Graphing workshop

Erica J. Crespi and Jesse L. Brunner

2019-06-12

Example 1: growth rates

Let us imagine we were interested in the growth of tadpoles in different environments. We knew that tadpoles grow faster (i.e., gained weight faster) at warmer temperatures, but suspected that those given a high-quality, high-protein diet would respond to elevated temperatures even more than those fed a typical diet. We thus conducted an experiment where we raised tadpoles at one of two temperatures (low or high temperature) and fed them one of two diets (typical or high protein) for a total of four treatments. We measured their initial and final sizes (in mg), which are given in the table, below.

With your partner(s), sketch a graph of these data that you think best conveys the salient points of the data.

Table 1: Initial and final masses (in mg) of tadpoles raised at one of two temperatures and provided one of two diets.

Temperature	Food	Initial	Final
Low	Typical	200	428
Low	Typical	200	393
Low	Typical	207	393
Low	Protein	189	423
Low	Protein	199	418
Low	Protein	200	459
High	Typical	208	417
High	Typical	198	431
High	Typical	198	454
High	Protein	194	492
High	Protein	204	467
High	Protein	196	495

Principles:

- Emphasize the comparison you want the viewer to see
  - key comparison along x or y axis
    - positions are easier to compare than angle or area[1] or subtle gradations in color.
  - ensure things to compared are adjacent
  - by convention, x *causes* y
  - consider presenting the effect of interest (e.g., difference, deviation from expected, slope, rate) instead of raw data[2]
  - consider transformed axes to make your message clearer[3]
  - Consider using color to highlight key comparisons or observations
- Let the data speak…
  - As much as possible, show the raw data[4] (e.g., points of individual observations in a scatter plot)
  - When showing means (or medians) provide an indication of the variance (e.g., +/- standard error, confidence interval, etc.)
- But avoid extraneous information or comparisons
  - too much information masks the key messages[5]
  - extra colors, sizes, etc. make the reader *look* for trends or comparisons, even if they are not there!
  - Maximize the information shown with minimal ink
- Be accurate
  - Ensure the ranges of the axes are scientifically meaningful and appropriate for the effect sizes.
    - Do not "zoom in" to make a small difference look large and important!
    - Include zero where appropriate[6]
  - Connect points *only* when they are from the same thing (e.g., individual or population)
  - Avoid plotting different data using a secondary y-axes[7]
  - *If* you use size to convey information (e.g., sample size), scale it with *area* rather than diameter.
- Be clear and avoid chart junk!
  - Ensure all labels are large enough to read!
  - Provide simple labels with units and legends as needed
  - Avoid 3-D graphs, shading, useless colors and words, extra crud[8]
- But, you may need to follow the conventions of your field for presenting data, including indicators of statistical significance.

[1] Hence the general prohibition on pie charts and 3-D figures.

[2] That is, don't make the reader do your work for you. But this will depend on discipline.

[3] E.g., when 1) variables span orders of magnitude (e.g., population sizes), 2) using ratios (try log_2), 3) effects increases with large changes in predictor variable (e.g., dose), 4) response scales indirectly with predictor variable (e.g., with surface area, but you measured mass).

[4] Some people think you should be able to reconstruct a data set from a graph.

[5] Exploratory graphs might be quite complex, but they need to be simplified for presentation. Figures for talks or posters need to be simpler than those in papers, which allow more time to digest and evaluate.

[6] Ensure bars start from zero!

[7] Secondary axes can be very misleading depending on the scaling. Instead, use two panels or facets with a common x-axis.

[8] Just because it is included in the output, does not mean you need to keep it.

Example 2: dose-response data

Imagine we are interested in the effect of a pesticide on *Daphnia* survival and so we expose twenty individuals each to one of four doses of the pesticide (in mg/L) and determine how many die from the exposure. Those data are presented in this table:

Table 2: Mortality in Daphnia exposed to one of four doses of a pesticide.

Dose	Died	Total
0.01	0	20
0.10	3	20
1.00	16	20
10.00	20	20

We also recorded *when* they died over the 10 day experiment, so we have the following data:

Table 3: Number of Daphnia surviving over 10 days post exposed to one of four doses of a pesticide.

Day	D0.01	D0.1	D1	D10
0	20	20	20	20
1	20	20	20	20
2	20	20	20	19
3	20	20	18	15
4	20	20	18	7
5	20	19	15	4
6	20	19	12	1
7	20	18	5	0
8	20	17	4	0
9	20	17	4	0
10	20	17	4	0

With your partner(s), sketch one or more graphs of these data that you think best conveys the salient points of the data.

Example 3: Exponential decay in fluorescences

Imagine we were interested in the rate at which some fluorescent molecules continued to emit light after excitation with a laser. We measured the amount of light (arbitrary units; au) from our sample every 10 minutes for an hour. We were interested in determining whether the data were a good fit to a theoretical expectation, which was that fluorescence would decline exponentially. This expectation is embodied in this equation:

$$\text{fluorescence} = \alpha \times e^{-\beta \times \text{time}},$$

where $\alpha = 1$ is the maximum fluorescence (set to one in our example) and $\beta = 1/20$ determines the rate of decline.

Table 4: Relative fluorescence (arbitrary units) of some neat material through time since excitation.

Minutes	Fluorescence
0	1.02
10	0.60
20	0.34
30	0.22
40	0.17
50	0.08
60	0.06

How would you plot these data to show the data relative to the theoretical expectation? With your partner, sketch out some ideas.

Resources

- Flowing Data's 7 basic rules for making charts and graphs (`https://flowingdata.com/2010/07/22/7-basic-rules-for-making-charts-and-graphs/`)
- Karl Broman's
  - very clear presentation of Dos and don'ts highlighting several key principles (`http://stat545.com/block015_graph-dos-donts.html`)
  - Top Ten Worst graphs (`https://www.biostat.wisc.edu/~kbroman/topten_worstgraphs/`)
- Wainer H (1984) How to display data badly. The American Statistician 38:137-147 (`https://www.jstor.org/stable/2683253?seq=8#metadata_info_tab_contents`)
- A series of columns on visualizing scientific data in Nature (`http://blogs.nature.com/methagora/2013/07/data-visualization-points-of-view.html`) breaks down the elements very nicely
- The Gallery of Data Visualization (`http://www.datavis.ca/gallery/index.php`) provides examples of the best and worst graphs
- Principles and examples of nice graphs from an online class on data visualization (`https://paldhous.github.io/ucb/2016/dataviz/week2.html`)

Constructing plots in R with `ggplot2`

It is possible to construct these graphs with base plotting functions in R, but we will use the `ggplot` package as it uses a consistent approach for various types of graphs, has reasonable defaults, and makes it easy to map aesthetics onto components of the data set. We will also need some components in the `dplyr` package. We can load both with the `tidyverse`.

```
# load the packages required
library(tidyverse)
```

Next we need to load in the data (found at `https://github.com/JesseBrunner/GraphingWorkshop`)

```
df1 <- read_csv("Example1.csv")
df2 <- read_csv("Example2A.csv")
df2B <- read_csv("Example2B.csv")
df3 <- read_csv("Example3.csv")
```

Example 1: plotting growth rates

We might want to plot the raw masses (or any other data) through time (when measured). However, `ggplot` works best when the response data is in a single column. Thus, we first use the `gather` function to **reshape** our data set from wide, where the initial and final masses are in different columns, to long, where these are in the same column and there is another column defining whether the mass is initial or final.

We also need to create a grouping variable so that we can get one line per animal. We use the `mutate` function to create this new variable and give it the row number for each line before reshaping things.

Finally, we re-factor the `Time` variable so that it starts with Initial instead of Final, which comes first in the alphabet

```
# re-organize data, from wide to long
# Note: We want animal number later as a grouping variable
df1long <- df1 %>%
  mutate(Animal = row_number()) %>%
  gather(key="Time", value="Mass", Initial, Final)

# get these in the right order
df1long$Time = fct_relevel(df1long$Time, "Initial")

# plot the mass against "Time" of the measurement by treatment
ggplot(df1long, aes(x=Time, y=Mass,
```

```
                  color = Temperature,
                  linetype=Food, shape=Food,
                  group=Animal)) +
  geom_line() +
  geom_point() +
  theme_bw()
```

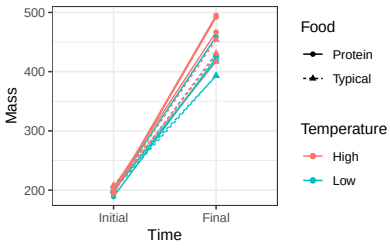

Figure 1: Plot of raw data from example 1.

But we are interested in growth, so that is what we should plot. We thus need to create a new column for the change in mass in the original, wide-formatted data set. We can then plot the change with temperature on the x-axis.

```
# create a new column for the change in mass
df1 <- df1 %>%
  mutate(Animal = row_number(),
         Change = Final-Initial)

df1$Temperature = fct_relevel(df1$Temperature, "Low", "High")

ggplot(df1, aes(x=Temperature, y=Change,
                color=Food,
                group=Animal)) +
  geom_point() +
  scale_y_continuous("Change in mass (mg)") +
  theme_bw()
```

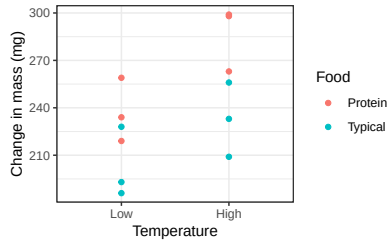

Figure 2: Plot of change in mass by individual among treatments from example 1

Notice that we are *not* connecting the dots with lines. That would imply that it is the same tadpole (or experimental unit) in both treatments, but in this case animals can be in only one treatment.

It might be cleaner to plot the mean or some other summary statistics. This is pretty simple to do with the `stat_summary` function, which calculates summary (by default, mean and se) at each unique x-values.

```
ggplot(df1, aes(x=Temperature, y=Change,
                color=Food)) +
  stat_summary() +
  scale_y_continuous("Change in mass (mg)") +
  theme_bw()
```

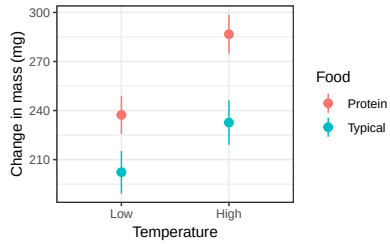

Figure 3: Plot of mean and standard error of the change in mass by treatment from example 1.

Example 2: plotting frequency and cummulative event data

These were the results of the dose-response study. Notice 1) I plotted the proportion that died rather than raw numbers, 2) I $\log_{10}$-tranformed the x-axis because we usually expect the response to contaminants or pathogens to increase with large changes in dose, and 3) I fit a logistic regression line (the `geom_smooth(method=glm)` bit).

```
ggplot(df2, aes(x=Dose, y=Died/Total, weight=Total)) +
  geom_smooth(method = glm,
              method.args=list(family = "binomial")) +
  geom_point() +
  scale_x_log10() +
  labs(x="Dose (mg/L)", y="Proportion died") +
  theme_bw()
```

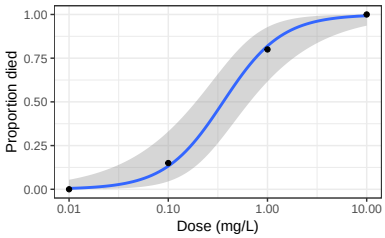

Figure 4: Plot of mortality by dose in example 2.

Here are the survival data. Again, we needed to re-organize the data from wide to long. I also used the `substr` function to get the numeric-part of the names of doses as it it is prettier. (Note: these are still strings in R rather than numeric. That is important because `ggplot` is applying colors to them as if they were factors, not particular values along some scale or metric.) Lastly, I am using a step function geometry to convey the fact that we know how many were alive only at the censuses, not in between. This is convention with survival analyses and is important when the intervals between measurements are variable.

```
# re-organize data, from wide to long
df2bLong <- df2B %>%
  gather(key=Dose, value=Surviving, -Day)

# extract numeric part of doses (after "D")
df2bLong$Dose <- substr(df2bLong$Dose, start=2, 5)

# plot survival through time by dose
ggplot(df2bLong, aes(x=Day, y=Surviving, color = Dose)) +
  geom_step() +
  geom_point() +
  scale_x_continuous("Days post exposure", breaks = 0:5*2) +
  scale_y_continuous("Number surviving",
                     minor_breaks = 0:20) +
  scale_color_brewer(palette = "Reds") +
theme_bw()
```

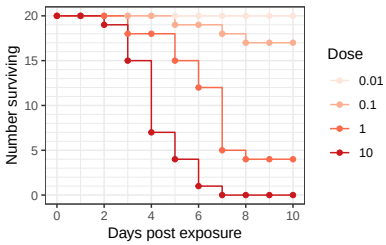

Figure 5: Plot of survivorship through time by dose from example 2.

Also, survival data are generally analyzed with survival analyses that account for individuals being censored or removed (e.g., Kaplan-Meyer survival curves, Cox proportional hazard models, etc.). These approaches often have their own plotting functions and styles. ## Example 3: plotting data against a theoretical expectation

In this example we may want to plot both the raw data (fluorescence against time) as points and the theoretical expecation using a line. We can use `stat_function()` for the latter where we supply the negative exponential as a function of x, where x is time.

```
ggplot(df3, aes(Minutes, Fluorescence)) +
  geom_point() +
  stat_function(fun=function(x) exp(-1/20*x)) +
  labs(x="Minutes from excitation",
       y="Relative fluorescence (au)") +
  theme_bw()
```

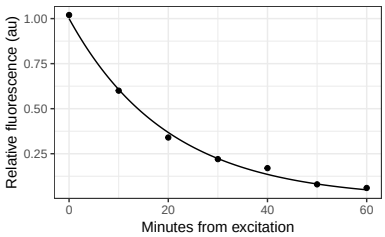

Figure 6: The decline in fluorescence with time since excitation compared with the theoretical expectation.

Alternatively, we might want to plot the deviations from this theoretical expecation to see where the data agree and disagree.

```
ggplot(df3, aes(Minutes, Fluorescence-exp(-1/20*Minutes) )) +
  geom_hline(yintercept=0) +
  geom_point() +
  labs(x="Minutes from excitation",
       y="Observed - expected\n fluorescence (au)") +
  theme_bw()
```

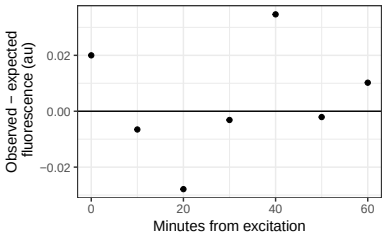

Figure 7: Deviations from the theoretical expectation of an exponential decline in fluorescence with time since excitation.

Notice I added a horizontal line to emphasize where zero deviation is. Also, I used the 'new line' escape, `\n`, to split the y-axis into two lines.

Suggestions for `ggplot` graphs

Use `ggsave()` function to save the last graph. It will guess the file type by the extension and you can provide `width=` and `height=` to control the image size saved.

Look to ColorBrewer (`http://colorbrewer2.org/`) for various color schemes that look nice and work well.

There are numerous helpful online guides for plotting. I would recommend Winston Chang's "cookbook" for graphs in R (`http://www.cookbook-r.com/Graphs/`) as a good starting place for most things you might want to try.