L03 ggplot II

Data Visualization (STAT 302) - Shay Lebovitz

Contents

Overview

The goal of this lab is to continue the process of unlocking the power of `ggplot2` through constructing and experimenting with a few basic plots.

Datasets

We'll be using data from the `BA_degrees.rda` and `dow_jones_industrial.rda` datasets which are already in the `/data` subdirectory in our **data_vis_labs** project. Below is a description of the variables contained in each dataset.

`BA_degrees.rda`

- `field` - field of study
- `year_str` - academic year (e.g. 1970-71)
- `year` - closing year of academic year
- `count` - number of degrees conferred within a field for the year
- `perc` - field's percentage of degrees conferred for the year

`dow_jones_industrial.rda`

- `date` - date
- `open` - Dow Jones Industrial Average at open
- `high` - Day's high for the Dow Jones Industrial Average
- `low` - Day's low for the Dow Jones Industrial Average
- `close` - Dow Jones Industrial Average at close
- `volume` - number of trades for the day

We'll also be using a subset of the BRFSS (Behavioral Risk Factor Surveillance System) survey collected annually by the Centers for Disease Control and Prevention (CDC). The data can be found in the provided `cdc.txt` file — place this file in your `/data` subdirectory. The dataset contains 20,000 complete observations/records of 9 variables/fields, described below.

- `genhlth` - How would you rate your general health? (excellent, very good, good, fair, poor)
- `exerany` - Have you exercised in the past month? (`1 = yes`, `0 = no`)
- `hlthplan` - Do you have some form of health coverage? (`1 = yes`, `0 = no`)
- `smoke100` - Have you smoked at least 100 cigarettes in your life time? (`1 = yes`, `0 = no`)
- `height` - height in inches
- `weight` - weight in pounds

- `wtdesire` - weight desired in pounds
- `age` - in years
- `gender` - `m` for males and `f` for females

```
library(tidyverse)
library(lubridate)
library(splines)
```

Exercises

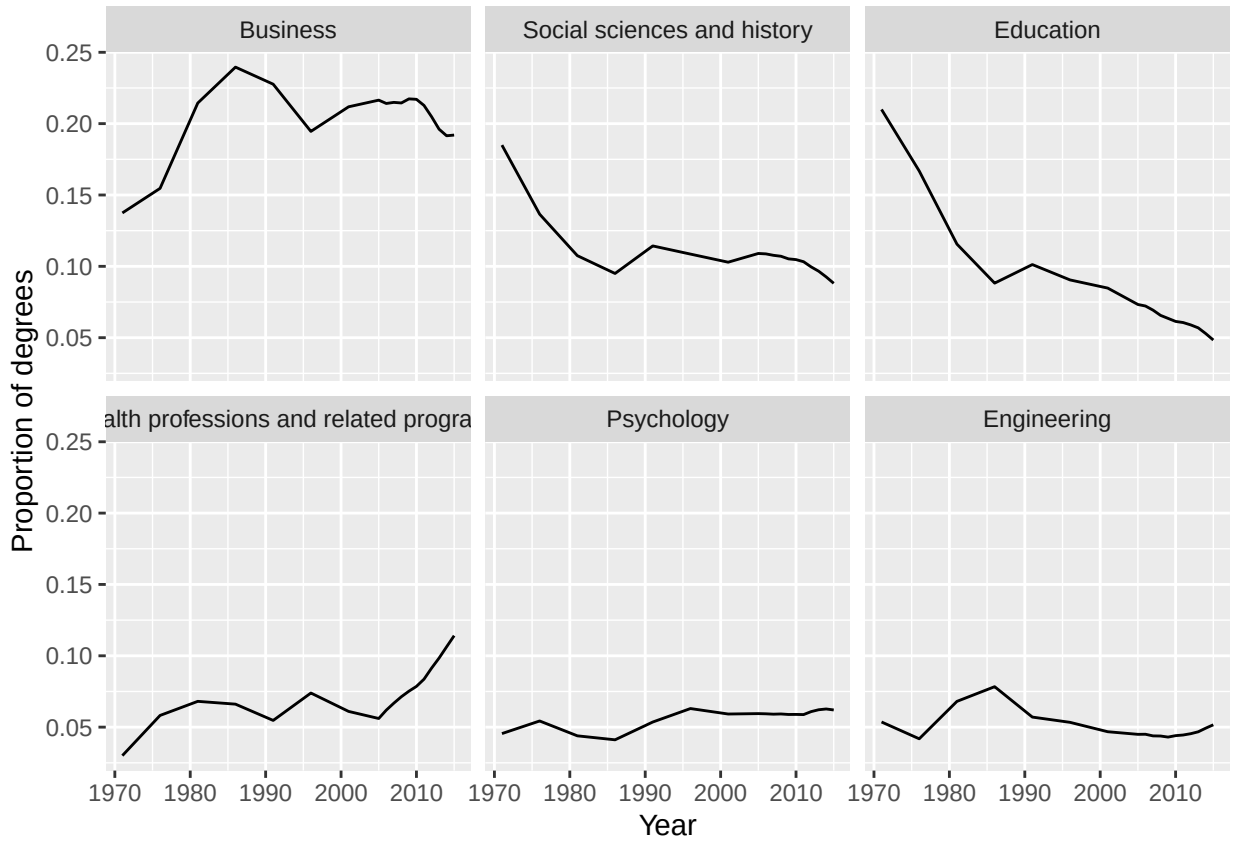
Complete the following exercises.

Exercise 1

Using `BA_degrees` dataset, recreate the following graphics as precisely as possible.

```
load('data/BA_degrees.rda')

# Wrangling for plotting
ba_dat <- BA_degrees %>%
  # mean % per field
  group_by(field) %>%
  mutate(mean_perc = mean(perc)) %>%
  # Only fields with mean >= 5%
  filter(mean_perc >= 0.05) %>%
  # Organizing for plotting
  arrange(desc(mean_perc), year) %>%
  ungroup() %>%
  mutate(field = fct_inorder(field))
```

```
ba_dat %>%
  ggplot(aes(year, perc)) +
  geom_line() +
  facet_wrap(~ field) +
  labs(x = 'Year', y = 'Proportion of degrees')
```

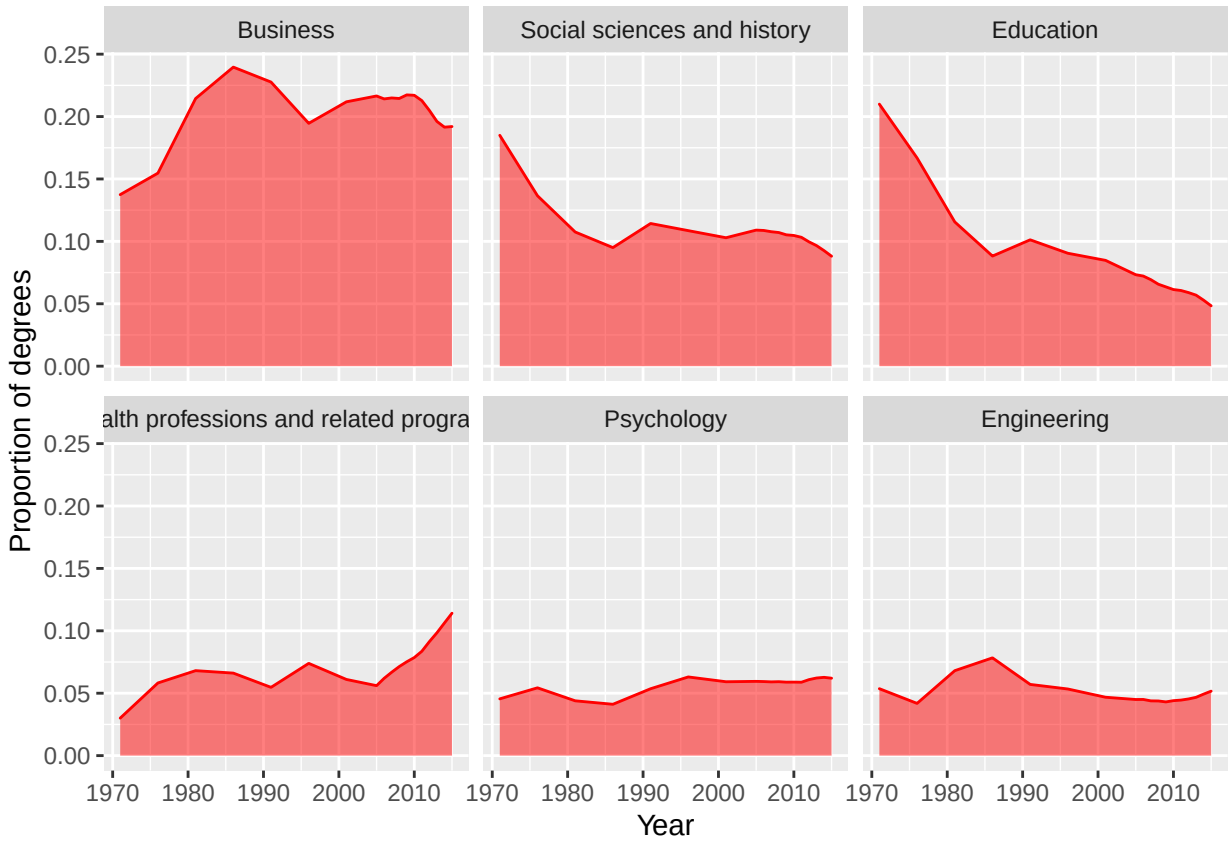


Plot 1

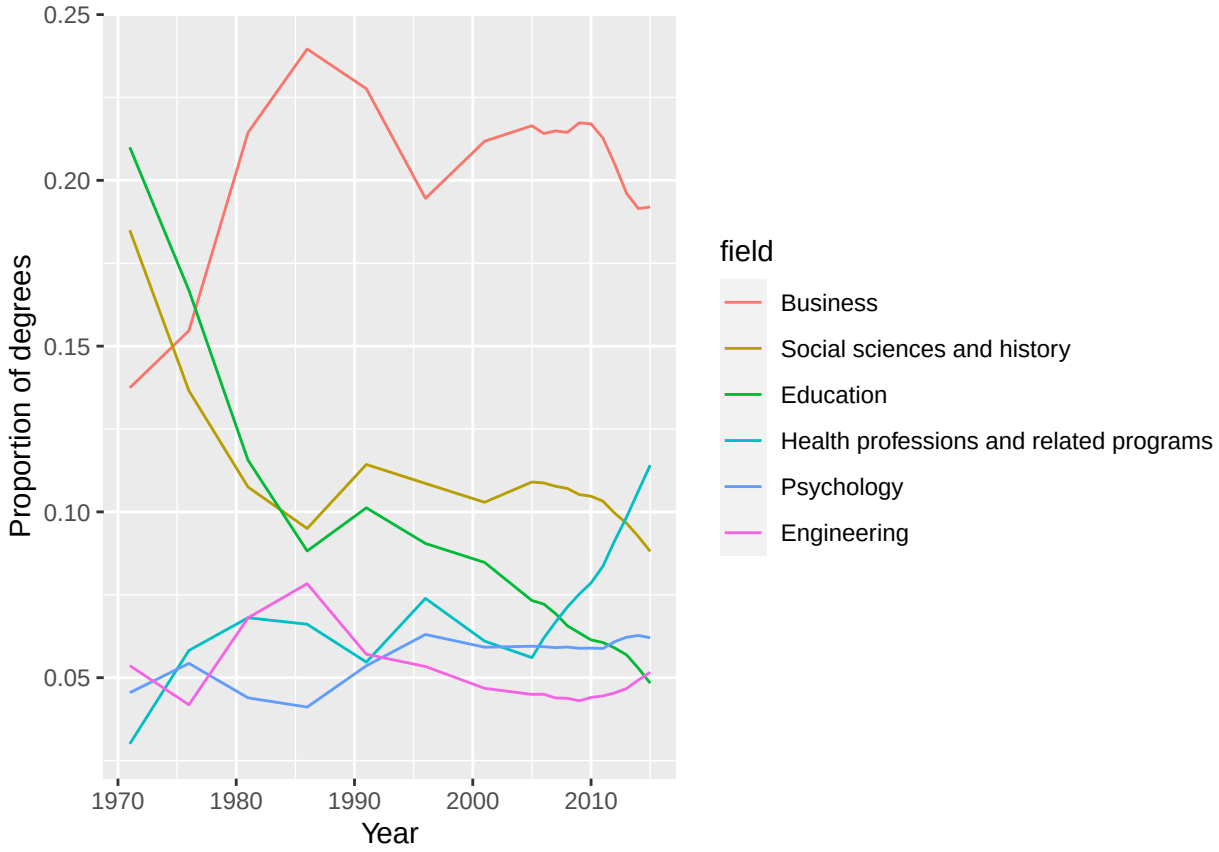
Plot 2 *Hints:*

- Transparency is 0.5
- Color used is`"red"`

```
ba_dat %>%
  ggplot(aes(year, perc)) +
  geom_area(color = 'red', fill = 'red', alpha = 0.5) +
  facet_wrap(~ field) +
  labs(x = 'Year', y = 'Proportion of degrees')
```




```
ba_dat %>%
  ggplot(aes(year, perc)) +
  geom_line(aes(color = field)) +
  labs(x = 'Year', y = 'Proportion of degrees')
```



Plot 3

Exercise 2

Using `dow_jones_industrial` dataset, recreate the following graphics as precisely as possible. *Hint:* Used `close`.

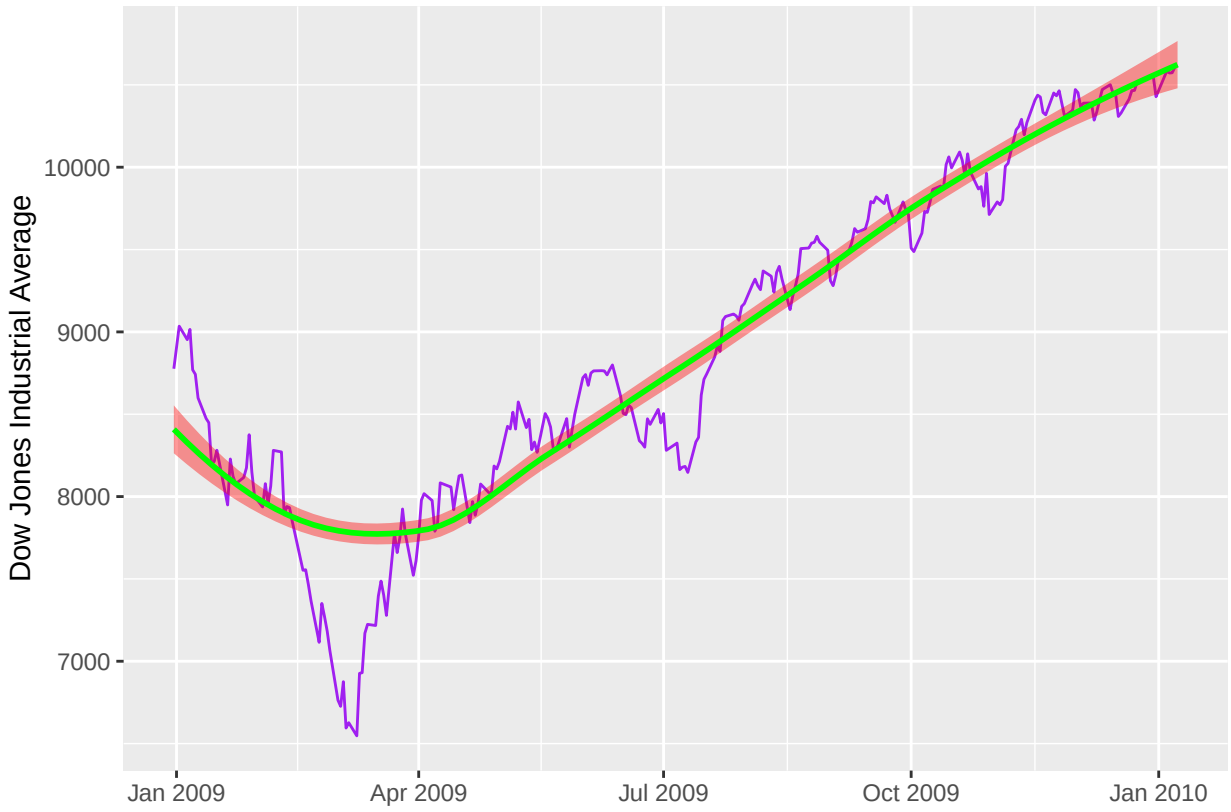
```
load('data/dow_jones_industrial.rda')

# Restrict data to useful range
djia_date_range <- dow_jones_industrial %>%
  filter(date >= ymd("2008/12/31") & date <= ymd("2010/01/10"))
```

Plot 1 *Hints:*

- Colors used `"red"`, `"purple"`, & `"green"`

```
djia_date_range %>%
  ggplot(aes(date, close)) +
  geom_line(color = 'purple') +
  geom_smooth(color = 'green', fill = 'red') +
  labs(x = '', y = 'Dow Jones Industrial Average')
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

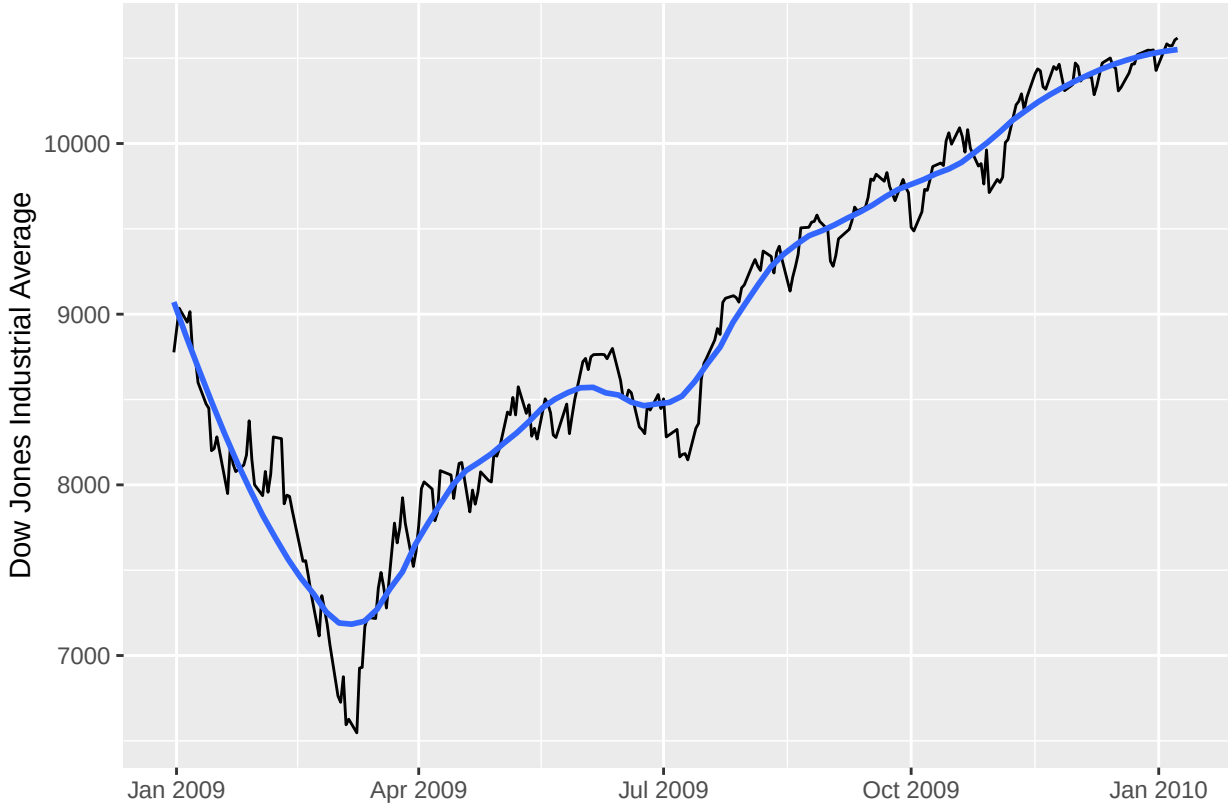


Plot 2 *Hints:*

- Wiggliness for `loess` is 0.3

```
djia_date_range %>%
  ggplot(aes(date, close)) +
  geom_line() +
  geom_smooth(span = 0.3, se = F) +
  labs(x = '', y = 'Dow Jones Industrial Average')
```

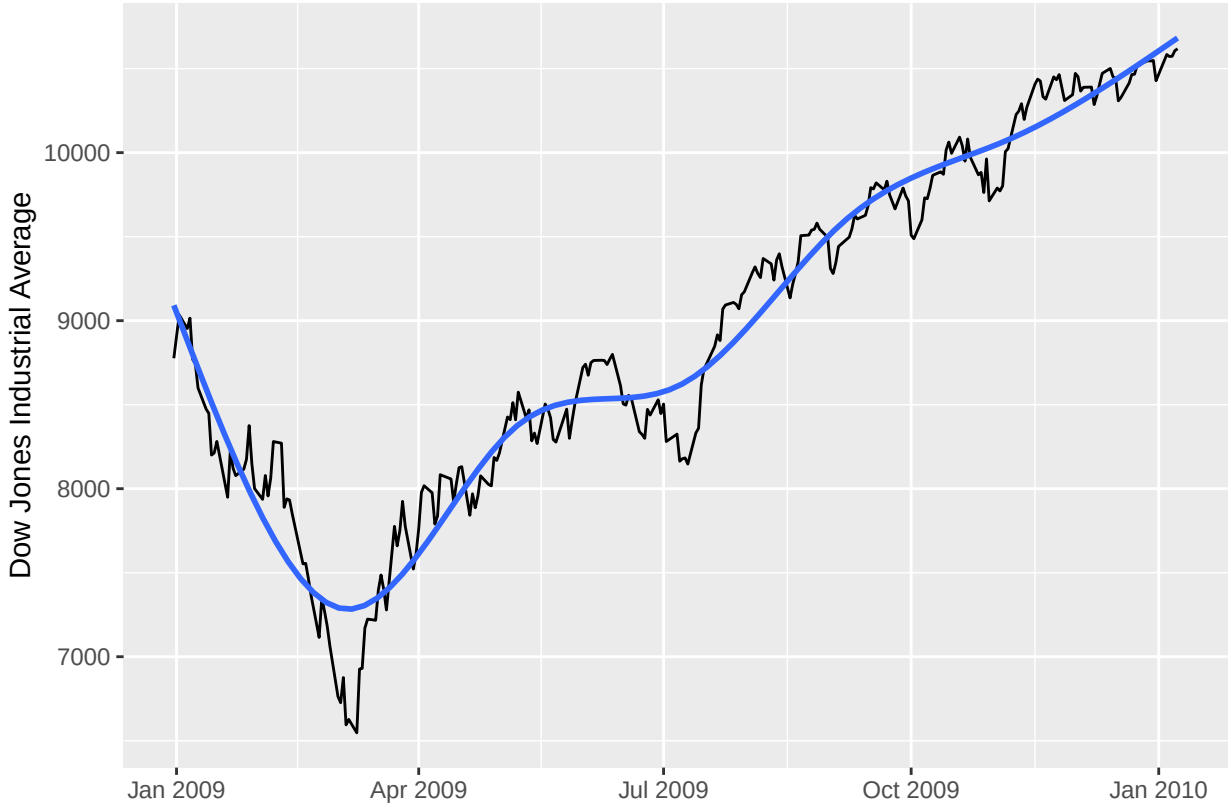
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Plot 3 *Hints:*

- `y ~ ns(x, 6)` will need `splines` package (`"lm"` will work)

```
djia_date_range %>%
  ggplot(aes(date, close)) +
  geom_line() +
  geom_smooth(method = lm, formula = y ~ ns(x, 6), se = F) +
  labs(x = '', y = 'Dow Jones Industrial Average')
```



Exercise 3

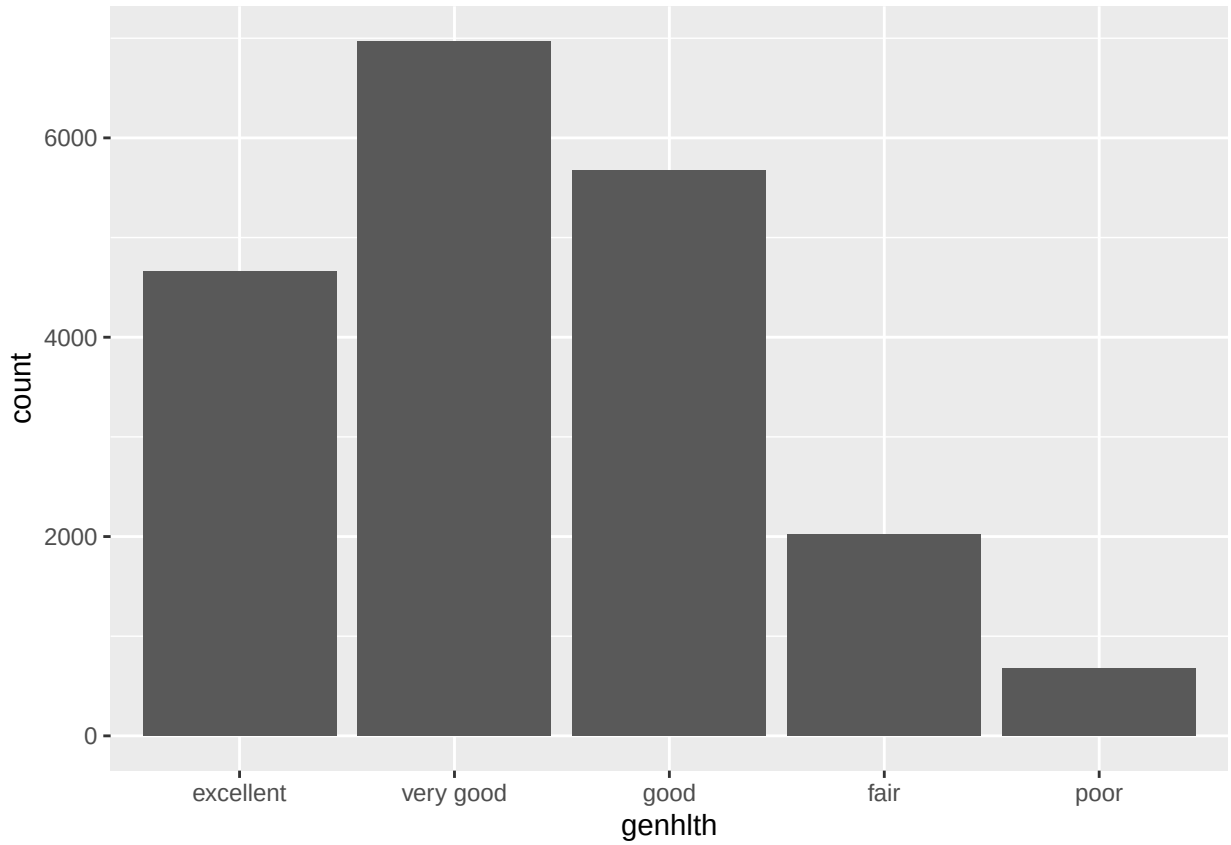
Using `cdc` dataset, recreate the following graphics as precisely as possible.

```
# 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")
  ))
```

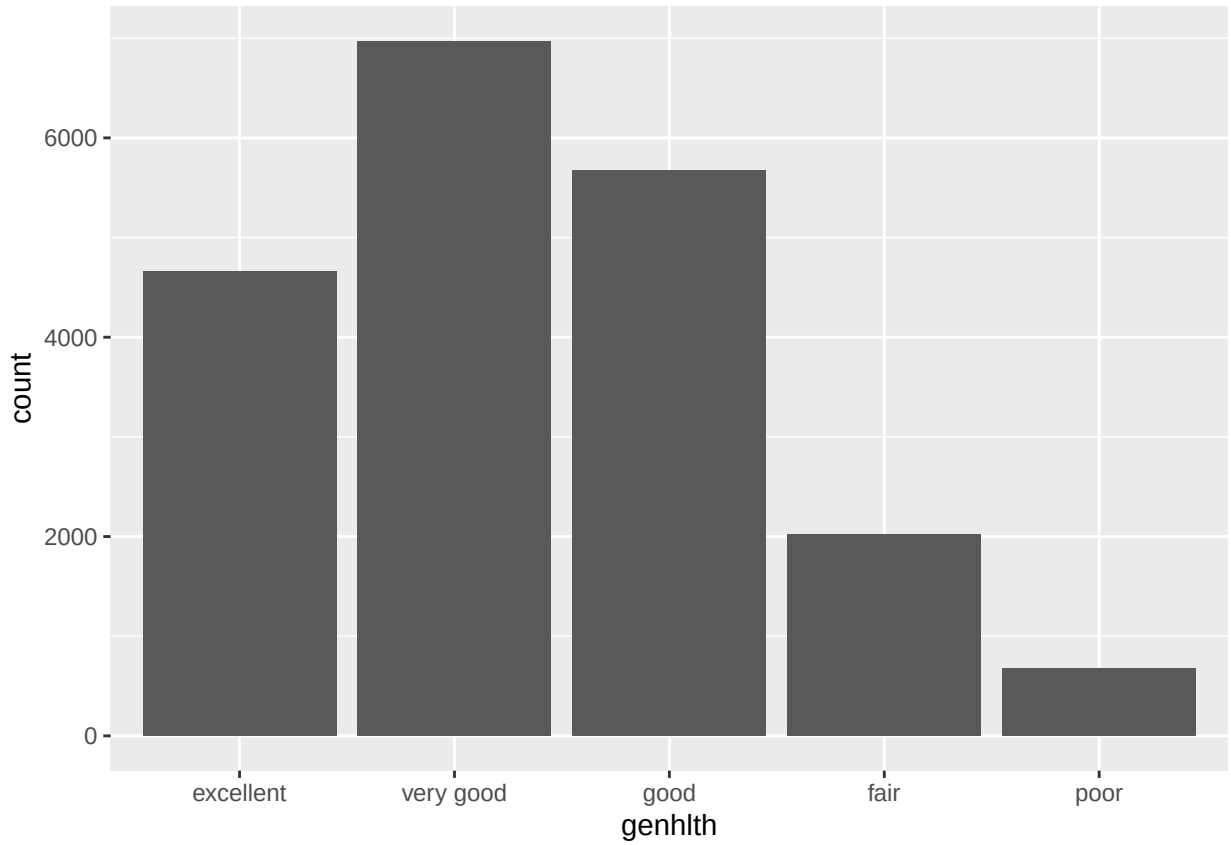
Plot 1 Construct this plot in two ways. Once using `cdc` and once using the `genhlth_count`.

```
genhlth_count <- cdc %>%
  count(genhlth)
```

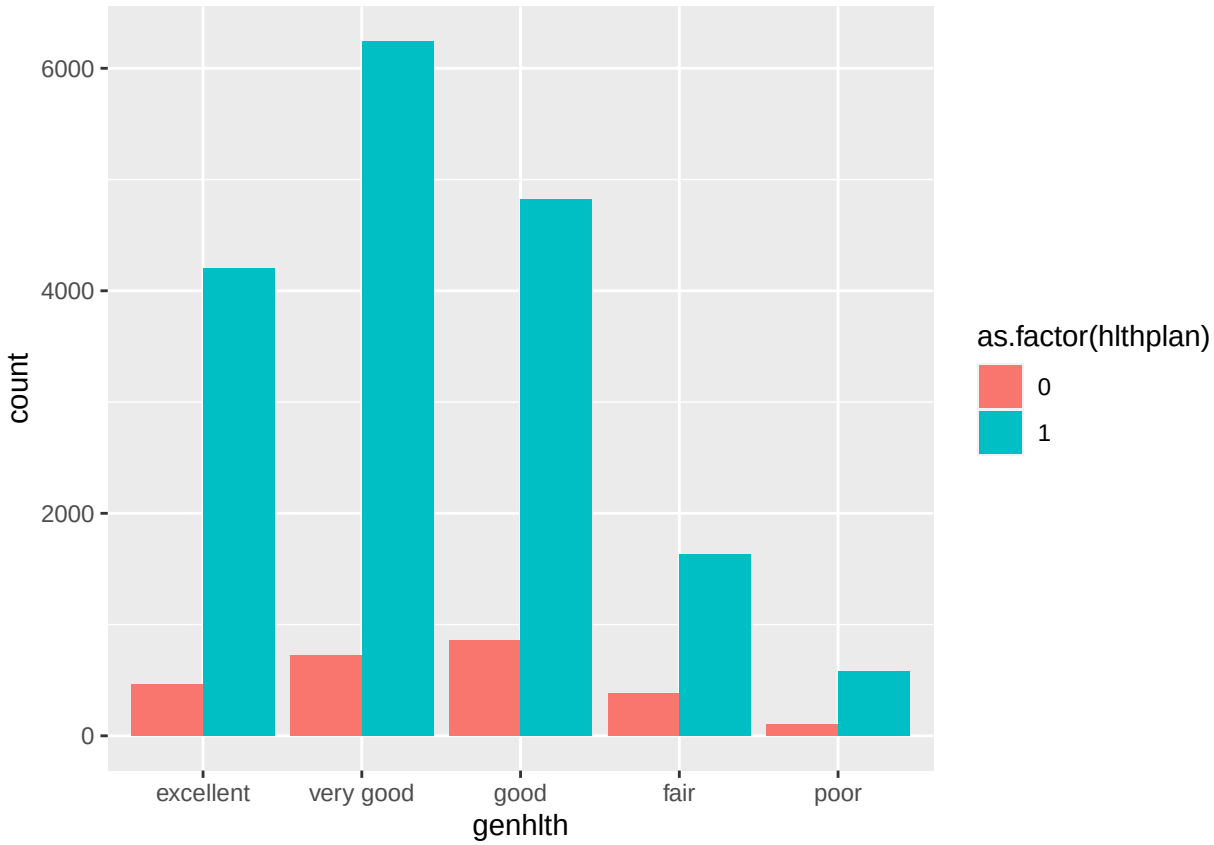
```
genhlth_count %>%
  ggplot(aes(genhlth, n)) +
  geom_bar(stat = 'identity') +
  labs(x = 'genhlth', y = 'count')
```




```
cdc %>%
  ggplot(aes(genhlth)) +
  geom_bar() +
  labs(x = 'genhlth', y = 'count')
```




```
cdc %>%
  ggplot(aes(genhlth)) +
  geom_bar(aes(fill = as.factor(hlthplan)), position = 'dodge') +
  labs(y = 'count')
```

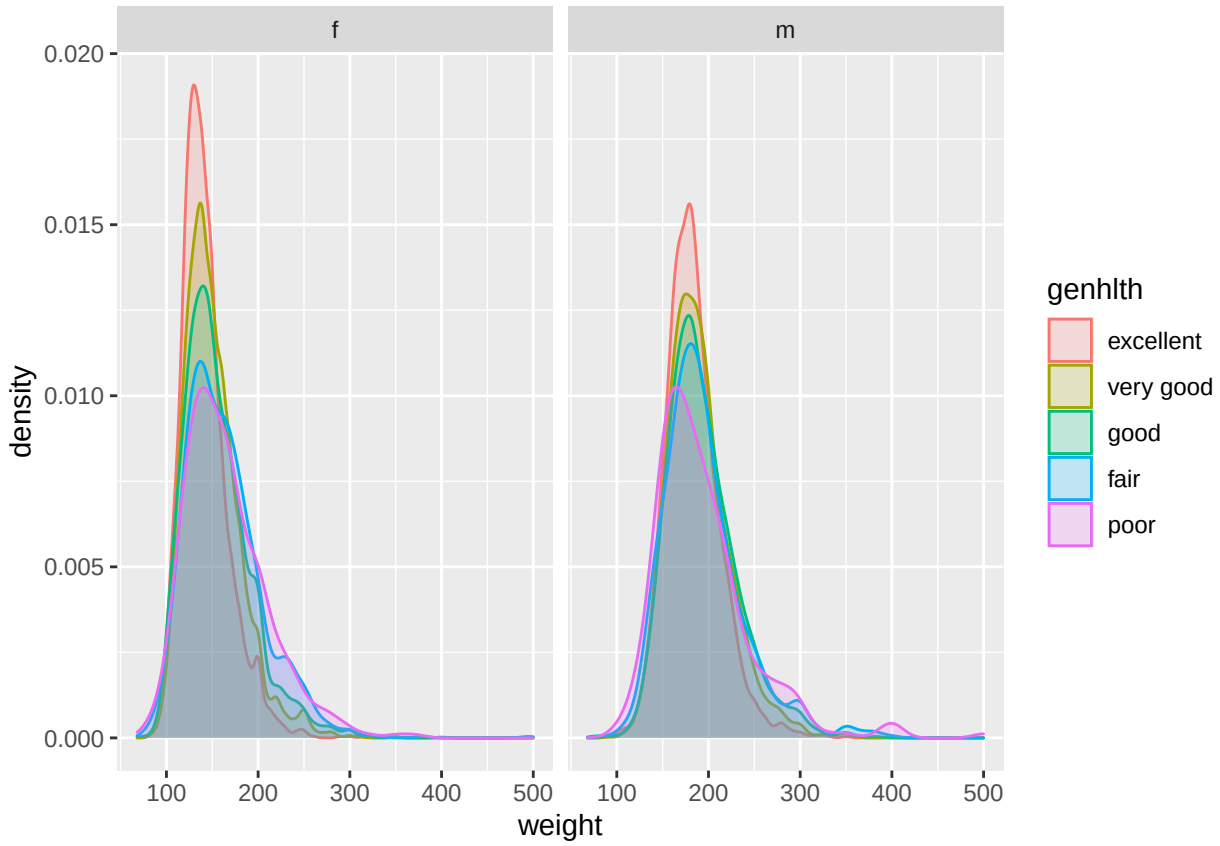


Plot 2

Plot 3 *Hints:*

- Transparency is 0.2

```
cdc %>%
  ggplot(aes(weight)) +
  geom_density(aes(color = genhlth, fill = genhlth), alpha = 0.2) +
  facet_wrap(~ gender)
```

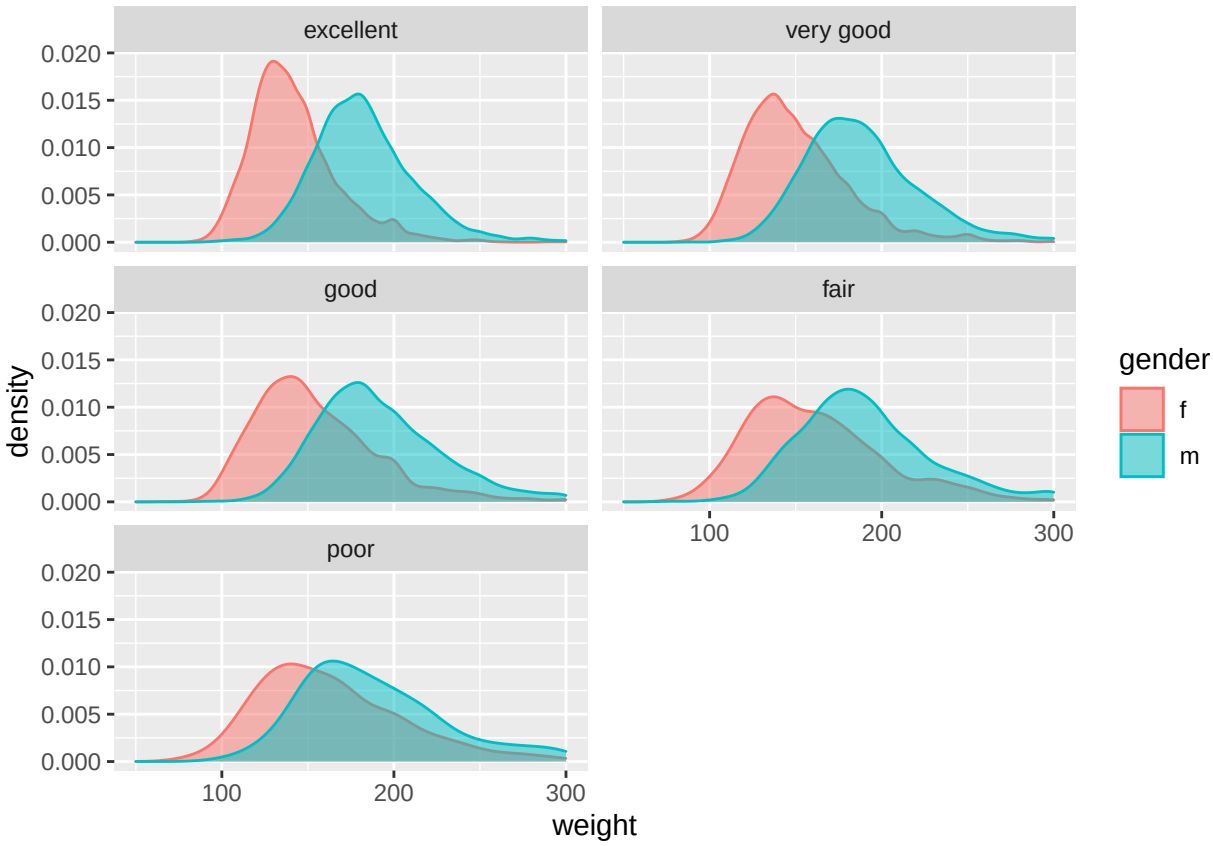


Plot 4 *Hints:*

- Transparency is 0.5
- Horizontal axis should have lower limit of 50 and upper limit of 300

```
cdc %>%
  ggplot(aes(weight)) +
  geom_density(aes(color = gender, fill = gender), alpha = 0.5) +
  facet_wrap(~ genhlth, nrow = 3) +
  xlim(50, 300)
```

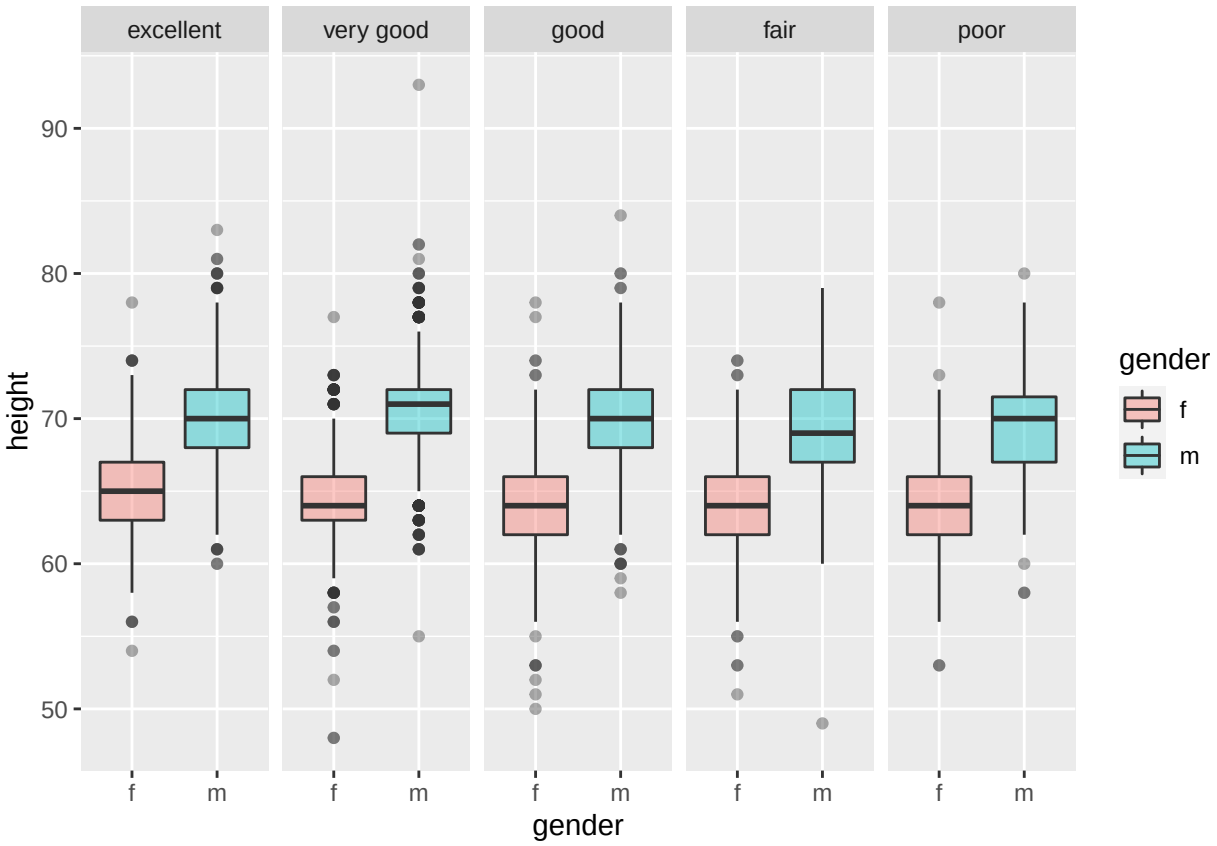
```
## Warning: Removed 103 rows containing non-finite values (stat_density).
```



Plot 5 *Hints:*

- Transparency is 0.4

```
cdc %>%
  ggplot(aes(gender, height)) +
  geom_boxplot(aes(fill = gender), alpha = 0.4) +
  facet_wrap(~ genhlth, nrow = 1)
```

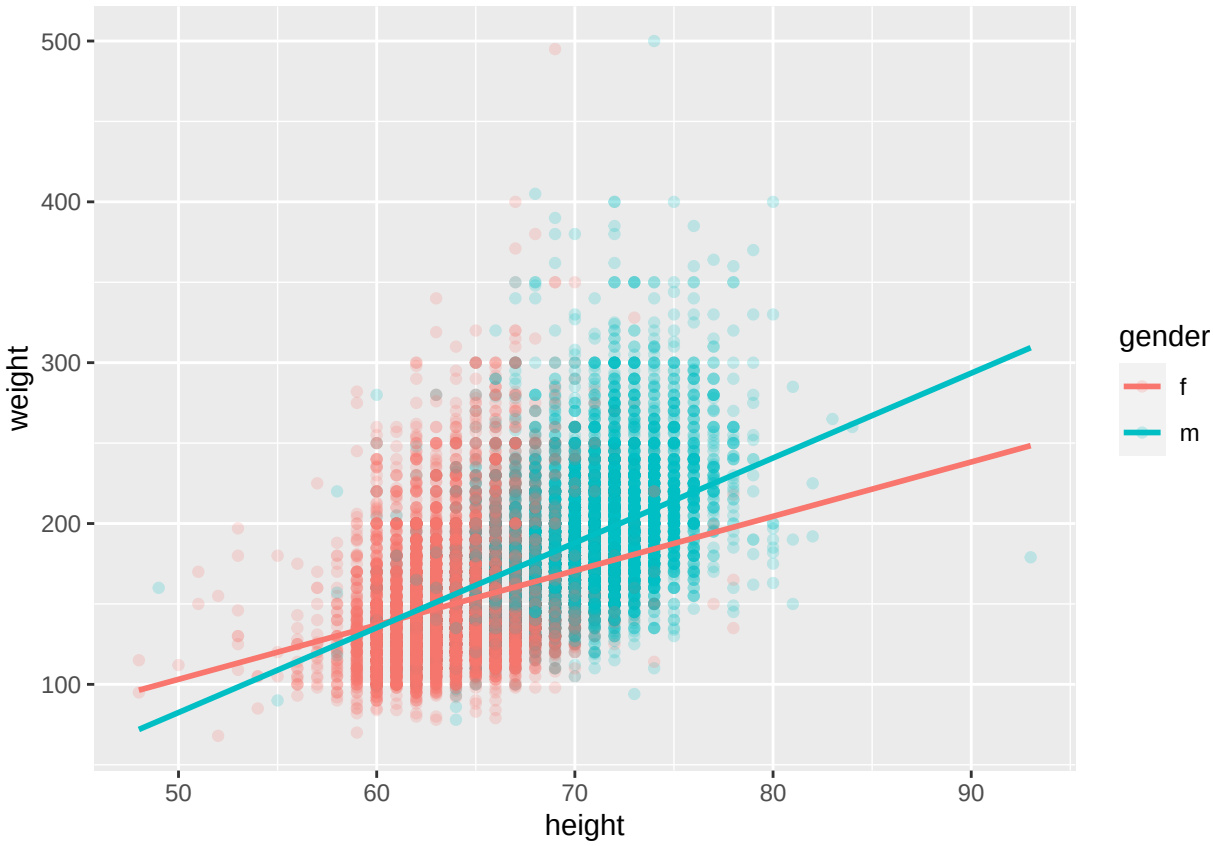


Plot 6 *Hints:*

- Transparency is 0.2

```
cdc %>%
  ggplot(aes(height, weight, group = gender)) +
  geom_point(aes(color = gender), alpha = 0.2) +
  geom_smooth(method = lm, aes(color = gender), se = F, fullrange = T)
```

```
## `geom_smooth()` using formula 'y ~ x'
```


500
400
300
200
100
weight
50
60
70
80
90
height
gender
f
m