Graphing numerical data

Put your name here

Put the date here

Introduction

In this module, we will use the `ggplot2` package for creating nicely formatted charts and graphs for numerical data.

Instructions

Presumably, you have already created a new project and downloaded this file into it. From the `Run` menu above, select `Run All` to run all existing code chunks.

When prompted to complete an exercise or demonstrate skills, you will see the following lines in the document:

ANSWER

These lines demarcate the region of the R Markdown document in which you are to show your work.

Sometimes you will be asked to add your own R code. That will appear in this document as a code chunk with a request for you to add your own code, like so:

```
# Add code here
```

Be sure to remove the line `# Add code here` when you have added your own code. You should run each new code chunk you create by clicking on the dark green arrow in the upper-right corner of the code chunk.

Sometimes you will be asked to type up your thoughts. That will appear in the document with the words, "Please write up your answer here." Be sure to remove the line "Please write up your answer here" when you have written up your answer. In these areas of the assignment, please use contextually meaningful full sentences/paragraphs (unless otherwise indicated) and proper spelling, grammar, punctuation, etc. This is not R code, but rather a free response section where you talk about your analysis and conclusions. You may need to use inline R code in these sections.

When you are finished with the assignment, knit to PDF and proofread the PDF file **carefully**. Do not download the PDF file from the PDF viewer; rather, you should export the PDF file to your computer by selecting the check box next to the PDF file in the Files pane, clicking the `More` menu, and then clicking `Export`. Submit your assignment according to your professor's instructions.

Load Packages

We load the `mosaic` package as well as the `MASS` package for working with data on risk factors associated with low birth weight. (Note that the `ggplot2` package we will use for graphing is automatically loaded alongside the `mosaic` package.)

```
library(MASS)
library(mosaic)
```

ggplot

The `ggplot` command is an all-purpose graphing utility. It uses a graphing philosophy derived from a book called *The Grammar of Graphics* by Leland Wilkinson. The basic idea is that each variable you want to plot should correspond to some element or "aesthetic" component of the graph. The obvious places for data to go are along the y-axis or x-axis, but other aesthetics are important too; graphs often use color, shape, or size to illustrate different aspects of data. Once these aesthetics have been defined, we will add "layers" to the graph. These are objects like dots, boxes, lines, or bars that dictate the type of graph we want to see.

In an introductory course, we won't get too fancy with these graphs. But be aware that there's a whole field of data visualization that studies clear and interesting ways to understand data graphically.

It will be easier to explain the `ggplot` syntax in the context of specific graph types, so let's proceed to the next section and start looking at ways to graph numerical data.

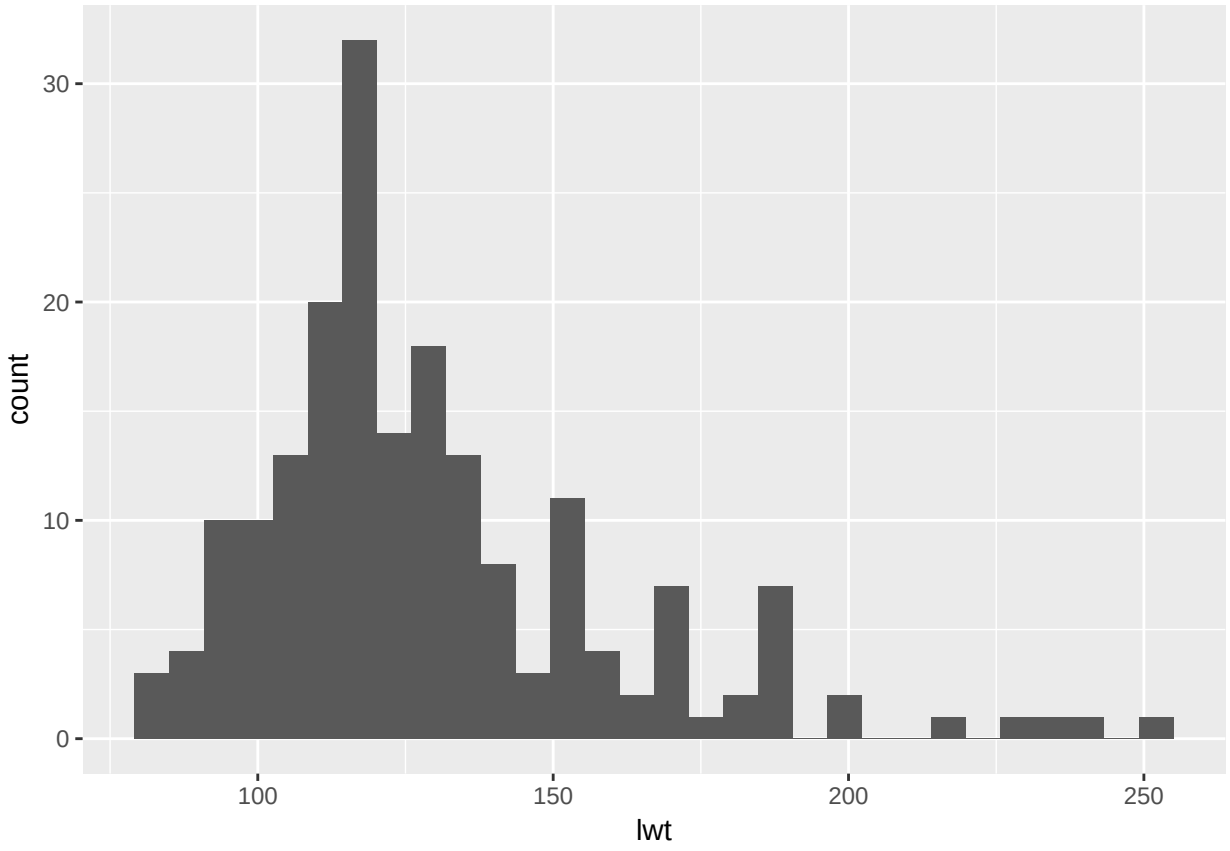
Graphing one numerical variable

From the birth weight data `birthwt`, let's consider the weight of the mother at her last menstrual period. This is clearly a numerical variable.

The single most useful display of a single numerical variable is a histogram. Here is the `ggplot` command to do that:

```
ggplot(birthwt, aes(x = lwt)) +
    geom_histogram()
```

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



Let's walk through this syntax step by step. The first argument of the `ggplot` command is the name of the data frame, in this case, `birthwt`. Next we define the aesthetics using `aes` and parentheses. Inside the parentheses, we assign any variables we want to plot to aesthetics of the graph. For this analysis, we are only interested in the variable `lwt` and for a histogram, the numerical variable typically goes on the x-axis. That's why it says `x = lwt` inside the `aes` argument. Next, `ggplot` needs to know what kind of graph we want. Graph types are called "geoms" in the `ggplot` world, and `geom_histogram()` tells `ggplot` to add a histogram layer. (Adding a layer is accomplished by literally typing a plus sign.)

Binwidths and boundaries

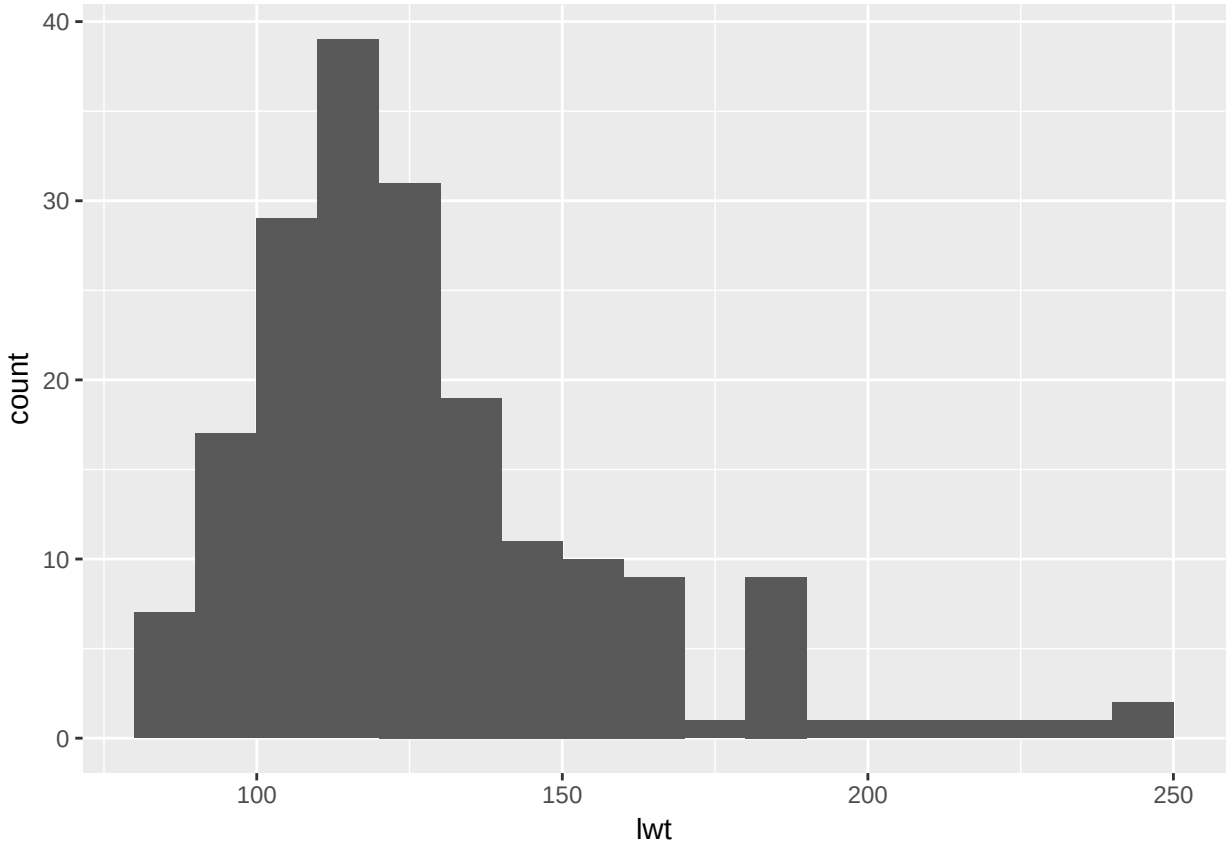
Generally, the default binning for `ggplot` histograms is not so great. In fact, if you look at the output from the graphing command above, you can see that `ggplot` informs you that you should pick a better value. You can also see that the bins aren't ideal. They are too narrow, which means that arbitrary differences between bins show up as "random" spikes all over the graph.

Instead, we should aim to use bins that show the overall shape of the data and smooth it out a bit. Look back at the scale of the x-axis to assess how wide each bar should be. There's no one correct answer. In this case, the bins ought to be a little wider. Since our x-axis goes from about 75 to 250, maybe we should try binwidths of 10. And if 10 doesn't look good, nothing prevents us from trying a different number.

It's also easier to interpret the histogram when the bins' edges line up with numbers that are easy to see in the plot. Use `boundary` to determine where you want one of the bin boundaries to fall. For example, if we set the boundary to 100, that means that one bar will start with its left edge at 100. The latter number is pretty arbitrary; once one boundary is set, it determines where all the other bins will line up. With a binwidth of 10, we'd get the same graph if the boundary were set to 110 or 150, or any other multiple of 10.

We use `binwidth` and `boundary` inside the parentheses of the `geom_histogram` to modify these parameters.

```
ggplot(birthwt, aes(x = lwt)) +
    geom_histogram(binwidth = 10, boundary = 100)
```



Exercise

Write a paragraph or so describing the shape of the distribution of the `lwt` variable, focusing on the three key features (modes, symmetry, and outliers). Be sure to speak about these in the context of the data; in other words, your answer should refer to women and their weight, and not just abstract numbers and stats words.

ANSWER

Please write up your answer here.

Your turn

Create a histogram of the baby's birth weight (in grams). Adjust the binwidth and boundary to see the shape of the distribution more clearly. Then describe the shape of the distribution as you did for the `lwt` variable above.

ANSWER

```
# Add code here to create a histogram for the distribution of
# the baby's birth weight (in grams).
# Don't forget to adjust the binwidth and boundary.
```

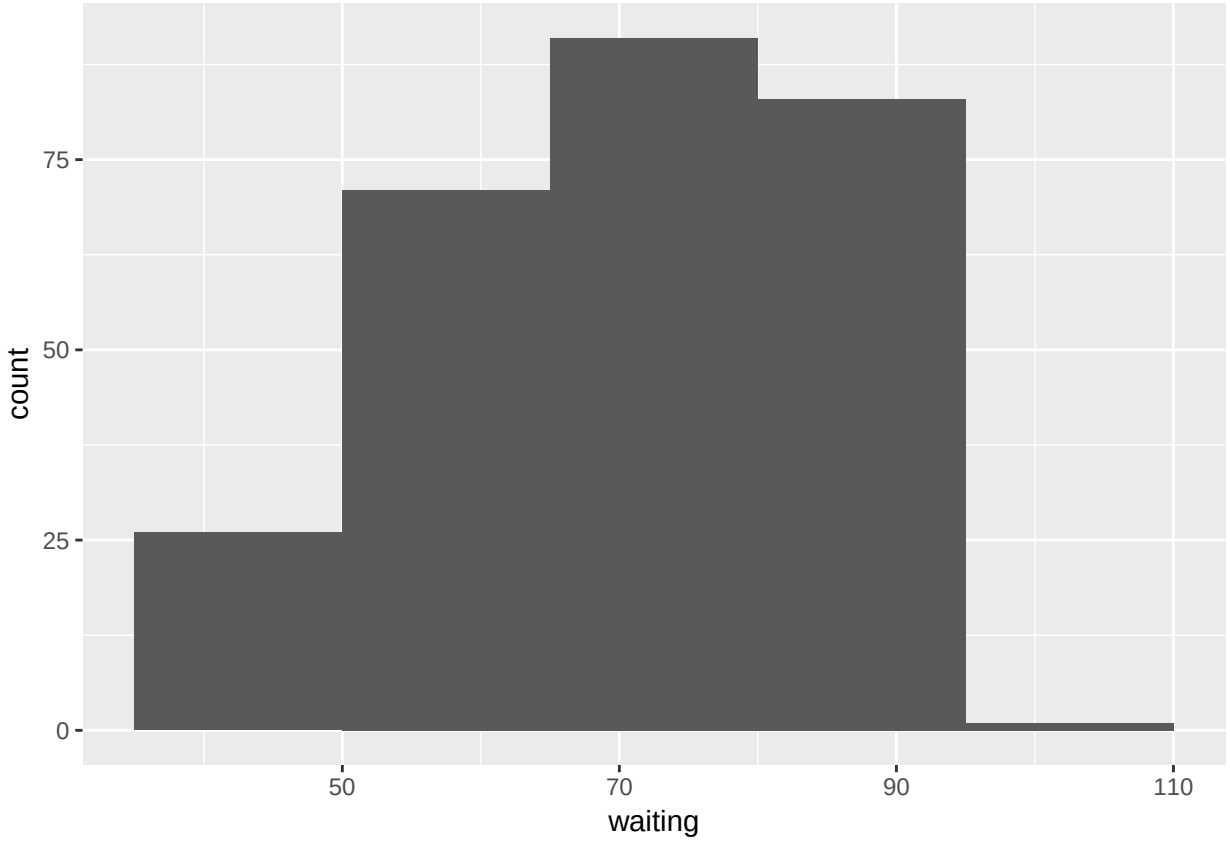
Please write up your answer here.

Exercise

The `faithful` data set has a variable called `waiting` that records the waiting times (in minutes) between eruptions of the Old Faithful geyser in Yellowstone National Park.

Here is a histogram of those eruptions.

```
ggplot(faithful, aes(x = waiting)) +
    geom_histogram(binwidth = 15, boundary = 50)
```



Write a paragraph or so describing the shape of the distribution of waiting times, focusing on the three key features (modes, symmetry, and outliers).

ANSWER

Please write up your answer here.

This is a trick question! According to one website[1],

[1] http://www.geyserstudy.org/geyser.aspx?pGeyserNo=OLDFAITHFUL

> "Old Faithful is currently bimodal. It has two eruption durations, either a long (over 4 minutes) or more rarely a short (about 2-1/2 minutes). Short eruptions lead to an interval of just over an hour and long eruptions lead to an interval of about 1-1/2 hours."

Change the binwidth (no need to change the boundary) to something more sensible to see the bimodal nature of the distribution.

ANSWER

```
# Change the binwidth of the last histogram to see
# see the bimodal nature of the distribution.
```

Less useful plot types

There are two other graph types that one might see for a single numerical variable: dotplots and boxplots. I'm not a big fan of dotplots as they are just a messier version of histograms. I do like boxplots, but they are typically less informative than histograms. Boxplots are much better for comparing groups, so we'll see them in another module. Besides, there isn't an easy way to make a boxplot for a single numerical variable using `ggplot`.

Graphing two numerical variables

The proper graph for two numerical variables is a scatterplot. We graph the response variable on the y-axis and the explanatory variable on the x-axis.

Exercise

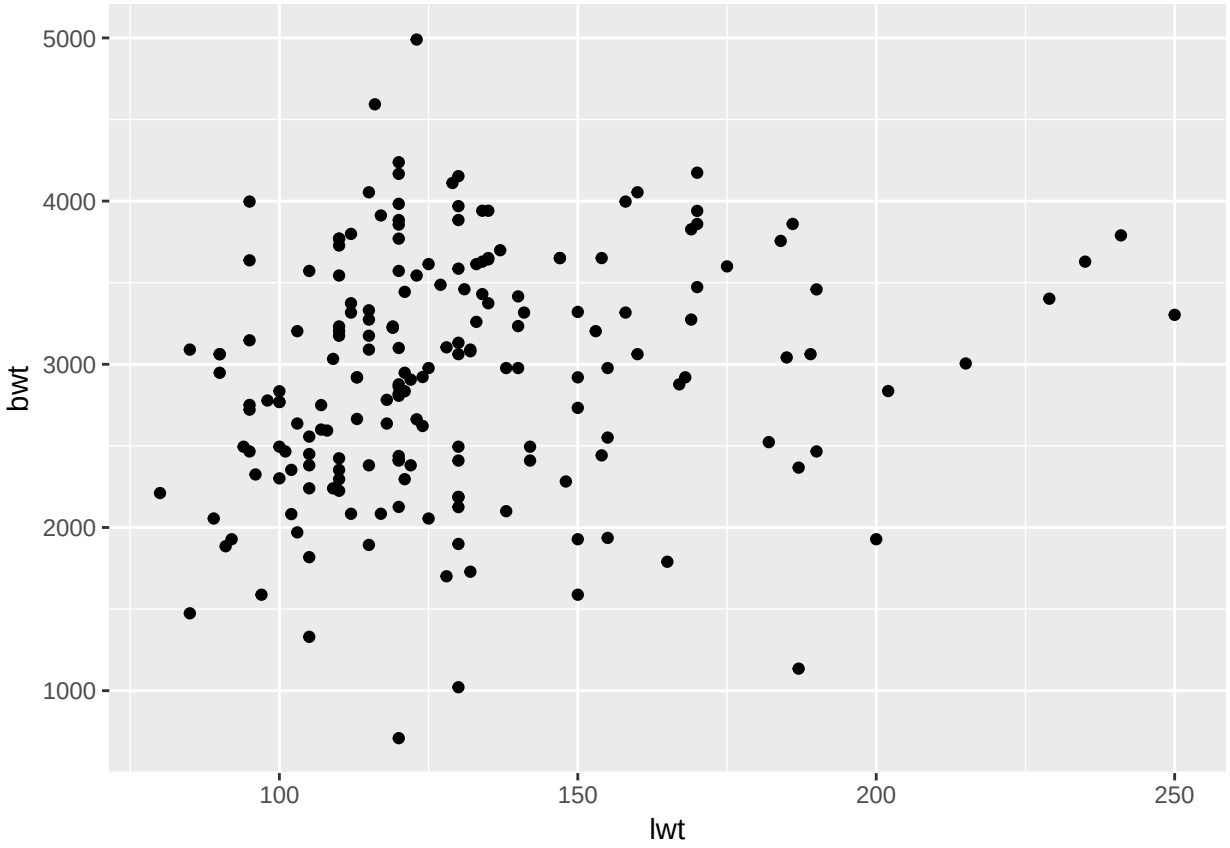
If you were interested in exploring a possible association between the weight of the mother at her last menstrual period and the birth weight of the baby, which variable would you consider to be the response variable and which would be the explanatory variable? Explain your reasoning. Be careful not to use language that suggests cause or effect. There may, in fact, be a causal relationship, but that is not going to be proven from observational data. Instead, when there is an association, we say that the explanatory variable might be used to "predict" the value of the response variable.

ANSWER

Please write up your answer here.

Now we'll create a scatterplot of the birth weight of the baby and the weight of the mother at her last menstrual period. Since we are now plotting two variables, we have two aesthetics, one on the y-axis (the response variable) and one on the x-axis (the explanatory variable). Since scatterplots use points to plot each data value, the correct layer to add is `geom_point()`.

```
ggplot(birthwt, aes(y = bwt, x = lwt)) +
    geom_point()
```



Exercise

Comment on the nature of the association. (Is is positive/negative, or are these two variables independent?) As always, be sure to word your answer in the context of the data.

ANSWER

Please write up your answer here.

Your turn

Consider the two variables `bwt` and `age`. Which variables would you consider as response and explanatory?

ANSWER

Please write up your answer here.

Now create a scatterplot to visualize the relationship between the birth weight of the baby and the mother's age. Be careful to make sure you put the variables on the right axes!

ANSWER

```
# Add code here to create a scatterplot using bwt and age.
```

Now comment on the nature of the association.

ANSWER

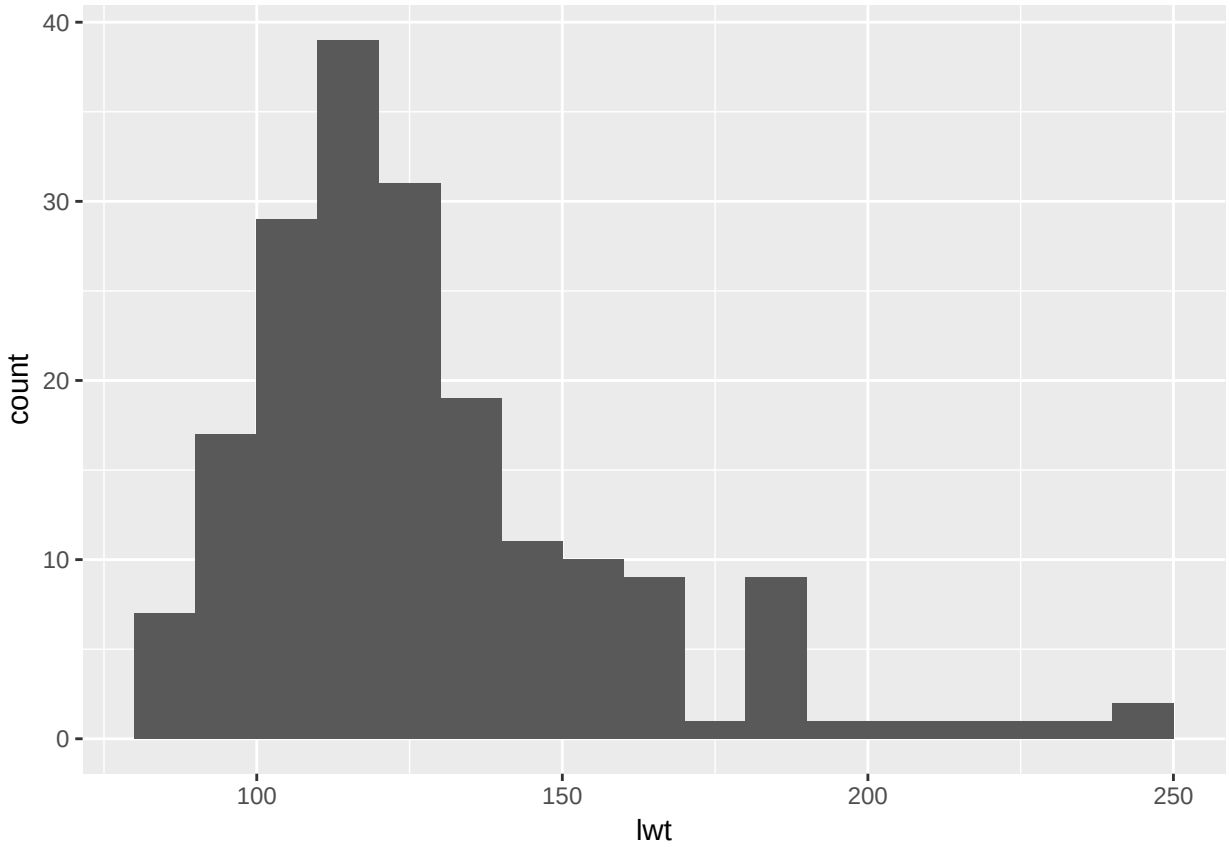
Please write up your answer here.

Publication-ready graphics

The great thing about `ggplot2` graphics is that they are already quite pretty. To take them from exploratory data analysis to the next level, there are a few things we can do to tidy them up.

Let's go back to the first histogram from this module.

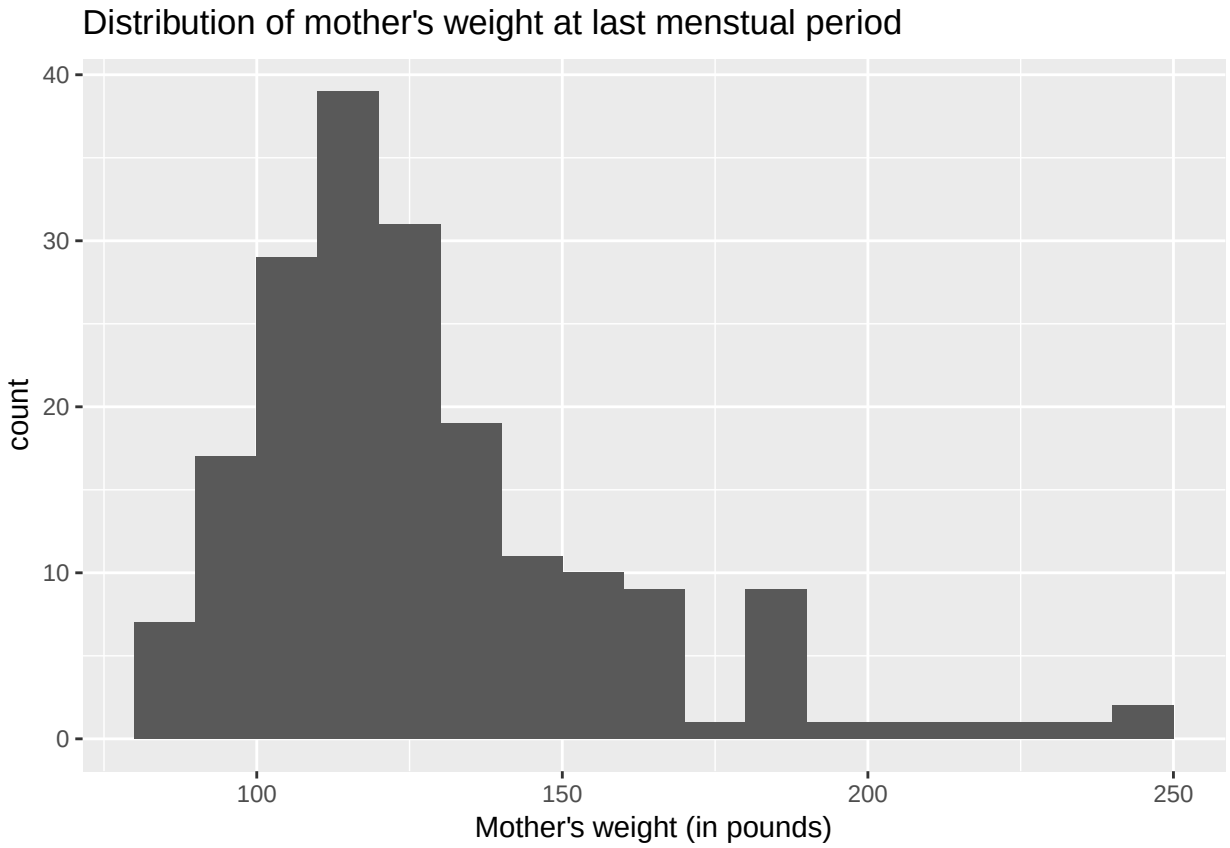
```
ggplot(birthwt, aes(x = lwt)) +
    geom_histogram(binwidth = 10, boundary = 100)
```



Note that the variable names of this data set are not terribly informative. In other words, if you were using

this graph in a publication or presentation for an audience, they would have no idea what `lwt` was. Also note that this graph could use a title. We can do all this with `labs` (for labels). Observe:

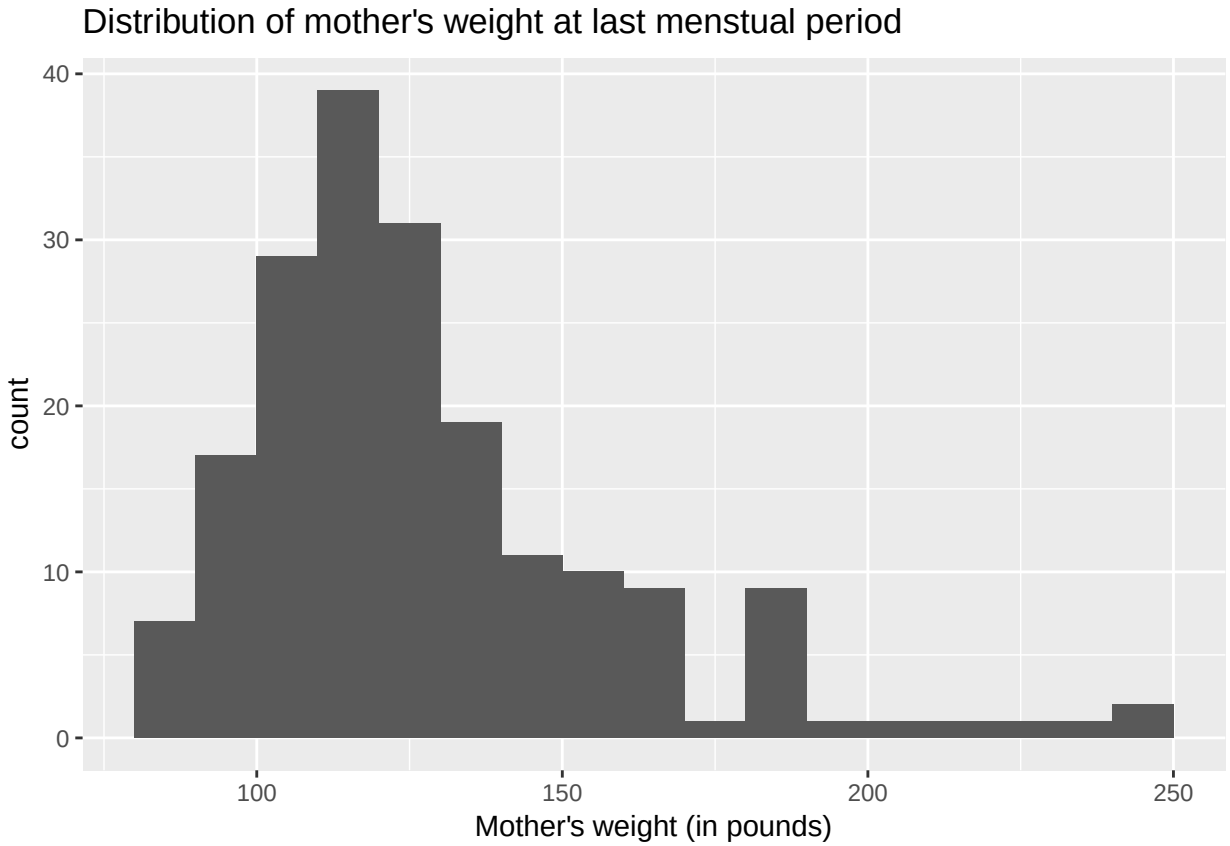
```
ggplot(birthwt, aes(x = lwt)) +
    geom_histogram(binwidth = 10, boundary = 100) +
    labs(title = "Distribution of mother's weight at last menstual period",
         x = "Mother's weight (in pounds)")
```



You can also see that we took the opportunity to mention the units of measurement (pounds) for our variable in the x-axis label. This is good practice.

A quick note about formatting in R code chunks. Notice that I put different parts of the last `ggplot` command on their own separate lines. The command would still work if I did this:

```
ggplot(birthwt, aes(x = lwt)) + geom_histogram(binwidth = 10, boundary = 100) + labs(title = "Distributi
```



In the R Markdown document, the code chunk "wraps" to the next line so it's all visible. But now knit the document and look at the PDF version of the code chunk above.

Do you see how it runs off the right edge of the page?

Now imagine your PDF output is being graded and the grader can only see the first few lines of code before it runs off the edge of the paper. Do you think you will earn full points for code that the grader can't even see?

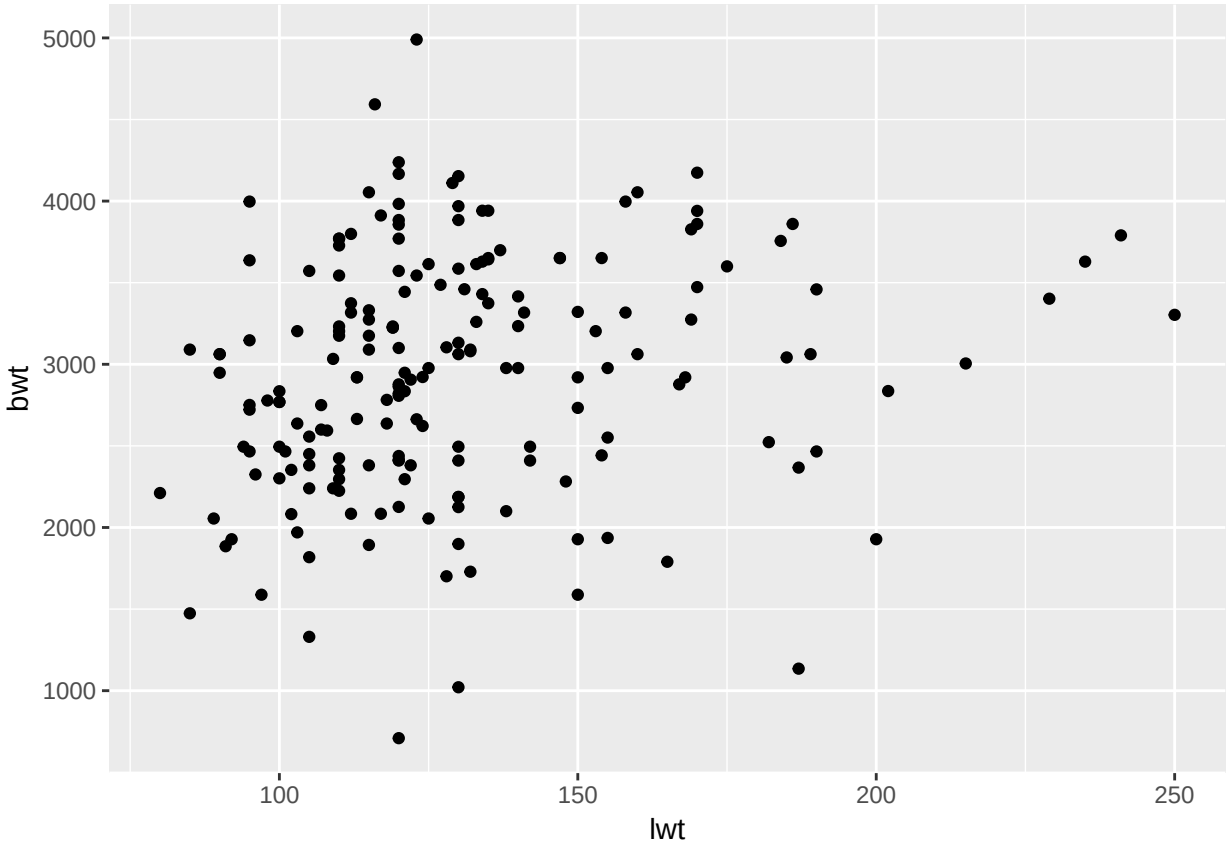
The moral of the story is this: use line breaks judiciously to format your code. If it wraps to the next line in RStudio, there's a good chance it will run off the page in the PDF.

Exercise

Modify the following scatterplot by adding a title and labels for both the y-axis and x-axis.

ANSWER

```
# Modify the following scatterplot by adding a title and
# labels for both the y-axis and x-axis.
ggplot(birthwt, aes(y = bwt, x = lwt)) +
    geom_point()
```



Every part of the graph can be customized, from the color scheme to the tick marks on the axes, to the major and minor grid lines that appear on the background. We won't go into all that, but you can look at the ggplot2 documentation online and search Google for examples if you want to dig in and figure out how to do some of that stuff. However, the default options are often (but not always) the best, so be careful that your messing around doesn't inadvertently make the graph less clear or less appealing.

Conclusion

The `ggplot2` package with its `ggplot` command is a very versatile tool for creating nice graphs relatively easily. For a single numerical variable, the standard graph type is a histogram. For two numerical variables, use a scatterplot.