L08 Layers

Data Visualization (STAT 302)

Anna Wagman

Contents

```r
library(knitr)
opts_chunk$set(warning = FALSE, message = FALSE, dpi = 300)
```

Overview

The goal of this lab is to explore more plots in `ggplot2` and continue to leverage the the use of various layers to build complex and well annotated plots.

Datasets

We'll be using the `tech_stocks.rda` dataset which is already in the `/data` subdirectory in our **data_vis_labs** project. We have a new dataset, `NU_admission_data.csv`, which will need to be downloaded and added to our `/data` subdirectory. We will also be using the `mpg` dataset which comes packaged with `ggplot2` — use `?ggplot2::mpg` to access its codebook.

```r
# Load package(s)
library(tidyverse)
library(skimr)
library(ggplot2)
library(lubridate)
library(dplyr)
library(cowplot)
library(janitor)

#seed
set.seed(9876)

# Load datasets

load(file = "data/tech_stocks.rda")
NU_dat <- read.csv("data/NU_admission_data.csv") %>%
  clean_names()
```

Exercises

Complete the following exercises.

Exercise 1

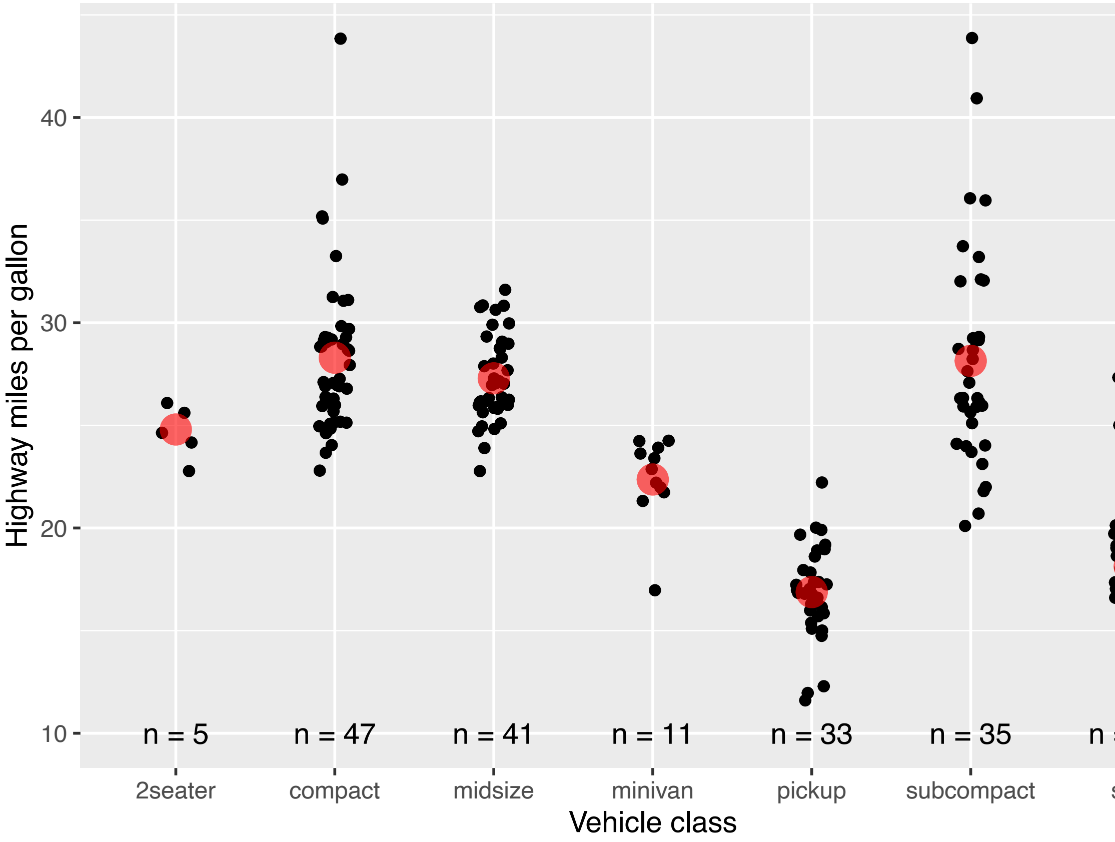
Using `mpg` and the `class_dat` dataset created below, recreate the following graphic as precisely as possible in two different ways.

Hints:

- Transparency is 0.6
- Horizontal position adjustment is 0.1
- Larger points are 5

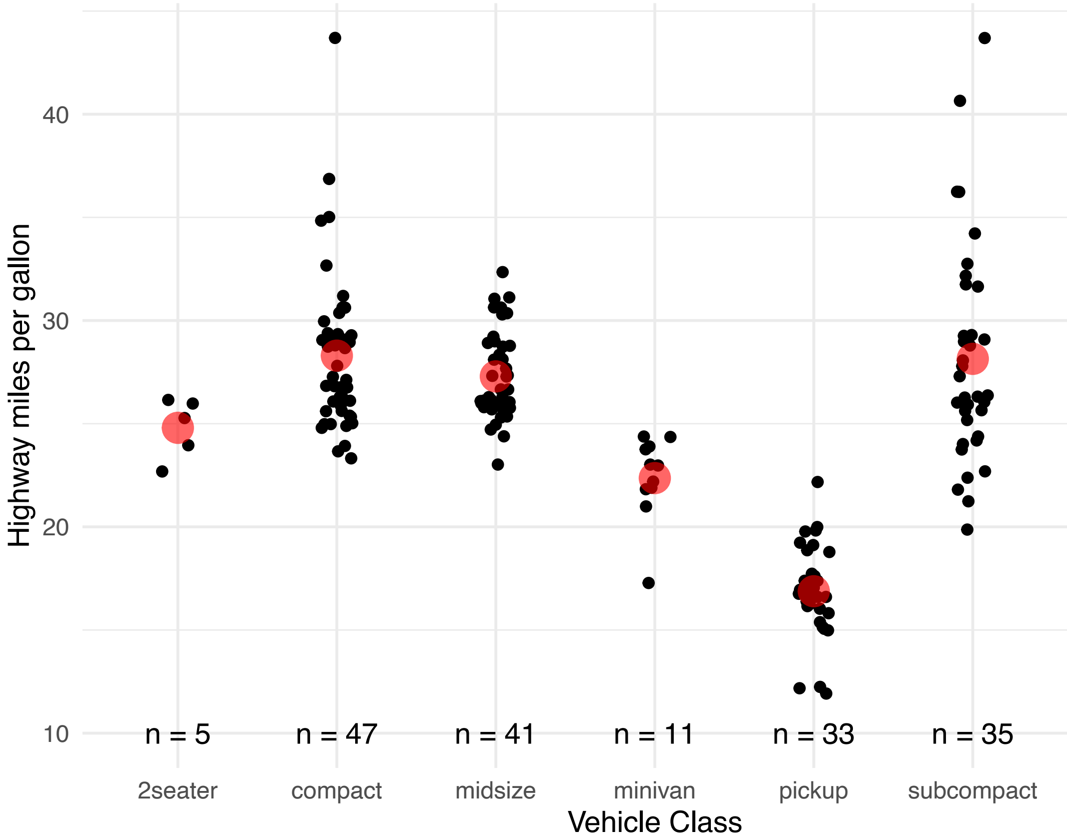
```
# Additional dataset for plot
class_dat <- mpg %>%
  group_by(class) %>%
  summarise(
    n = n(),
    mean_hwy = mean(hwy),
    label = str_c("n = ", n, sep = "")
  )
```

```
ggplot(data = mpg, aes(class, hwy)) +
  ##add the black points
  #horiztonal position adj = .1
  geom_jitter(width = 0.1) +
  ##red dots
  stat_summary(geom = "point", fun.y = mean, colour = "red", size = 5, alpha = 0.6) +
  geom_text(data = class_dat, aes(y = 10, label = label)) +
  labs(x = "Vehicle class", y = "Highway miles per gallon")
```



Plot 1 – using `mean_hwy`

```r
ggplot(mpg, aes(class, hwy)) +
  geom_jitter(width = 0.1) +
  geom_point(
    stat = "summary",
    fun = mean,
    color = "red",
    alpha = 0.6,
    size = 5
  ) +
  geom_text(
    data = class_dat,
    mapping = aes(y = 10, label = label)
  ) +
  labs(
    x = "Vehicle Class",
    y = "Highway miles per gallon"
  ) +
  theme_minimal()
```



Plot 2 – not using `mean_hwy`

Exercise 2

Using `perc_increase` dataset derived from the `tech_stocks` dataset, recreate the following graphic as precisely as possible.

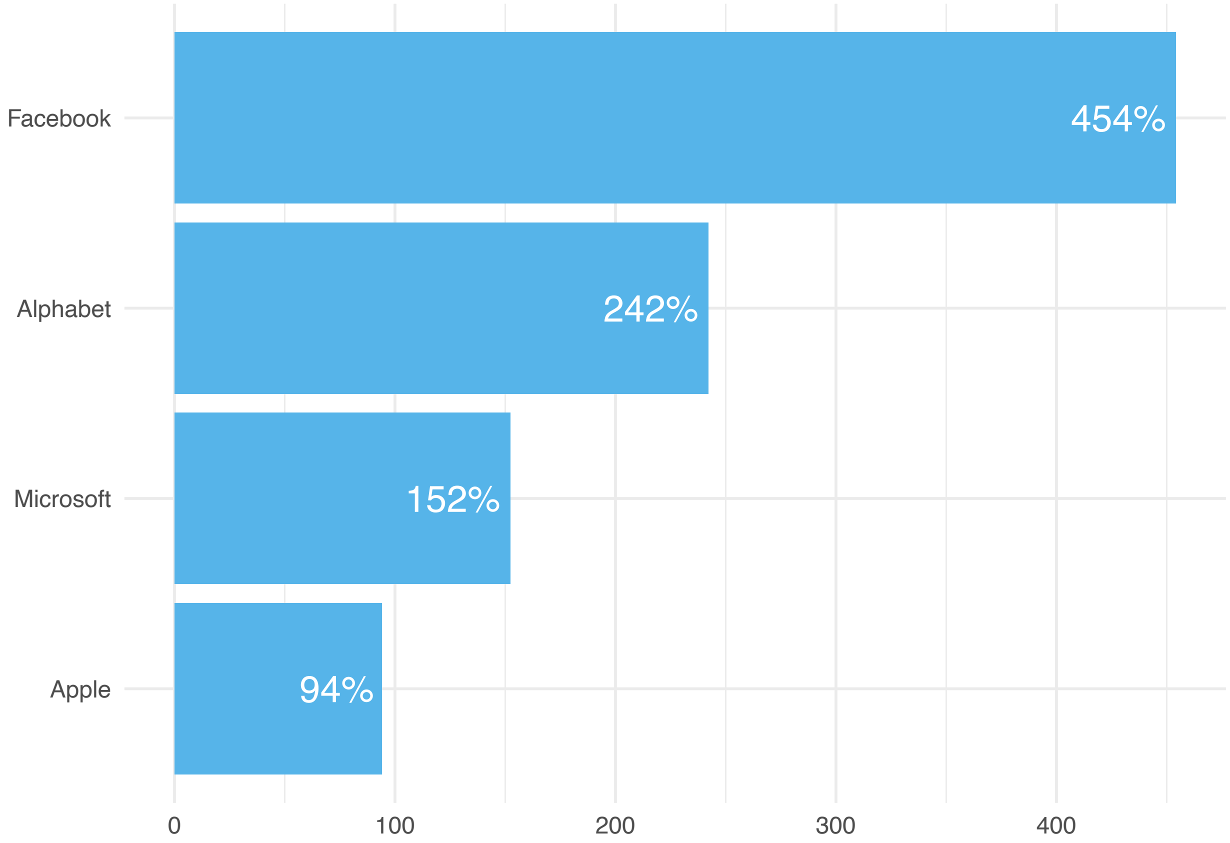
Hints:

- Hex color code `#56B4E9`

- Justification of 1.1
- Size is 5

```
# percentage increase data
perc_increase <- tech_stocks %>%
  arrange(desc(date)) %>%
  distinct(company, .keep_all = TRUE) %>%
  mutate(
    perc = 100 * (price - index_price) / index_price,
    label = str_c(round(perc), "%", sep = ""),
    company = fct_reorder(factor(company), perc)
  )
```

```
ggplot(perc_increase, aes(perc, company)) +
  #color bars
  geom_col(fill = "#56B4E9") +
  #label percentages
  geom_text(aes(label = label), size = 5, hjust = 1.1, color = "white") +
  theme_minimal() +
  #remove axes labels set by default
  labs(x = NULL, y = NULL)
```



Exercise 3

We will need to do some data wrangling, which we will do together in class

---

Using `NU_admission_data.csv` create two separate plots derived from the single plot depicted in `undergraduate-admissions-statistics.pdf` — this visual and data has been collected from https://www.adminplan.northwestern.edu/ir/data-book/. They overlaid two plots on one another by using two y-axes. Create two separate plots that display the same information instead of trying to put it all in one single plot — consider stacking them using `patchwork` or `cowplot`. There is one major error they make with the bars in their graphic. Explain what it is.

When including detailed labeling like this take care to pick label fonts and colors so the text doesn't take away the from the message of the data (the trend in these plots). With these labels you could image removing the y-axes altogether so they don't distract the reader/consumer.

Which approach do you find communicates the information better, their single plot or the two plot approach? Why?

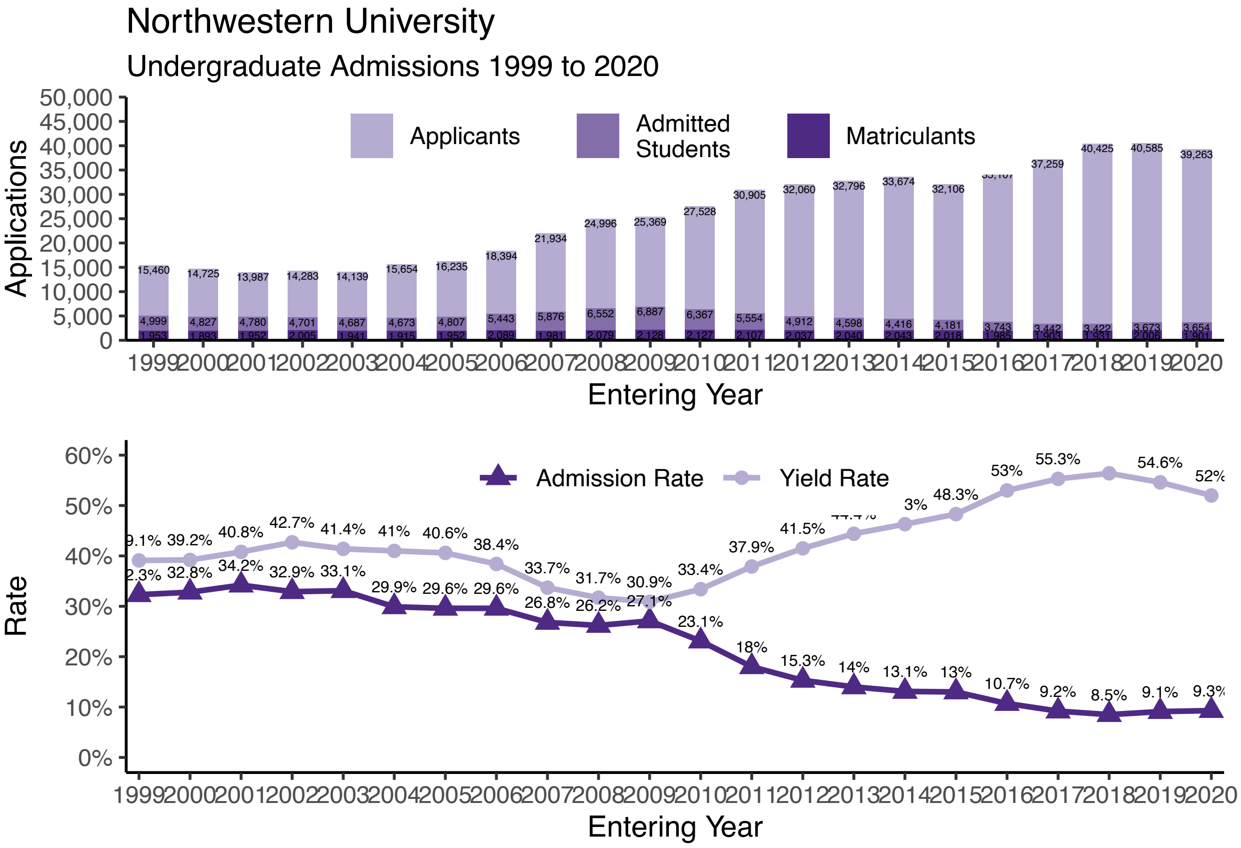
Hints:

- Form 4 datasets (helps you get organized, but not entirely necessary):
  - 1 that has bar chart data,
  - 1 that has bar chart label data,
  - 1 that has line chart data, and
  - 1 that has line chart labels
- Consider using `ggsave()` to save the image with a fixed size so you it is easier to pick font sizes.

```r
#graph data as bar graph
bar_dat <- NU_dat %>%
  mutate(
    cat_a = applications - admitted_students,
    cat_b = admitted_students - matriculants,
    cat_c = matriculants
  ) %>%
  select(year, contains("cat_")) %>%
  pivot_longer(
    cols = -year,
    names_to = "category",
    values_to = "value"
    )

#add labels to bars
bar_labels <- NU_dat %>%
  select(-contains("rate")) %>%
  pivot_longer(
    cols = -year,
    names_to = "category",
    values_to = "value"
    ) %>%
  mutate(
    col_label = prettyNum(value, big.mark = ",")
  )

##ggplot the bar plot
bar_plot <- ggplot(bar_dat, aes(year, value)) +
  geom_col(width = 0.6, aes(fill = category)) +
  geom_text(
    data = bar_labels,
    mapping = aes(label = col_label),
    size = 1.3,
    color = "black",
    vjust = 1,
    nudge_y = -200
    ) +
  scale_y_continuous(
    name = "Applications",
    breaks = seq(0, 50000, 5000),
    limits = c(0, 50000),
    labels = scales::label_comma(),
    expand = c(0, 0)
  ) +
  scale_x_continuous(
    name = "Entering Year",
    breaks = 1999:2020,
    expand = c(0, .25)
  ) +
  scale_fill_manual(
    name = NULL,
    values = c("#B6ACD1", "#836EAA", "#4E2A84"),
    labels = c("Applicants      ",
               "Admitted\nStudents      ",
               "Matriculants      "
               )
  ) +
  theme_classic() +
  theme(
    legend.justification = c(0.5, 1),
    legend.position = c(0.5, 1),
    legend.direction = "horizontal"
  ) +
  labs(
    title = "Northwestern University",
    subtitle = "Undergraduate Admissions 1999 to 2020"
  )
```

```
#Data percentages
rate_dat <- NU_dat %>%
  select(year, contains("_rate")) %>%
  gather(key = rate_type, value = pct, -year)

#Labels
rate_label <- rate_dat %>%
  mutate(pct_label = str_c(pct, "%"))

##Line plot
line_plot <- rate_dat %>%
  ggplot(
    aes(year, pct, color = rate_type)
    ) +
  geom_point(aes(shape = rate_type), size = 3) +
  geom_line(size = 1) +
  scale_shape_manual(
    name = NULL,
    labels = c("Admission Rate", "Yield Rate"),
    values = c(17,20),
    guide = guide_legend(direction = 'horizontal')
    ) +
  scale_color_manual(
    name = NULL,
    labels = c("Admission Rate", "Yield Rate"),
    values = c("admission_rate" = "#4E2A84", "yield_rate" =  "#B6ACD1"),
    guide = guide_legend(direction = 'horizontal')
    ) +
  theme_classic() +
  theme(
    legend.justification = c(0.5, 1),
    legend.position = c(0.5, 1)
    ) +
  scale_x_continuous(
    name = "Entering Year",
    breaks = seq(1999, 2020, 1),
    expand = c(0, 0.25)
    ) +
  scale_y_continuous(
    "Rate",
    limits = c(0, 60),
    breaks = seq (0, 60, 10),
    labels = scales::unit_format(suffix = '%')
    ) +
  geom_text(
    data = rate_label,
    aes(label = pct_label),
    size = 2,
    position = position_nudge(y = 4),
    color = "black"
    )
```

```
#Combine andd create the plots as a grid
plot_grid(bar_plot, line_plot, nrow = 2, align = 'v')
```



When the data is faceted into 2 seperate plots, it is much easier to interpret and compare. There are so many different variables and numbers that are much easier to understand when split into seperate graphs. Specifically, having the same variable on the x axes allows the data to be easily compared.