Checking assumptions

Key

Mon Apr 18 11:40:52 2022

Assumption checking

Before now we have treated checking of assumptions as a pre-flight check; that is, something that we do with the data before doing any analysis. We will learn today to use **model criticism** instead, which is an interactive approach of fitting models and inspecting residuals to find a combination of model structure and data scale that meet the assumptions of GLM.

We will work with the bacteria dataset from your book.

First, import the data:

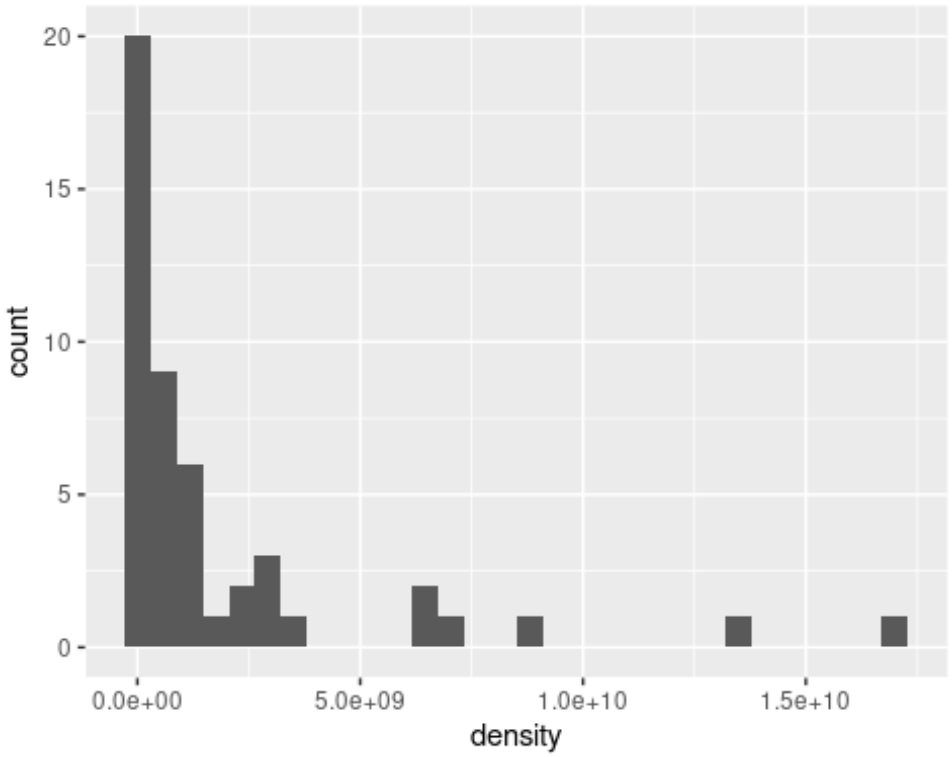
```
library(readxl)
read_excel("bacteria.xls","bacteria") -> bacteria
```

Convert the categorical predictors, day, sucrose, and leucine, to factors:

```
bacteria$day <- factor(bacteria$day)
bacteria$sucrose <- factor(bacteria$sucrose)
bacteria$leucine <- factor(bacteria$leucine)
```

Plot a histogram:

```
library(ggplot2)
ggplot(bacteria, aes(x = density)) + geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Question: do the data look normally distributed based on this histogram? How do you know?

No, they have a long right tail, so they look positively skewed.

First analysis

Fit a linear model of density as a funciton of day + sucrose + leucine - no transformation, and no interactions between the predictors.

```
dens.nointer.lm <- lm(density ~ day + sucrose + leucine, data = bacteria)
```

Get the predicted values from this model:

```
predict(dens.nointer.lm)
```

```
##           1           2           3           4           5           6
## -2098253021 -1660590708  1817703042 -1469504938 -1031842625  2446451125
##           7           8           9          10          11          12
##   282183396   719845708  4198139458  1926983396  2364645708  5842939458
##          13          14          15          16          17          18
## -1368711354  -931049042  2547244708  -739963271  -302300958  3175992792
##          19          20          21          22          23          24
##  1011725062  1449387375  4927681125  2656525062  3094187375  6572481125
##          25          26          27          28          29          30
##  -722763021  -285100708  3193193042   -94014937   343647375  3821941125
##          31          32          33          34          35          36
```

```
##  1657673396  2095335708  5573629458  3302473396  3740135708  7218429458
##          37          38          39          40          41          42
## -1255042771  -817380458  2660913292  -626294687  -188632375  3289661375
##          43          44          45          46          47          48
##  1125393646  1563055958  5041349708  2770193646  3207855958  6686149708
```

Question: what is the problem with these predicted values? They are supposed to be means for the combinations of day, leucine, and sucrose found in each row of the table, why do you know these are poor estimates for mean densities?

Many of them are negative numbers, and since densities can't be negative the mean of density can't be negative either.

Now calculate the residuals for the model:

```
residuals(dens.nointer.lm)
```

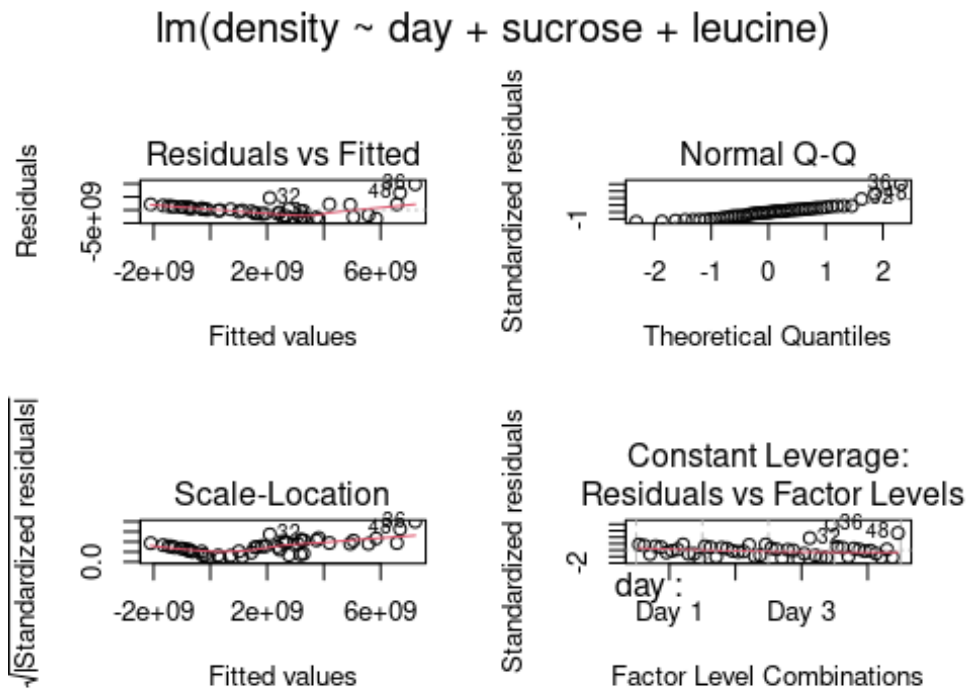
```
##           1           2           3           4           5           6
##  2122953021  1698690708 -1512703042  1481704937  1121142625  -906451125
##           7           8           9          10          11          12
##  -258783396   238154292  2161860542 -1426983396 -1396645708 -3322939458
##          13          14          15          16          17          18
##  1460011354  1064049042 -2484344708   907963271   658300958  -325992792
##          19          20          21          22          23          24
##  -681725062  -512387375  1992318875 -2291525062 -1934187375  2147518875
##          25          26          27          28          29          30
##   725643021   301600708 -3042193042   126214937   -81647375 -3441941125
##          31          32          33          34          35          36
## -1357673396  4584664292 -1983629458 -2797473396 -2815135708  9781570542
##          37          38          39          40          41          42
##  1255825771   846380458 -2435913292   629434687   240932375  -409661375
##          43          44          45          46          47          48
##  -175393646 -1276055958 -2731349708   279806354 -2737855958  6513850292
```

Question: is it a problem that some of the residuals are negative? Why or why not?

No, residuals are observed - predicted, so any observed value that falls below the mean will have a negative residual. Residuals can be negative.

Plot the residuals:

```
oldpar <- par()
par(oma = c(0,0,3,0), mfrow = c(2,2))
plot(dens.nointer.lm)
```



Question: do the plots give you any reason to think the residuals are not meeting the normality or HOV assumptions? Explain.

The predicted values don't go through the middle of the data, which you can see by the residuals vs. fitted value plot in the upper left. The data also appear to be more variable at larger predicted values.

Confirm that you are violating normality and/or HOV with quantitative tests:

```
shapiro.test(residuals(dens.nointer.lm))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(dens.nointer.lm)
## W = 0.87854, p-value = 0.0001368
```

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
bptest(dens.nointer.lm)

##
##  studentized Breusch-Pagan test
##
## data:  dens.nointer.lm
## BP = 16.979, df = 8, p-value = 0.03033
```

Question: are the residuals normally distributed? How do you know?

No, the Shapiro-Wilk normality test has p less than 0.05, so the residuals are not normal.

Question: do the groups have equal variances? How do you know?

The BP test has a p-valeu below 0.05, so the variances are not the same.

Try adding an interaction between sucrose and leucine:

```
dens.inter.lm <- lm(density ~ day + sucrose * leucine, data = bacteria)
summary(dens.inter.lm)

##
## Call:
## lm(formula = density ~ day + sucrose * leucine, data = bacteria)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -7.103e+09 -6.675e+08 -6.523e+07  7.018e+08  6.002e+09
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)

## (Intercept)                       -707144729 1217078470  -0.581 0.565176

## dayDay 2                           729541667  888828443   0.821 0.417656

## dayDay 3                          1375490000  888828443   1.548 0.131273

## dayDay 4                           843210250  888828443   0.949 0.349682

## sucroseSucrose 2                    23969250 1539496023   0.016 0.987672

## sucroseSucrose 3                   370934250 1539496023   0.241 0.811088

## sucroseSucrose 4                  1075084250 1539496023   0.698 0.489860

## leucineLeucine 2                    24234250 1539496023   0.016 0.987535

## leucineLeucine 3                   156059250 1539496023   0.101 0.919870
```

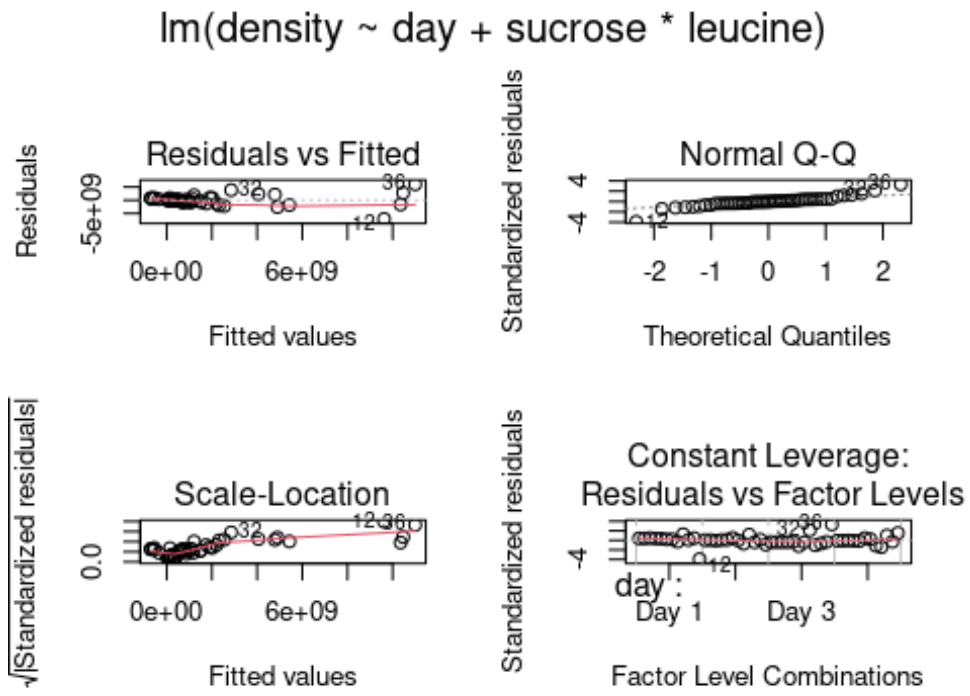
```
## sucroseSucrose 2:leucineLeucine 2  111780750 2177176155   0.051 0.959362

## sucroseSucrose 3:leucineLeucine 2 1790415750 2177176155   0.822 0.416777

## sucroseSucrose 4:leucineLeucine 2 -248484250 2177176155  -0.114 0.909825

## sucroseSucrose 2:leucineLeucine 3 1702555750 2177176155   0.782 0.439789

## sucroseSucrose 3:leucineLeucine 3 4238090750 2177176155   1.947 0.060137 .

## sucroseSucrose 4:leucineLeucine 3 9098940750 2177176155   4.179 0.000202
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.177e+09 on 33 degrees of freedom
## Multiple R-squared:  0.7248, Adjusted R-squared:  0.6081
## F-statistic: 6.208 on 14 and 33 DF,  p-value: 8.188e-06

anova(dens.inter.lm)

## Analysis of Variance Table
##
## Response: density
##                 Df     Sum Sq    Mean Sq F value    Pr(>F)
## day              3 1.1546e+19 3.8487e+18  0.8119 0.4964173
## sucrose          3 1.1872e+20 3.9574e+19  8.3488 0.0002865 ***
## leucine          2 1.4733e+20 7.3666e+19 15.5411 1.755e-05 ***
## sucrose:leucine  6 1.3440e+20 2.2400e+19  4.7257 0.0014198 **
## Residuals       33 1.5642e+20 4.7401e+18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question: although it is not a good idea to interpret results of models that do not meet assumptions, we will revisit the interaction in this model in a later question. Is the interaction between sucrose and leucine significant?

Yes it is, the p-value for sucrose:leucine is less than 0.05.

Plot the residuals:

```
par(oma = c(0,0,3,0), mfrow=c(2,2))
plot(dens.inter.lm)
```



Question: are there still signs that the data violate the normality and HOV assumptions? Explain.

Yes, the predicted values still don't go through the middle of the data, and there is a change in variance as predicted values increase.

Test the assumptions quantitatively:

```
shapiro.test(residuals(dens.inter.lm))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(dens.inter.lm)
## W = 0.86808, p-value = 6.812e-05

bptest(dens.inter.lm)

##
##  studentized Breusch-Pagan test
##
## data:  dens.inter.lm
## BP = 28.267, df = 14, p-value = 0.01311
```

Question: do you meet the normality and HOV assumptions now that an interaction term has been added? How do you know?

Both tests fail, since both have p less than 0.05.

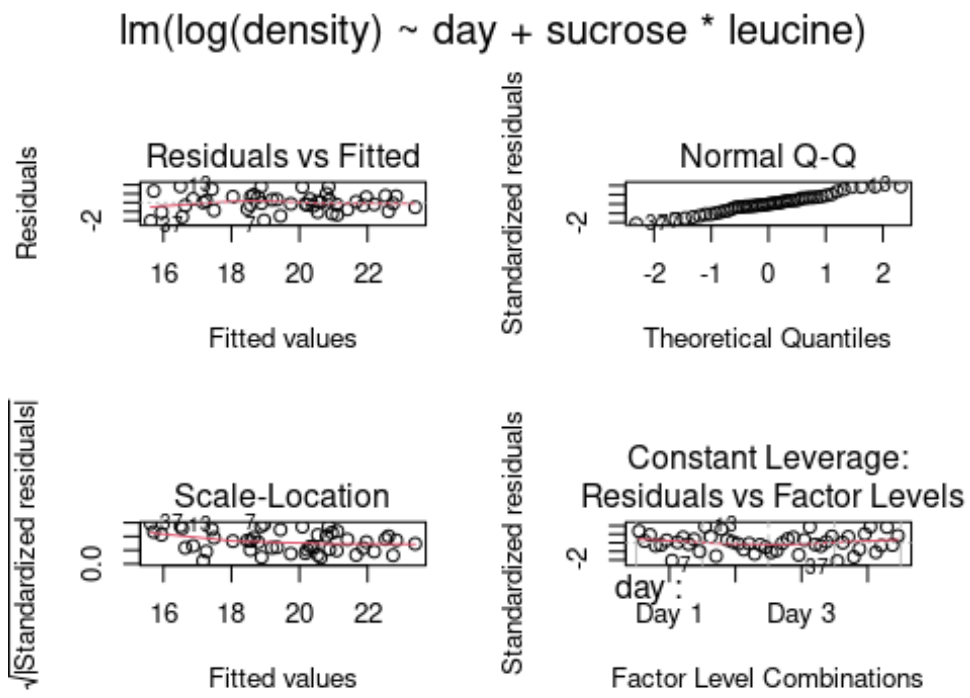
Try log-transforming density:

```
dens.inter.log.lm <- lm(log(density) ~ day + sucrose * leucine, data =
bacteria)
summary(dens.inter.log.lm)
```

```
##
## Call:
## lm(formula = log(density) ~ day + sucrose * leucine, data = bacteria)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.05791 -0.55914 -0.02123  0.62018  1.82725
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        15.71776    0.64648  24.313  < 2e-16 ***
## dayDay 2                            0.78466    0.47212   1.662 0.105987
## dayDay 3                            0.22944    0.47212   0.486 0.630195
## dayDay 4                           -0.08895    0.47212  -0.188 0.851707
## sucroseSucrose 2                    0.92687    0.81773   1.133 0.265190
## sucroseSucrose 3                    3.24449    0.81773   3.968 0.000369 ***
## sucroseSucrose 4                    4.45696    0.81773   5.450 4.88e-06 ***
## leucineLeucine 2                    1.54177    0.81773   1.885 0.068203 .
## leucineLeucine 3                    2.94028    0.81773   3.596 0.001043 **
## sucroseSucrose 2:leucineLeucine 2   0.37090    1.15645   0.321 0.750445
## sucroseSucrose 3:leucineLeucine 2   0.12368    1.15645   0.107 0.915479
## sucroseSucrose 4:leucineLeucine 2  -1.40378    1.15645  -1.214 0.233414
## sucroseSucrose 2:leucineLeucine 3   1.29941    1.15645   1.124 0.269284
## sucroseSucrose 3:leucineLeucine 3   0.06442    1.15645   0.056 0.955914
## sucroseSucrose 4:leucineLeucine 3  -0.49719    1.15645  -0.430 0.670045
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.156 on 33 degrees of freedom
## Multiple R-squared:  0.8211, Adjusted R-squared:  0.7452
## F-statistic: 10.82 on 14 and 33 DF,  p-value: 1.338e-08
```

Produce the plots:

```
par(oma = c(0,0,3,0), mfrow=c(2,2))
plot(dens.inter.log.lm)
```



Question: now do the residual plots look like they should if you meet the normality and HOV assumptions? Explain.

The residual plots are much better. The predicted values are in the middle of the data, and the amount of variation is the same from low to high fitted values. The points follow the straight diagonal line on the normal probability plot.

Confirm your impression that we meet the assumptions - run the quantitative tests:

```
shapiro.test(residuals(dens.inter.log.lm))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(dens.inter.log.lm)
## W = 0.97908, p-value = 0.541

bptest(dens.inter.log.lm)

##
##  studentized Breusch-Pagan test
##
## data:  dens.inter.log.lm
## BP = 21.096, df = 14, p-value = 0.09921
```

Question: did you pass the normality and HOV assumption tests?

Yes, both tests have p greater than 0.05 now.

Second analysis

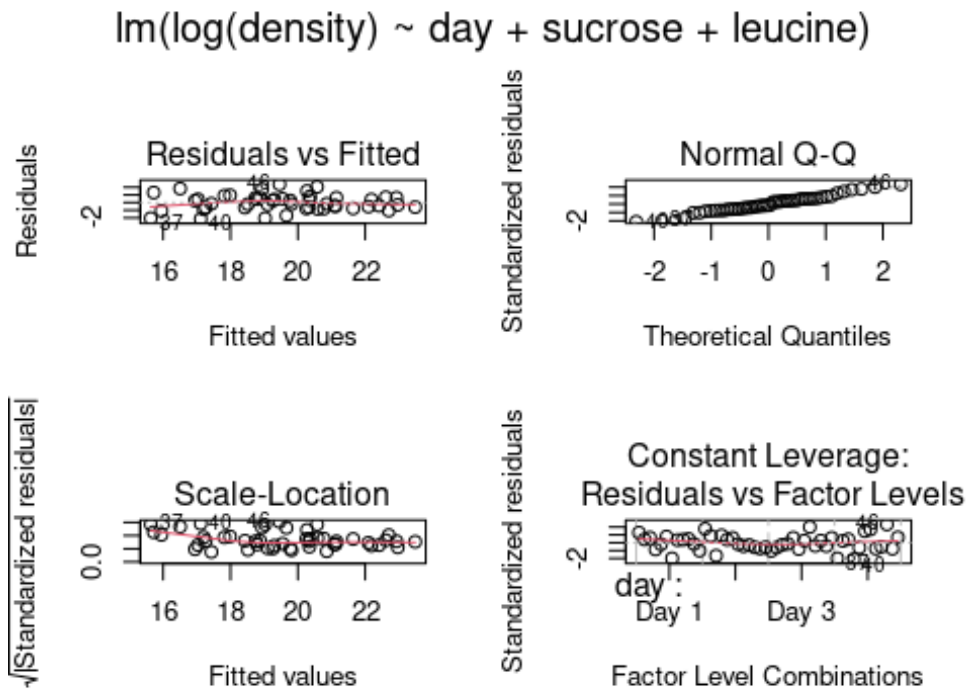
Fit one more model that log-transforms density, but does not include the interaction between sucrose and leucine:

```
logdens.nointer.lm <- lm(log(density) ~ day + sucrose + leucine, data =
bacteria)
summary(logdens.nointer.lm)
```

```
##
## Call:
## lm(formula = log(density) ~ day + sucrose + leucine, data = bacteria)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1562 -0.6492 -0.1576  0.6279  2.3828
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      15.72130    0.49139  31.994  < 2e-16 ***
## dayDay 2          0.78466    0.46329   1.694  0.09830 .
## dayDay 3          0.22944    0.46329   0.495  0.62321
## dayDay 4         -0.08895    0.46329  -0.192  0.84873
## sucroseSucrose 2  1.48363    0.46329   3.202  0.00271 **
## sucroseSucrose 3  3.30719    0.46329   7.139 1.38e-08 ***
## sucroseSucrose 4  3.82330    0.46329   8.253 4.37e-10 ***
## leucineLeucine 2  1.31447    0.40122   3.276  0.00221 **
## leucineLeucine 3  3.15694    0.40122   7.868 1.42e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.135 on 39 degrees of freedom
## Multiple R-squared:  0.7964, Adjusted R-squared:  0.7547
## F-statistic: 19.07 on 8 and 39 DF,  p-value: 2.871e-11
```

Plot the residuals:

```
par(oma = c(0,0,3,0), mfrow=c(2,2))
plot(logdens.nointer.lm)
```



Question: do these graphs look different from the graph of the log-transformed data with an interaction between sucrose and leucine that you made in the previous step?

Yes the residuals look nearly the same with or without an interaction included on a log scale.

Confirm that these graphs look okay - check assumptions quantitatively:

```
shapiro.test(residuals(logdens.nointer.lm))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(logdens.nointer.lm)
## W = 0.97844, p-value = 0.5154

bptest(logdens.nointer.lm)

##
##  studentized Breusch-Pagan test
##
## data:  logdens.nointer.lm
## BP = 14.618, df = 8, p-value = 0.067
```

Question: do you pass the normality and HOV tests?

Yes, both tests are passed on a log scale, even though no interaction was included.

Question: given that both of models that used log-transformed data met the assumptions, why might you want to try using an interaction first and only transform if that isn't sufficient, like the book did?

> The interaction assesses whether the response to leucine depends on sucrose level, which is a scientifically interesting part of the analysis. The experiment uses crossed factor levels for leucine and sucrose, and it is thus well designed to test for main effects and interactions between them. Given that one would want to test for an interaction anyway, it makes sense to include the interaction term first, and only transform if the model with an interaction doesn't fit the data.

Getting the right stats for interpreting your results

We will use the emmeans library to get the predicted means for leucine.

Load the emmeans library:

```
library(emmeans)
```

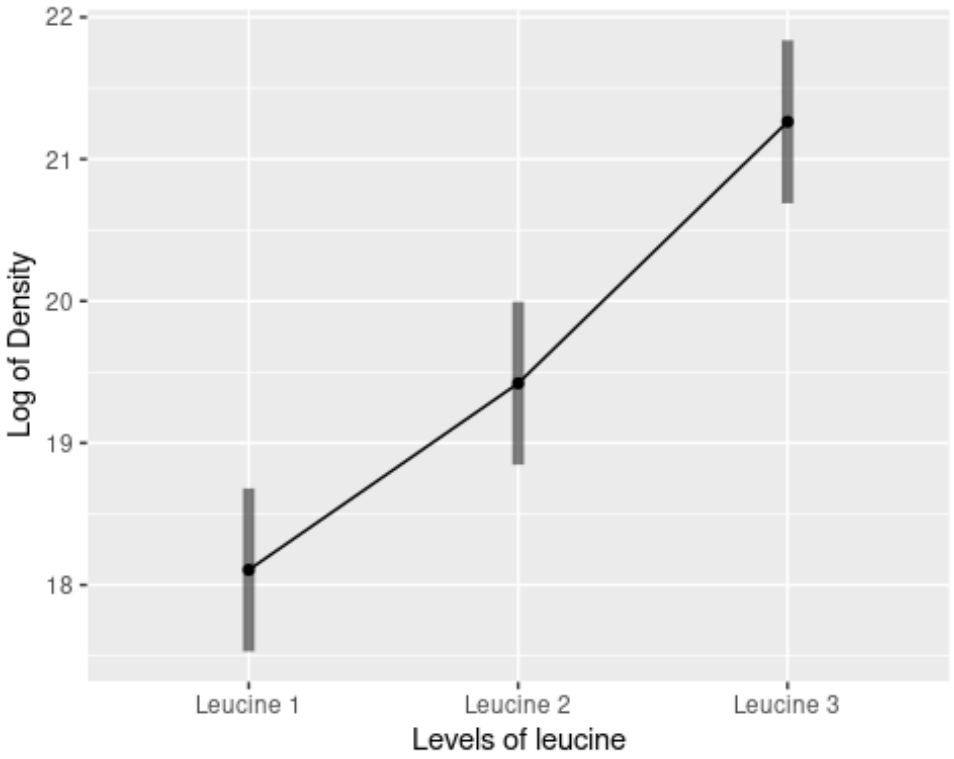
Estimate the means on the log density scale used to fit the logdens.nointer.lm model:

```
emmeans(logdens.nointer.lm, specs = "leucine")
```

```
##  leucine   emmean    SE df lower.CL upper.CL
##  Leucine 1   18.1 0.284 39     17.5     18.7
##  Leucine 2   19.4 0.284 39     18.8     20.0
##  Leucine 3   21.3 0.284 39     20.7     21.8
##
## Results are averaged over the levels of: day, sucrose
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
```

Plot these means using a y-axis showing the log of density. The plot symbols will be means of log bacterial density:

```
emmip(logdens.nointer.lm, formula = ~ leucine, CIs = T) + labs(y = "Log of
Density")
```



Get the means as back-transformed, geometric means:

```
emmeans(logdens.nointer.lm, specs = "leucine", type = "response")
```

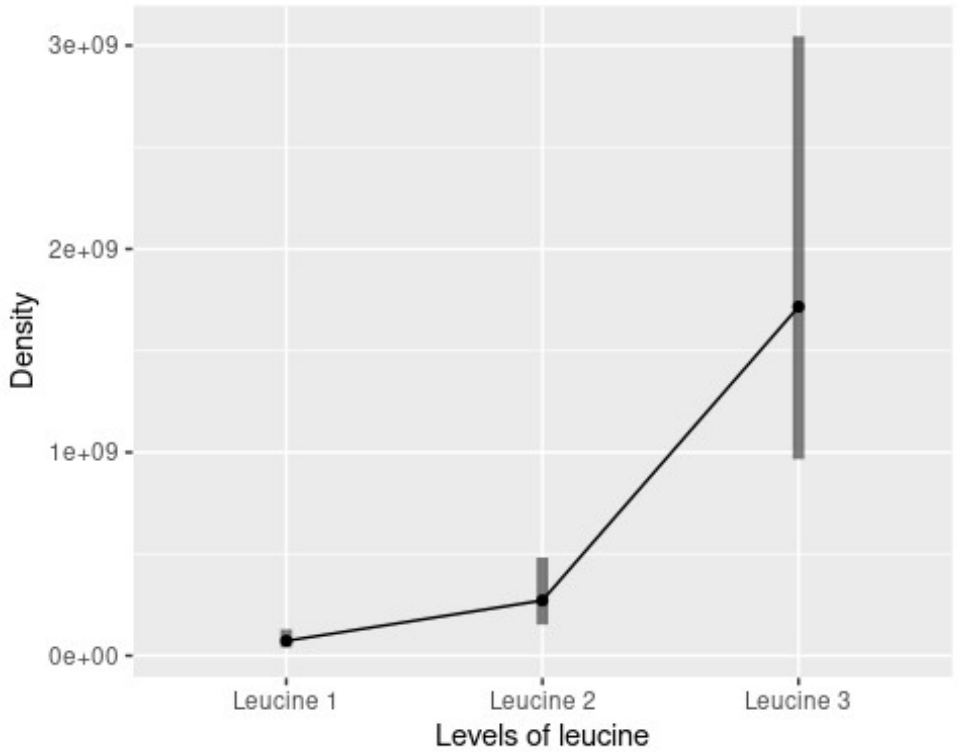
```
##  leucine   response       SE df lower.CL upper.CL
##  Leucine 1 7.30e+07 2.07e+07 39 4.11e+07 1.30e+08
##  Leucine 2 2.72e+08 7.71e+07 39 1.53e+08 4.82e+08
##  Leucine 3 1.72e+09 4.87e+08 39 9.67e+08 3.05e+09
##
## Results are averaged over the levels of: day, sucrose
## Confidence level used: 0.95
## Intervals are back-transformed from the log scale
```

Question: what kind of means are these back-transformed values, arithmetic or geometric?

These are geometric means.

Plot the leucine effect from logdens.nointer.eff again, but this time using density as the y-axis values.

```
emmip(logdens.nointer.lm, formula = ~ leucine, CIs = T, type = "response") +
labs(y = "Density")
```

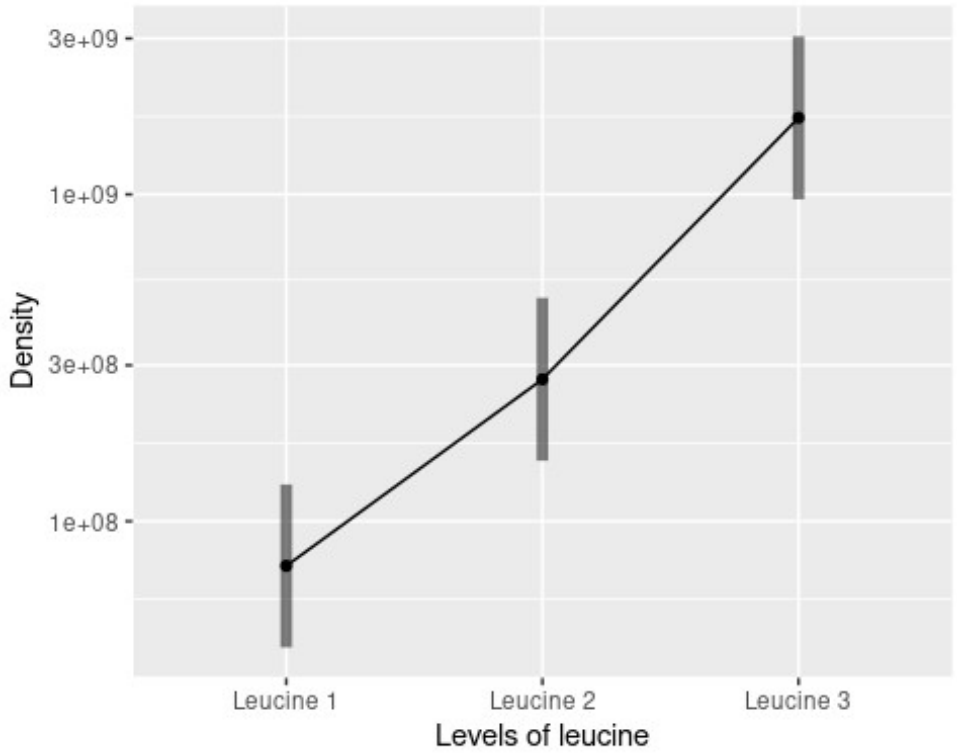


Question: why are the confidence intervals asymmetrical?

> The confidence intervals are calculated on a log scale and back-transformed, so symmetrical intervals on a log scale are asymmetrical on the original data scale.

Finally, plot the leucine effect from logdens.nointer.eff using densities, but log-scale the y-axis.

```
emmip(logdens.nointer.lm, formula = ~leucine, CIs = T, type = "response") +
labs(y = "Density") + scale_y_log10()
```



Question: the y-axis is a log axis - does that mean that the tick labels are the log of density, or are they density?

> They are density - the spacing between the ticks are log scale, but the numbers are un-transformed density.

What do interactions on a log scale mean?

Question: we found that there is no interaction between sucrose and leucine on a log scale, but there was one on the linear scale. Why would the interaction be significant on the original data scale but not on a log scale?

> The plot of the data on the original data scale shows the lines diverging, which is why the interaction is significant. However, the lines were diverging because the treatment effects were constant multipliers. On a log scale these multiplicative effects become additive effects, and constant multipliers become constant additive effects. When the effects of predictors are additive only the main effects will be significant, so on a log scale there are only significant main effects in the model.