STAT 33A Workbook 5

CJ HINES (3034590053)

Oct 1, 2020

This workbook is due **Oct 1, 2020** by 11:59pm PT.

The workbook is organized into sections that correspond to the lecture videos for the week. Watch a video, then do the corresponding exercises *before* moving on to the next video.

Workbooks are graded for completeness, so as long as you make a clear effort to solve each problem, you'll get full credit. That said, make sure you understand the concepts here, because they're likely to reappear in homeworks, quizzes, and later lectures.

As you work, write your answers in this notebook. Answer questions with complete sentences, and put code in code chunks. You can make as many new code chunks as you like.

In the notebook, you can run the line of code where the cursor is by pressing `Ctrl` + `Enter` on Windows or `Cmd` + `Enter` on Mac OS X. You can run an entire code chunk by clicking on the green arrow in the upper right corner of the code chunk.

Please do not delete the exercises already in this notebook, because it may interfere with our grading tools.

You need to submit your work in two places:

- Submit this Rmd file with your edits on bCourses.
- Knit and submit the generated PDF file on Gradescope.

If you have any last-minute trouble knitting, **DON'T PANIC**. Submit your Rmd file on time and follow up in office hours or on Piazza to sort out the PDF.

R Graphics Overview

Watch the "R Graphics Overview" lecture video.

No exercises for this section. Keep at it!

The Tidyverse

Watch the "The Tidyverse" lecture video.

Exercise 1

How many packages are in the Tidyverse? Explore the website to find out. You can count the tidymodels packages as a single package.

YOUR ANSWER GOES HERE:

> There are 22 packages in Tidyverse.

Tibbles

Watch the "Tibbles" lecture video.

Exercise 2

1. Read the documentation for the tibble package on the website. What's the name of the function that creates a new tibble from column vectors?
2. Create a tibble with 4 rows and 3 columns. You can make up the data in the columns, but use a different data type for each one.
3. Show how to convert the tibble from step 2 into an ordinary data frame.

YOUR ANSWER GOES HERE:

1. The function that creates a new tibble from column vectors is `tibble()`.

```
library(tibble)
tibble()
```

```
## # A tibble: 0 x 0
```

2.

```
atib = tibble(x = 1:4, y = c(3.3, 4.0, 5.0, 6.2), z = "canada")
atib
```

```
## # A tibble: 4 x 3
##       x     y z
##   <int> <dbl> <chr>
## 1     1   3.3 canada
## 2     2   4   canada
## 3     3   5   canada
## 4     4   6.2 canada
```

3.

```
class(atib)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

```
class(as.data.frame(atib))
```

```
## [1] "data.frame"
```

The Grammar of Graphics

Watch the "The Grammar of Graphics" lecture video.

For exercises that mention the dogs data, you can use either `dogs.rds` or `dogs_tibble.rds` from the bCourse (do not use `dogs_sample.rds`).

Exercise 3

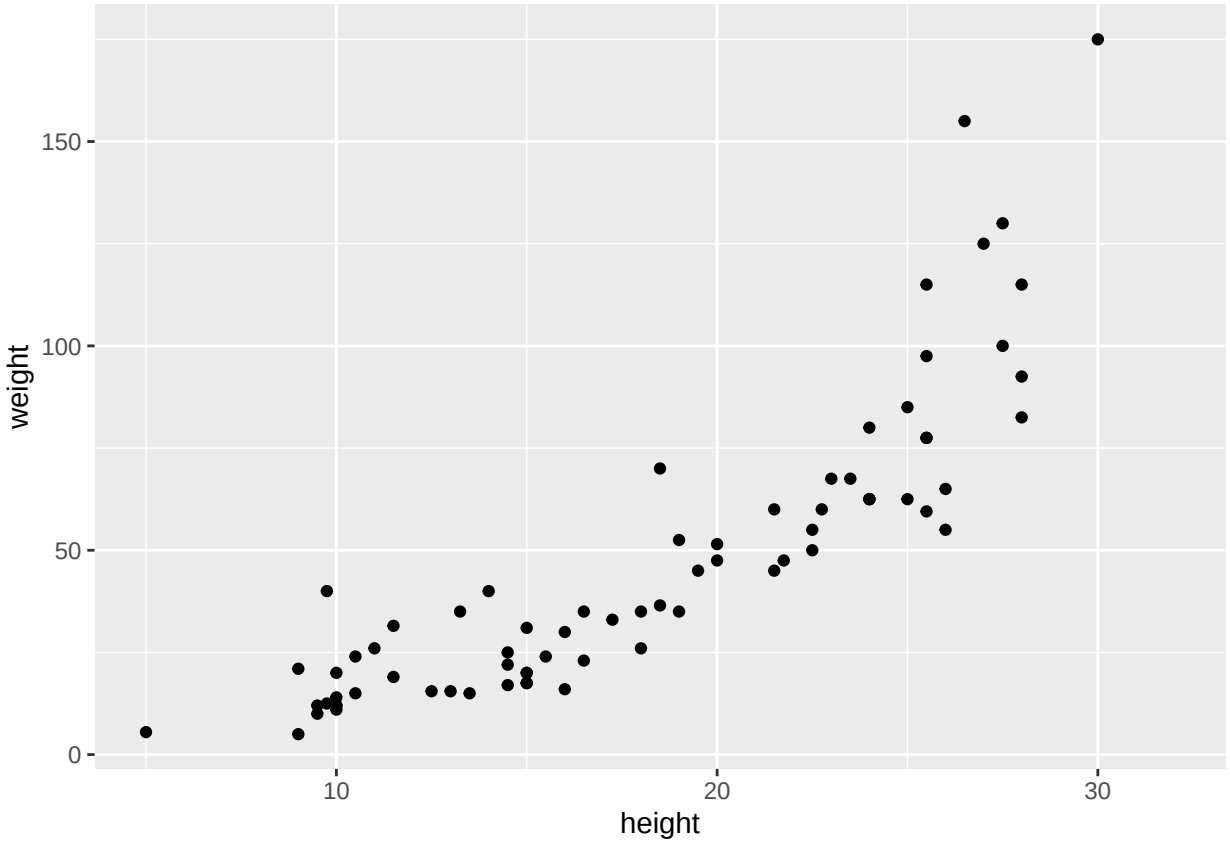
Use ggplot2 and the dogs data to make a scatterplot that shows the relationship between height and weight. In 2-3 sentences, discuss any patterns you see in the plot.

YOUR ANSWER GOES HERE:

```
library(ggplot2)
dogs = readRDS("dogs.rds")

ggplot(dogs, aes(x = height, y = weight)) + geom_point()
```

```
## Warning: Removed 98 rows containing missing values (geom_point).
```



A pattern from the plot is that as height increases, weight increases. It's an almost linear graph.

Exercise 4

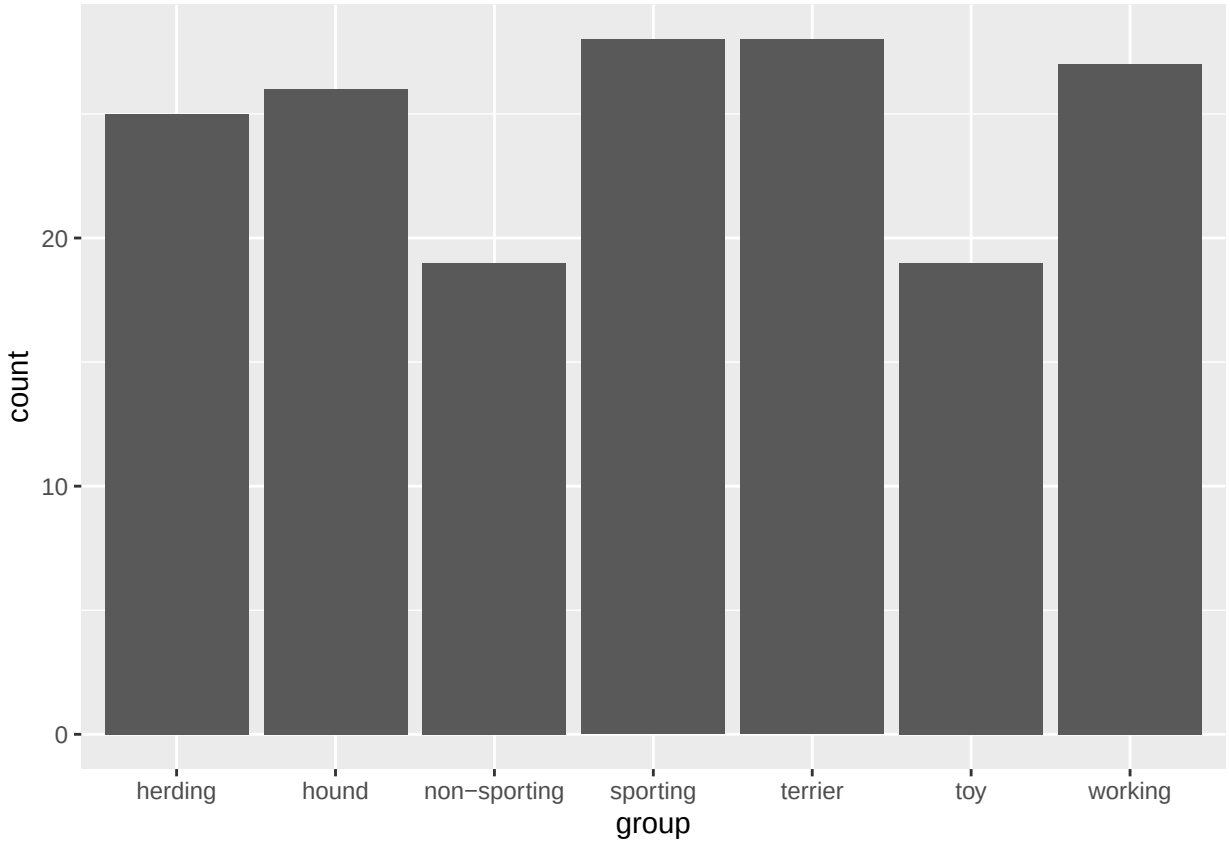
The `geom_bar()` geometry function makes a bar plot. Even though a bar plot has two axes, by default `geom_bar()` only requires one aesthetic: `x`. The y-axis is computed automatically and shows the frequencies of the values on the x-axis. This makes `geom_bar()` especially convenient for displaying categorical data.

1. Make a bar plot that shows the number of dogs in each "group" of dogs.
2. Are any groups much larger or smaller than the others?

YOUR ANSWER GOES HERE:

1.

```
ggplot(dogs, aes(x = group)) + geom_bar()
```



2. The sporting,terrier, and working groups are the largest groups. The non-sporting and toy groups are the smallest.

Exercise 5

1. Which geometry function makes a histogram? Use the ggplot2 website or cheat sheet to find out.
2. Use ggplot2 to make a histogram of longevity for the dogs data. How long do most dogs typically live? How spread out is the distribution of lifespans?

   Hint 1: If you're not familiar with histograms, OpenIntro Statistics 4th Edition (link on Piazza) explains how to interpret them in Section 2.1.3.

   Hint 2: Describe how "spread out" the distribution is at whatever level of statistics you're comfortable with. Range is the simplest, but standard deviation or other statistics are also fine.

YOUR ANSWER GOES HERE:

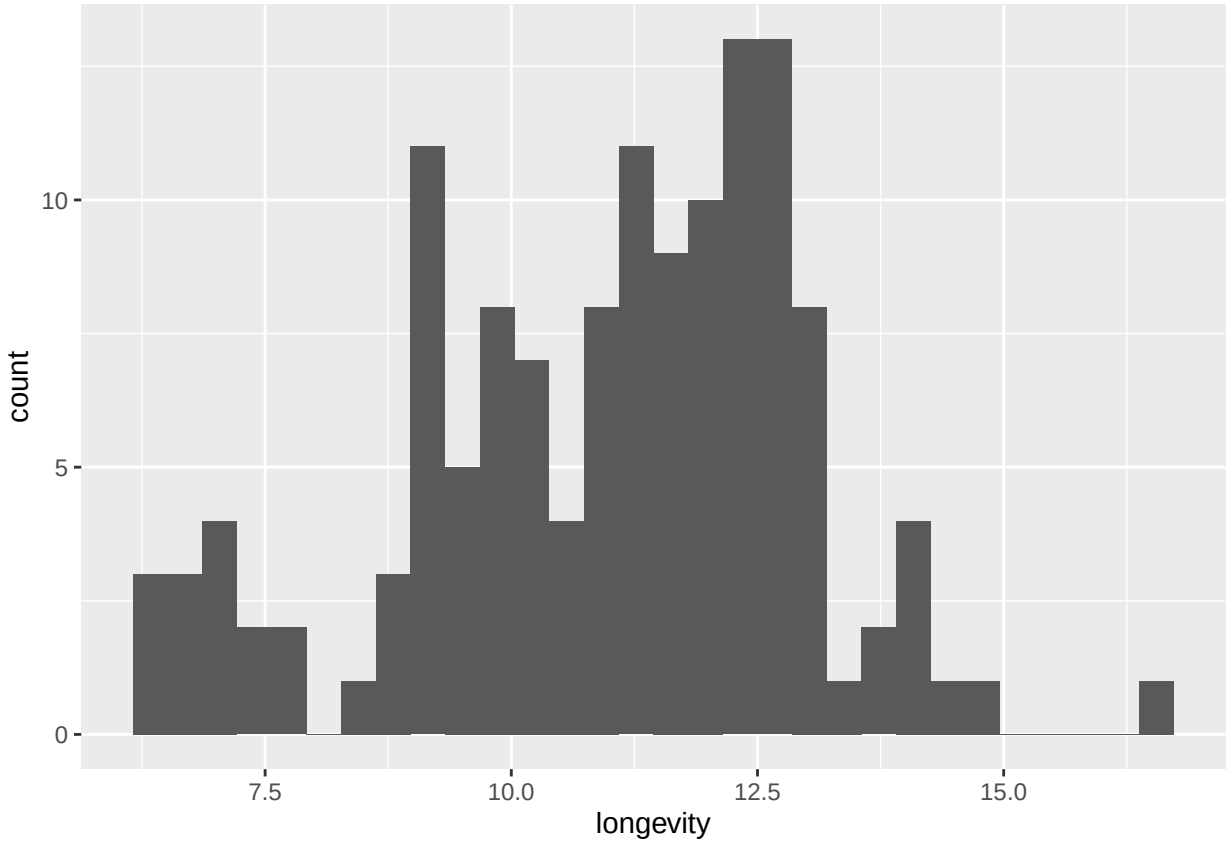
1. `geom_histogram()`

2. Most dogs typically live around 9-12 years. The spread out of the distribution of lifespans is 10.21.

```
ggplot(dogs, aes(x = longevity)) + geom_histogram()
```

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

```
## Warning: Removed 37 rows containing non-finite values (stat_bin).
```




```
range(dogs$longevity, na.rm = TRUE)
```

```
## [1]  6.29 16.50
```

```
#sort(dogs$longevity)
```

Saving Plots

Watch the "Saving Plots" lecture video.

No exercises for this section. Almost done!

Customizing Plots

Watch the "Customizing Plots" lecture video.

Exercise 6

1. Modify your plot from Exercise 3 so that the shape of the points is determined by the "group" of the dog.

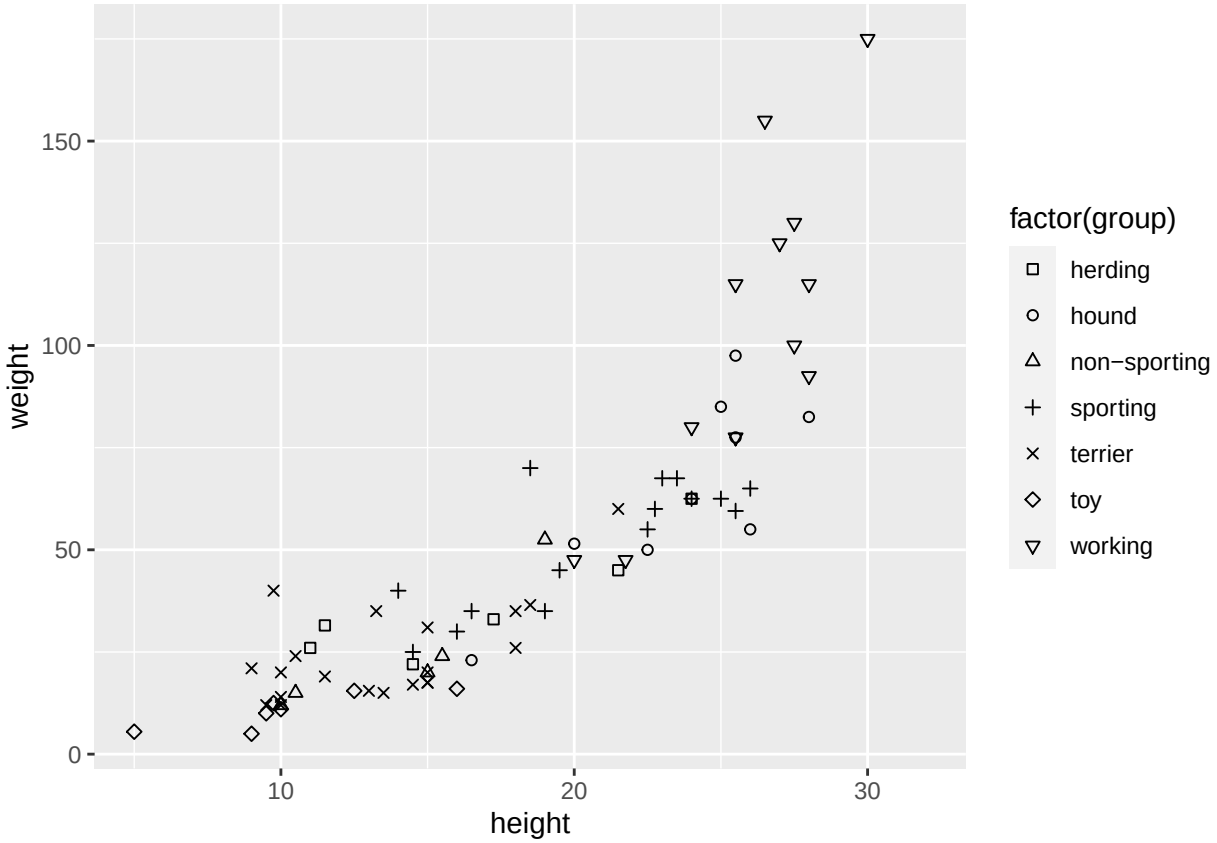
   Hint: To do this, you need to set another aesthetic. The documentation for `geom_point()` *includes an example that shows how to change the shape of the points.*

YOUR ANSWER GOES HERE:

1.

```
ggplot(dogs, aes(x = height, y = weight, shape = factor(group))) + geom_point() + scale_shape_manual(va
```

```
## Warning: Removed 98 rows containing missing values (geom_point).
```



Exercise 7

1. Modify your plot from Exercise 6 so that the color of the points is *also* determined by the "group" of the dog.
2. Features that separate different categories of observations are useful for building models that predict those categories.

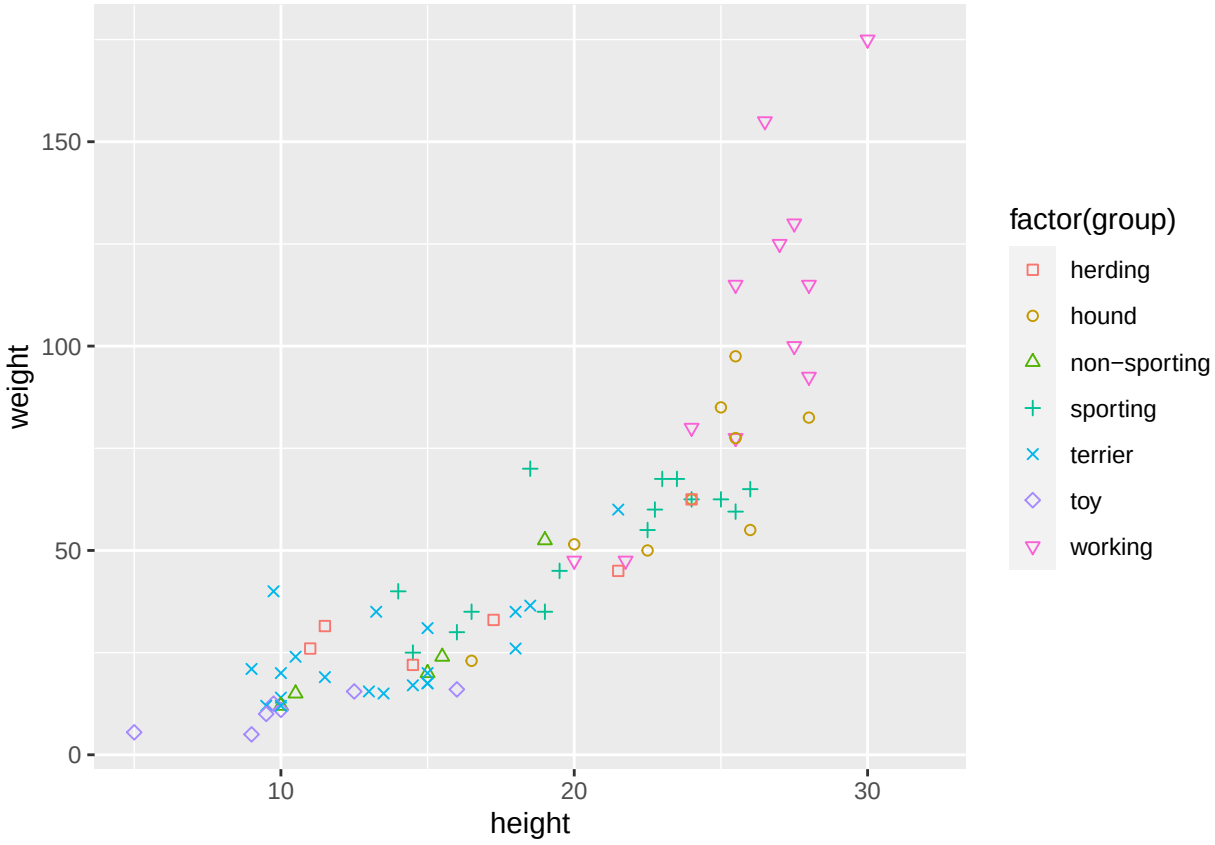
   Do height and weight effectively separate the different groups of dogs? In other words, are there clear boundaries between the groups in the plot (as opposed to being mixed together)? Are some groups better separated than others?

YOUR ANSWER GOES HERE:

1.

```
ggplot(dogs, aes(x = height, y = weight, shape = factor(group))) + geom_point(aes(color = factor(group)))
```

```
## Warning: Removed 98 rows containing missing values (geom_point).
```



2. Height and weight separate some groups more effectively than others. For example, dogs of the working group are mostly very heavy and tall and dogs of the toy group are mostly short and light. For dog groups like hound and non-sporting, though, there is a bit more uncertainty and mix.