BMR 617

Graphing Quantitative Variables

February 4th 2021

Principles of graphing

- Producing clear, informative graphs which present the facets of interest in your data is a challenging task
- There is always a trade-off between providing detail and clarity
- There are some basic rules which will help when graphing quantitative data:
  - A “typical value” (average) should always be apparent from the graph
  - The spread of the data should always be apparent from the graph
  - If possible, show all the data
    - But if there’s too much data this will hide the average and spread

Graph types for quantitative data

- There are a few basic graph types for graphing a single set of quantitative data:
  - Bar chart
    - Often too simplistic
  - Box-and-whisker plot
    - Good for graphing the basic summary: mean/median, interquartile range, outliers
  - Column scatter plot
    - Plots all the data; good if there are a reasonably small number of data points
- These can be combined when appropriate

Using color

- Using color in graphs can be very powerful
- Can distinguish between variables that are not represented by x- or y-axes
- However, there are many pitfalls
- Color is subjective
  - Not everyone perceives color the same way
  - Avoid using colors that are indistinguishable to color-blind people

Graphing with R: the ggplot2 library

- The ggplot2 library is part of the tidyverse package
  - The “gg” stands for “graphical grammar”
- Very powerful, produces publication-quality images
- Based on the concept of layers
  - If you’re familiar with photoshop or similar tools you may be familiar with this concept
  - Build portions of the graph and add them on top of each other
  - Each layer has “aesthetics” associated with it

Loading and preparing data for plotting

- Open RStudio, load the tidyverse package, and download our usual data set and wrangle it to get the strain and diet columns:

```
library(tidyverse)
met <- read_csv("https://denvirlab.marshall.edu/BMR617-2021/data/TH-B6-metabolic.csv")
met <- separate(met, MouseID, sep="-", into=c("Strain", "Diet", "ID"))
```

- Filter for one group (Tallyho mice fed Chow) to experiment with:

```
th_chow <- filter(met, Strain=="TH" & Diet == "Chow")
```

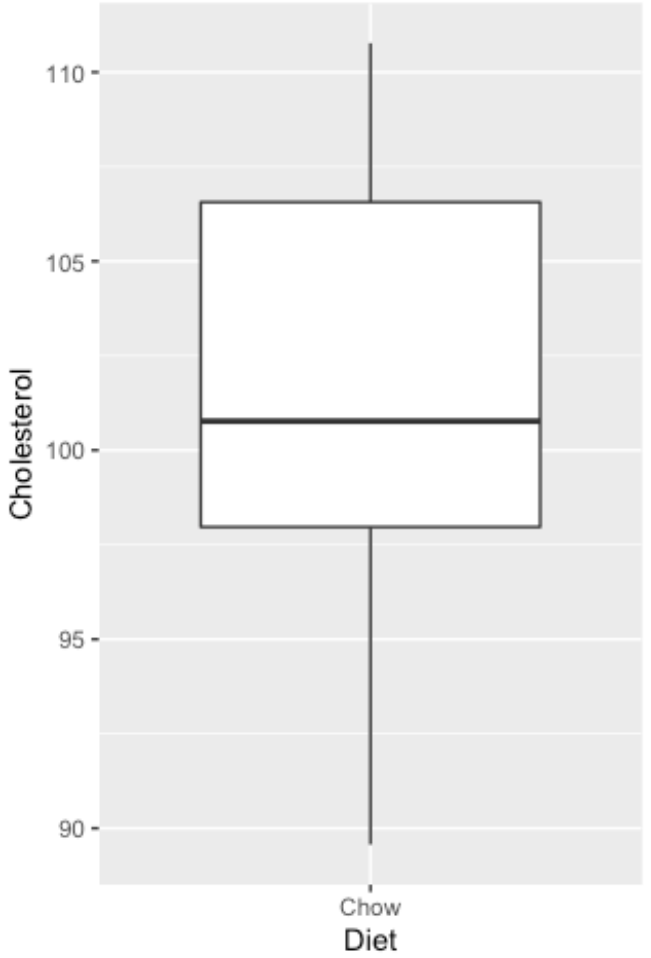
Experimenting with ggplot2

- The ggplot function creates a base for a plot
- Takes a data set and a set of *aesthetics*
  - Aesthetics must include the variables that represent the x- and y-axes
- Try the following:
  `ggplot(th_chow, aes(x=Diet, y=Cholesterol))`
- Note how this just creates the axes, but doesn't plot any data
  - We haven't told ggplot how to display the data yet

Box and whisker plot with ggplot2

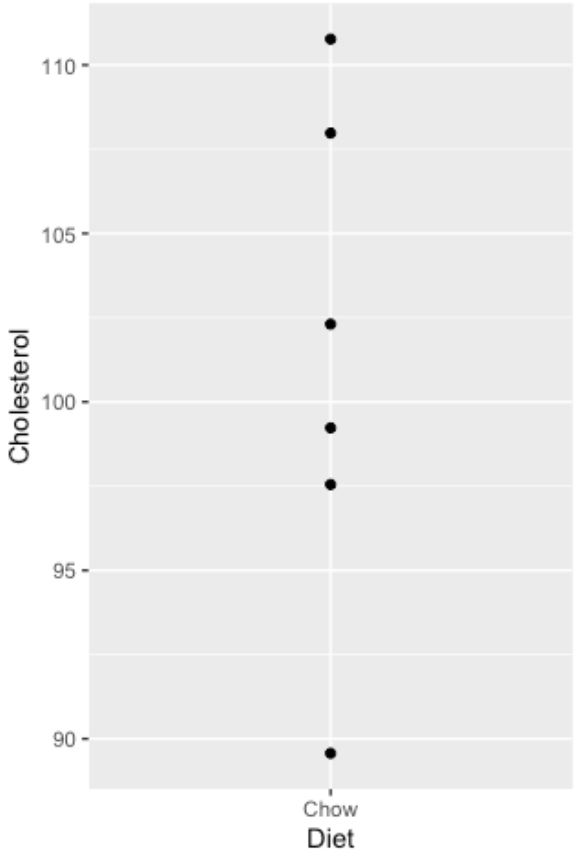
- To create additional data *layers*, we can add various “geometries” to the plot
- The geometries are represented by functions whose names start `geom_`
  - Using RStudio, if you type `geom_`, it will show a list of options available
- To show a “box and whisker plot”, add a geom_boxplot() to the plot:

```
ggplot(th_chow, aes(x=Diet, y=Cholesterol)) +
geom_boxplot()
```



Plotting points with `geom_point()`

- How many data points ("observations") are there in `th_chow`?
- Would it be sensible to plot all of these?
- We can plot the individual points with `geom_point()`
- Try
```
ggplot(th_chow, aes(x=Diet, y=Cholesterol)) +
geom_point()
```
- What are the pros and cons of this plot versus the previous one?
- Is there a plot which would improve on both?



Combining plots by adding them

- Try the following:

```
ggplot(th_chow, aes(x=Diet, y=Cholesterol)) + geom_boxplot() +
geom_point()
```

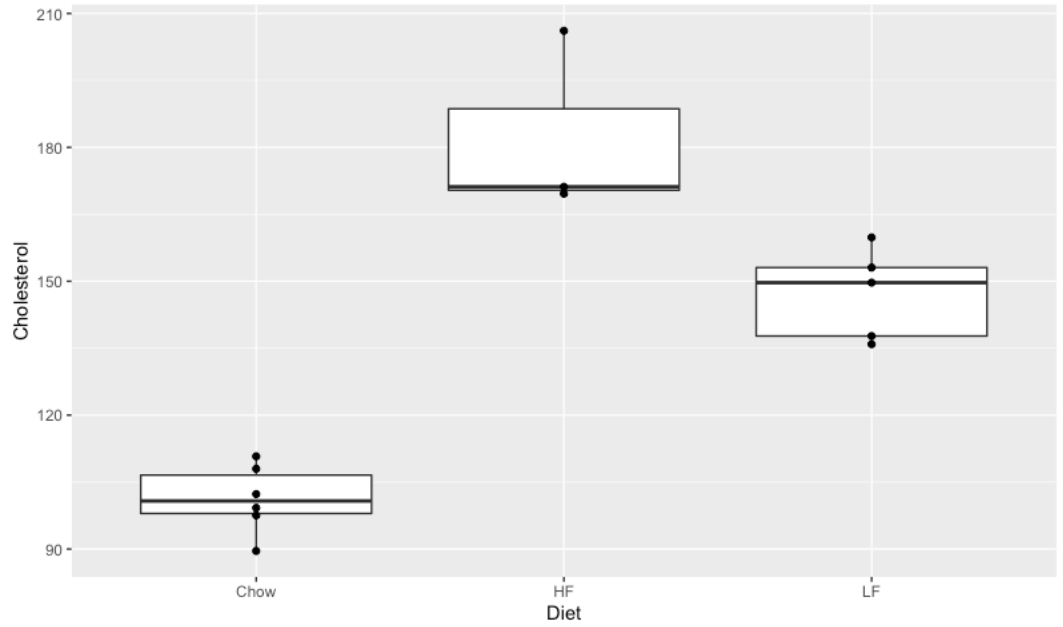
- What happens if you reverse the order of plots?

```
ggplot(th_chow, aes(x=Diet, y=Cholesterol)) + geom_point() +
geom_boxplot()
```

- Can you explain what is happening here?

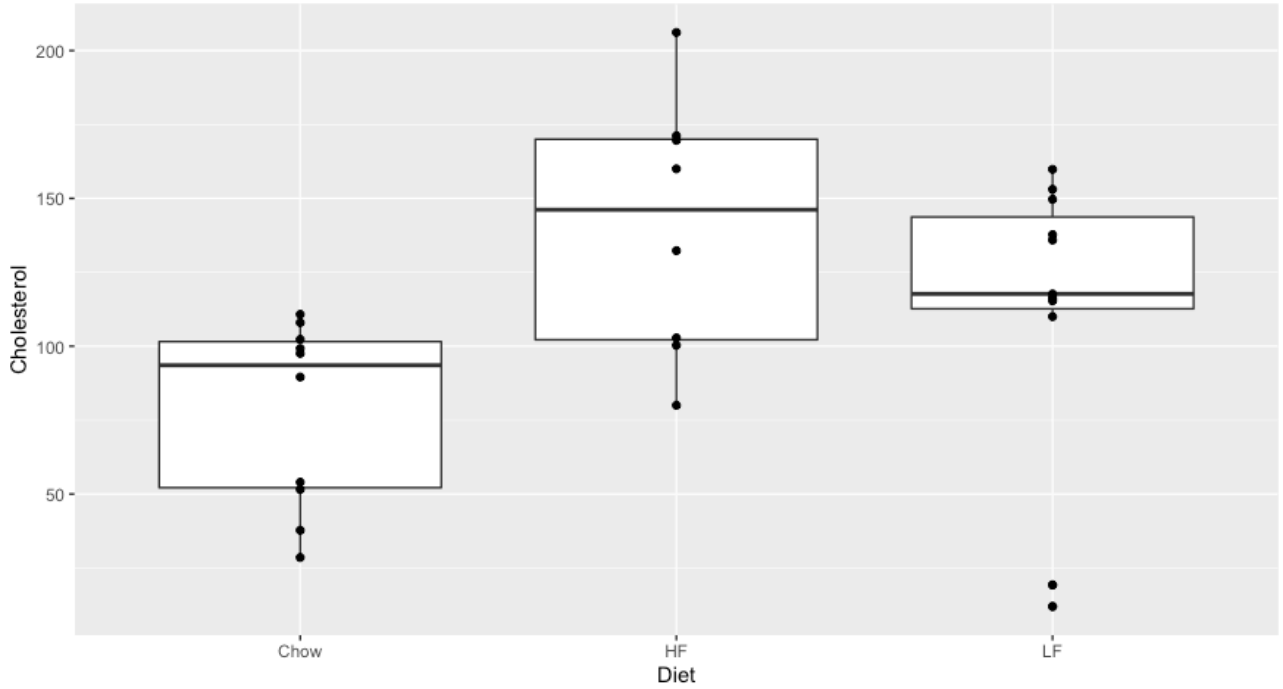
Plotting multiple groups

- Let's look at all the Tallyho mice, over all three diets
  `th <- filter(met, Strain=="TH")`
- Try plotting them with the same commands
  `ggplot(th, aes(x=Diet, y=Cholesterol)) + geom_boxplot() + geom_point()`
- Does the graph tell you what's going on?



Plotting the full data set

- What if we do
```
ggplot(met, aes(x=Diet,
y=Cholesterol)) +
geom_boxplot() +
geom_point()
```
- Is there a problem with this plot?
- How can we fix it?



Using color

- We can color the points by mouse strain by adding an aesthetic to the `geom_point` layer:

```
ggplot(met, aes(x=Diet, y=Cholesterol)) + geom_boxplot() +
geom_point(color=Strain)
```

- Note this doesn't split the box plots. We can do this by applying the color to the entire plot:

```
ggplot(met, aes(x=Diet, y=Cholesterol, color=Strain)) +
geom_boxplot() + geom_point()
```

- And we can adjust the position of the points using a function for the position parameter of geom_point:

```
ggplot(met, aes(x=Diet, y=Cholesterol, color=Strain)) +
geom_boxplot() +
geom_point(position=position_jitterdodge(jitter.width = 0))
```