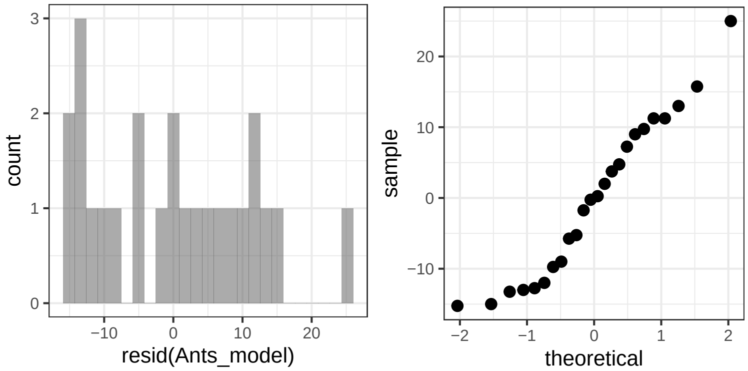
```
df_stats(Ants ~ Filling, data = SandwichAnts)
```

```
##   response       Filling min    Q1 median    Q3 max  mean        sd n missing
## 1     Ants Ham & Pickles  34 42.00   51.0 55.25  65 49.25 10.793517 8       0
## 2     Ants Peanut Butter  19 21.75   30.5 44.00  59 34.00 14.628739 8       0
## 3     Ants      Vegemite  18 24.00   30.0 39.00  42 30.75  9.254343 8       0
```

The sample sizes here are exactly the same in each group, so this check is less important than it would be if the group sizes were quite different, but the smallest is more than half of the largest, so we're good to go anyway.

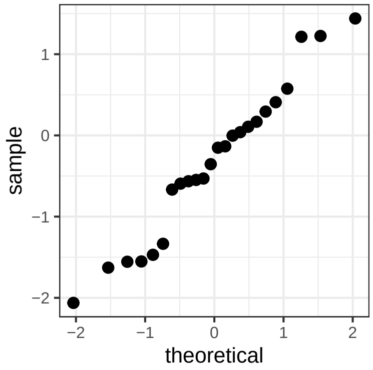
Finally, we can look at the residuals. There are several ways to do this. The function `resid()` will extract residuals from our model. Then we can create a histogram or normal-quantile plot.

```
Ants_model <- lm(Ants ~ Filling, data = SandwichAnts)
gf_histogram(~resid(Ants_model))
# normal quantile plot is a special case of quantile-quantile plot, so qq.
gf_qq(~resid(Ants_model))
```



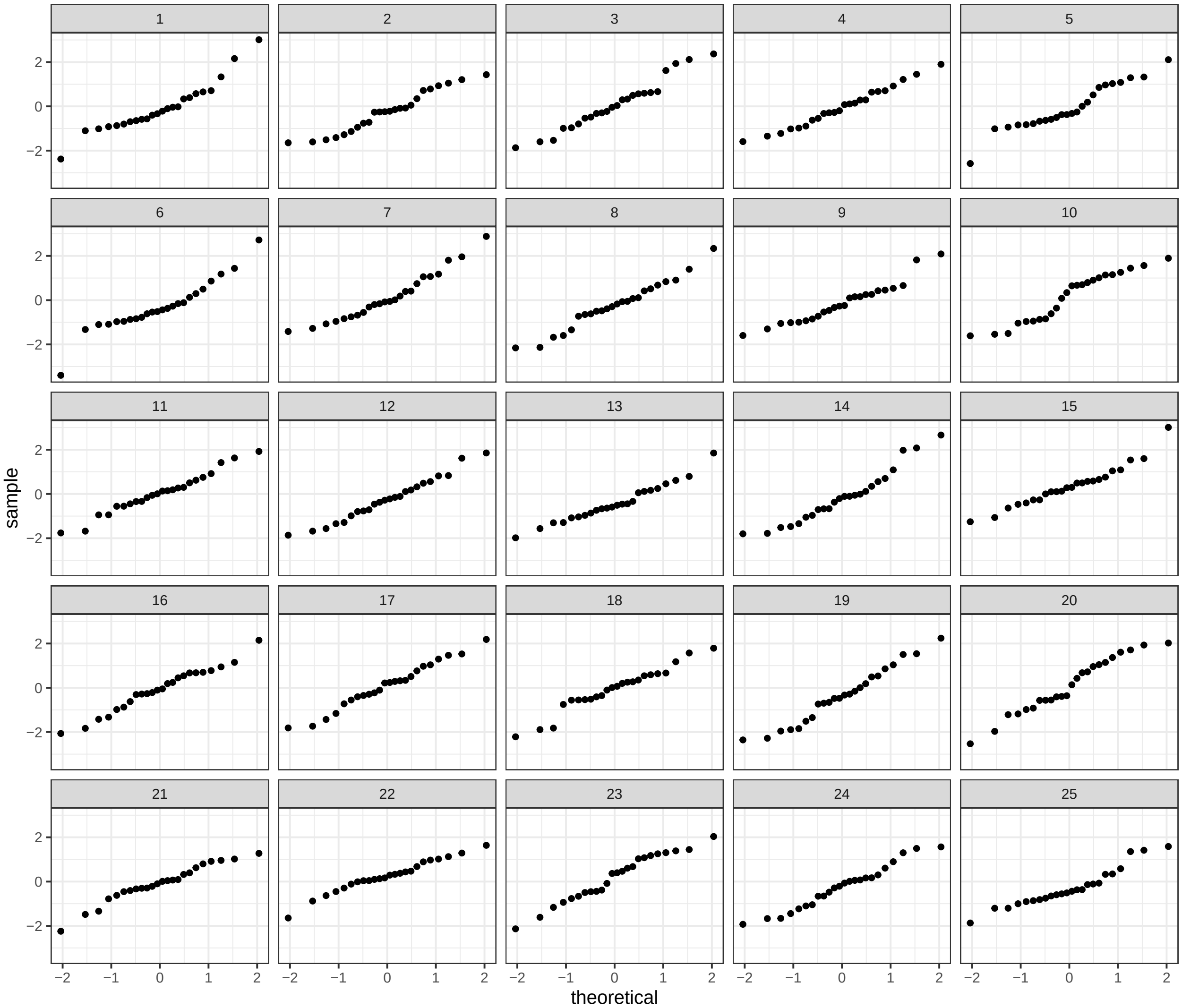
Assessing a histogram or normal-quantile plot is pretty challenging with small data sets, and we shouldn't expect an exactly straight line. How straight should the line be in a qq-plot? We can generate some random examples and compare.

```
gf_qq( ~ rnorm(24))
```



Let's look at 25 of these.

```
expand.grid(rep = 1:25, i = 1:24) %>%
  mutate(x = rnorm(25 * 24)) %>%
  gf_qq(~x | rep, size = 0.7)
```



As we see, we the patterns is not exactly a straight line, and the last couple values on each end often fit the linear pattern worst. What we are looking for is a pattern that clearly departs from a line, especially in the center of the distribution where there dots are closest together. That would be an indication that the normality assumption is not met.

If we are concerned about these conditions, we can always choose to use the randomization method instead.

Your turn

32. Check the ANOVA conditions for the FAMuSS example.

Which means differ? By how much?

If we obtain a small p-value and can reject the null hypothesis that the means are all the same, the natural question is which means are different from which, and by how much.

We almost know how to answer this already: two sample t tests or confidence intervals for the difference in means. We will focus on confidence intervals primarily.

The confidence interval for a 2-sample t-test has the form

$$\overline{x_1} - \overline{x_2} \pm t_* \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$$

When we compute all of the pairwise intervals, we will make two adjustments.

1. Since we are assuming all the standard deviations are the same, we will compute one standard deviation and use it in both places. This **pooled estimate for standard deviation** is

   $$s = \sqrt{MSE}$$

   so

   $$s^2 = MSE$$

   We can read this number directly off of the ANOVA table output. We will plug this in for both s_1^2 and s_2^2. [Note: s will always be somewhere between the largest and smallest of the group standard deviations.]

2. To adjust for the number of intervals we are computing, we will use a larger critical value You won't need to know how that is computed, since we will let R take care of that for us. But the effect will be to make all of the intervals wider. (The more groups there are, the bigger the adjustment.)

   The goal of this adjustment is to make the confidence intervals wide enough that the probability is 95% (or some other confidence level) that **all** of the intervals contain the parameter of interest (in this case, the parameter of interest is the difference between two population means). This is referred to as a **family-wise** confidence level.

   To do this in R, we use the `TukeyHSD()` function. Tukey is a famous statistician. HSD stands for "honest significant difference" – honest because it acknowledges that we are looking at all possible pairs of groups and not computing just one interval.

Fortunately R does all of the work for us and displays the results in a table or a plot.

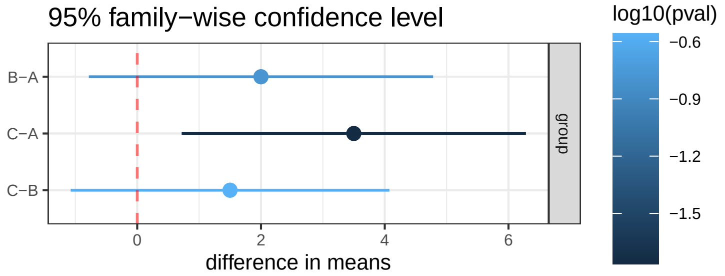
```r
small_model <- lm(response ~ group, data = Small)
small_model %>% anova() # or anova(small_model)
```

```
## Analysis of Variance Table
##
## Response: response
##           Df Sum Sq Mean Sq F value  Pr(>F)
## group      2     21  10.500  6.4615 0.02137
## Residuals  8     13   1.625
```

```
small_model %>% TukeyHSD() # or TukeyHSD(small_model)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = x)
##
## $group
##     diff        lwr      upr     p adj
## B-A  2.0 -0.7820367 4.782037 0.1613927
## C-A  3.5  0.7179633 6.282037 0.0172056
## C-B  1.5 -1.0756655 4.075665 0.2759368
```

```
small_model %>% TukeyHSD() %>% mplot()
# or mplot(TukeyHSD(small_model))
```



In this case we see that our data give us evidence that groups A and C have different means.

33. How do we conclude this from the output above?
34. What can we conclude about groups A and B? B and C?

Your Turn

35. Return to the sandwich ants experiment. If the ANOVA p-value was small enough, use Tukey's honest significant differences to determine which means appear to be different from which.
36. Return to the FAMuSS example. If the ANOVA p-value was small enough, use Tukey's honest significant differences to determine which means appear to be different from which.
37. Return to the blisters experiment. If the ANOVA p-value was small enough, use Tukey's honest significant differences to determine which means appear to be different from which.

The small example – check your work

```
SmallTable <-
  Small %>%
  mutate(
    `grand mean` = mean(response)
  ) %>%
  group_by(group) %>%
  mutate(
    `group mean` = mean(response),
    `group mean - grand mean` = `group mean` - `grand mean`,
    `(group mean - grand mean)^2` = (`group mean` - `grand mean`)^2,
    `response - group mean` = response - mean(response),
    `(response - group mean)^2` = (response - mean(response))^2
  ) %>% ungroup()
SmallTable %>% pander()
```

Table 4: Table continues below

group	response	grand mean	group mean	group mean - grand mean	(group mean - grand mean)^2	response - group mean	(response - group mean)^2
A	4	7	5	-2	4	-1	1
A	5	7	5	-2	4	0	0
A	6	7	5	-2	4	1	1
B	7	7	7	0	0	0	0
B	8	7	7	0	0	1	1
B	8	7	7	0	0	1	1
B	5	7	7	0	0	-2	4
C	10	7	8.5	1.5	2.25	1.5	2.25
C	8	7	8.5	1.5	2.25	-0.5	0.25
C	7	7	8.5	1.5	2.25	-1.5	2.25
C	9	7	8.5	1.5	2.25	0.5	0.25