L07 Scales, Axes & Legends

Data Visualization (STAT 302)

Shay Lebovitz

Contents

Overview

The goal of this lab is to explore ways to manage and manipulate scales, axes, and legends within `ggplot2`.

Datasets

We'll be using the `tech_stocks.rda`, `corruption.rda`, `cdc.txt`, and a few toy datasets.

```r
# Load package(s)
library(tidyverse)
library(scales)
library(lubridate)
library(ggrepel)
library(scales)

# Load datasets
load('data/tech_stocks.rda')
load('data/corruption.rda')

# Read in the cdc dataset
cdc <- read_delim(file = "data/cdc.txt", delim = "|") %>%
  mutate(genhlth = factor(genhlth,
    levels = c("excellent", "very good", "good", "fair", "poor")
  ))

# Set seed
set.seed(8221984)

# Selecting a random subset of size 100
cdc_small <- cdc %>% slice_sample(n = 100)

# Generating toy datasets for exercise 2
dat1 <- tibble(theta = c(0, 2 * pi))

dat2 <- tibble(
  theta = seq(0, 2 * pi, length.out = 100),
  obs = rnorm(100, sin(theta), 0.1),
  larger_than = ifelse(abs(obs) < abs(sin(theta)), "1", "0")
)
```

Exercises

Complete the following exercises.

Exercise 1

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

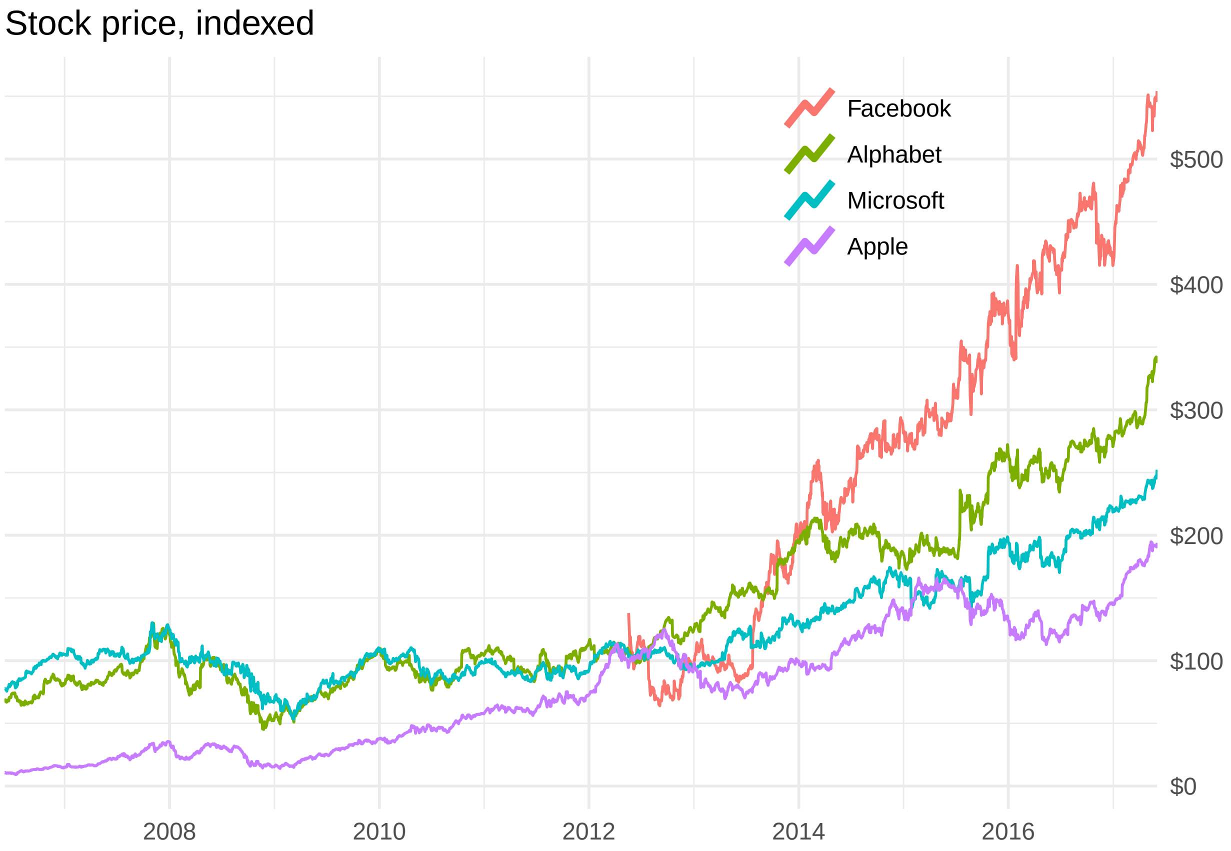
Hints:

- Size of lines in legend is 1.3
- 0.75 and 0.85 will be useful
- Eliminated extra space in horizontal direction
- `key_glyph`

```
xrng <- range(tech_stocks$date)
yrng <- range(tech_stocks$price_indexed)

tech_stocks$company <- factor(tech_stocks$company,
                              levels = c('Facebook', 'Alphabet',
                                         'Microsoft', 'Apple'))

tech_stocks %>%
  ggplot(aes(x = date, y = price_indexed)) +
  geom_line(aes(color = company), key_glyph = 'timeseries') +
  scale_x_date(expand = c(0, 0)) +
  scale_y_continuous(labels = scales::label_dollar(),
                     breaks = c(0, 100, 200, 300, 400, 500),
                     position = 'right') +
  theme_minimal() +
  labs(x = NULL, y = NULL,
       title = 'Stock price, indexed') +
  scale_color_manual(values = c(
    'Alphabet' = '#7CAE00',
    'Facebook' = '#F8766D',
    'Microsoft' = '#00BFC4',
    'Apple' = '#C77CFF'
  )) +
  theme(legend.position = c(0.75, 0.85),
        legend.title = element_blank()) +
  guides(color = guide_legend(override.aes = list(size = 1.3)))
```



Exercise 2

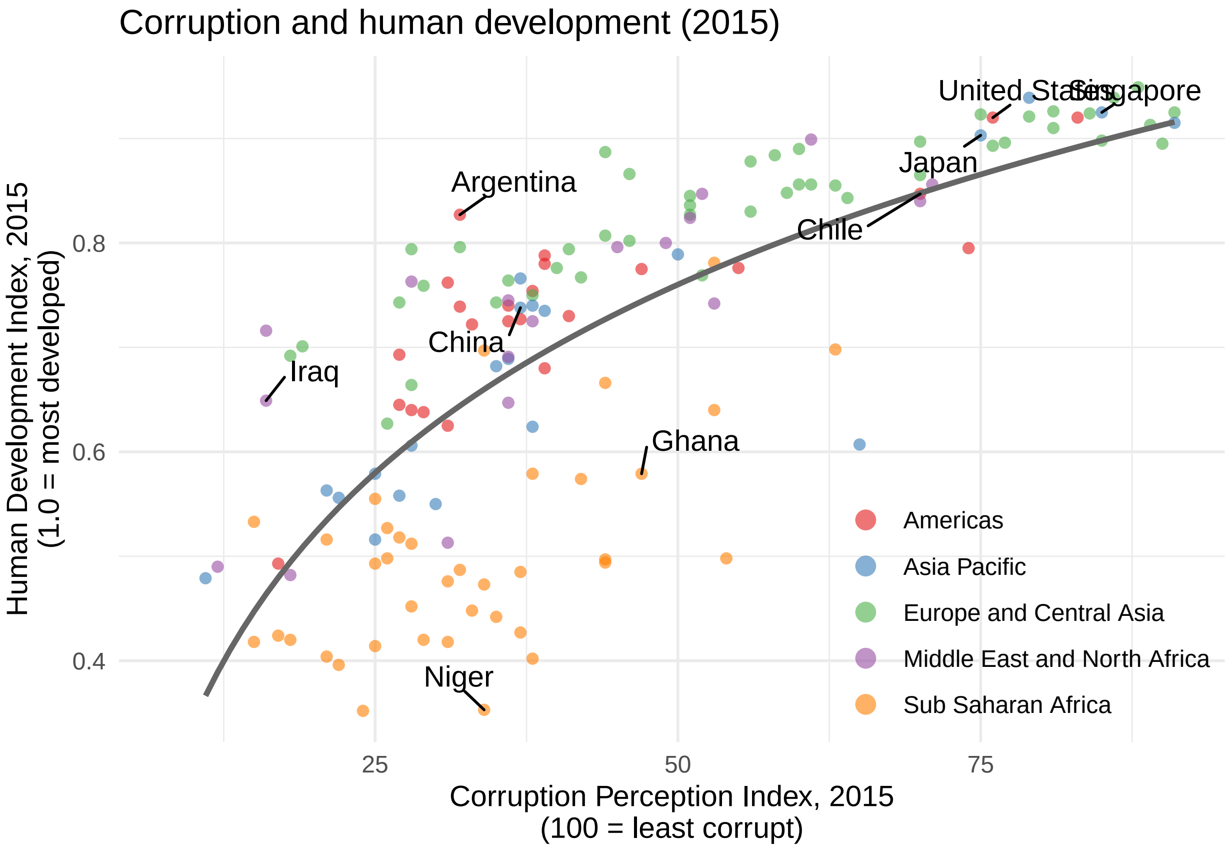
Using `corruption.rda` dataset, recreate the following graphic as precisely as possible.

Hints:

- Transparency is 0.6
- Only use 2015 data
- Point size is 3 in legend
- Color used is `grey40` & color palette `"Set2"`
- `"y ~ log(x)"` (method `"lm"`)
- Package `ggrepel`
- `box.padding` is 0.6
- Minimum segment length is 0
- seed is 9876

```
label_info <- corruption %>%
  filter(year == 2015) %>%
  filter(country %in% c('Singapore', 'United States', 'Japan',
                        'Chile', 'Ghana', 'Argentina', 'China',
                        'Iraq', 'Niger'))

corruption %>%
  filter(year == 2015) %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region), alpha = 0.6) +
  geom_smooth(formula = y ~ log(x), method = 'lm', se = F,
             color = 'grey40') +
  labs(x = 'Corruption Perception Index, 2015\n(100 = least corrupt)',
       y = 'Human Development Index, 2015\n(1.0 = most developed)',
       title = 'Corruption and human development (2015)') +
  theme_minimal() +
  scale_colour_brewer(palette = 'Set1') +
  geom_text_repel(data = label_info, aes(label = country),
                  box.padding = 0.6, min.segment.length = 0, seed = 9876) +
  theme(legend.position = c(1, 0),
        legend.justification = c('right', 'bottom'),
        legend.title = element_blank()) +
  guides(color = guide_legend(override.aes = list(size = 3)))
```



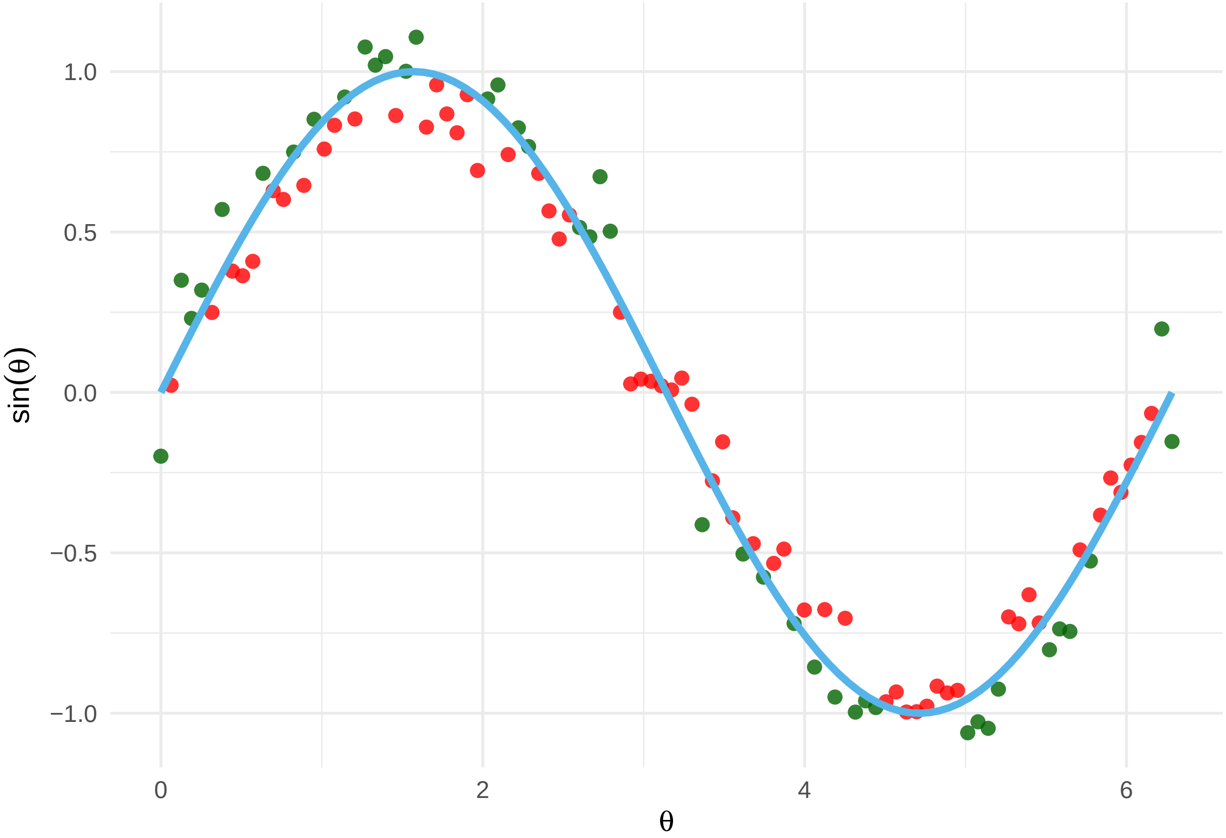
Exercise 3

Use the toy datasets `dat1` and `dat2` to recreate the following graphic as precisely as possible.

Hints:

- Sizes used 2 (points) and 1.3 (line)
- Transparency 0.8
- Colors: `#56B4E9`, `darkgreen`, & `red`

```
dat2 %>%
  ggplot(aes(x = theta, y = obs, color = larger_than)) +
  geom_point(size = 2, alpha = 0.8) +
  geom_function(fun = function(x) sin(x), color = '#56B4E9', size = 1.3) +
  labs(x = quote(theta), y = quote(sin(theta))) +
  theme_minimal() +
  theme(legend.position = 'none') +
  scale_color_manual(values = c('0' = 'darkgreen', '1' ='red'))
```



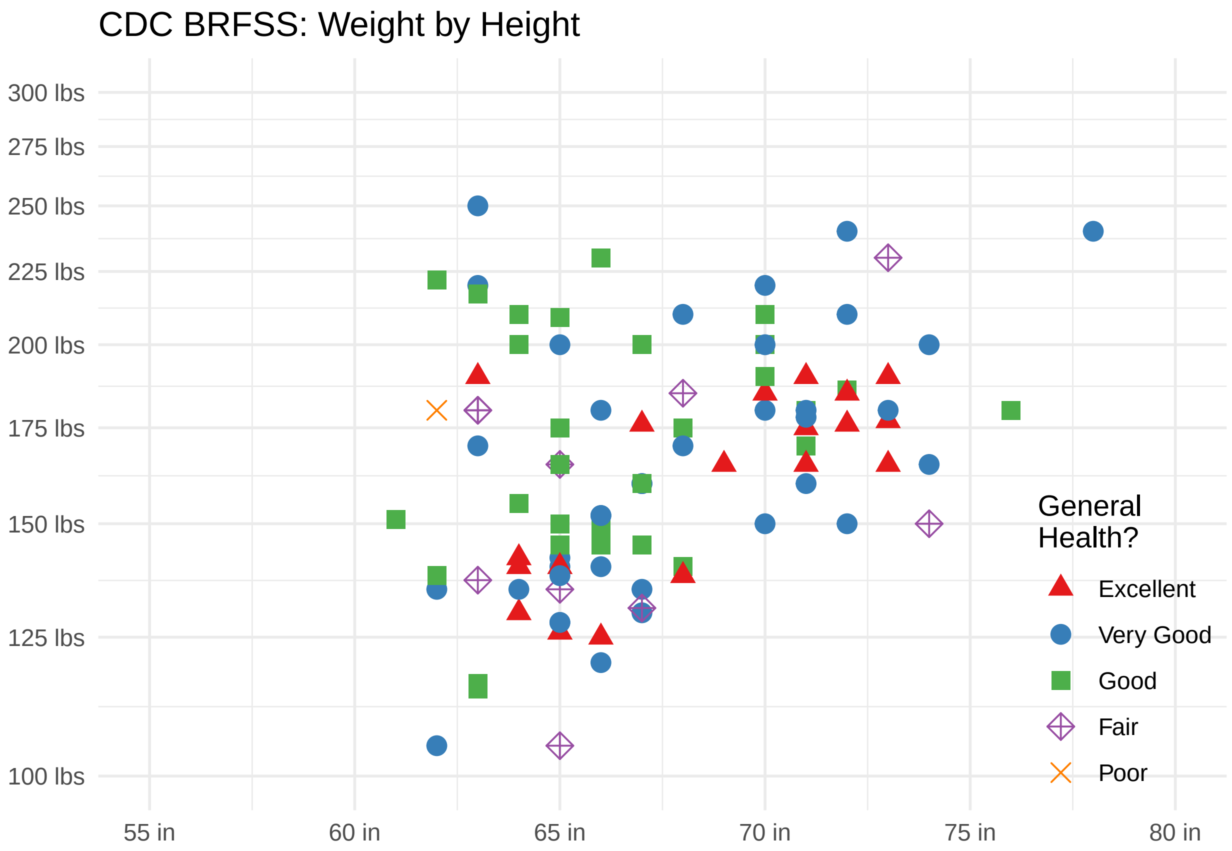
Exercise 4

Using `cdc_small`construct a scatterplot of `weight` by `height` with the following requirements:

- Size of plotting characters should be 3.
- Color and shape should both identify `genhlth`.
- One legend for both color and shape.
- Legend title should be “General Health?” with a newline starting after general.
- Legend categories should be ordered from excellent (top) to poor (bottom) with each word in category capitalized in the legend.
- Legend should be placed in the lower right-hand corner of the plotting area.
- Color should follow the `"Set1"` palette.
- Shape should have a solid triangle (17) for excellent, solid circle (19) for very good, an `x` (4) for poor, an hollow rotated square with an `x` in it (9) for fair, and a solid square (15) for good.
- `height` values should be limited between 55 and 80.
- `height` axis should display every 5th number between 55 and 80 and be appropriately labeled (i.e. `55 in`, `60 in`, . . . , `80 in`). No axis title is necessary.
- `weight` values should be limited between 100 and 300.
- `weight` axis should be on log base 10 scale, but still display weights in pounds starting at 100 and displaying every 25 pounds until 300. Must be appropriately labeled (i.e. `100 lbs`, `125 lbs`, . . . , `300 lbs`). No axis title is necessary.

- Graph title should be `CDC BRFSS: Weight by Height`.
- Minimal theme.

```
cdc_small %>%
  ggplot(aes(x = height, y = weight)) +
  geom_point(aes(color = genhlth, shape = genhlth), size = 3) +
  theme_minimal() +
  scale_shape_manual(values = c(17, 19, 15, 9, 4),
                     labels = c('Excellent', 'Very Good', 'Good',
                                'Fair', 'Poor'),
                     name = 'General\nHealth?') +
  theme(#legend.title = 'General\nHealth?',
        legend.position = c(1, 0),
        legend.justification = c(1, 0)) +
  scale_color_brewer(palette = 'Set1',
                     name = 'General\nHealth?',
                     labels = c('Excellent', 'Very Good', 'Good',
                                'Fair', 'Poor')) +
  labs(x = NULL, y = NULL, title = 'CDC BRFSS: Weight by Height') +
  scale_y_log10(breaks = c(100, 125, 150, 175,
                                    200, 225, 250, 275, 300),
                limits = c(100, 300),
                labels = c('100 lbs', '125 lbs', '150 lbs', '175 lbs',
                           '200 lbs', '225 lbs', '250 lbs', '275 lbs', '300 lbs')) +
  scale_x_continuous(breaks = c(55, 60, 65, 70, 75, 80),
                     limits = c(55, 80),
                     labels = c('55 in', '60 in', '65 in', '70 in', '75 in', '80 in'))
```


CDC BRFSS: Weight by Height
300 lbs
275 lbs
250 lbs
225 lbs
200 lbs
175 lbs
150 lbs
125 lbs
100 lbs
55 in
60 in
65 in
70 in
75 in
80 in
General
Health?
Excellent
Very Good
Good
Fair
Poor