1-Factor GLM Homework Answers

Crispin Jordan

22/09/2021

Question 1 - Caffeine

The data for this question address the question of whether caffeine produced in nectar affects the amount of nectar collected by pollinators.

Before we begin to analyze any dataset we much first determine whether the data meet the assumptions of our expected analysis. So, what kind of analysis do we anticipate for the `Caffeine` dataset? The Caffeine experiment has four treatments: `50a`, `100a`, `150a`, and `200a`, which specify the four levels of caffeine examined in the experiment. The dataset also includes a series of measurements that were calculated as *difference* in the amount of nectar consumed when a pollinator was offered a flower with caffeine (either 50, 100, 150 or 200 ppm (parts per million)) and the amount of nectar consumed from a 'control' flower, which had no caffeine. Due to the fact that each measurement is a *difference* between two values (with a specified caffeine level vs. no caffeine), we can analyse these data from two perspectives. From the **first perspective**, we can view each measurement as a measure of preference (with vs. without caffeine) and **aim to compare preference levels between the four caffeine treatments.** From this perspective, it would be natural to analyze these data with a 1-factor general linear model. From a **second perspective**, we might be interested in **whether the average difference (i.e., the average datapoint) estimated for a given caffeine treatment differs from zero. If it does, this indicates that pollinators tended to remove different quantities of nectar from caffeinated vs. un-caffeinated flowers**. We can address this perspective by obtaining 95% CI's for the mean value of each treatment, and determine whether zero (i.e., no difference) is a plausible value. If the mean value (i.e., the mean difference between caffeine vs. no caffeine) for a given treatment is close to zero (or if the 95% CI's) include zero, we may conclude that we have little evidence that pollinators can detect a difference between the flowers with vs. without caffeine for the given treatment.

We can use a 1-factor glm to address both perspectives. Before we proceed, however, we need to check whether the data meet two important assumptions: random allocation to treatments and independence of data within treatments. If the data do not meet these assumptions, we cannot analyze the data with 1-factor glm. (Moreover, if subjects were not assigned randomly to treatments then we cannot analyze these data with any method and expect to obtain trustworthy results).

- **Randomization**: Positions of caffeinated vs. uncaffeinated flowers were altered randomly. Subjects (honeybees) experienced all treatments, so we needn't worry about randomly allocating subjects to treatment levels. We're satisfied that the data meet the assumption of randomization.
- **Independence**: In this experiment, the question of independence is a bit complicated. To save space (and save you a lot of reading), I will simply say at this point that the data meet the assumption of independence. **Please continue reading if you're interested in the details; if not, you can skip the rest of this paragraph.** The researchers created 5 'stations' where honeybees could visit the two flower types ((un)caffeinated flowers). Each station had both flower types, and all 5 stations presented the same treatment level (e.g., `50a`) at the same time on a given day. The researchers presented a different treatment level on each day. Stations are independent from one another because the bees only tended to visit a single station. *(i.e., Data from each station results from a different set of subjects - the authors explain that this is not 100% true because some bees visit more than one station, and this is a failing with the experiment. We will ignore this only for teaching purposes).* Therefore, the data *within* treatment levels are independent, and this is what matters. The data *among*

> treatment levels are not independent, because the bees at each station visited all treatment levels; but this kind of non-independence does not interfere with a 1-factor glm. You can find the paper here: https://link.springer.com/content/pdf/10.1007/s10886-005-8394-z.pdf

Now let's begin our analysis.

We beging by importing the data:

```
caf <- read.table("caffeine.csv", header = TRUE, sep = ',')
```

We should always start our analysis by inspecting the dataset, even if we collected the data and created the dataset file, ourselves. When we created the dataset, ourselves, we shoudl check the spreadsheet to inspect for any obvious error; we also do this by plotting the data (which we will do shortly)! If we're analyzing data that someone else collected, we need to familiarize ourselves with the dataset before we can begin.

Let's start by simply examining the entire dataset. We do this by entering the name of the object that contains the data:

```
caf
```

```
##    ppmCaffeine consumptionDifferenceFromControl
## 1          50a                            -0.40
## 2         100a                             0.01
## 3         150a                             0.65
## 4         200a                             0.24
## 5          50a                             0.34
## 6         100a                            -0.39
## 7         150a                             0.53
## 8         200a                             0.44
## 9          50a                             0.19
## 10        100a                            -0.08
## 11        150a                             0.39
## 12        200a                             0.13
## 13         50a                             0.05
## 14        100a                            -0.09
## 15        150a                            -0.15
## 16        200a                             1.03
## 17         50a                            -0.14
## 18        100a                            -0.31
## 19        150a                             0.46
## 20        200a                             0.05
```

The dataset has two columns, `ppmCaffeine`, which indicates which treatment a measurement came from, and `consumptionDifferenceFromControl`, where we find the measurments of the difference in consumption in flowers with caffeine vs. a control that lacked caffeine. We also see that the dataset is pretty small: we have only 20 observations, total, for 4 treatments (5 measurements per treatment, on average). Small datasets like this are more difficult to analyze for a few reasons. One reason, as we'll see below, is that it becomes harder to assess assumptions with small datasets. Other than a small sample size, no obvious problems jump out.

We would also like to know how **R** *sees* these data. What do I mean by this? I'll show you using the `str()` function:

```
str(caf)
```

```
## 'data.frame':    20 obs. of  2 variables:
##  $ ppmCaffeine                     : chr  "50a" "100a" "150a" "200a" ...
##  $ consumptionDifferenceFromControl: num  -0.4 0.01 0.65 0.24 0.34 -0.39 0.53 0.44 0.19 -0.08 ...
```

This output reveals how **R** has classified the data in each column. We see that **R** currently treats the data in the column, `consumptionDifferenceFromControl`, as a number (`num`). This is good: this is what we want. However, we see that **R** currently views the data in column, `ppmCaffeine`, as a *character* variable (see `chr`). We want, instead, for these data to be a `factor`. Usually it is OK to run analyses of factors when the data are classified as `chr`, but I have heard of occasions where this problems arise. Therefore, to be safe, we'll convert the data in the column, `ppmCaffeine` into a factor. We do this with the `factor()` function:

```
caf$ppmCaffeine <- factor(caf$ppmCaffeine)
```

Now, let's check whether we have effectively converted the data in the first column into a factor:

```
str(caf)
```

```
## 'data.frame':    20 obs. of  2 variables:
##  $ ppmCaffeine                     : Factor w/ 4 levels "100a","150a",..: 4 1 2 3 4 1 2 3 4 1 ...
##  $ consumptionDifferenceFromControl: num  -0.4 0.01 0.65 0.24 0.34 -0.39 0.53 0.44 0.19 -0.08 ...
```

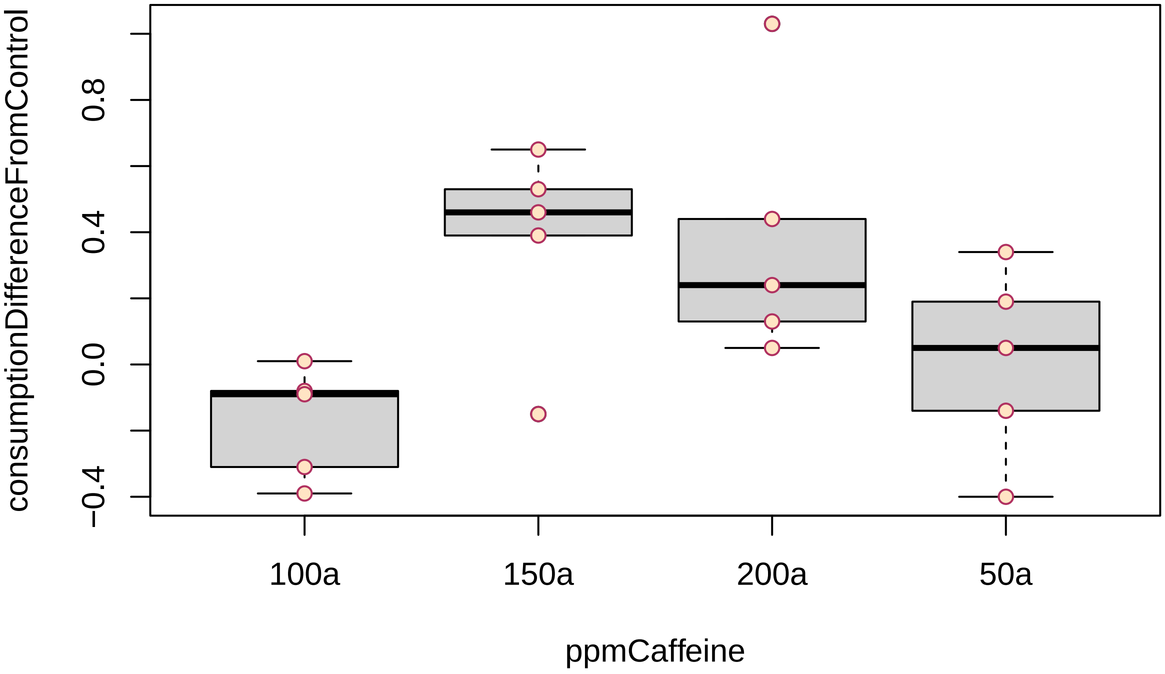
Looks good! Out output now indicates that `ppmCaffeine` is a factor with four levels, which is exactly what we expect.

Now that we're familiar with the dataset and are convinced that the dataframe is set up as we require, we can start our analysis.

We always begin our analysis by plotting the data.

In order to plot our data, we need to decide which column contains the *dependent* variable (which we plot on the y-axis) and which column contains the *independent* variable (x-axis). Our hypothesis is that the amount of caffeine will determine how much nectar a pollinator consumes. In other words, we expect that nectar consumption *depends* on the caffeine treatment. Therefore, we conclude that `consumptionDifferenceFromControl` is the *dependent* variable and `ppmCaffeine` is the *independent* variable. We place the independent variable on the left of the tilda (~) in our function to plot the data (`boxplot` and `stripchart`), and we place the independent variable to the right of the ~. Therefore, our plotting functions look like this:

```
boxplot(consumptionDifferenceFromControl ~ ppmCaffeine, data = caf)
stripchart(consumptionDifferenceFromControl ~ ppmCaffeine, data = caf, add = TRUE,
           vertical = TRUE, pch = 21, col = "maroon", bg = "bisque")
```



What do we see? Well, the fist thing to notice is that, as expected, the small sample size makes it difficult to anticipate any problems with assumptions. Let's look at the figure in detail:

- **Outliers?** Plotting our data can help us spot unusual datapoints, or so-called, 'outliers'. Outliers could represent true, real measurements that happend to be unusual, our outliers can represent errors when typeing the data into a spreadsheet. The latter case is obvious when we find values that are truly impossible; for example, if we measured a subject's mass and recorded it as negative (this violates physics). In our case, we see slightly unusual measurements in the two middle treatments (`150a` has an unusually low measurement, and `200a` has an unusually high measurement). But, with so few data it is difficult to discern whether these are truly 'unusual'. So, we'll proceed, assuming that there are not obvious outlies, but we'll grumble to ourselves abotu low sample sizes.
- **Normality?** We can examine the shape of the boxplots (and individual values) to get a preliminary sense of whether the data are likely to meet the assumption of normality. To my eye, the boxplots all look fairly symmetrical, which is consistent with normality. But again, it is hard to tell with so few data. Please note that, even when we assess boxplots to check 'normality', we must still formally inspect the residual plots for a better assessment of normality. We inspect the boxplots simply to set our expectations for what we'll find later in our analysis. **This is useful because, if we find that our conclusions later in the analysis do not match our expectations from inspecting a plot of the data, this can signal that we've made a mistake. In this case, we should check our work. Often, when discrepancies occur between expectations and what we actually find, this occurs due to forming poor expectations (this is easy to do). But it is still wise to check.**
- **Equal variance?** Again, small sample size complicates this assessment. To my eye, I woudl guess that variance is likely equal. But we face the same problems as we did when assessing outliers.
- **Expected differences between treatments?** It is always useful to use the plot to guess the size of expected differences between treatments; we can compare our guesses to the model output. Again, we do this as a check of whether we've made any big errors during our analysis. Let's compare each treatment against the first treatment on the plot, `100a`.
  - I'll guess that the mean of `100a` equals (approximately) `-0.15`.

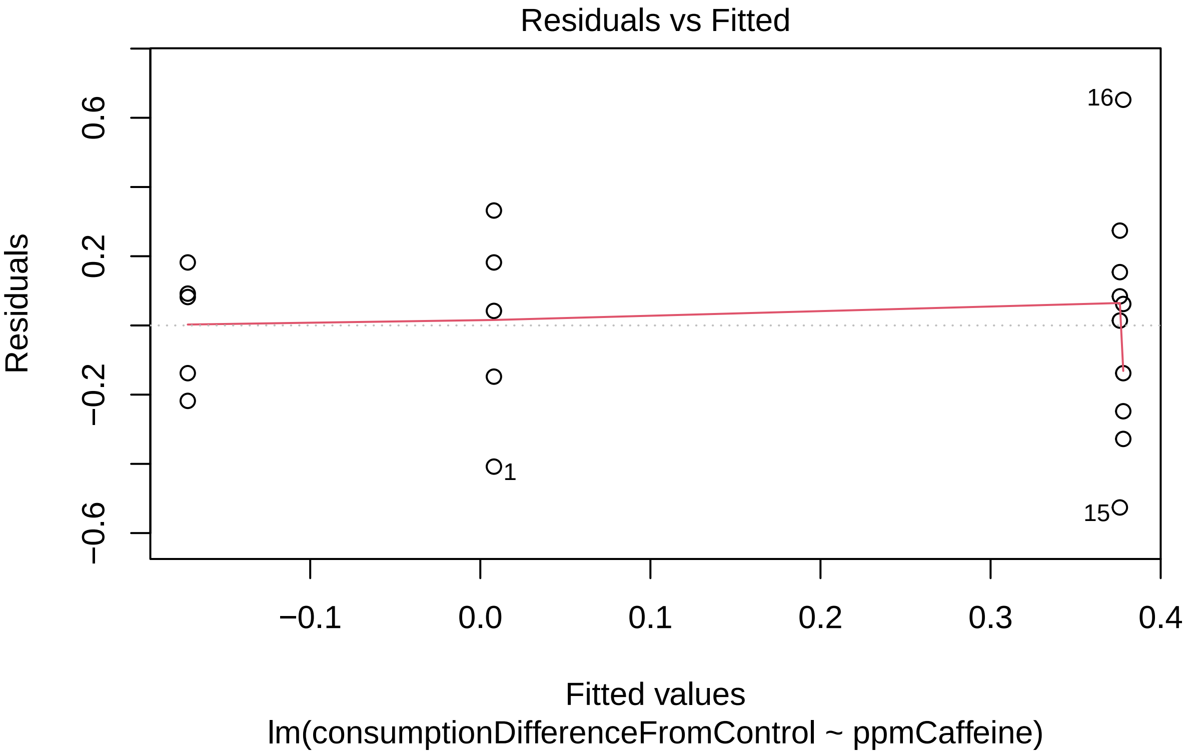
  - If the mean of `150a` is about positive 0.3 (guessing again), then we expect that the difference between `100a` and `150a` equals about 0.45.
  - Guessing again, I think the mean of `200a` equals 0.40. Therefore I anticipate a difference between `100a` and `200a` of 0.55.
  - Finally, the mean of `50a` looks close to 0.1. Therefore I anticipate a difference of 0.25 between `100a` and `50a`.

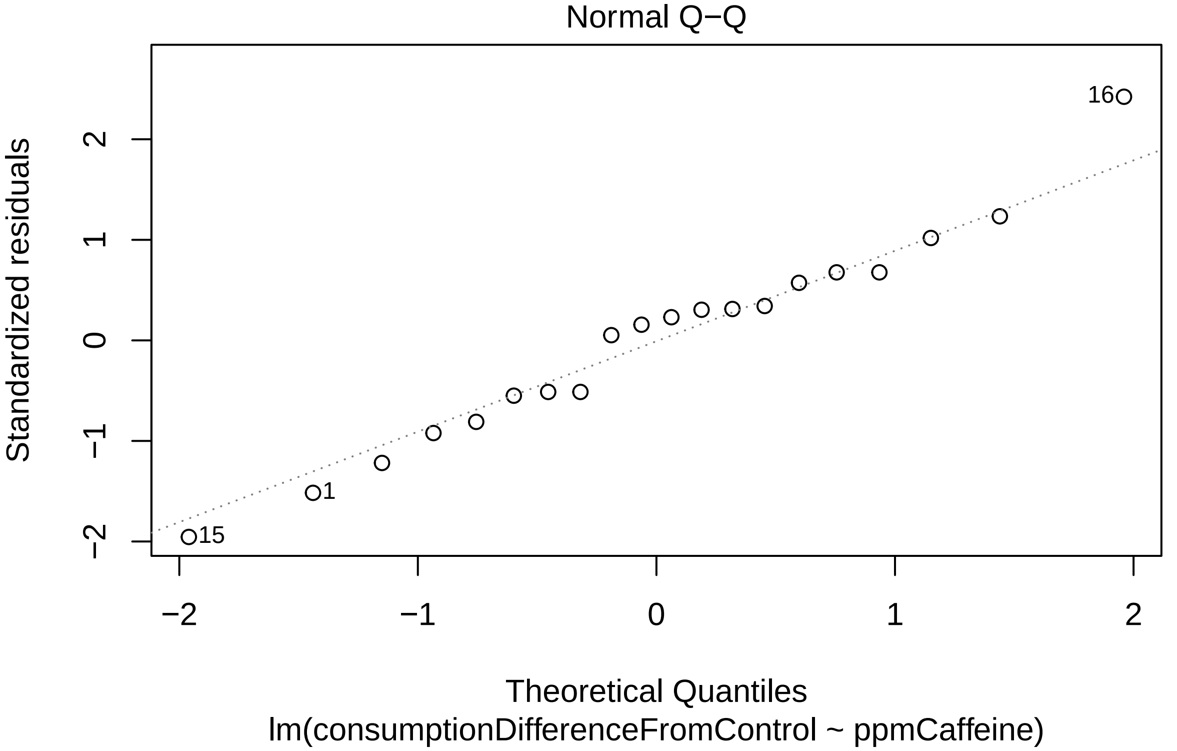
With these expectations in mind, let's proceed with our analysis. We need to formulate our model. To do so, we place the dependent variable (`consumptionDifferenceFromControl`) to the left of the tilda, and the independent variable (`ppmCaffeine`) on the right (exactly as we did when plotting the data). We use the `lm()` function for our 1-factor glm. Our model looks like this:

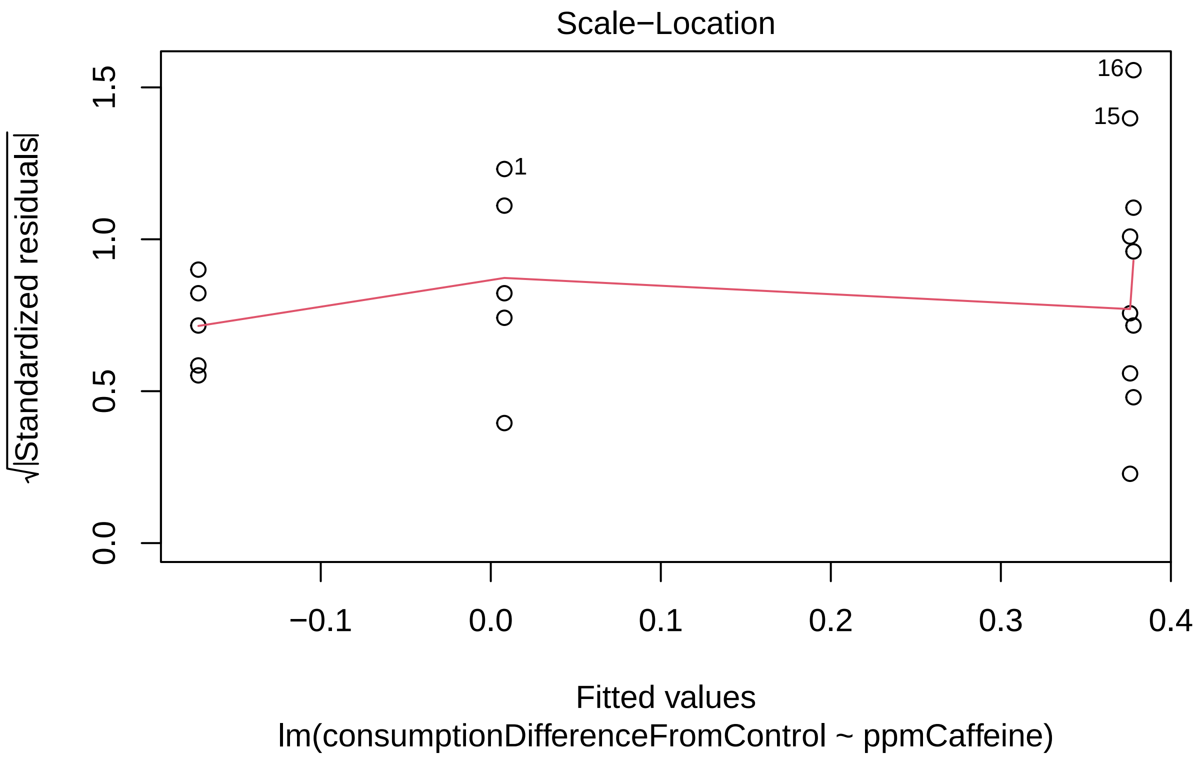
```r
caf.lm <- lm(consumptionDifferenceFromControl ~ ppmCaffeine, data = caf)
```

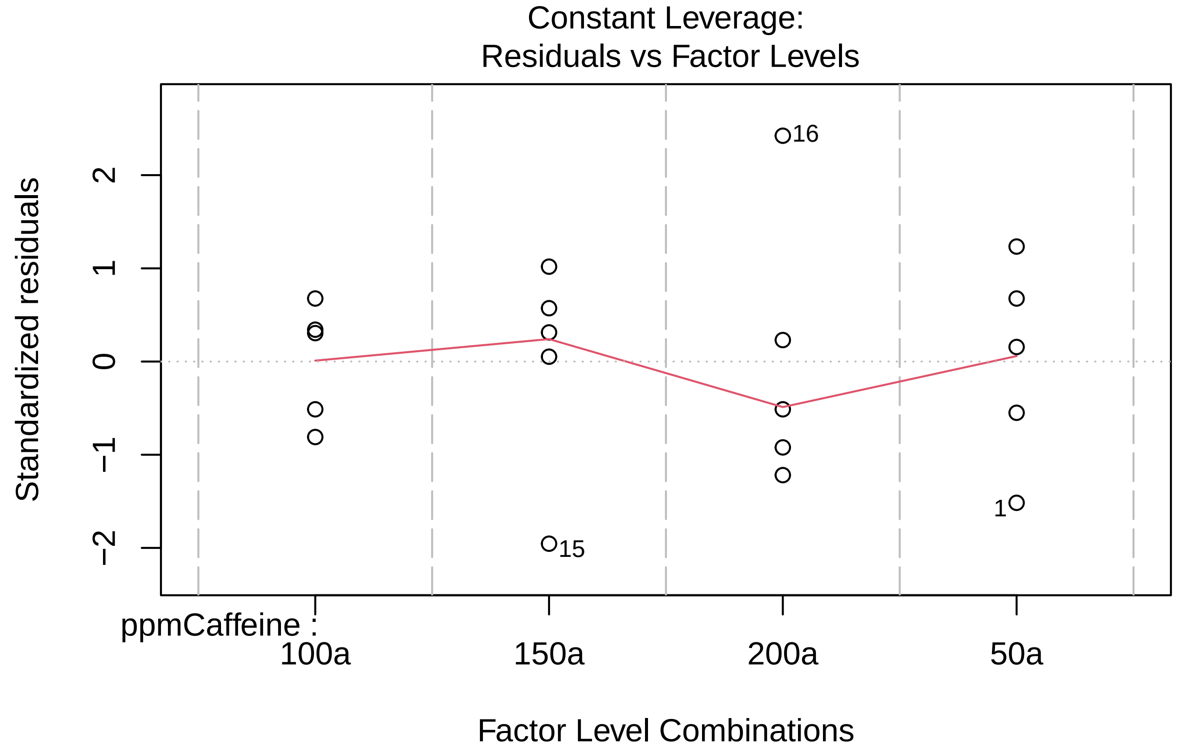
After running the model, we always check our assumptions next. We do this as follows:

```r
plot(caf.lm)
```


Residuals vs Fitted
Residuals
0.6
0.2
−0.2
−0.6
16
1
15
−0.1
0.0
0.1
0.2
0.3
0.4
Fitted values
lm(consumptionDifferenceFromControl ~ ppmCaffeine)


Normal Q−Q
Standardized residuals
2
1
0
−1
−2
16
1
15
−2
−1
0
1
2
Theoretical Quantiles
lm(consumptionDifferenceFromControl ~ ppmCaffeine)





How do we interpret this output? Let's consider the four plots:

- **Plot 1:** This plot provides one way to assess the assumption of equal variance. Ideally (when variance is equal), we would not see any pattern in the points, they would be evenly distributed above and below the dotted line, and the red line would be straight. To my eye, it looks like the points on the left side of the plot are closer together (vertically), and the points become more spread out (vertically) as we proceed to the right of the plot. This would suggest a problem with variance. However, this trend is slight, and it is difficult to confidently claim that this trend exists due to the low sample size. We also see that the red line is mostly straight, except at the far right. Overall, this plot provides ambiguous evidence for equal variance, as we anticipated from the boxplots. There's just not enough data to confidently check this assumption.

- **Plot 2:** This qq plot allows us to check the assumption of normally distributed residuals. In our case, the data fall nicely along the dotted line, overall. This indicates that the data meet the assumption of normality. Recall that this was difficult to anticipate when we examined the data in the boxplots. The qq plot provides a much more reliable assessment of this assumption.
- **Plot 3:** Like plot 1, plot 3 allows us to check the assumption of equal variance. In this case, the data meet the assumption of equal variance when the red line is flat and horizontal, the points lie above and below the red line equally. In our case, the red line is certainly not perfectly flat due to the sudden upwards shift at the right. However, this amount of "wobble" in the red line is perfectly expected with such a small sample size. Therefore, this plot provides no evidence that the variances differ between treatment levels. Good!
- **Plot 4:** We will ignore this plot.

OK. We've now checked our assumptions and we're happy that they are all met, we will carry on.

Our next step is to examine a summary of the output. This summary indicates differences between some of the treatment combinations. We obtain the summary with:

```
summary(caf.lm)
```

```
##
## Call:
## lm(formula = consumptionDifferenceFromControl ~ ppmCaffeine,
##     data = caf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5260 -0.1655  0.0520  0.1610  0.6520
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -0.1720     0.1345  -1.278   0.2193
## ppmCaffeine150a    0.5480     0.1903   2.880   0.0109 *
## ppmCaffeine200a    0.5500     0.1903   2.891   0.0106 *
## ppmCaffeine50a     0.1800     0.1903   0.946   0.3582
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3008 on 16 degrees of freedom
## Multiple R-squared:  0.4393, Adjusted R-squared:  0.3341
## F-statistic: 4.178 on 3 and 16 DF,  p-value: 0.02308
```

That's a lot of output. Let's focus on the part called, `Coefficients:`. Notice that we have four rows of information: `(Intercept)`, `ppmCaffeine150a`, `ppmCaffeine200a` and `ppmCaffeine50a`. The only treatment not listed is `ppmCaffeine100a`; therefore, `(Intercept)` provides information about `ppmCaffeine100a`. (Notice that treatment, therefore, are listed in that same order as in our boxplot; alphanumerically.) Let's not interpret each row:

- `(Intercept)`: This row provides the mean value for treatment `ppmCaffeine100a`, provided under `Estimate`. This value (-0.1720) closely matches our guess of -0.15, which is reassuring. The column `Std. Error` provides the standard error for this estimate of the mean of level, `ppmCaffeine100a`. The values in columns `t value` and `Pr(>|t|)` provide statistics for a test of whether the `Estimate` (-0.1720) differs from zero. The p-value is large (0.2193) indicating that we have no evidence that the `Estimate` differs from zero. We will return to this when we use the `emmeans` function, below.

- `ppmCaffeine150a`: This row provides information about the estimated difference between the mean of the `Intercept` (i.e., the mean of treatment `ppmCaffeine100a`) and the mean of level, `ppmCaffeine150a`. The `Estimate` indicated that this difference equals 0.5480. Recall that when we examined the boxplot, we predicted a difference of 0.45 for these two treatments. Our guess was not too bad, which is reassuring (again). The column, `Std. Error`, nor provides the standard error for the *difference* between `(Intercept)` and the mean of `ppmCaffeine150a`. Likewise, columns `t value` and `Pr(>|t|)` provide statistics for the comparison of these two means. The p-value (0.0109) indicates that we have *moderate* or *suggestive* evidence of differences between these two means (see below for more explanation).
- `ppmCaffeine200a`: Following the logic from above, this row provides information about the comparison between the `(Intercept)` and the mean of level, `ppmCaffeine200a`, which equals 0.5500. Again, this matches our guess of 0.55, exactly. (Not bad!)
- `ppmCaffeine50a`: Finally, we guessed a difference between `(Intercept)` and `ppmCaffeine50a` of 0.25. It is reassuring that the `Estimate` of this difference provided in this row equals 0.1800, which is not a great difference.

Overall, our predictions match the output fairly well, especially given that our guesses were admittedly poor due to the small amount of data. This is reassuring, and we can confidently continue our analysis.

We will now obtain an ANOVA table to determine the statistics for our overall test of differences among groups. We do so like this:

```
anova(caf.lm)
```

```
## Analysis of Variance Table
##
## Response: consumptionDifferenceFromControl
##             Df Sum Sq Mean Sq F value  Pr(>F)
## ppmCaffeine  3 1.1344 0.37814  4.1779 0.02308 *
## Residuals   16 1.4482 0.09051
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Three values interest us:

- `Df` (which equal 3 and 16; we report them both);*
- `F value`, which equals 4.1779; we will report this value;
- `Pr(>F)`, which is our p-value, equaling 0.02308. How should we interpret this p-value?

How do we interpret the p-value?

- It used to be convention that p-values less than 0.05 were considered 'significant' effects and those above 0.05 were 'non-significant'. In 2019, the *American Statistical Association* (ASA) decided that this practice led to too many poor conclusions and that the concept of 'statistical significance' should be abolished. See here: https://www.tandfonline.com/doi/full/10.1080/00031305.2019.1583913
- The ASA suggests p-values be interpreted along a 'sliding scale', where small p-values constitute 'strong' evidence for an effect and large p-values provide 'weak' evidence for an effect. But how small should a p-value be to provide 'strong' evidence?
- Benjamin et al. (2018; DOI: 10.1038/s41562-017-0189-z) argue that p-values near 0.005 and smaller constitute "substantial" or "strong" evidence for an effect. Publications that adopt this perspective (e.g., https://ecoevorxiv.org/n4g5z/) may suggest that p-values between 0.005 and 0.05 are 'suggestive' of effects; similarly, I sometimes say that p-values near 0.05 provide 'moderate' evidence for an effect. We will adopt this perspective when interpreting p-values.
- The ASA also cautions against over-relying on p-values to interpret results; they suggest using other sources of evidence, e.g., effect sizes (which we use, below).

The p-value equals 0.02308, which is *not* strong evidence for an effect; it provides 'moderate' or 'suggestive' evidence for an effect. With this in mind, we will now conduct post-hoc tests to examine evidence for differences between treatments; more important, we will estimate effect sizes.

We will use functions in the `emmeans` library for our post-hoc tests. Open the library like this (or install it first, using the `install.packages()` function if you do not have it):

```
library(emmeans)
```

We examine our effect sizes and post-hoc tests in two steps in `emmeans`. First, we calculate the mean values for each treatment. We do this as follows:

```
caf.emmeans <- emmeans(caf.lm, "ppmCaffeine")
```

We used the `emmeans` functions to calculate means for each level of `ppmCaffeine`. We first provided the name of the object that contained out model output (`caf.lm`), and then we specified the independent variable for which we wished to obtain estimates (`ppmCaffeine`). We stored the output of this work in the object, `caf.emmeans`. Let's examine these results now:

```
caf.emmeans
```

```
##  ppmCaffeine emmean    SE df lower.CL upper.CL
##  100a        -0.172 0.135 16  -0.4572    0.113
##  150a         0.376 0.135 16   0.0908    0.661
##  200a         0.378 0.135 16   0.0928    0.663
##  50a          0.008 0.135 16  -0.2772    0.293
##
## Confidence level used: 0.95
```

Let's examine this output: - The first column, `emmean`, provides the 'estimated marginal mean' for each group; **we will report this mean for each group** - `SE` provides the standard error for each estimated mean value; again, **we will report this mean for each group** - `df` indicated the degrees of freedom used to calculate the `SE`; we do not need to report this value, specifically, because we will already report this value when we report the `F value`. - `lower.CL` and `upper.CL`: These provide the upper and lower 95% confidence limits for each estimated mean; **It is optional to report these values, but they are often very useful**.

We will report many of these values when we describe our results. In this particular experiment, however, the 95% CI's for the mean of each group provide an extra degree of insight into the data. This is because, as you will recall, each measurement equals the *difference* between nectar consumed for flowers with caffeine vs. nectar consumed in flowers lacking caffeine: therefore, the 95% CI's for the mean of each treatment level allow us to evaluate evidence whether these differences are likely different from zero (i.e., whether pollinators have different preferences between caffeinated flowers vs. non-caffeinated flowers for a given treatment level). Due to this special use of 95% CI's in this particular experiment, we'll now take a moment to interpret them more closely.

Let's begin with treatment `100a`. Here, the 95% CI's range from -0.4572 to 0.113. Importantly, these 95% CI's span the value zero; this implies that 'zero' is a plausible difference between nectar consumed from caffeinated vs. non-caffeinated flowers in the `100a` treatment. In other words, we have no evidence that pollinators had a preference for one of the two flower types in this treatment. This situation might arise for a variety of reasons: one possible reason could be that pollinators could not perceive caffeine at this low concentration (100 ppm). We would need further experiments to determine whether this was true.

We come to a similar conclusion when examining the 95% CI's for the treatment, `50a`. However, the 95% CI's do not include zero for treatments `150a` and `200a`; therefore, we have evidence that pollinators extracted more nectar from one of the flower types than the other (caffeinated vs. non-caffeinated) when the caffeine concentration equaled 150ppm or 200ppm. These results can help us interpret the outcome of the experiment.

Now that we have obtained mean values for each treatment level and interpreted them, we will make comparisons among the treatment levels and estimate effect sizes. We do so using the `pairs()` function:

```
caf.pairs <- pairs(caf.emmeans)
caf.pairs
```

```
##  contrast    estimate   SE df t.ratio p.value
##  100a - 150a   -0.548 0.19 16 -2.880  0.0482
##  100a - 200a   -0.550 0.19 16 -2.891  0.0472
##  100a - 50a    -0.180 0.19 16 -0.946  0.7809
##  150a - 200a   -0.002 0.19 16 -0.011  1.0000
##  150a - 50a     0.368 0.19 16  1.934  0.2534
##  200a - 50a     0.370 0.19 16  1.945  0.2494
##
## P value adjustment: tukey method for comparing a family of 4 estimates
```

Each row of the output provides information about a different comparison between groups, specified by the column heading, `contrast`. For example, the first row provides information about the comparison between levels `100a` and `150a`, indicated by `100a - 150a`. This terminology, `100a - 150a` should be interpreted literally: this row provides results that come from subtracting the mean of level `150a` from that of `100a`. The column, `estimate`, provides the differences between the means. In the first row, the estimate is negative, because the mean of `150a` is larger than the mean of `100a` (and we get a negative number when we subtract a larger number from a smaller number). **We will report these differences in `estimate`, which are the effect sizes. We will also report the values in `SE`**, which are the standard errors of the estimated differences. `df` provides the degrees of freedom for this comparison between means, and `t.ratio` and `p.value` provide test statistics for the comparisons between these means. In this example, all of the p-values are large; the smallest p-values are near 0.05 (e.g., 0.0472) which only provide 'moderate' or 'suggestive' evidence of an effect. Therefore, we never have strong evidence for differences between any of the treatment levels. **NOTE that these p-values were adjusted via the Tukey method to account for multiple conparisons (i.e., the p-values come from a Tukey test).**

What can the effect sizes tell us? Personally, without knowing more about how pollinators respond to chemicals like caffeine I find it difficult to interpret these results. From a naive perspective, we might expect that pollinators' responses to the treatments would increase or decrease systematically with the caffeine concentration. However, that's not what we see. Based on the means alone (i.e., ignoring uncertainty given by SE's), the lowest mean value occurred with `100a`; the next highest mean was for `50a`, and then there was virtually no difference between the levels `150a` and `200a`. So, the 'story' told by these data is not as simple as 'mean values increase with the caffeine concentration'. Of course, we cannot ignore uncertainty in these estimates: for example, even though the mean of `100a` appears smaller than that for `50a`, the 95% CI's for this `contrast` suggests that the difference between these treatments may range from -0.946 (i.e., the mean of `50a` is greater than that of `100a` by 0.946) to 0.7809 (i.e., the mean of `100a` exceeds that is `50a` by 0.78). We notice that these 95% CI's include zero, indicating that we have little evidence that the mean values of `50a` and `100a` differ.

Overall, the effect sizes provide few easy insights into the pollinators' behaviour. We can explain this when we report the results.

Finally, we would like obtain 95% CI's for the effect sizes. We do so with:

```
confint(caf.pairs)
```

```
##  contrast    estimate   SE df lower.CL upper.CL
##  100a - 150a   -0.548 0.19 16   -1.092 -0.00362
##  100a - 200a   -0.550 0.19 16   -1.094 -0.00562
##  100a - 50a    -0.180 0.19 16   -0.724  0.36438
##  150a - 200a   -0.002 0.19 16   -0.546  0.54238
##  150a - 50a     0.368 0.19 16   -0.176  0.91238
##  200a - 50a     0.370 0.19 16   -0.174  0.91438
##
## Confidence level used: 0.95
## Conf-level adjustment: tukey method for comparing a family of 4 estimates
```

Now we have everything we need to report our results. We might report them as follows.

(Discuss assumptions) We analyzed the effect of caffeine concentration on the difference in nectar consumed

in caffeinated vs. non-caffeinated flowers for four levels of caffeine (50, 100, 150 and 200ppm) using a 1-factor GLM. The data met the assumptions of random allocation of subjects to treatments and independence, reflected in the experimental design. The model residuals indicated that the data met the assumptions of normally distributed residuals and equal variance.

(Provide overall test result.) Our analysis revealed moderate evidence that caffeine concentration affected nectar consumption (1-factor GLM, F(3,16) = 4.1779, p = 0.02308).

(Here, I describe the means of each treatment.) Estimated treatment means provide little evidence that pollinators removed different amounts of nectar from caffeinated vs. non-caffeinated flowers when caffeine concentrations were low (`50a` and `100a`) (Figure 1 *(always plot of your data, as we did at the top of this document.)*). The mean difference between nectar removal from caffeinated vs. non-caffeinated flowers equaled (mean `+/-` SE) 0.008 `+/-` 0.135; (95% CI's: -0.2772 to 0.293) and -0.172 `+/-` 0.135 (-0.4572 to 0.113) for levels, `50a` and `100a`, respectively. The 95% CI's provide little evidence that pollinators removed different quantities fo nectar from caffeinated vs. non-caffeinated flowers for these two treatment levels. On the other hand, 95% CI's provide suggest that pollinators removed different quantities of nectar from caffeinated vs. non-caffeinated flowers for levels `150a` and `200a`. Their means (`+/-` SE) are, respectively, 0.376 `+/-` 0.135 (0.0908 to 0.661) and 0.378 `+/-` 0.135 (0.0928 to 0.663).

(Discuss differences between factor levels) **Given that we have many comparisons to report, I would provide these results in a table. The table would include columns that indicate: i) the `contrast` being made (i.e., which levels are compared); ii) the `estimate` of that contrast; iii) the SE for this `estimate`; iv) the `df`; v) the `t.ratio`, vi) the `p.value`, and the 95% CI's for the estimated differences. These results are all found in the output of `pairs()` and `confint()`. The table should clearly state that the p-values and 95% CI's are adjusted for multiple comparisons via the Tukey method. We can call this Table, Table 1.** The mean values of each treatment level do not tend to increase with the concentration of caffeine (although we did not explicitly test for this tend). For example, the mean of `100a` tends to be lower than that for `50a`, but post-hoc Tukey comparisons provide little evidence that their means differ (Table 1). Therefore, little evidence suggests that the extent of the difference in nectar collection from caffeinated vs. un-caffeinated flowers differs between `50a` and `100a` levels. Similarly, we find little evidence that the mean of `150a` differs from that of level `200a` (Table 1). Only comparisons between level `100a` vs. `150a` or `200a` provide even moderate evidence for pairwise differences (Table 1).

Please note:

- We did not discuss the magnitude of the effects. The reason is simple: in this case, I do not know how to biologically interpret the magnitude of differences of differences.
- It is important to note whether the p-values and 95% CI's for post-hoc comparisons are adjusted for mutiple comparisons (and if so, how (e.g., Tukey method)).
- Not all researcher agree that analyses should correct for multiple comparisons. The reasons for this are beyond the scope of this exercise. If you decide to conduct post-hoc comparisons and measure effect sizes without correcting for multiple comparisons you can do so by adding `adjust = "none"` to the `pairs()` function: e.g., `caf.pairs.noAdjust <- pairs(caf.emmeans, adjust = "none")`

Question 2 - Aphids

Much of the explanation provided for answer of Question 1 will apply to Question 2. Therefore, we provide less commentary for Question 2.

As in Question 1, we must always consider whether the data are appropriate for a given analysis. This experiment involves three levels of one factor and we aim to compare the mean values among the three levels. Therefore, it would be natural to analyze these data with a 1-factor GLM. Do the data meet the assumptions of randomization and independence?

- **Randomization**: As far as I can tell, the original paper (and supplementary online materials) do not provide details to know whether the data meet the assumption of randomization. Disappointing.
- **Independence**: As for randomization, I could not find information to assess the assumption of independence. Disappointing.

OK. We d not know whether these assumptions are met. If we had been reviewers for this paper we would have demanded these details. We will continue with our analysis, even though we're deeply disappointed. (Sigh.)

Let's import the data:

```
aphid <- read.table("aphid.csv", header = TRUE, sep = ',')
```

Let's examine the data:

```
aphid
```

```
##    infectionStatus color
## 1         original  30.7
## 2         original  25.4
## 3         original  26.2
## 4         original  23.0
## 5         original  20.9
## 6         original  20.7
## 7         original  15.8
## 8         original  17.4
## 9         original  17.6
## 10        original  17.0
## 11        original  16.5
## 12        original  15.3
## 13      uninfected  25.2
## 14      uninfected  22.3
## 15      uninfected  18.5
## 16      uninfected  15.4
## 17      uninfected  15.3
## 18      uninfected  17.0
## 19      uninfected  16.6
## 20      uninfected  18.6
## 21      uninfected  19.0
## 22        infected  43.3
## 23        infected  42.3
## 24        infected  40.7
## 25        infected  41.2
## 26        infected  39.6
## 27        infected  39.5
## 28        infected  36.2
## 29        infected  36.2
## 30        infected  34.4
## 31        infected  30.7
## 32        infected  31.9
```

We see two columns: `infectionStatus` and `color`. `infectionStatus` indicates which of the three treatments a measure of `color` belongs to: `original`, `uninfected` or `infected`. We have about 10 observations for each level, which is an better than we had for Question 1. (Although, at least for Question 1 we knew the data were independent, but we do not know this for Question 2. Sigh (again).)

Let's examine the columns in more detail:

```
str(aphid)
```

```
## 'data.frame':    32 obs. of  2 variables:
##  $ infectionStatus: chr  "original" "original" "original" "original" ...
##  $ color          : num  30.7 25.4 26.2 23 20.9 20.7 15.8 17.4 17.6 17 ...
```

Currently, the column `infectionStatus` is a 'character' type variable. Let's change it to a `factor`:

```
aphid$infectionStatus <- factor(aphid$infectionStatus)
```

Let's check that our code worked:

```
str(aphid)
```

```
## 'data.frame':    32 obs. of  2 variables:
##  $ infectionStatus: Factor w/ 3 levels "infected","original",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ color          : num  30.7 25.4 26.2 23 20.9 20.7 15.8 17.4 17.6 17 ...
```

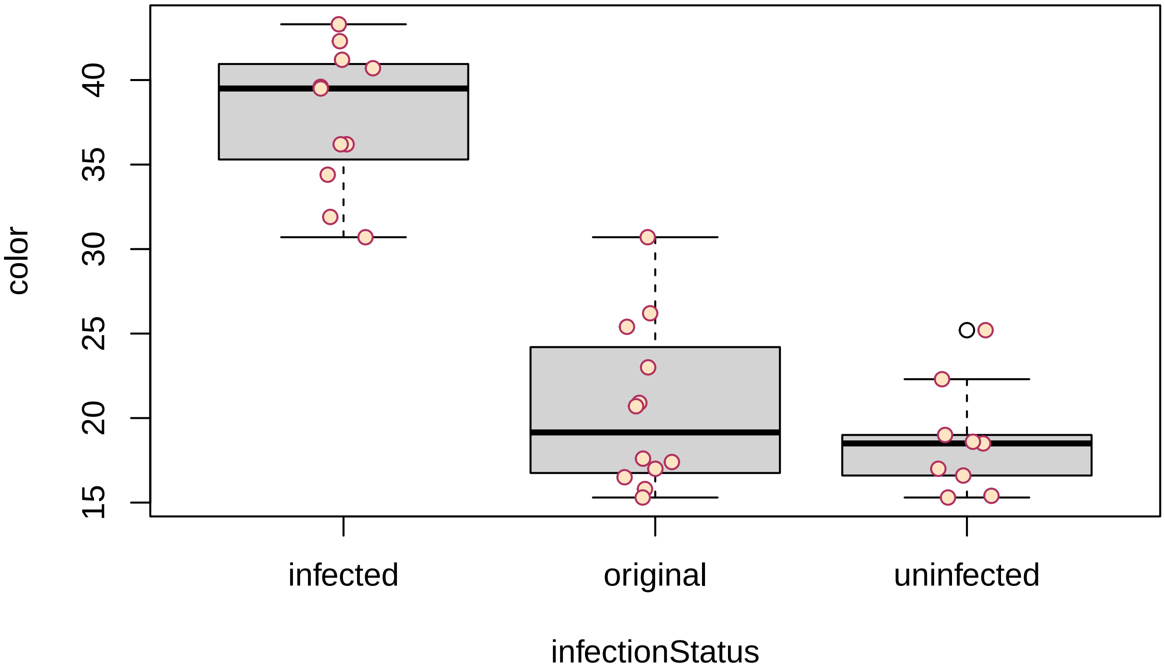
Yes! `infectionStatus` is now a `factor`.

Now that we're happy with the dataset, let's plot the data.

We hypothesize that `infectionStatus` affects `color`; i.e., we hypothesize that `color` depends on `infectionStatus`. Therefore, `color` is the *dependent* variable and we will place it on the left of the tilda (`~`) both when we plot the data and when we run our linrar model (`lm()`). Similarly, `infectionStatus` is the *independent* variable and we will place it to the right of the tilda in these functions.

Plot the data:

```
boxplot(color ~ infectionStatus, data = aphid)
stripchart(color ~ infectionStatus, data = aphid, add = TRUE, vertical = TRUE, method = "jitter", pch =
```



What do we see?

- **Outliers?** No obvious unusual datapoints. Nice.
- **Normality?** Boxplots are roughly symmetrical; we expect the data to meet the assumption of normality (we still need to check this by plotting the residuals.)
- **Equal variance?** The data are similarly 'spread out' for levels `infected` and `original`, but seem to be slightly closer together for `uninfected`. It is possible that the data might violate the assumption of equal variance.
- **Estimated differences?** The mean value of `infected` appears to be about 37. The mean of `original` appears to be about 20; therefore we expect the difference between the mean of `infected` and `original`

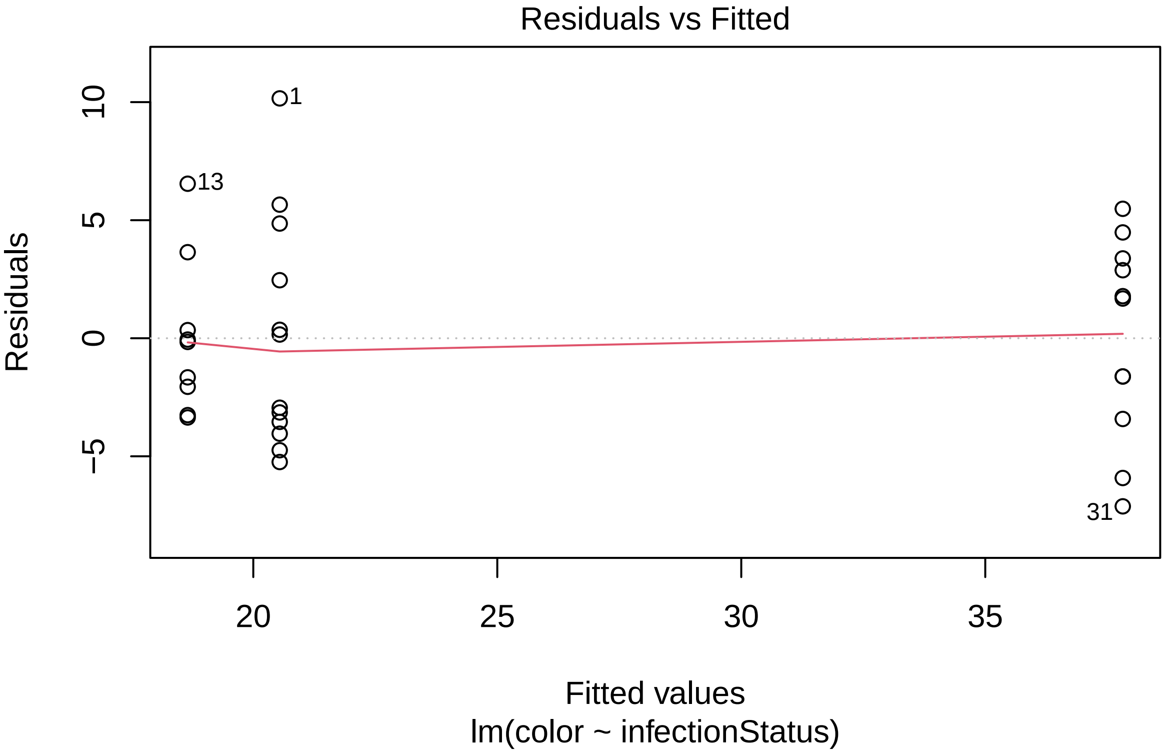
to be 20 - 37 = -17. Similarly, the mean of `uninfected` appears to be about 18, so we expect the difference between `uninfected` and `infected` to equal 18 - 37 = -19.

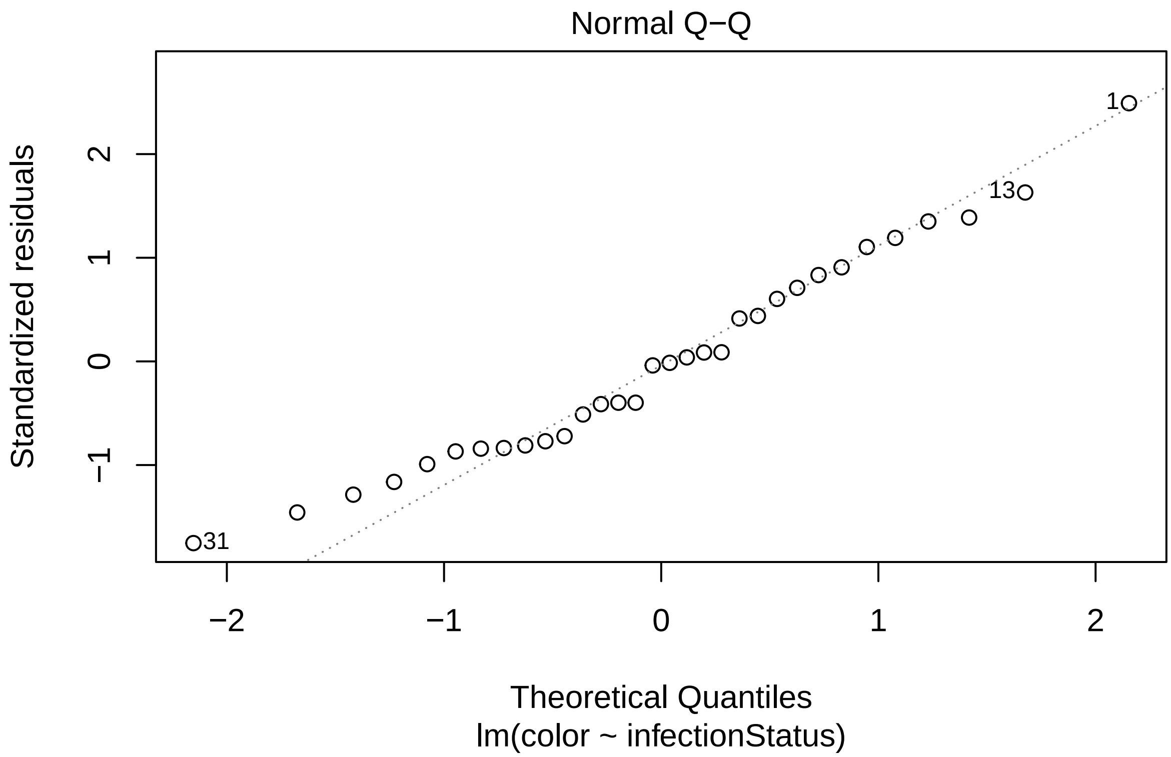
With these predictions in mind, let's run our model. We decided, above that `color` is the dependent variable. Therefore, our model looks like this:

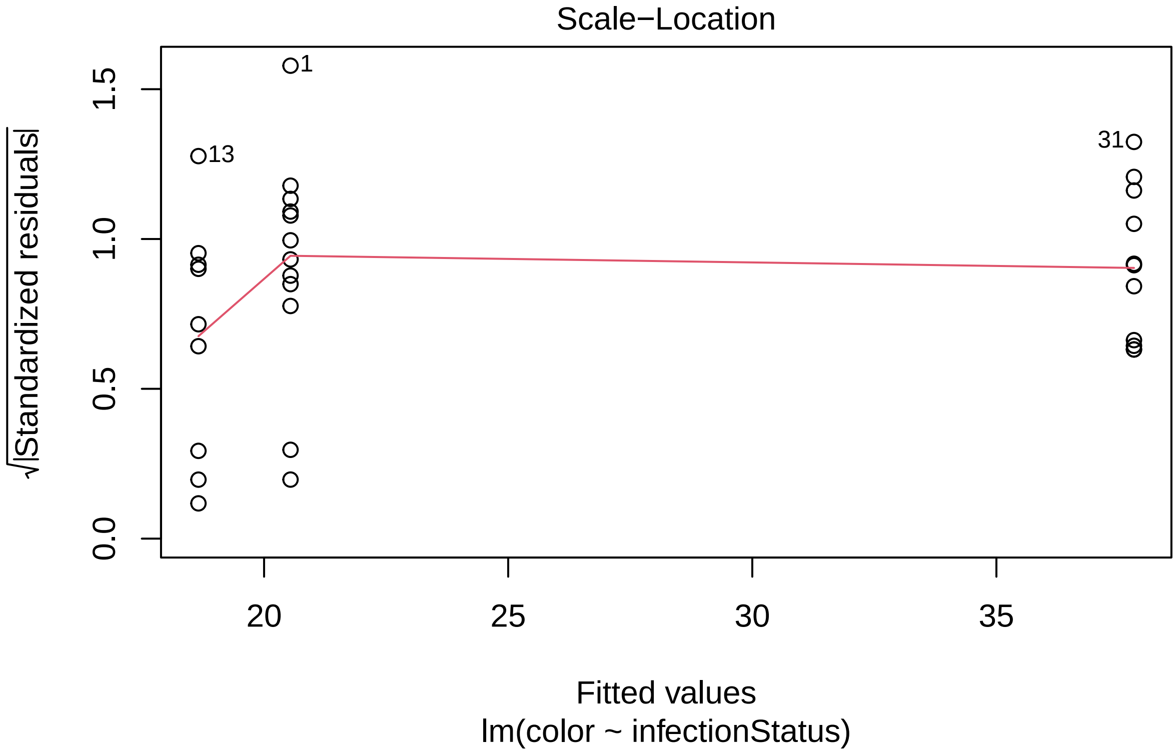
```
aphid.lm <- lm(color ~ infectionStatus, data = aphid)
```

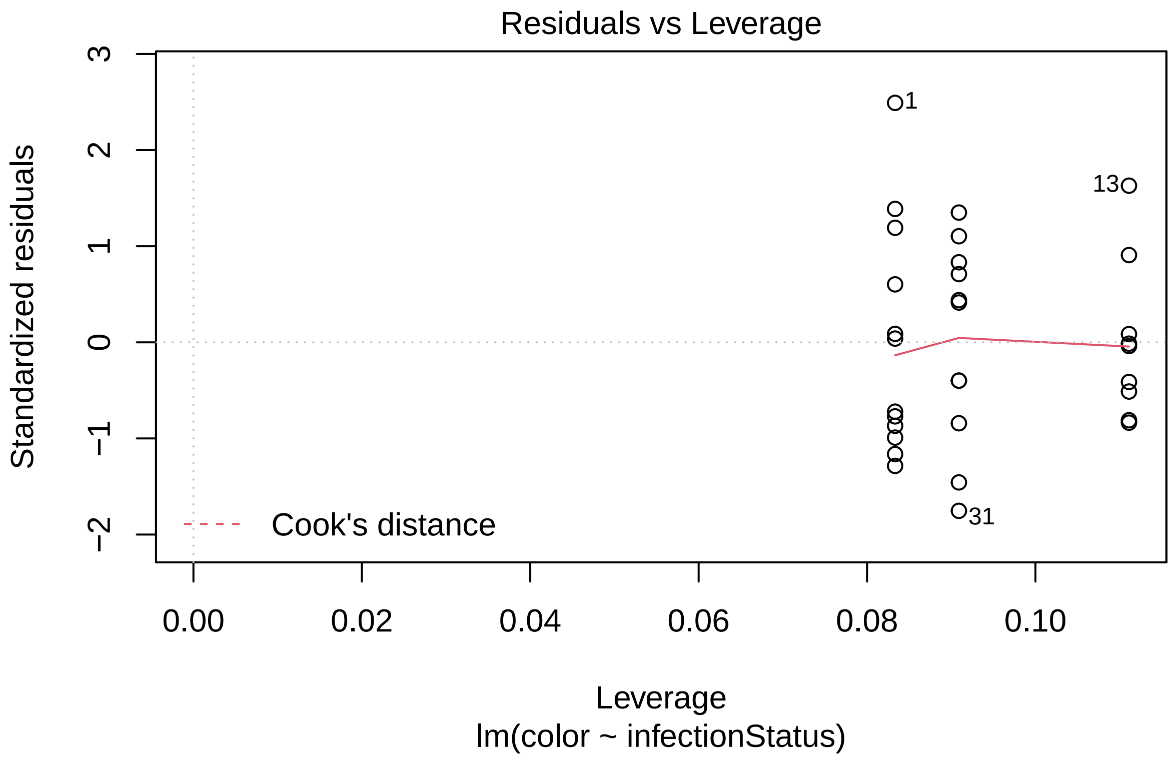
We always plot the data to test the assumptions of equal variance an normality **before** we look at the results:

```
plot(aphid.lm)
```




Normal Q−Q
Standardized residuals
−1
0
1
2
−2
−1
0
1
2
31
13
1
Theoretical Quantiles
lm(color ~ infectionStatus)


Scale−Location
$\sqrt{|\text{Standardized residuals}|}$
0.0
0.5
1.0
1.5
20
25
30
35
1
13
31
Fitted values
lm(color ~ infectionStatus)



We see:

- **Plot 1:** No strong pattern in the residuals; suggests data meet assumption of equal variance.
- **Plot 2:** Points generally fall along the line: suggests that the data meet the assumption of normally distributed residuals.
- **Plot 3:** The red line is not as flat as we'd like, but it is not too bad. Moreover, *if* there was a problem with equal variance, it would arise due to one level (`uninfected`) having unusually small variance. It turns out that having a single level with unusually small variance has less severe impact on conclusions than other possible situations (e.g., if one level had unusually high variance). Given that the plot looks alright, and given the pattern of variance, we'll say we're happy that the data meet the assumptions of equal variance.
- **Plot 4:** Skip this plot.

We're happy the data meet the assumptions. Let's look at the model estimates to see whether they match our expectations:

```
summary(aphid.lm)
```

```
##
## Call:
## lm(formula = color ~ infectionStatus, data = aphid)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.1182 -3.2806 -0.1056  3.0068 10.1583
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   37.818      1.284  29.453  < 2e-16 ***
## infectionStatusoriginal      -17.277      1.778  -9.719 1.26e-10 ***
## infectionStatusuninfected    -19.163      1.914 -10.011 6.43e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.259 on 29 degrees of freedom
## Multiple R-squared:  0.819,  Adjusted R-squared:  0.8065
## F-statistic: 65.59 on 2 and 29 DF,  p-value: 1.728e-11
```

Here, the `(Intercept)` represents the mean value of the level, `infected`; the two rows beneath provide the difference between the mean of `infected` and the means of `original` (second row) and `uninfected` (third row). Notice that the values in the column, `Estimate` match our predictions from the boxplots very well. This provides confidence in our analysis. Nice!

Now, let's obtain our test statistics:

```
anova(aphid.lm)
```

```
## Analysis of Variance Table
##
## Response: color
##                 Df  Sum Sq Mean Sq F value    Pr(>F)
## infectionStatus  2 2379.27 1189.64  65.595 1.728e-11 ***
## Residuals       29  525.95   18.14
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We note these values: `Df` = 2, 29; `F value` = 65.595; `Pr(>F)` = 1.728e-11. **We will report these values later.** The p-value is extremely small, indicateding that we have strong evidence that mean `color` differens among levels of `infectionStatus`.

We now want to examine comparisons between the treatment levels and examine effect sizes:

```
library(emmeans)
```

We begin by calculating the mean values of each level:

```
aphid.emmeans <- emmeans(aphid.lm, "infectionStatus")
aphid.emmeans
```

```
##  infectionStatus emmean   SE df lower.CL upper.CL
##  infected          37.8 1.28 29     35.2     40.4
##  original          20.5 1.23 29     18.0     23.1
##  uninfected        18.7 1.42 29     15.8     21.6
##
## Confidence level used: 0.95
```

We will report everything is the output above; note however that the degrees of freedom is the same as we obtained for the `F value` (the denominator degrees of freedom, 29), and we want to avoid presenting the degrees of freedom twice.

Notice that the mean values of `original` and `uninfected` are very similar and their 95% CI's overlap greatly (suggesting little evidence for a difference between them). On the other hand, the 95% CI's of `infected` do not overlap with the 95% CI's of the other two levels. This will be reflected in the pairwise comparisons:

```
aphid.pairs <- pairs(aphid.emmeans)
aphid.pairs
```

```
##  contrast              estimate   SE df t.ratio p.value
##  infected - original      17.28 1.78 29   9.719  <.0001
##  infected - uninfected    19.16 1.91 29  10.011  <.0001
##  original - uninfected     1.89 1.88 29   1.004  0.5801
```

```
##
## P value adjustment: tukey method for comparing a family of 3 estimates
```

Again, we will report all of the output, above (being careful to not repeat our reporting of the degrees of freedom). Notice that the p-values provide strong evidence that mean `color` of `infected` individuals differs from that of `original` and `uninfected` individuals, but little evidence that mean `color` differs between `original` and `uninfected` individuals.

Now we obtain 95% CI's for these effect sizes (i.e., differences between levels):

```
confint(aphid.pairs)
```

```
##  contrast              estimate   SE df lower.CL upper.CL
##  infected - original      17.28 1.78 29    12.89    21.67
##  infected - uninfected    19.16 1.91 29    14.44    23.89
##  original - uninfected     1.89 1.88 29    -2.75     6.52
##
## Confidence level used: 0.95
## Conf-level adjustment: tukey method for comparing a family of 3 estimates
```

We will report the 95% CI's from this output (the remaining output is identical to the output from `pairs()`).

Now we must report our results.

We analyzed the effect of `infectionStatus` on hue angle (`color`) with a 1-factor GLM; low hue angles correspond to red, and larger angles correspond to green. It is unclear whether the data meet the assumptions of (1) random allocation of treatments and (2) independence were met because the published article appeared to lack these details. Plots of the model residuals indicate that the data meet the assumptions of equal variance and normally distributed residuals.

Our analysis revealed strong evidence that mean `color` differs among levels of `infectionStatus` (1-factor GLM; F(2,29) = 65.595; p = 1.728e-11).

(We would include a figure like the boxplot and individual values, provided, above. We refer to this as Figure 1.)

Post-hoc Tukey tests that adjust p-values and 95% CI's of contrasts for multiple comparisons revealed little evidence for a difference in mean hue angle between the `original` level (mean `+/-` SE; 95% CI's: 20.5 `+/-` 1.23; 18.0 to 23.1; Figure 1) and the `uninfected` level (18.7 `+/-` 1.42; 15.8 to 21.6; Figure 1) (contrast estimate mean `+/-` SE: 1.89 `+/-` 1.88; t ratio = 1.004; p = 0.5801). This result is consistent with the hypothesis that injecting bacteria, per se, causes little change in hue angle (95% CI's suggest that injection increases hue angle by 2.75 to -6.52).

By comparison, strong evidence suggests mean hue angle of the `infected` level (37.8 `+/-` 1.28; 95% CI's 35.2 to 40.4) exceeded that of the `original` level (contrast estimate: 17.28 `+/-` 1.78; 95% CI's: 12.89 to 21.67; t ratio = 9.719; p < 0.0001) and the `uninfected` level (contrast estimate: 19.16 `+/-` 1.91; 95% CI's: 14.44 to 23.89; t ratio = 10.011; p < 0.0001). The 95% CI's for these two contrasts are similar, and suggest that infection increases hue angle (yielding more green) by approximately 13 to 23 degrees (alternatively, by approximately 50% to 100%) compared to mean `original` and `uninfected` levels.

Note that we used 95% CI's of effect sizes to describe a plausible range of effects.