DATA2002

ANOVA post hoc tests

Garth Tarr

THE UNIVERSITY OF SYDNEY

Multiple comparisons

Checking for normality with residuals

Bonferroni method

Tukey's method

Scheffé's method

Multiple comparisons: simultaneous confidence intervals

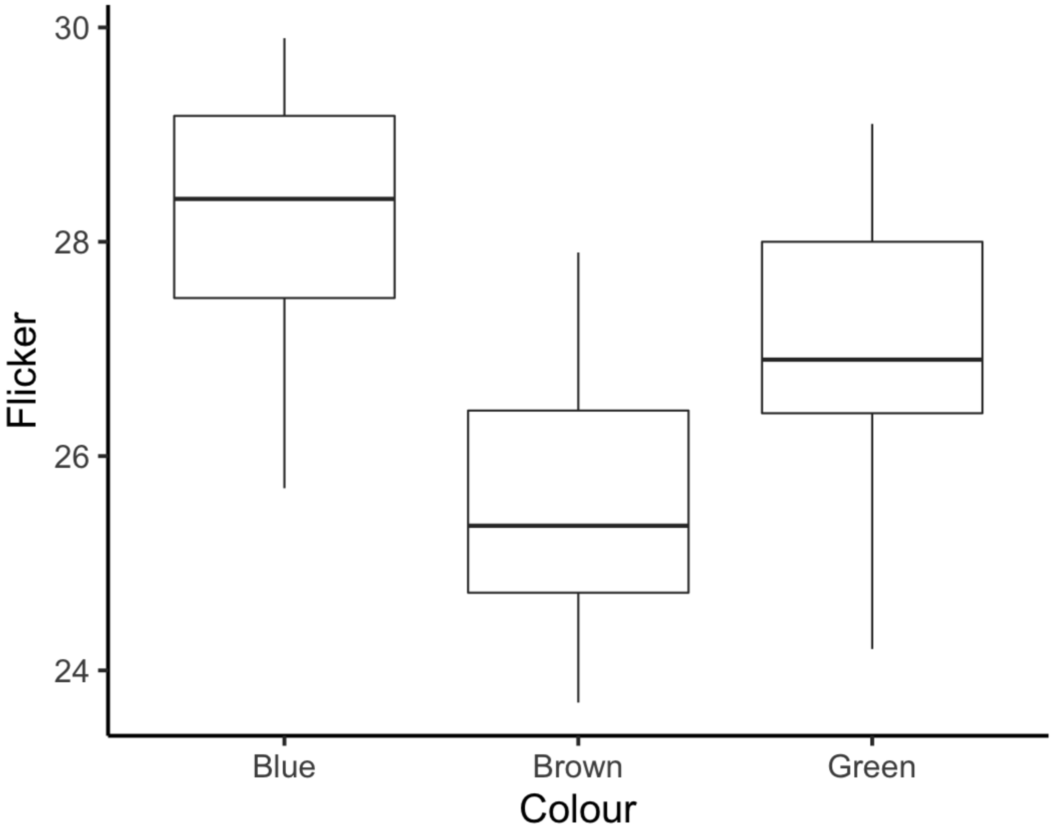
Multiple comparisons: simultaneous confidence intervals

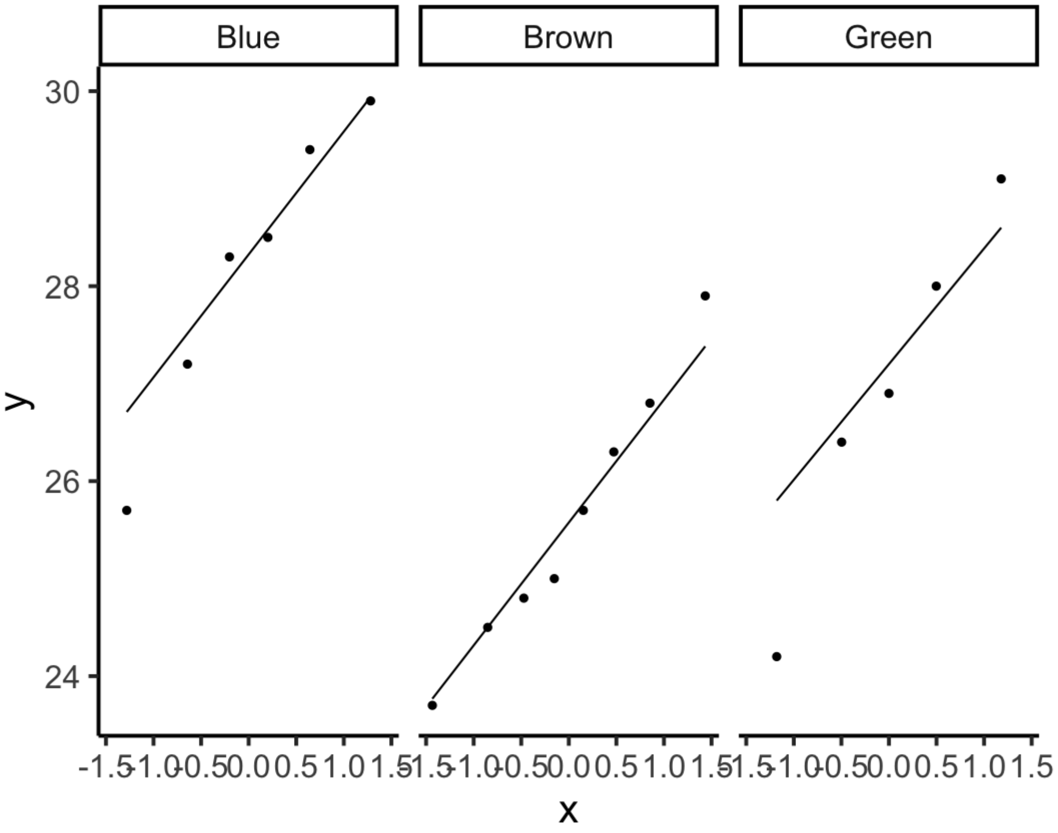
- In general, there may be more than one "contrast of interest".
- Consider the "flicker frequency" data, considered in this week's tutorial:

```
path = "https://raw.githubusercontent.com/DATA2002/data/master/flicker.txt"
flicker = read_tsv(path)
glimpse(flicker)
```

```
## Rows: 19
## Columns: 2
## $ Colour  <chr> "Brown", "Brown", "Brown", "Brown", "Brown…
## $ Flicker <dbl> 26.8, 27.9, 23.7, 25.0, 26.3, 24.8, 25.7, …
```

```
p1 = ggplot(flicker, aes(x = Colour, y = Flicker)) +
  geom_boxplot() + theme_classic(base_size = 20)
p2 = ggplot(flicker, aes(sample = Flicker)) +
  geom_qq() + geom_qq_line() + facet_wrap(~ Colour) + theme_classic(base_size = 20)
gridExtra::grid.arrange(p1, p2, ncol=2)
```





```r
sum_stat = flicker %>%
  group_by(Colour) %>%
  summarise(n_i = n(),
            ybar_i = mean(Flicker),
            v_i = var(Flicker))
sum_stat
```

```
## # A tibble: 3 × 4
##   Colour   n_i ybar_i   v_i
##   <chr>  <int>  <dbl> <dbl>
## 1 Blue       6   28.2  2.33
## 2 Brown      8   25.6  1.86
## 3 Green      5   26.9  3.40
```

```r
n_i = sum_stat %>% pull(n_i)
ybar_i = sum_stat %>% pull(ybar_i)
v_i = sum_stat %>% pull(v_i)
```

Aside: checking for normality with residuals

The population model is,

$$Y_{ij} = \mu_i + \varepsilon_{ij},$$

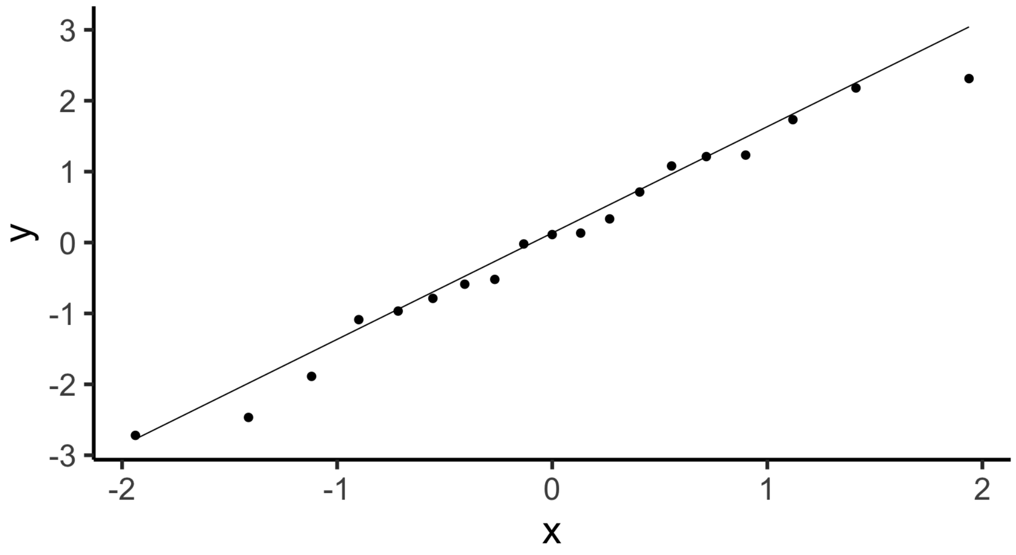
where $\varepsilon_{ij} \sim N(0, \sigma^2)$.

Rather than looking at QQ plots for each sample, we can instead consider the ANOVA residuals,

$$r_{ij} = y_{ij} - \bar{y}_{i\bullet}.$$

If the ANOVA assumptions hold true, then the residuals should be normally distributed.

```r
flicker_anova = aov(Flicker~Colour,
                    data = flicker)
flicker_resid = flicker_anova$residuals
ggplot(data.frame(flicker_resid),
       aes(sample = flicker_resid)) +
  geom_qq(size=3) + geom_qq_line() +
  theme_classic(base_size = 32)
```



All pairwise differences

- When no single group is "special" or notable, so that each pairwise difference is equally interesting, we can consider each pairwise difference as a contrast of interest.
- In this case,
- a t-statistic can be constructed for each pairwise difference;
- a t-based confidence interval can be constructed for each pairwise "population" difference (contrast).
- Let's focus on confidence intervals for the moment.

Individual 95% confidence intervals

- We now construct 95% confidence intervals for each pairwise comparison *individually*.
- the standard error for $\bar{y}_{i\bullet} - \bar{y}_{h\bullet}$ is $\hat{\sigma}\sqrt{\frac{1}{n_i} + \frac{1}{n_h}}$.

```
N = length(flicker_resid)
g = 3
sig_sq_hat = sum(flicker_resid^2)/(N-g) # Mean square resiudal
sig_sq_hat
```

```
## [1] 2.39438
```

```
# alternatively
# sig_sq_hat = sum((n_i - 1) * v_i)/sum(n_i - 1)
t_star = qt(.975, df = sum(n_i - 1))
t_star
```

```
## [1] 2.119905
```

Blue vs Brown

```
se.Bl.Br = sqrt(sig_sq_hat * ((1/n_i[1]) + (1/n_i[2])))
(int.Bl.Br.95.indiv = ybar_i[1] - ybar_i[2] + c(-1,1) * t_star * se.Bl.Br)
```

```
## [1] 0.8076044 4.3507289
```

Blue vs Green

```
se.Bl.Gr = sqrt(sig_sq_hat * ((1/n_i[1]) + (1/n_i[3])))
(int.Bl.Gr.95.indiv = ybar_i[1] - ybar_i[3] + c(-1, 1) * t_star * se.Bl.Gr)
```

```
## [1] -0.7396511  3.2329845
```

Green vs Brown

```
se.Gr.Br = sqrt(sig_sq_hat*((1/n_i[2]) + (1/n_i[3])))
(int.Gr.Br.95.indiv = ybar_i[2] - ybar_i[3] + c(-1, 1) * t_star * se.Gr.Br)
```

```
## [1] -3.2025564  0.5375564
```

The emmeans package

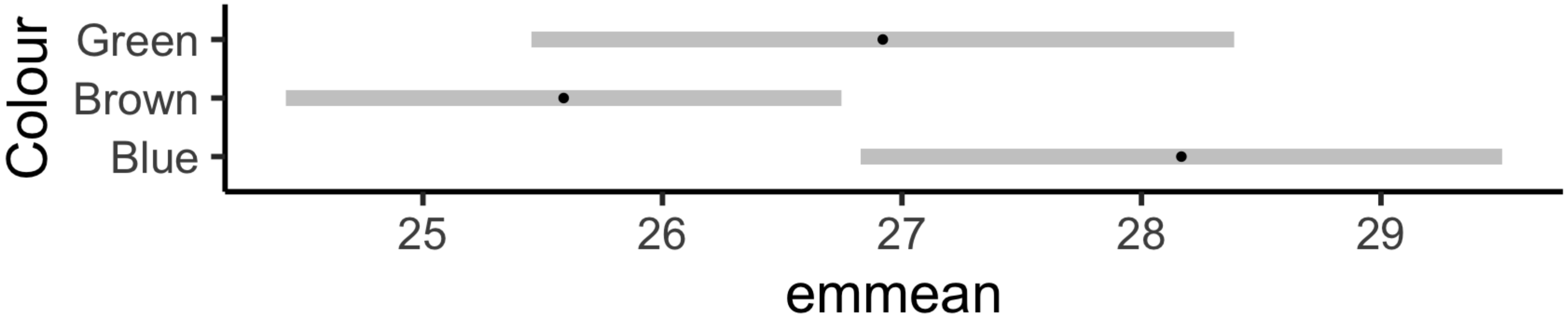
```r
# install.packages("emmeans")
suppressPackageStartupMessages(library(emmeans))
flicker_anova = aov(Flicker ~ Colour, data = flicker)
flicker_em = emmeans(flicker_anova, ~ Colour)
confint(flicker_em, adjust = "none")
```

```
##  Colour emmean    SE df lower.CL upper.CL
##  Blue     28.2 0.632 16     26.8     29.5
##  Brown    25.6 0.547 16     24.4     26.7
##  Green    26.9 0.692 16     25.5     28.4
##
## Confidence level used: 0.95
```

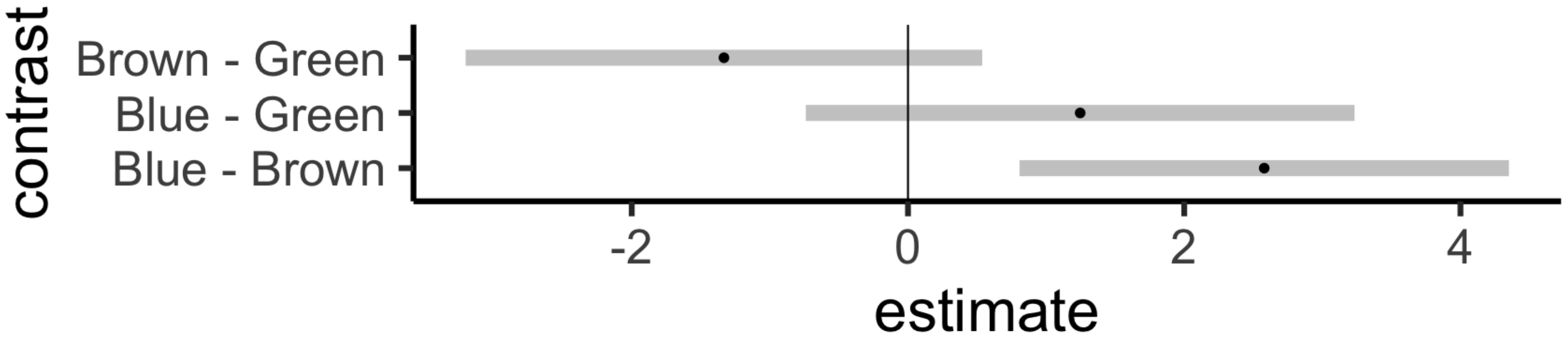
```r
confint(pairs(flicker_em, adjust = "none"))
```

```
##  contrast      estimate    SE df lower.CL upper.CL
##  Blue - Brown      2.58 0.836 16    0.808    4.351
##  Blue - Green      1.25 0.937 16   -0.740    3.233
##  Brown - Green    -1.33 0.882 16   -3.203    0.538
##
## Confidence level used: 0.95
```

```r
confint(flicker_em, adjust = "none") %>% plot(colors = "black") + theme_classic(base_size = 30)
```




```r
confint(pairs(flicker_em, adjust = "none")) %>% plot(colors = "black") +
  geom_vline(xintercept = 0)
```



Summary of *individual* intervals

- So it would appear that *individually* the only "significantly different" pair is Blue and Brown.
- **However**, we have constructed each interval *without taking any regard of the others*.
- More precisely:
  - each interval has been constructed using a procedure so that *when the model is correct*, the probability that the "correct" population contrast is covered is 0.95... *individually*.
- **But**, what is the probability that **all** intervals cover their corresponding true values **simultaneously**?

The Bonferroni method

- Let A_1, A_2 A_3 denote the events where each of the 3 intervals above cover the corresponding "true" value.
- Then, under our normal-equal-variance model, we have

$$P(A_1) = P(A_2) = P(A_3) = 0.95\,.$$

- However, what is $P\left(A_1 \cap A_2 \cap A_3\right)$?
- This is "a bit hard"[1], but we can derive a *lower bound* a bit more easily using the relation

$$\left(A_1 \cap A_2 \cap A_3\right)^c = A_1^c \cup A_2^c \cup A_3^c\,.$$

This is solved *exactly* in the case of equal sample sizes using Tukey's method.

- Recall that $P(A \cup B) \leq P(A) + P(B)$, so we get

$$\begin{aligned}
1 - P\left(A_1 \cap A_2 \cap A_3\right) = P\left\{\left(A_1 \cap A_2 \cap A_3\right)^c\right\} &= P\left(A_1^c \cup A_2^c \cup A_3^c\right) \\
&\leq P(A_1^c) + P(A_2^c) + P(A_3^c) \\
&= 0.05 + 0.05 + 0.05 = 0.15.
\end{aligned}$$

- Therefore, $P(A_1 \cap A_2 \cap A_3) \geq 0.85.$

The *simultaneous coverage probability* of all 3 intervals is therefore *at least* 85%.

Make the individual intervals *a little bit wider*

- This method shows us how to get a lower bound of 0.95:
- make each interval have *individual* coverage probability $1 - (0.05)/3 = 59/60 = 0.98\dot{3}$ (this requires the $1 - (0.05/6)$ quantile!):

```
t_simul = qt(1 - (0.05)/6, df = sum(n_i - 1))
t_simul
```

```
## [1] 2.673032
```

Simultaneous (at least) 95% confidence intervals

Blue vs Brown

```
(int.Bl.Br.95.simul = ybar_i[1] - ybar_i[2] + c(-1,1) * t_simul * se.Bl.Br)
```

```
## [1] 0.3453673 4.8129660
```

Blue vs Green

```
(int.Bl.Gr.95.simul = ybar_i[1] - ybar_i[3] + c(-1,1) * t_simul * se.Bl.Gr)
```

```
## [1] -1.257922  3.751256
```

Green vs Brown

```
(int.Gr.Br.95.simul = ybar_i[2] - ybar_i[3] + c(-1, 1) * t_simul * se.Gr.Br)
```

```
## [1] -3.690493  1.025493
```

emmeans package

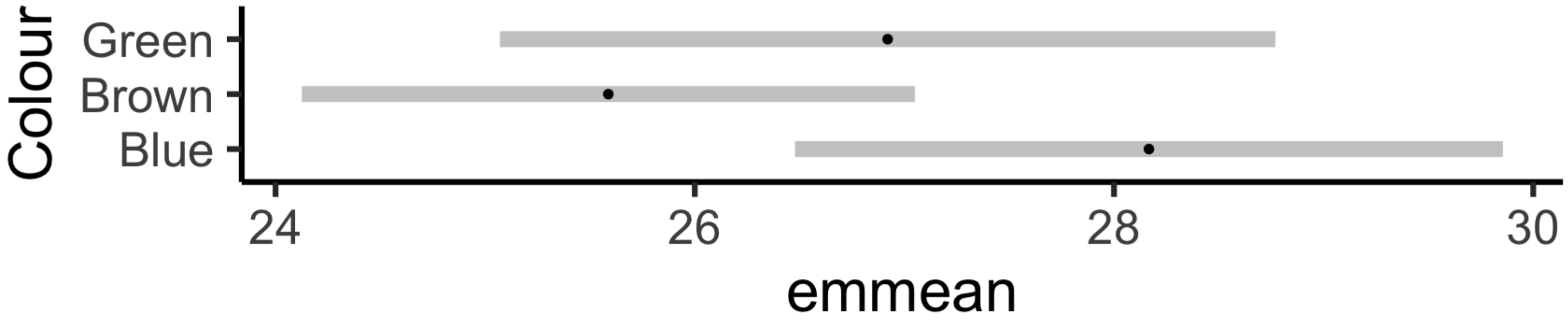
```r
flicker_em = emmeans(flicker_anova, ~ Colour)
confint(flicker_em, adjust = "bonferroni")
```

```
##  Colour emmean    SE df lower.CL upper.CL
##  Blue     28.2 0.632 16     26.5     29.9
##  Brown    25.6 0.547 16     24.1     27.0
##  Green    26.9 0.692 16     25.1     28.8
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
```

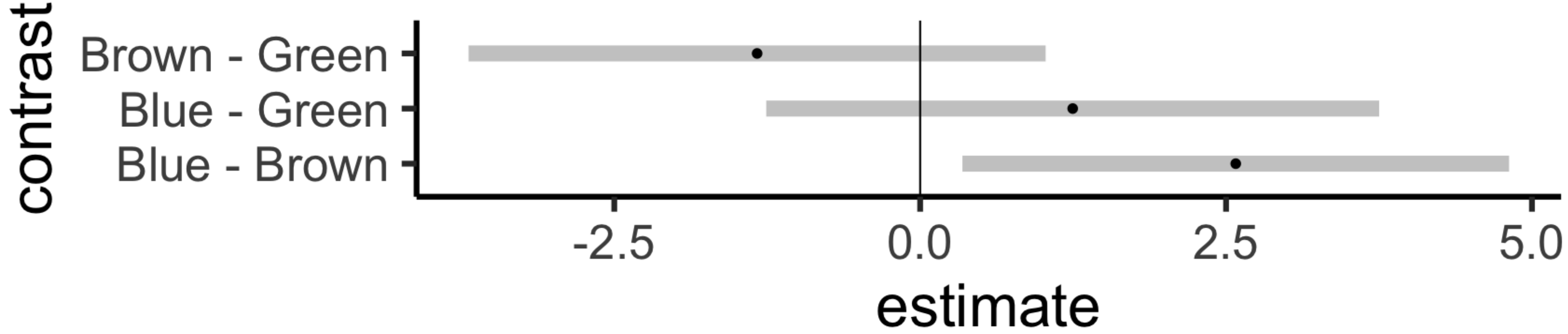
```r
confint(pairs(flicker_em, adjust = "bonferonni"))
```

```
##  contrast      estimate    SE df lower.CL upper.CL
##  Blue - Brown      2.58 0.836 16    0.345     4.81
##  Blue - Green      1.25 0.937 16   -1.258     3.75
##  Brown - Green    -1.33 0.882 16   -3.690     1.03
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
```

```r
confint(flicker_em, adjust = "bonferroni") %>% plot(colors = "black")
```




```r
confint(pairs(flicker_em, adjust = "bonferonni")) %>% plot(colors = "black") +
  geom_vline(xintercept = 0)
```



"Simultaneous" conclusions

- So, even though we "adjusted for multiplicity", the "Blue–Brown" difference is *still significant*, in the sense that the corresponding interval does **not** include zero.
- By increasing the confidence level of each *individual* comparison, we are able to make "simultaneous" valid statments about them all.

The general Bonferroni approach for k simultaneous comparisons

- In general, if we have k confidence intervals that we wish to have *simultaneous* coverage probability of (*at least*) $100(1-\alpha)\%$, we can achieve this (possibly conservatively!) by constructing each interval to have *individual* coverage probability $100(1-\alpha/k)\%$.
- If we have g groups, then there are $k = \binom{g}{2} = \frac{g(g-1)}{2}$ possible pairs.
- For moderate-to-large g, this grows "quadratically" i.e. like g^2;
- other approaches e.g. Tukey's method, Scheffé's method can give "less conservative" (i.e. smaller) multipliers.

Pairwise t tests

A general t test for a contrast takes the form:

$$t_0 = \frac{\sum_{i=1}^{g} c_i \bar{y}_{i\bullet}}{\hat{\sigma}\sqrt{\sum_{i=1}^{g} c_i^2/n_i}}$$

Blue vs Brown

$$t_0 = \frac{\bar{y}_{1\bullet} - \bar{y}_{2\bullet}}{\hat{\sigma}\sqrt{1/n_1 + 1/n_2}}$$

```
se.Bl.Br = sqrt(sig_sq_hat *
                 ((1/n_i[1]) + (1/n_i[2])))
t_stat.Bl.Br = (ybar_i[1]-ybar_i[2])/se.Bl.Br
2*(1-pt(abs(t_stat.Bl.Br), df = sum(n_i-1)))
```

```
## [1] 0.007079982
```

Blue vs Green

```
t_stat.Bl.Gr=(ybar_i[1]-ybar_i[3])/se.Bl.Gr
2*(1-pt(abs(t_stat.Bl.Gr),df=sum(n_i-1)))
```

```
## [1] 0.2020033
```

Brown vs Green

```
t_stat.Gr.Br=(ybar_i[2]-ybar_i[3])/se.Gr.Br
2*(1-pt(abs(t_stat.Gr.Br),df=sum(n_i-1)))
```

```
## [1] 0.1504046
```

Pairwise t tests using emmeans

No adjustment

```
test(pairs(flicker_em, adjust = "none"))
```

```
##  contrast      estimate    SE df t.ratio p.value
##  Blue - Brown      2.58 0.836 16   3.086  0.0071
##  Blue - Green      1.25 0.937 16   1.331  0.2020
##  Brown - Green    -1.33 0.882 16  -1.511  0.1504
```

Bonferroni adjustment (multiply unadjusted p-values by 3)

```
test(pairs(flicker_em, adjust = "bonferroni"))
```

```
##  contrast      estimate    SE df t.ratio p.value
##  Blue - Brown      2.58 0.836 16   3.086  0.0212
##  Blue - Green      1.25 0.937 16   1.331  0.6060
##  Brown - Green    -1.33 0.882 16  -1.511  0.4512
##
## P value adjustment: bonferroni method for 3 tests
```

Summary

Contrast	Unadjusted p-value	Adjusted p-value
Blue–Brown	0.007	0.021
Blue–Green	0.202	0.606
Brown–Green	0.150	0.450
Overall p-value		0.021

Tukey's method

- John Tukey derived the *exact* multiplier needed for simultaneous confidence intervals for all pairwise comparisons **when the sample sizes are equal**.
- It was later shown that when sample sizes are *unequal*, Tukey's procedure is *conservative*, thus yielding valid simultaneous intervals that may be *narrower* than those using the Bonferroni method.
- Multiplicity-adjusted p-values can be obtained in the same way by inverting the intervals.
- The "overall ANOVA null hypothesis" can be tested using the *smallest* of these.
- Tukey named his method "Honest Significant Differences"; it is implemented in the function `TukeyHSD()`, which takes as argument an `aov()` fit or using the `emmeans` package

Tukey's method

```r
# TukeyHSD(flicker_anova, conf.level = 0.95)
confint(pairs(flicker_em, adjust = "tukey"))
```

```
##  contrast      estimate    SE df lower.CL upper.CL
##  Blue - Brown      2.58 0.836 16    0.423    4.735
##  Blue - Green      1.25 0.937 16   -1.171    3.664
##  Brown - Green    -1.33 0.882 16   -3.609    0.944
##
## Confidence level used: 0.95
## Conf-level adjustment: tukey method for comparing a family of 3 estimates
```

```r
test(pairs(flicker_em, adjust = "tukey"))
```

```
##  contrast      estimate    SE df t.ratio p.value
##  Blue - Brown      2.58 0.836 16   3.086  0.0184
##  Blue - Green      1.25 0.937 16   1.331  0.3994
##  Brown - Green    -1.33 0.882 16  -1.511  0.3124
##
## P value adjustment: tukey method for comparing a family of 3 estimates
```

Scheffé's simultaneous confidence interval method

- If we choose the special multiplier

$$t^{\star}_{\mathrm{Sch}}(\alpha) = \sqrt{(g-1)F_{g-1,N-g}(\alpha)} = \sqrt{(g-1)\texttt{*qf(1}-\alpha\texttt{,}g-1,N-g\texttt{)}}$$

  and construct simultaneous confidence intervals for **all possible contrasts** according to

$$\sum_{i=1}^{g} c_i \bar{Y}_{i\bullet} \pm t^{\star}_{\mathrm{Sch}}(\alpha)\,\hat{\sigma}\sqrt{\sum_{i=1}^{g}\frac{c_i^2}{n_i}}$$

  then the probability that **all** sample contrasts include their true population values is **exactly** $1-\alpha$.

- We effectively compare each contrast t-statistic to the $\sqrt{(g-1)F}$ distribution.
- Any which exceeds *that* critical value is significant in this "simultaneous" sense.
- The *smallest* such p-value is the F-test p-value!

Scheffé's simultaneous confidence intervals using emmeans

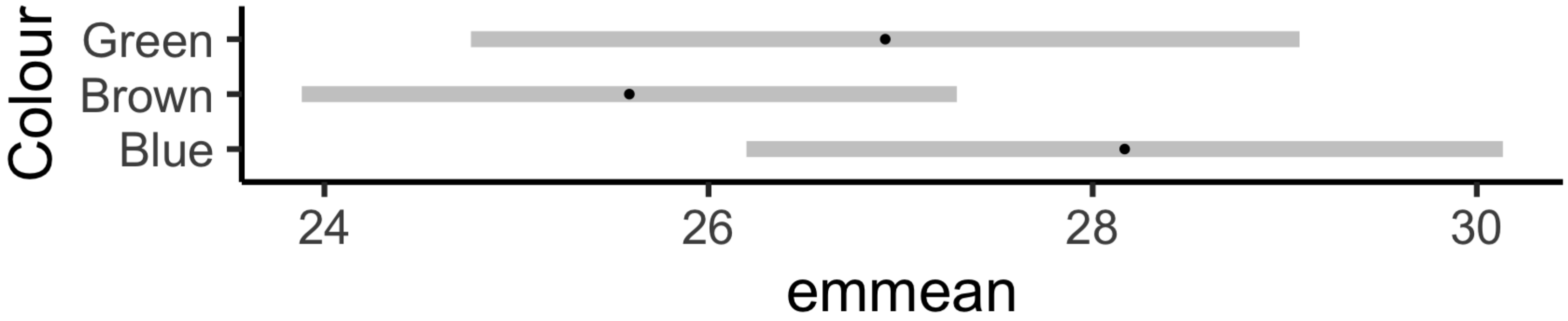
```
confint(pairs(flicker_em, adjust = "scheffe"))
```

```
##  contrast      estimate    SE df lower.CL upper.CL
##  Blue - Brown      2.58 0.836 16    0.326     4.83
##  Blue - Green      1.25 0.937 16   -1.279     3.77
##  Brown - Green    -1.33 0.882 16   -3.711     1.05
##
## Confidence level used: 0.95
## Conf-level adjustment: scheffe method with rank 2
```

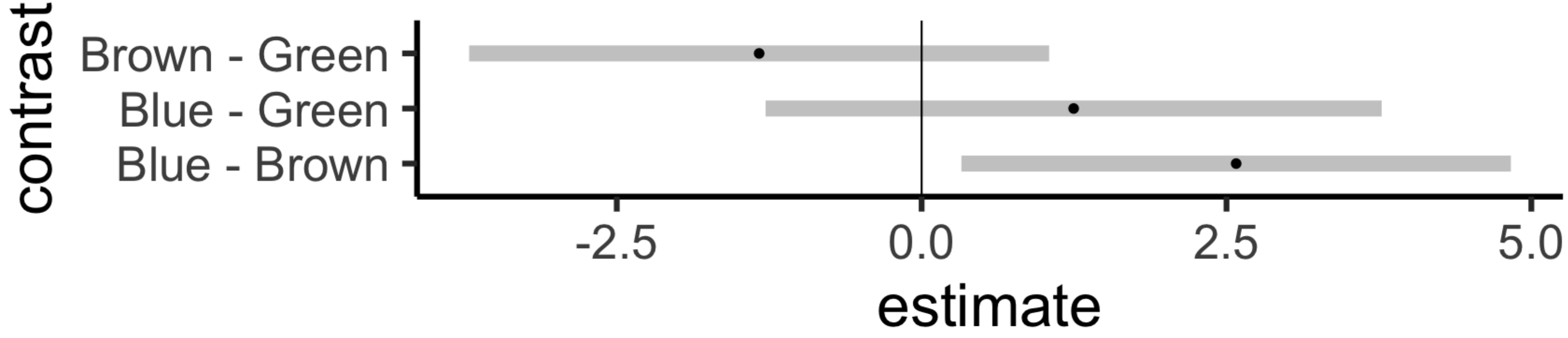
```
test(pairs(flicker_em, adjust = "scheffe"))
```

```
##  contrast      estimate    SE df t.ratio p.value
##  Blue - Brown      2.58 0.836 16   3.086  0.0238
##  Blue - Green      1.25 0.937 16   1.331  0.4319
##  Brown - Green    -1.33 0.882 16  -1.511  0.3442
##
## P value adjustment: scheffe method with rank 2
```

```r
confint(flicker_em, adjust = "scheffe") %>% plot(colors = "black")
```




```r
confint(pairs(flicker_em, adjust = "scheffe")) %>% plot(colors = "black") +
  geom_vline(xintercept = 0)
```



Concluding remarks

- The ANOVA F-test alone may or may not address the important scientific questions in each example.
- Depending on the context, a test based on the most significant contrast(s) may be *more* useful than a straight F-test.
- Bonferroni procedures are in general *conservative* i.e. p-values and confidence intervals may be larger than they really need to be.
    - alternative methods which may be more accurate i.e. less conservative exist: e.g. Tukey's method.
- Any contrasts must be decided upon **before looking at the data**. Otherwise we are **data snooping**.
- If we "snoop" until we find a significant contrast, we *must take account of that*:
    - Scheffé's method permits unlimited data snooping
    - If we snoop only across k fixed contrasts e.g. all pairwise comparisons, we can use the Bonferroni method to adjust for that (but for large k Tukey's method or Scheffé's method may give smaller intervals).

References

Lenth, R. (2018). *emmeans: Estimated Marginal Means, aka Least-Squares Means*. R package version 1.2.3. URL: https://CRAN.R-project.org/package=emmeans.