Modern Dive 5 and 6 HW

Emilio Horner

2023-01-18

```r
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──
## ✓ ggplot2 3.4.0      ✓ purrr   0.3.5
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.5.0
## ✓ readr   2.1.3      ✓ forcats 0.5.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
```

```r
library(broom)
library(skimr)

twitch_data <- read_csv("https://raw.githubusercontent.com/vaiseys/223_course/main/Data/twitchdat
a-update.csv")
```

```
## Rows: 1000 Columns: 11
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (2): Channel, Language
## dbl (7): Watch time(Minutes), Stream time(minutes), Peak viewers, Average vi...
## lgl (2): Partnered, Mature
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
library(tidyverse)
library(moderndive)
```

```r
library(janitor)
```

```
##
## Attaching package: 'janitor'
```

```
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
twitch_data <- clean_names(twitch_data)

# Inspect new names
colnames(twitch_data)
```

```
##  [1] "channel"              "watch_time_minutes"  "stream_time_minutes"
##  [4] "peak_viewers"         "average_viewers"     "followers"
##  [7] "followers_gained"     "views_gained"        "partnered"
## [10] "mature"               "language"
```

1.

```
twitch_data %>%
  sample_n(size = 5)
```

```
## # A tibble: 5 × 11
##   channel watch…¹ strea…² peak_…³ avera…⁴ follo…⁵ follo…⁶ views…⁷ partn…⁸ mature
##   <chr>     <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <lgl>   <lgl>
## 1 dizzy    1.60e8   49125   19262    2997  795869  122263 4572893 TRUE    TRUE
## 2 Northe…  2.43e8   60855    9312    3968  359260   22629 3384340 TRUE    FALSE
## 3 dasMEH…  1.17e9  231465   47683    5013  299048   76568 7422911 TRUE    TRUE
## 4 HAchub…  2.04e8   73110   13675    2779  221461  159519 5075885 TRUE    FALSE
## 5 Tomato   2.06e8   76080    5358    2675   92634   36010 2200922 TRUE    TRUE
## # … with 1 more variable: language <chr>, and abbreviated variable names
## #   ¹watch_time_minutes, ²stream_time_minutes, ³peak_viewers, ⁴average_viewers,
## #   ⁵followers, ⁶followers_gained, ⁷views_gained, ⁸partnered
```

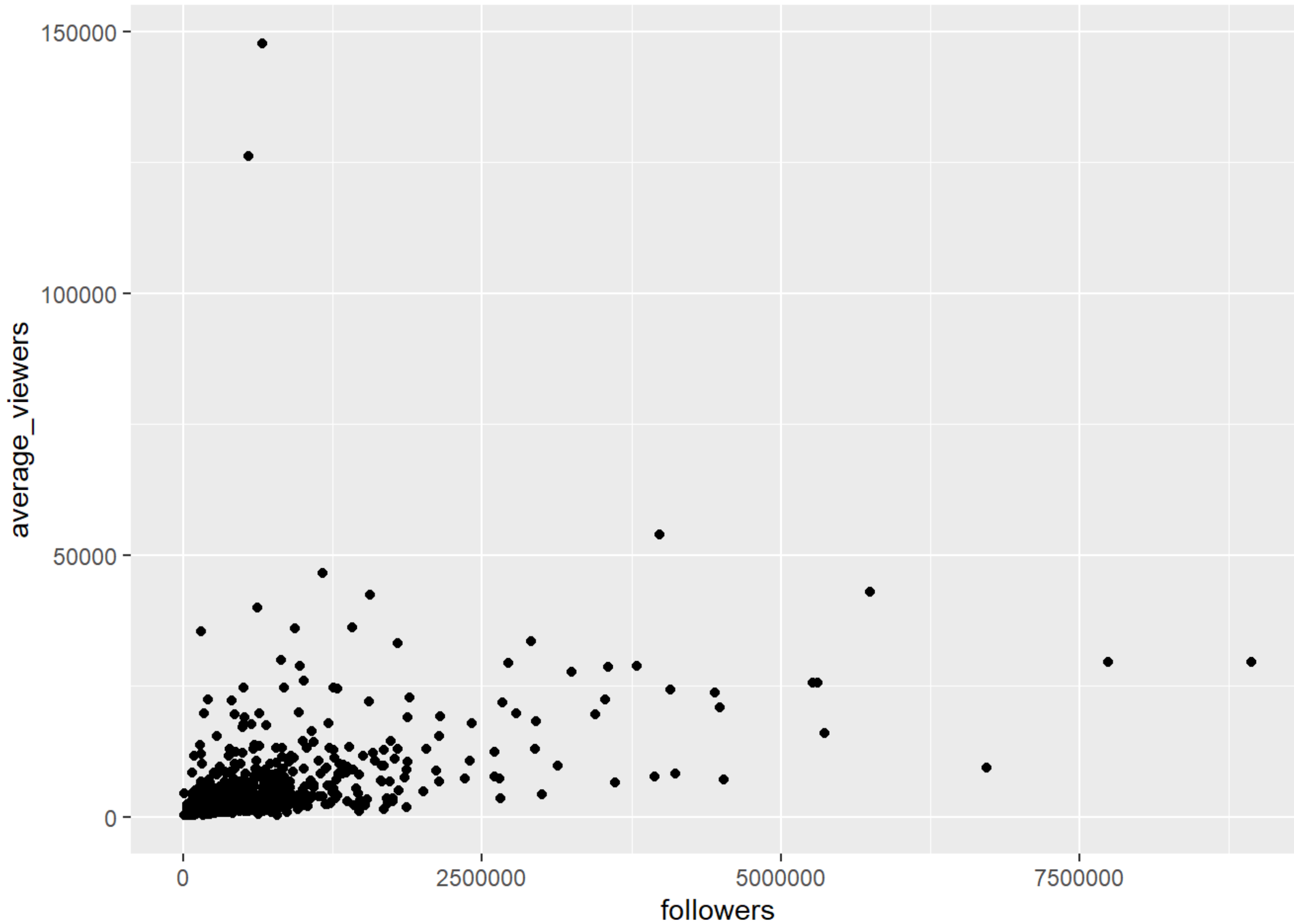
Generally, each streamer that has more followers has more average viewers.

```
twitch_data %>%
select(followers, average_viewers) %>%
  summary()
```

```
##    followers       average_viewers
##  Min.   :   3660   Min.   :   235
##  1st Qu.: 170546   1st Qu.:  1458
##  Median : 318063   Median :  2425
##  Mean   : 570054   Mean   :  4781
##  3rd Qu.: 624332   3rd Qu.:  4786
##  Max.   :8938903   Max.   :147643
```

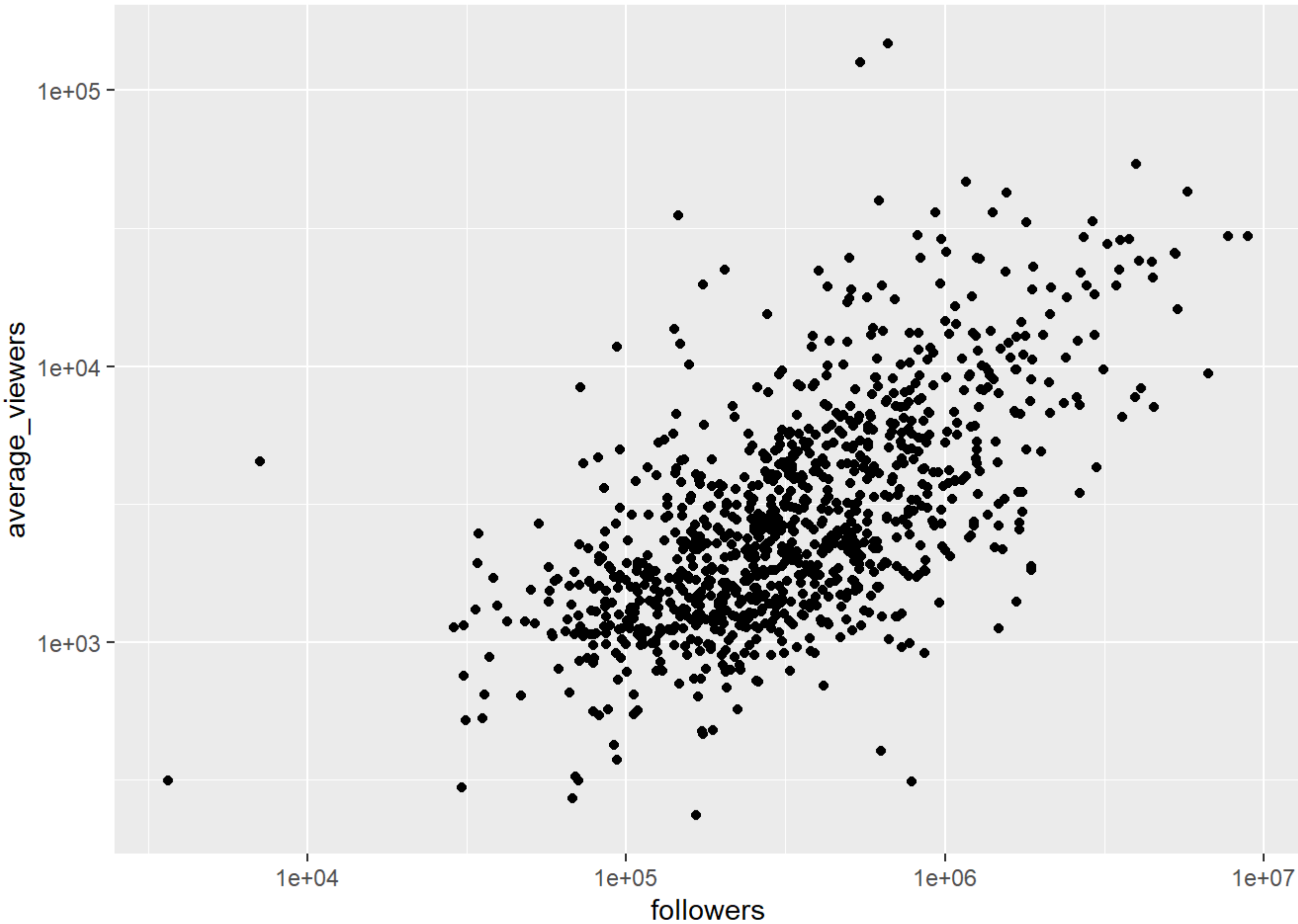
The average switch streamer in the dataset has 570054 followers and about 4781 average viewers.

```
ggplot(data = twitch_data, mapping = aes(x = followers, y = average_viewers)) +
  geom_point()
```




```
P <- ggplot(data = twitch_data, mapping = aes(x = followers, y = average_viewers)) +
  geom_point()
```

```
P + scale_x_log10() + scale_y_log10()
```



The graph shows the positive relationship between number of followers and average viewers.

```
twitch_data <- twitch_data %>%
  mutate(log_viewers = log10(average_viewers),
         log_followers = log10(followers))
```

2.

```
fit1 <- lm(log_viewers ~ log_followers, data = twitch_data)
```

```
tidy(fit1)
```

```
## # A tibble: 2 × 5
##   term          estimate std.error statistic   p.value
##   <chr>            <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)      0.198    0.125       1.58 1.15e-  1
## 2 log_followers    0.588    0.0226     26.0  1.69e-114
```

the coefficient is .59

$$1.1^{.59} = 1.058$$

A ten percent increase is associated with a 5.8 increase in average number of viewers.

3.

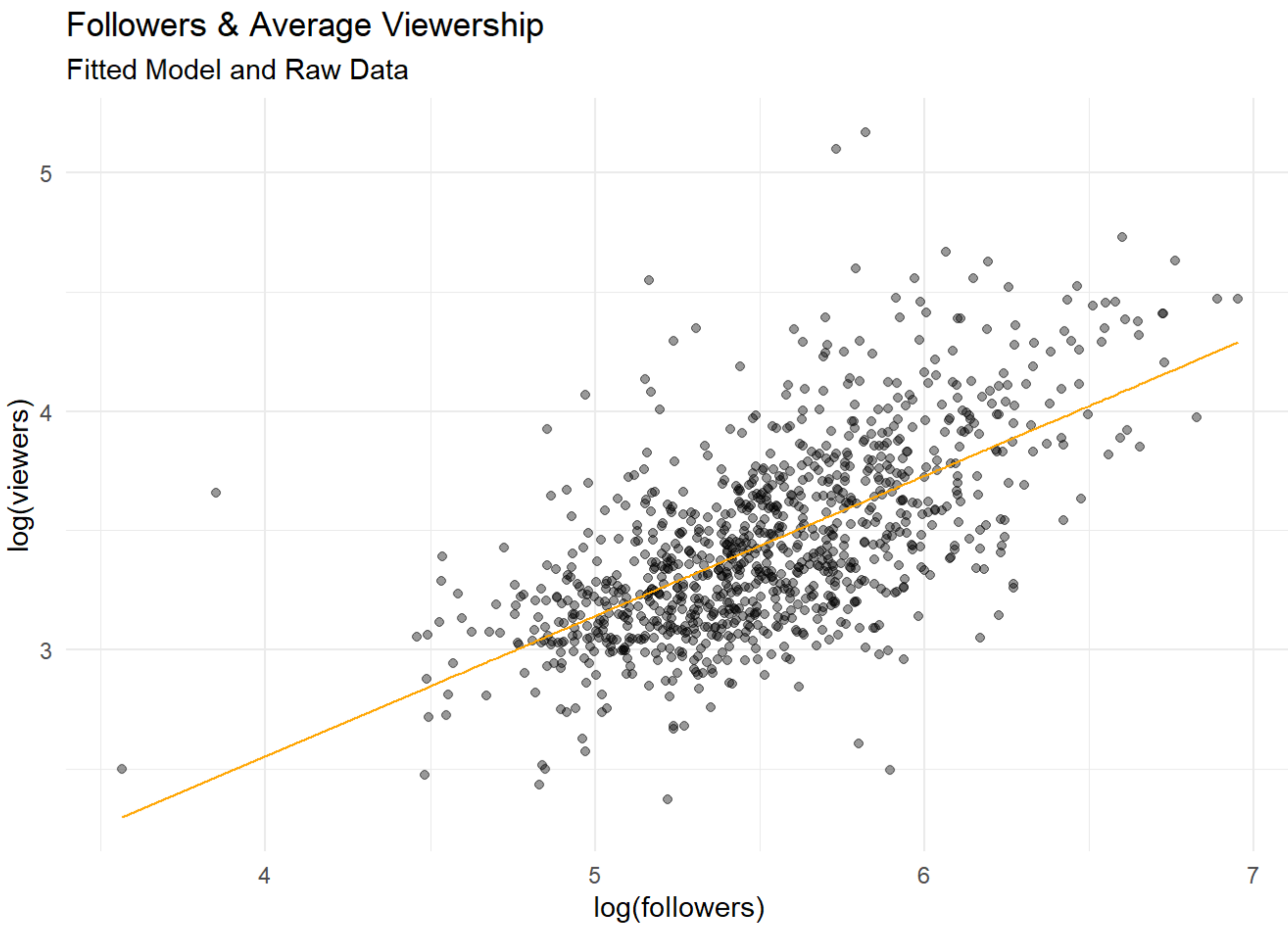
```r
library(broom)

pred_data <- augment(fit1)

# glimpse our new data
glimpse(pred_data)
```

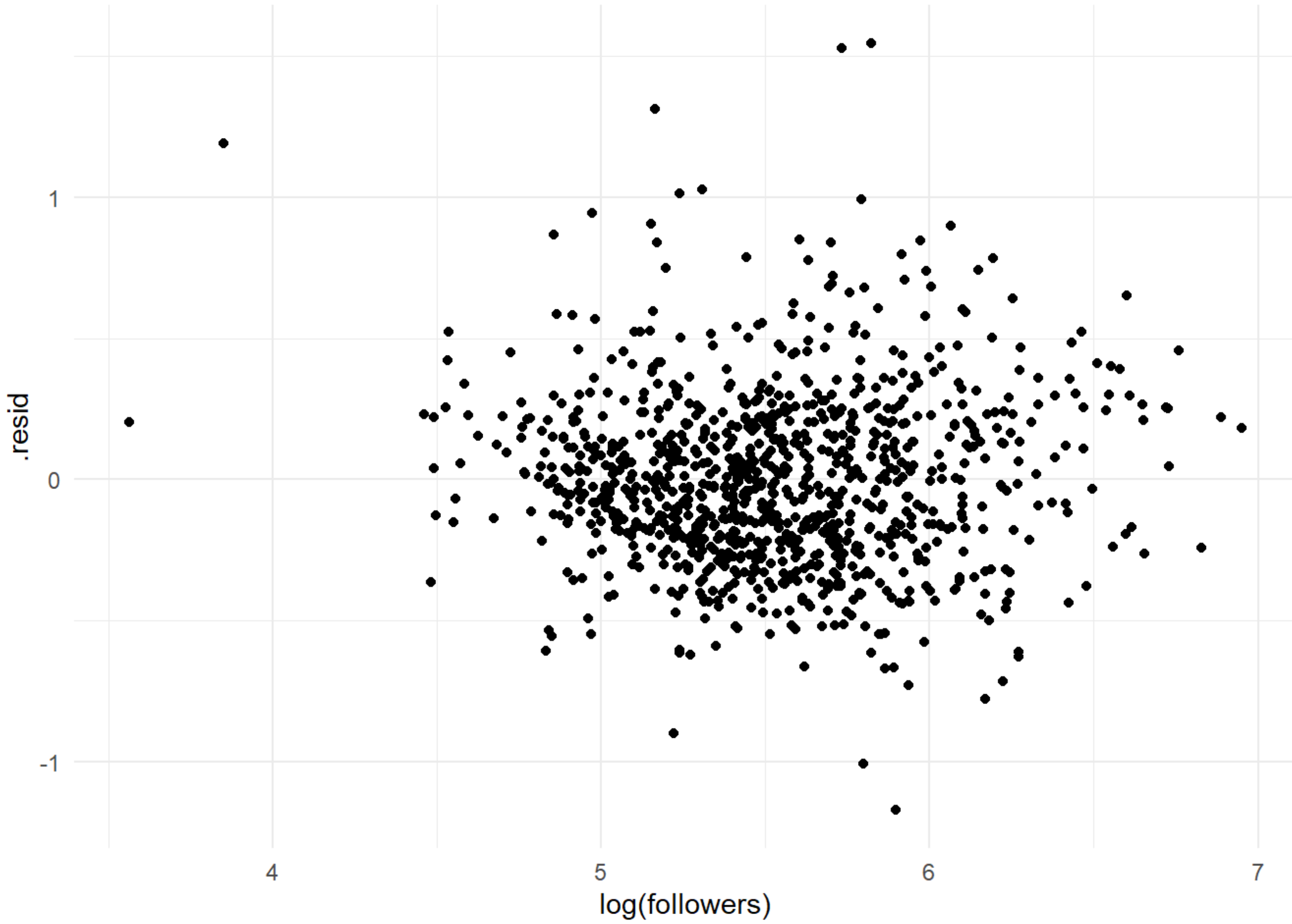
```
## Rows: 1,000
## Columns: 8
## $ log_viewers   <dbl> 4.442731, 4.408410, 4.040444, 3.887280, 4.471321, 4.6275…
## $ log_followers <dbl> 6.511388, 6.725108, 6.247393, 6.596030, 6.951284, 6.1940…
## $ .fitted       <dbl> 4.029761, 4.155534, 3.874400, 4.079572, 4.288638, 3.8430…
## $ .resid        <dbl> 0.4129697, 0.2528757, 0.1660436, -0.1922928, 0.1826833, …
## $ .hat          <dbl> 0.006194481, 0.008694557, 0.003782169, 0.007126066, 0.01…
## $ .sigma        <dbl> 0.3085580, 0.3087321, 0.3087919, 0.3087764, 0.3087820, 0…
## $ .cooksd       <dbl> 0.0056128779, 0.0029688873, 0.0005513456, 0.0014026033, …
## $ .std.resid    <dbl> 1.3420109, 0.8227954, 0.5389316, -0.6251793, 0.5953620, …
```

```r
pred_data %>%
  ggplot(aes(x = log_followers,
             y = log_viewers)) +
  geom_jitter(alpha = 0.4) +
  geom_line(aes(x = log_followers,
                y = .fitted),
            col = "orange") +
  theme_minimal() +
  labs(subtitle = "Fitted Model and Raw Data",
       title = "Followers & Average Viewership",
       x = "log(followers)",
       y = "log(viewers)")
```



The model shows the general positive slope of followers to viewers, but there is some variation.

```
pred_data %>%
  ggplot(aes(x = log_followers,
             y = .resid)) +
  geom_jitter(alpha = 0.4) +
  geom_point() +
  theme_minimal() +
  labs( x = "log(followers)", y = ".resid")
```



I don't think the model is that accurate because the residuals are all over the place. It doesn't seem to be more extreme at a particular x value though.

4.

```
twitch_data %>%
  select(language, average_viewers)
```

```
## # A tibble: 1,000 × 2
##    language   average_viewers
##    <chr>                <dbl>
##  1 English              27716
##  2 English              25610
##  3 Portuguese           10976
##  4 English               7714
##  5 English              29602
##  6 English              42414
##  7 English              24181
##  8 English              18985
##  9 English              22381
## 10 English              12377
## # … with 990 more rows
```

```
glimpse
```

```
## function (x, width = NULL, ...)
## {
##     UseMethod("glimpse")
## }
## <bytecode: 0x00000188dec34f58>
## <environment: namespace:pillar>
```

```
SubsetTwitchData <- twitch_data %>%
  select(language, average_viewers)
```

```
SubsetTwitchData %>%
  skim()
```

Data summary

Name	Piped data
Number of rows	1000
Number of columns	2
________________________	
Column type frequency:	
character	1
numeric	1
________________________	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
language	0	1	4	10	0	21	0

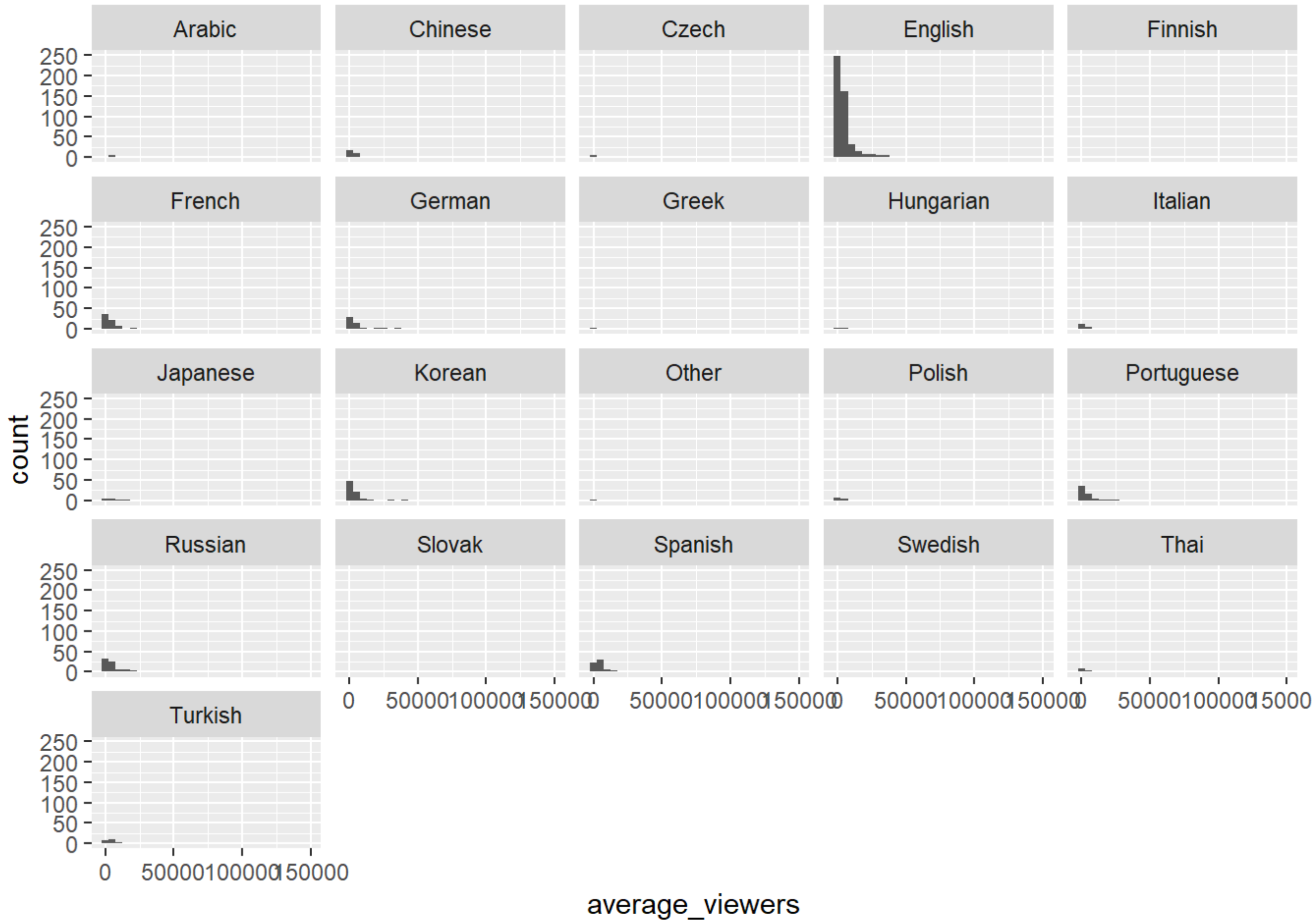
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
average_viewers	0	1	4781.04	8453.68	235	1457.75	2425	4786.25	147643	▇▁▁▁▁

The average number of viewers is 4781.04.

```
ggplot(SubsetTwitchData, aes(x = average_viewers)) +geom_histogram()+facet_wrap("language")
```

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



5.

```
twitch_data <- twitch_data %>%
  mutate(language = as.factor(language),
         language = relevel(language, ref = "English"))
```

```
ViewersModel <- lm(average_viewers ~ language, data = twitch_data)
get_regression_table(ViewersModel)
```

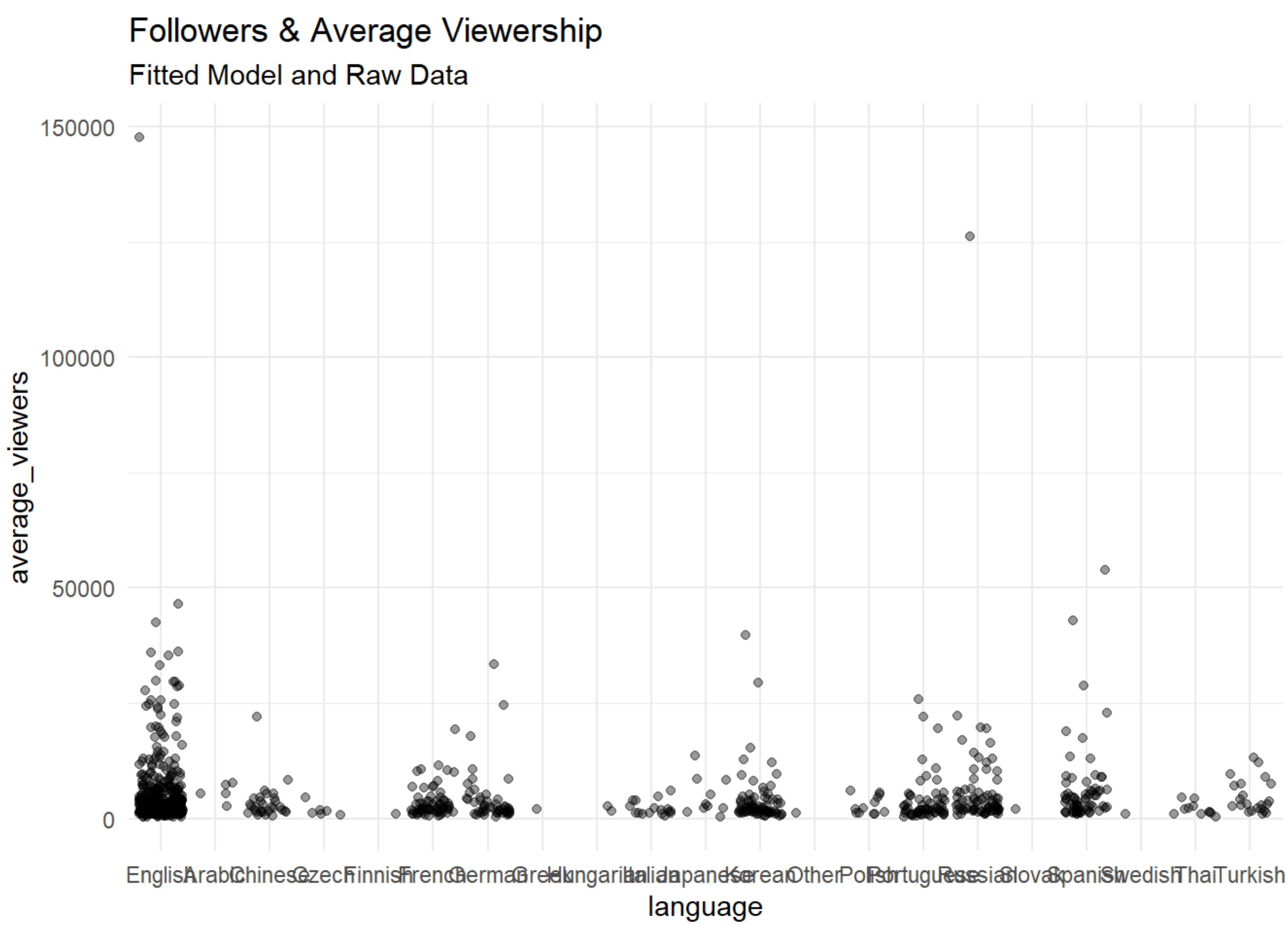
```
## # A tibble: 21 × 7
##    term                estimate std_error statistic p_value lower_ci upper_ci
##    <chr>                  <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
##  1 intercept              5113.      385.    13.3     0         4358.    5868.
##  2 language: Arabic        569.     3808.     0.15    0.881    -6903.    8042.
##  3 language: Chinese     -1688.     1594.    -1.06    0.29     -4815.    1439.
##  4 language: Czech       -3285.     3480.    -0.944   0.345   -10113.    3543.
##  5 language: Finnish     -4086.     8480.    -0.482   0.63    -20726.   12555.
##  6 language: French      -1606.     1111.    -1.44    0.149    -3787.     575.
##  7 language: German       -835.     1270.    -0.657   0.511    -3326.    1657.
##  8 language: Greek       -3152.     8480.    -0.372   0.71    -19792.   13489.
##  9 language: Hungarian   -2972.     6002.    -0.495   0.621   -14751.    8806.
## 10 language: Italian     -2907.     2090.    -1.39    0.165    -7009.    1194.
## # … with 11 more rows
```

The prediction that English streamers have more viewers is predicted to be accurate except for Arabic for some reason.

6.

```
pred_data2 <- augment(ViewersModel)
```

```
pred_data2 %>%
  ggplot(aes(x = language,
             y = average_viewers)) +
  geom_jitter(alpha = 0.4) +
  geom_line(aes(x = language,
                y = .fitted),
            col = "orange") +
  theme_minimal() +
  labs(subtitle = "Fitted Model and Raw Data",
       title = "Followers & Average Viewership",
       x = "language",
       y = "average_viewers")
```



Chapter 6

1.

```r
library(tidyverse)
# Set our ggplot theme from the outset
theme_set(theme_light())
# Read in the data
gender_employment <- read_csv("https://raw.githubusercontent.com/vaiseys/223_course/main/Data/gen
der_employment.csv")
```

```
## Rows: 2088 Columns: 12
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (3): occupation, major_category, minor_category
## dbl (9): year, total_workers, workers_male, workers_female, percent_female, ...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# Glimpse at the data
glimpse(gender_employment)
```

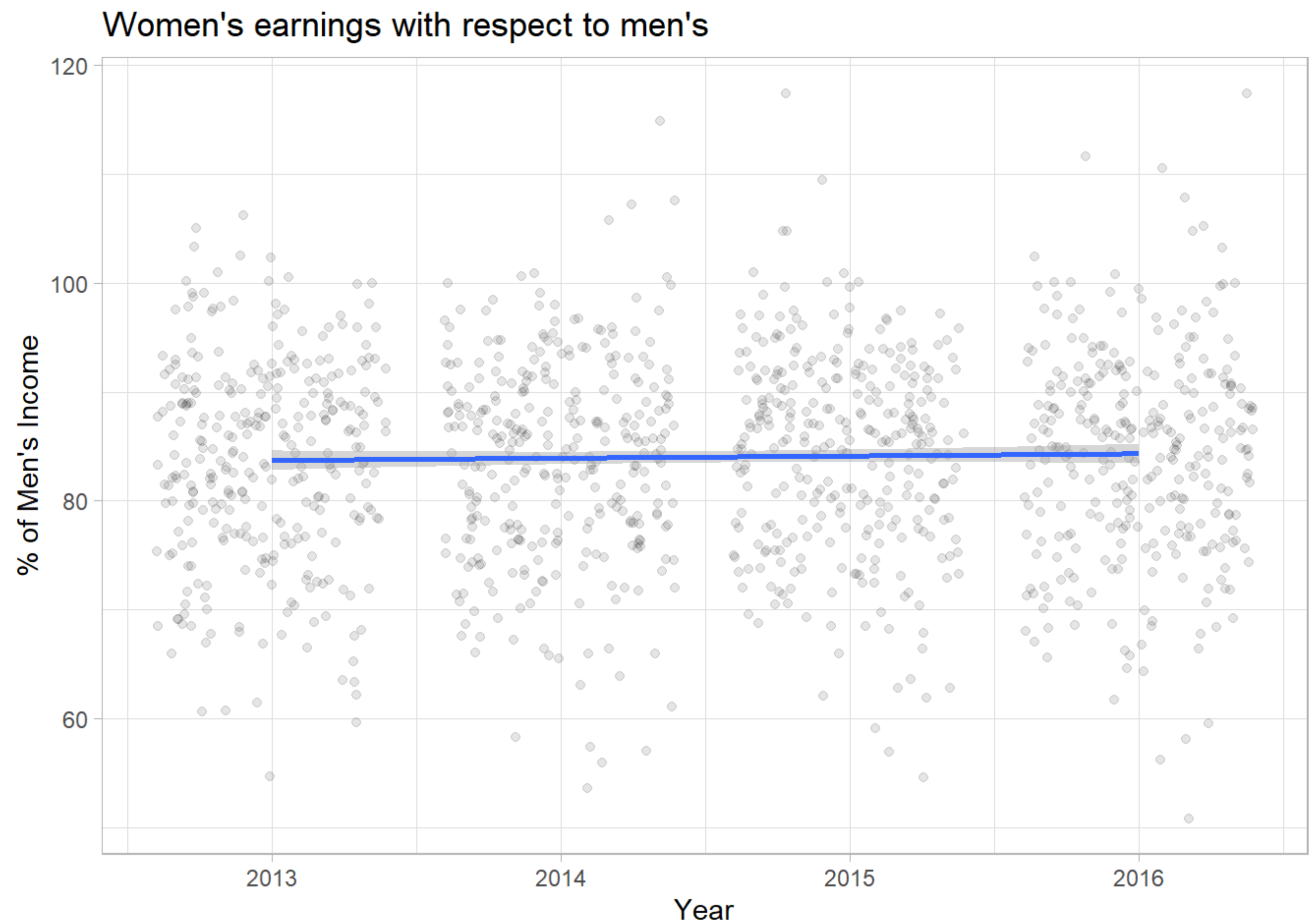
```
## Rows: 2,088
## Columns: 12
## $ year                  <dbl> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, …
## $ occupation            <chr> "Chief executives", "General and operations mana…
## $ major_category        <chr> "Management, Business, and Financial", "Manageme…
## $ minor_category        <chr> "Management", "Management", "Management", "Manag…
## $ total_workers         <dbl> 1024259, 977284, 14815, 43015, 754514, 44198, 10…
## $ workers_male          <dbl> 782400, 681627, 8375, 17775, 440078, 16141, 7287…
## $ workers_female        <dbl> 241859, 295657, 6440, 25240, 314436, 28057, 3683…
## $ percent_female        <dbl> 23.6, 30.3, 43.5, 58.7, 41.7, 63.5, 33.6, 27.5, …
## $ total_earnings        <dbl> 120254, 73557, 67155, 61371, 78455, 74114, 62187…
## $ total_earnings_male   <dbl> 126142, 81041, 71530, 75190, 91998, 90071, 66579…
## $ total_earnings_female <dbl> 95921, 60759, 65325, 55860, 65040, 66052, 55079,…
## $ wage_percent_of_male  <dbl> 76.04208, 74.97316, 91.32532, 74.29179, 70.69719…
```

```r
gender_employment%>%
  ggplot(aes(x = year, y = wage_percent_of_male)) +
  geom_jitter(alpha = 0.1) +
  geom_smooth(method = "lm") +
  labs(title = "Women's earnings with respect to men's",
       y = "% of Men's Income",
       x = "Year")
```

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

```
## Warning: Removed 846 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 846 rows containing missing values (`geom_point()`).
```




```
gender_employment <- gender_employment %>%
  mutate(major_category = as.factor(major_category),
         major_category = relevel(major_category, ref = "Management, Business, and Financial"))
```

```
# Fit regression model:
parallel_model <- lm(wage_percent_of_male ~ year + major_category, data = gender_employment)

tidy(parallel_model)
```

```
## # A tibble: 9 × 5
##   term                                      estimate std.e…¹ stati…²  p.value
##   <chr>                                        <dbl>   <dbl>   <dbl>    <dbl>
## 1 (Intercept)                               -307.     459.    -0.669 5.04e- 1
## 2 year                                         0.192   0.228   0.844 3.99e- 1
## 3 major_categoryComputer, Engineering, and Sc…    6.32    0.946   6.68  3.56e-11
## 4 major_categoryEducation, Legal, Community S…    5.76    0.985   5.84  6.53e- 9
## 5 major_categoryHealthcare Practitioners and …    5.52    1.10    5.00  6.41e- 7
## 6 major_categoryNatural Resources, Constructi…    4.91    1.24    3.95  8.15e- 5
## 7 major_categoryProduction, Transportation, a…   -1.31    0.960  -1.37  1.72e- 1
## 8 major_categorySales and Office               3.33    0.858   3.88  1.11e- 4
## 9 major_categoryService                        6.08    0.885   6.87  1.03e-11
## # … with abbreviated variable names ¹std.error, ²statistic
```

intercept = -306.72 service offset 6.09 slope = 0.19

equation

-306.72+ 6.09 = -300.63

-300.63 + .19(2016) = 82.41

Women in the service industry made 82.41 percent what men made

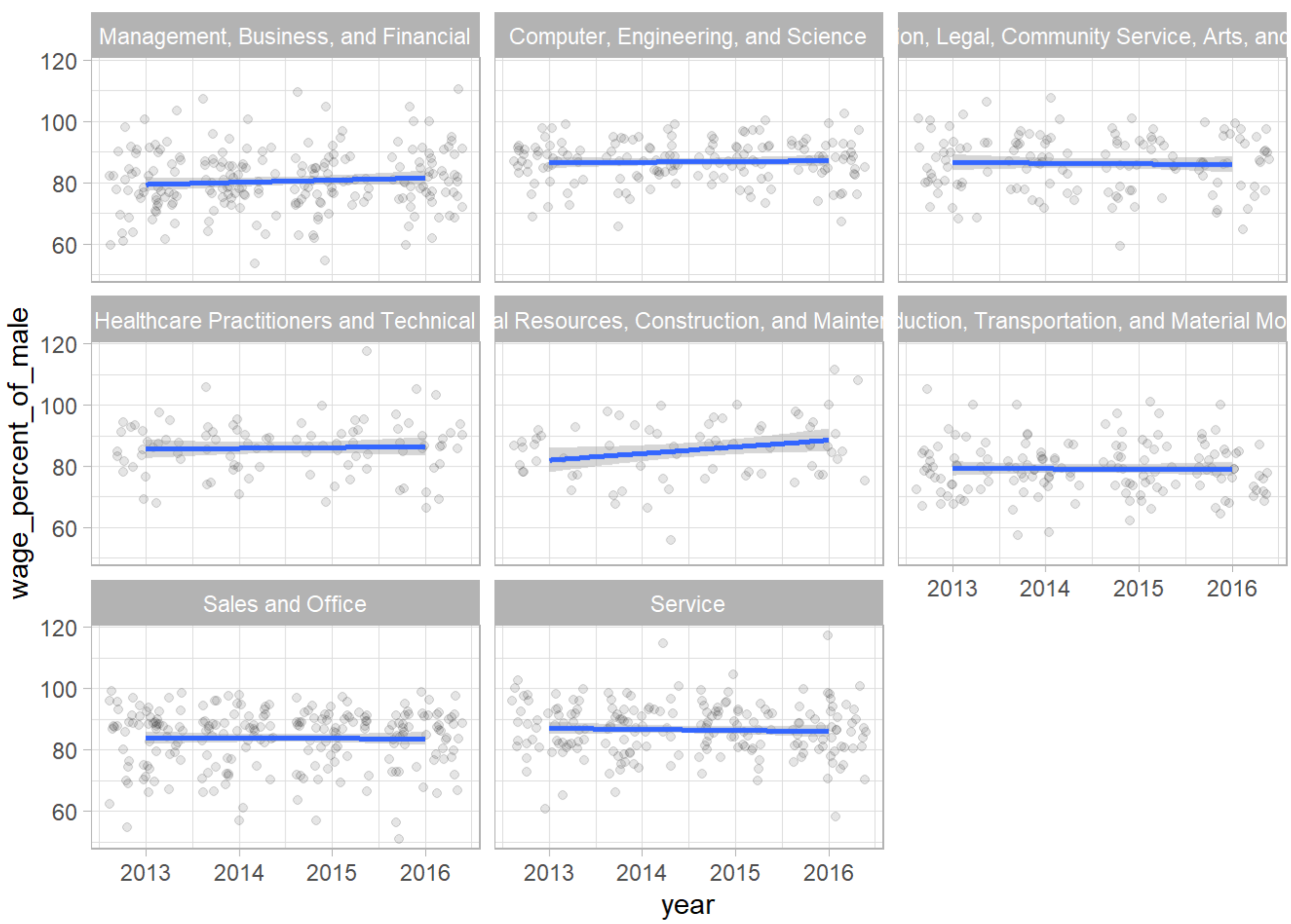
2.

```
gender_employment%>%
  ggplot(aes(x = year, y = wage_percent_of_male)) +
  geom_jitter(alpha = 0.1) +
  geom_smooth(method = "lm") + facet_wrap("major_category")
```

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

```
## Warning: Removed 846 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 846 rows containing missing values (`geom_point()`).
```




```
  labs(title = "Women's earnings with respect to men's",
       y = "% of Men's Income",
       x = "Year")
```

```
## $y
## [1] "% of Men's Income"
##
## $x
## [1] "Year"
##
## $title
## [1] "Women's earnings with respect to men's"
##
## attr(,"class")
## [1] "labels"
```

I think it is relatively similar accross the major categories. Wages have gotten more equal in the Construction category.

3.

```
# Fit regression model:
interaction_model <- lm(wage_percent_of_male ~ year * major_category, data = gender_employment)

tidy(interaction_model)
```

```
## # A tibble: 16 × 5
##    term                                         estimate std.e…¹ stati…² p.value
##    <chr>                                           <dbl>   <dbl>   <dbl>   <dbl>
##  1 (Intercept)                                 -1.37e+3 1.11e+3  -1.24    0.216
##  2 year                                         7.20e-1 5.49e-1   1.31    0.190
##  3 major_categoryComputer, Engineering, and Sc…  1.00e+3 1.70e+3   0.589   0.556
##  4 major_categoryEducation, Legal, Community S…  1.94e+3 1.77e+3   1.09    0.275
##  5 major_categoryHealthcare Practitioners and …  9.06e+2 1.99e+3   0.456   0.649
##  6 major_categoryNatural Resources, Constructi… -2.89e+3 2.23e+3  -1.29    0.196
##  7 major_categoryProduction, Transportation, a…  1.58e+3 1.73e+3   0.909   0.363
##  8 major_categorySales and Office                1.61e+3 1.54e+3   1.05    0.296
##  9 major_categoryService                         2.14e+3 1.59e+3   1.34    0.180
## 10 year:major_categoryComputer, Engineering, a… -4.95e-1 8.45e-1  -0.585   0.559
## 11 year:major_categoryEducation, Legal, Commun… -9.59e-1 8.80e-1  -1.09    0.276
## 12 year:major_categoryHealthcare Practitioners… -4.47e-1 9.86e-1  -0.453   0.651
## 13 year:major_categoryNatural Resources, Const…  1.44e+0 1.11e+0   1.30    0.195
## 14 year:major_categoryProduction, Transportati… -7.84e-1 8.61e-1  -0.910   0.363
## 15 year:major_categorySales and Office          -7.98e-1 7.65e-1  -1.04    0.297
## 16 year:major_categoryService                   -1.06e+0 7.92e-1  -1.34    0.182
## # … with abbreviated variable names ¹std.error, ²statistic
```

intercept = -1370.47 year = 0.72

offsets 1002.85 -0.49

for computers, engineering, and science the equation is

(-1370.47+1002.85)+ (.72-.49)(year)

-367.62+ 0.23(2016) -367.62+ 463.68=96.06

Women in Computers, Engineering, and Science made 96.06% of the wages that men made

intercept = -1370.47 year = 0.72 Offsets for Service 2137.65 -1.058

Equation for Service Industry (-1370.47+2137.65)+((.72-1.058)*2016)= 85.77200

Women in the service industry make 85.77% as much as men.

Therefore there is more pay equality in Computers than in the service industry.

4.

The book discusses this question through an explanation of occam's razor, essentially the simpliest explanation is the most likely, and that the differing slopes do not actually add all that much to our understanding of pay inequality.

5.

```
gender_employment %>%
  select(year, wage_percent_of_male, percent_female) %>%
  cor(use = "complete.obs")
```

```
##                             year wage_percent_of_male percent_female
## year                 1.000000000           0.02403895    0.004998286
## wage_percent_of_male 0.024038950           1.00000000    0.111464461
## percent_female       0.004998286           0.11146446    1.000000000
```

```
simple_fit <- lm(wage_percent_of_male ~year, data=gender_employment)
tidy(simple_fit)
```

```
## # A tibble: 2 × 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept) -322.      479.       -0.671   0.502
## 2 year           0.201     0.238     0.847   0.397
```

```
multiple_fit <- lm(wage_percent_of_male ~ year + percent_female, data=gender_employment)
tidy(multiple_fit)
```

```
## # A tibble: 3 × 5
##   term            estimate std.error statistic   p.value
##   <chr>              <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)    -314.      477.        -0.660 0.510
## 2 year              0.197     0.237      0.832 0.406
## 3 percent_female    0.0425    0.0108     3.94  0.0000843
```

For every percent an industry is more female the percent of wages that women have compared to men gets 4% more equal.

6.

R squared is a measure of how much of the variaton in the dependent variables is explained by variation in the independent variable.

```
simple_glanced <- glance(simple_fit)
simple_glanced$r.squared
```

```
## [1] 0.0005778711
```

```
multiple_glanced <- glance(multiple_fit)
multiple_glanced$r.squared
```

```
## [1] 0.01297574
```

Chapter 6 Extra

1.

```
library(tidyverse)
library(moderndive)
theme_set(theme_minimal())
```

```
data(bikes, package = "bayesrules")
glimpse(bikes)
```

```
## Rows: 500
## Columns: 13
## $ date        <date> 2011-01-01, 2011-01-03, 2011-01-04, 2011-01-05, 2011-01-0…
## $ season      <fct> winter, winter, winter, winter, winter, winter, winter, wi…
## $ year        <int> 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2011, 2011…
## $ month       <fct> Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan…
## $ day_of_week <fct> Sat, Mon, Tue, Wed, Fri, Sat, Mon, Tue, Wed, Thu, Fri, Sat…
## $ weekend     <lgl> TRUE, FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, FALSE, FALS…
## $ holiday     <fct> no, no, no, no, no, no, no, no, no, no, no, no, no, yes, n…
## $ temp_actual <dbl> 57.39952, 46.49166, 46.76000, 48.74943, 46.50332, 44.17700…
## $ temp_feel   <dbl> 64.72625, 49.04645, 51.09098, 52.63430, 50.79551, 46.60286…
## $ humidity    <dbl> 80.5833, 43.7273, 59.0435, 43.6957, 49.8696, 53.5833, 48.2…
## $ windspeed   <dbl> 10.749882, 16.636703, 10.739832, 12.522300, 11.304642, 17.…
## $ weather_cat <fct> categ2, categ1, categ1, categ1, categ2, categ2, categ1, ca…
## $ rides       <int> 654, 1229, 1454, 1518, 1362, 891, 1280, 1220, 1137, 1368, …
```

```
bikes %>%
  get_correlation(formula = rides ~ temp_feel)
```

```
##         cor
## 1 0.5824898
```

2.

```
bikes$wind_kph <- (bikes$windspeed)*1.61
```

3.

```
RidesWindspeedMPH <- lm(rides ~ windspeed, data=bikes)
tidy(RidesWindspeedMPH)
```

```
## # A tibble: 2 × 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   4205.      177.       23.8  5.99e-84
## 2 windspeed      -55.5      12.5     -4.44 1.13e- 5
```

```
RidesWindspeedKPH <- lm(rides ~ wind_kph, data=bikes)
tidy(RidesWindspeedKPH)
```

```
## # A tibble: 2 × 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   4205.     177.        23.8  5.99e-84
## 2 wind_kph       -34.5      7.78     -4.44 1.13e- 5
```

4.

4205.06 + -55.52(20)= 3094.66

5.

```
bikes$temp_c <- (((bikes$temp_feel)-30)/2)
```

```
WindTempRides <- lm(rides ~ temp_c + wind_kph, data = bikes)
tidy(WindTempRides)
```

```
## # A tibble: 3 × 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)    783.     264.        2.96 3.19e- 3
## 2 temp_c         159.      10.3      15.5  1.65e-44
## 3 wind_kph       -19.8      6.46     -3.07 2.24e- 3
```

6. SITUATION 1: temp = 25C, wind = 15 KPH SITUATION 2: temp = 15C, wind = 5 KPH SITUATION 3: temp = 10C, wind = 40 KPH

783.27 +159.15(25) + -19.84(15)=4464.42 Rides 783.27 +159.15(15) + -19.84(5)=3071.32 Rides 783.27 +159.15(10) + -19.84(40)=1581.17 Rides

7.

```
WeekendWindTempRides <- lm(rides ~ temp_c + wind_kph + weekend, data = bikes)
tidy(WeekendWindTempRides)
```

```
## # A tibble: 4 × 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   1059.     260.        4.07 5.53e- 5
## 2 temp_c         156.       9.96     15.7  3.26e-45
## 3 wind_kph       -20.4      6.26     -3.26 1.20e- 3
## 4 weekendTRUE   -714.     122.       -5.83 1.02e- 8
```

On the weekend’s ridership declines

8.

```
WeekendRiderShip <- lm(rides~ weekend, data = bikes)
tidy(WeekendRiderShip)
```

```
## # A tibble: 2 × 5
##   term        estimate std.error statistic   p.value
##   <chr>          <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)    3712.      80.9     45.9  5.42e-181
## 2 weekendTRUE    -815.     152.      -5.35 1.33e-  7
```

I think that if wind and temperature are not factors they are not included in the model.

9.

```
WeekendWindTempRides <- lm(rides ~ temp_c + wind_kph + weekend, data = bikes)
tidy(WeekendWindTempRides)
```

```
## # A tibble: 4 × 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   1059.     260.        4.07 5.53e- 5
## 2 temp_c         156.       9.96     15.7  3.26e-45
## 3 wind_kph       -20.4      6.26     -3.26 1.20e- 3
## 4 weekendTRUE   -714.     122.       -5.83 1.02e- 8
```

```
get_regression_points(WeekendWindTempRides)
```

```
## # A tibble: 500 × 7
##       ID rides temp_c wind_kph weekend rides_hat residual
##    <int> <int>  <dbl>    <dbl> <lgl>       <dbl>    <dbl>
##  1     1   654  17.4      17.3 TRUE        2700.   -2046.
##  2     2  1229   9.52     26.8 FALSE       1998.    -769.
##  3     3  1454  10.5      17.3 FALSE       2351.    -897.
##  4     4  1518  11.3      20.2 FALSE       2413.    -895.
##  5     5  1362  10.4      18.2 FALSE       2309.    -947.
##  6     6   891   8.30     28.8 TRUE        1053.    -162.
##  7     7  1280   7.79     24.1 FALSE       1783.    -503.
##  8     8  1220   9.62     13.2 FALSE       2290.   -1070.
##  9     9  1137   8.22     32.9 FALSE       1671.    -534.
## 10    10  1368   7.79     32.5 FALSE       1612.    -244.
## # … with 490 more rows
```