MATH 154 - HW3 - Wrangling

your name here

due: Thursday, Septemper 23, 2021

summary

In this assignment we will work with the packages in the **tidyverse** (namely **dplyr** and **tidyr**). The data for the assignment are given in the packages **openintro** (`seattlepets` and `ucla_textbooks_f18`) and **babynames**.

requisites

Chapters 5 (Data Transformation), 12 (Tidy data), and 13 (Relational data) in R for Data Science are all extremely helpful. You also may want to look at the RStudio cheatsheets.

data background

```
library(tidyverse)
library(openintro)  # for seattlepets & ucla_textbooks_f18
library(babynames)  # for babynames
```

assignment

0. `reprex()` A problem which is not due. On your exam, you will be required to email me a reproducible example. That is, you'll ask a question (e.g., "why don't the lines connect to the dots?") and have code which demonstrates what the code is doing. Good idea to practice now in creating reprexes! that is, as you get stuck, try to come up with a question you could ask which would help getting unstuck. You don't have to use the `reprex()`function, but sometimes it helps. Here is some advice: https://stackoverflow.com/help/minimal-reproducible-example Email me reprexes! Post reprexes to Discord!

1. **Pod Q** Describe one thing you learned from someone in your pod this week (it could be: content, logistical help, background material, R information, something fun, etc.) 1-3 sentences.

I learned from Brian and JJ that I should aim to reduce the number of verbs used when writing code as this is an aspect of good programming.

2. **pets** Use the `seattlepets` dataset from the **openintro** R package to do some wrangling:

a. How many pets are included in the dataset? (Print the answer to the screen, that is, show the R code which is your work, and write a complete sentence with the answer.)

Row index is by liscence number and so by individual pet

```
seattlepets %>% nrow()
```

```
## [1] 52519
```

There are 52519 Pets represented in this data.

b. How many variables are there for each pet? Again, show your work using R code and write a complete sentence.

```
seattlepets %>% ncol()
```

```
## [1] 7
```

There are 7 variables for each pet in the data set.

c. What are the three most common pet names in Seattle? (You'll need to use the function `n()` which counts the number of rows.)

```
seattlepets  %>% group_by(animal_name) %>% summarise(occurance = n()) %>% arrange(desc(occurance)) %>% h
```

```
## # A tibble: 3 x 2
##   animal_name occurance
##   <chr>           <int>
## 1 Lucy              439
## 2 Charlie           387
## 3 Luna              355
```

Lucy, charlie, Luna are the 3 most popular names.

d. What are the three most common names for each of the cat and dog species? Your initial code may only tell you about dogs (and the poor unnamed kitties). The `slice_` family of functions pulls out a specified number of rows. For example, `slice_min()` pulls out the smallest rows, `slice_max()` pulls out the largest rows, `slice_head()` pulls out the first row, ... (see the cheatsheets! https://www.rstudio.com/resources/cheatsheets/).

```
seattlepets %>% filter(species == "Cat") %>% group_by(animal_name) %>% summarise(occurance = n()) %>% fi
```

```
## # A tibble: 3 x 2
##   animal_name occurance
##   <chr>           <int>
## 1 Luna              111
## 2 Lucy              102
## 3 Lily               86
```

Luna, Lucy, and Lily are the 3 most popular cat names

```
seattlepets %>% filter(species == "Dog") %>% group_by(animal_name) %>% summarise(occurance = n()) %>% fi
```

```
## # A tibble: 3 x 2
##   animal_name occurance
##   <chr>           <int>
## 1 Lucy              337
## 2 Charlie           306
## 3 Bella             249
```

Lucy, Charlie, and Bella are the 3 most popular dog names.

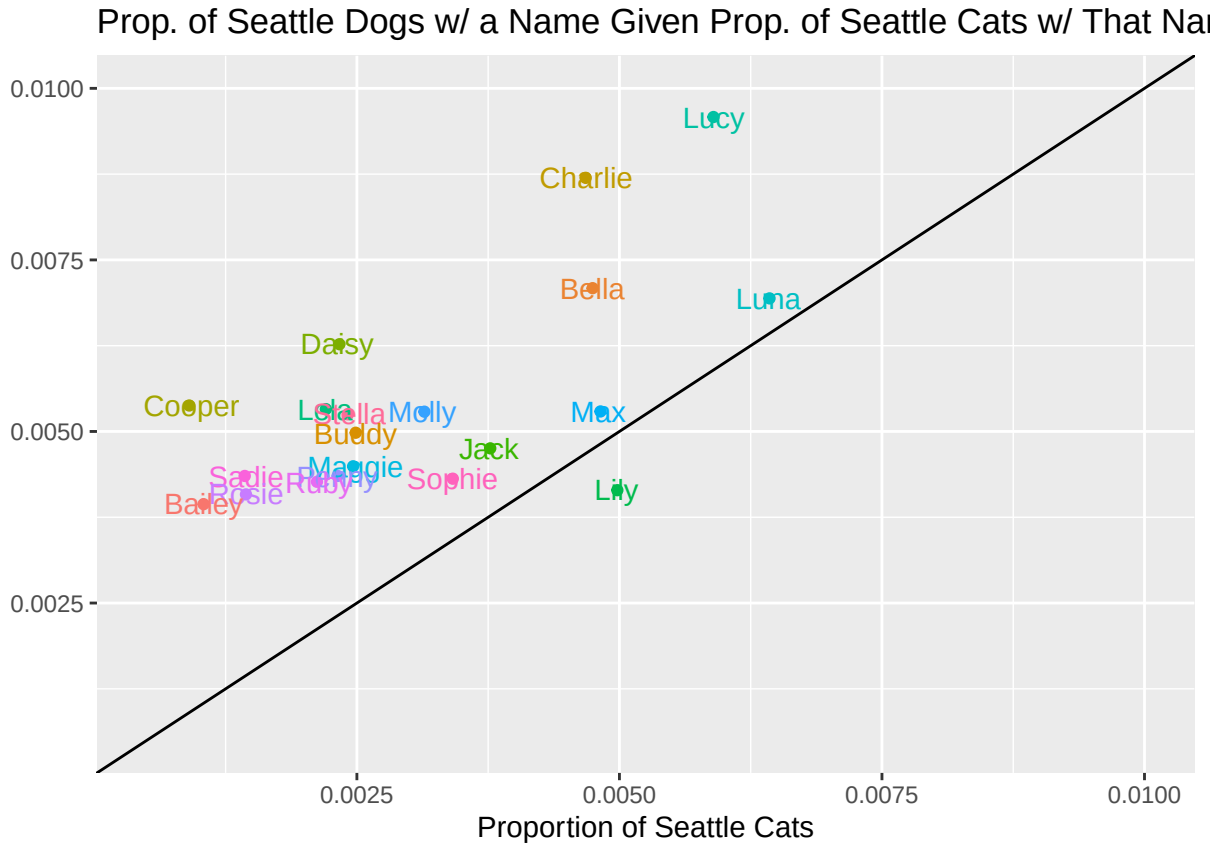
e. I've added a new column to the dataset which gives the proportion of a particular species with the given name. Create a scatterplot of the 20 most popular pet names (as measured by the proportion of the species with that name - the value calculated below). The x-axis will represent the proportion of cats with that name, the y-axis will represent the proportion of dogs with that name.

```
seattlepets_w_prop <- seattlepets %>%
  group_by(species, animal_name) %>%
  summarize(n_names = n()) %>%
  mutate(prop_names = n_names / sum(n_names)) %>%
  ungroup()

head(seattlepets_w_prop)
```

```
## # A tibble: 6 x 4
##   species animal_name     n_names prop_names
##   <chr>   <chr>             <int>      <dbl>
## 1 Cat     "__"                  1  0.0000578
## 2 Cat     "-"                   1  0.0000578
## 3 Cat     "'Alani"              1  0.0000578
## 4 Cat     "\"Mama\" Maya"       1  0.0000578
## 5 Cat     "\"Mo\""              1  0.0000578
## 6 Cat     "1"                   1  0.0000578
```

```
seattlepets_w_prop %>%
  filter(!is.na(animal_name))%>%
  pivot_wider(id_cols = animal_name, names_from = species, values_from = prop_names) %>%
  mutate( maximum_proportion = pmax(Cat, Dog, na.rm = TRUE)) %>% arrange(desc(Dog)) %>%
  slice_head(n = 20) %>%
  ggplot(mapping = aes(x = Cat , y = Dog, color = animal_name))+ geom_point(size = .5)+
  labs(x = "Proportion of Seattle Cats", y = "", title = "Prop. of Seattle Dogs w/ a Name Given Prop. o
  xlim(0.0005,.01)+
  ylim(0.0005,.01)+
  geom_jitter()+
  geom_text(aes(label = animal_name, key = FALSE))+
  theme(legend.position = "none")+
  geom_abline()
```



Hint 1: you'll need to `pivot_` (wider or longer?)
Hint 2: after pivoting, you'll need to sort based on the proportion. But you have two columns of proportions! Sort on the maximum of the two columns. In order to do a piece-wise maximum (element by element) in your `mutate()` call, use the function `pmax(first column, second column, na.rm = TRUE)`.
Hint3: after you get the basic scatterplot made, clean it up in the following ways:

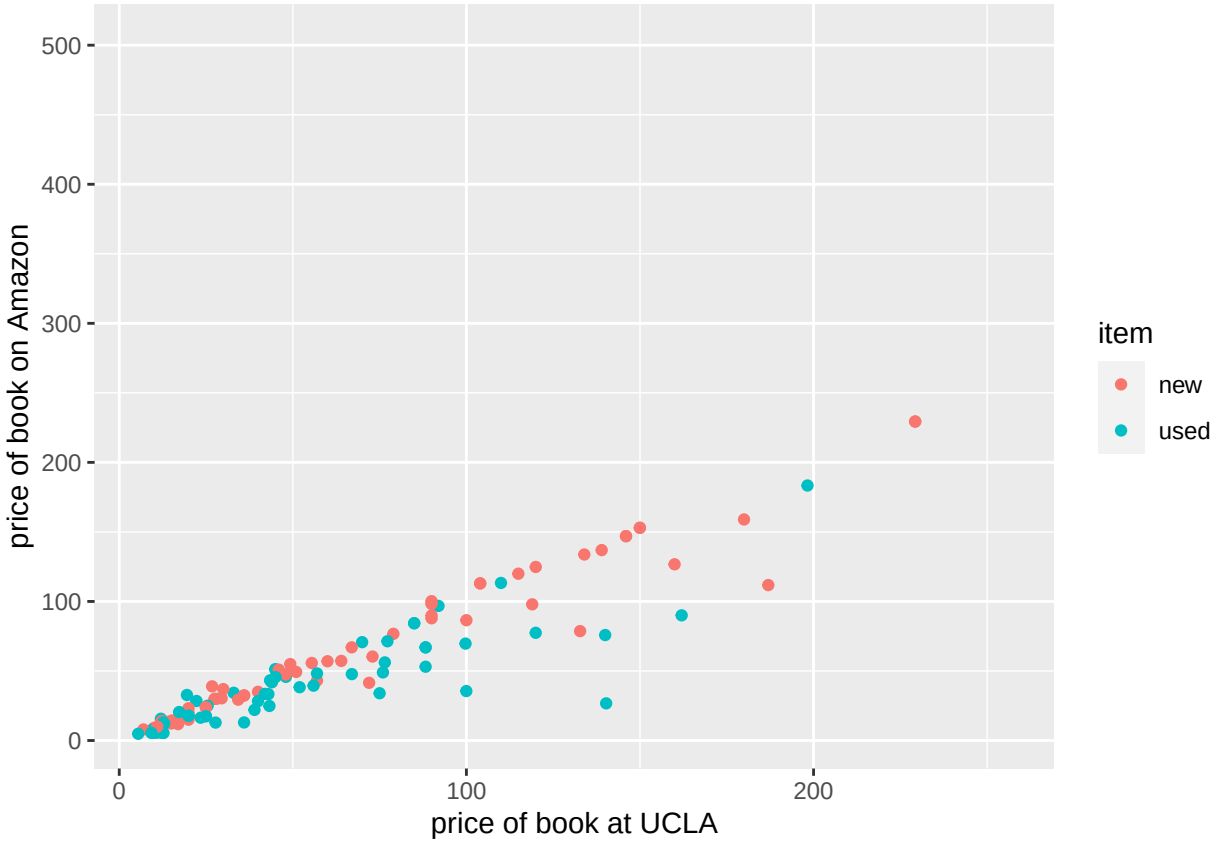
- add pet name labels using `geom_text()`
- add the line y = x using `geom_abline()`
- make the x-axis something better
- remove the y-axis and use the title to provide the y-axis (so that the letters are written horizontally instead of vertically)

3. **books** Using the `ucla_textbooks_f18` dataset from the **openintro** R package, create a plot with price of book at UCLA on the x-axis, price of book on Amazon on the y-axis, and color by whether the book was used or new.

Hint 1: a `pivot_` function will need to be used.
Hint 2: practice using help and reproducible examples. Someone asked a question about this on stackoverflow. You'll likely need to **run** their code (woo hoo!!! they posted a reprex!!) to figure out how to solve the problem: https://stackoverflow.com/questions/61940984/using-pivot-longer-with-multiple-paired-columns-in-the-wide-dataset

#ucla_textbooks_f18 %>% select(textbook_isbn, bookstore_new, bookstore_used, amazon_new, amazon_used) %>% filter(!is.na(textbook_isbn))

```
ucla_textbooks_f18 %>% select(textbook_isbn, bookstore_new, bookstore_used, amazon_new, amazon_used) %>%
    -textbook_isbn,
    names_to = c(".value", "item"),
    names_sep = "_"
  ) %>% ggplot(mapping = aes(x = bookstore , y = amazon, color = item))+ geom_point()+labs( x= "price o
```



4. **babynames** For each of the questions below, fix the R chunk so that it can compile and proivde the needed information. Note: Chunks in this template are headed with ```` ```{r eval=FALSE} ````. Change them to ```` ```{r} ```` when you are ready to compile / check your code. [No sentences needed, but show both the R code and the result from the call to the R code.]

Note: the column which represents the number of names is called `n` (unfortunately).

a. How many babies are represented? sum the column n

```
sum(babynames$n)
```

```
## [1] 348120517
```

b. How many babies are there in each year? group by year and sum the column in the groups

```
    babynames %>%
        group_by(year) %>%
    summarise(total = sum(n))
```

```
## # A tibble: 138 x 2
##     year  total
##    <dbl>  <int>
##  1  1880 201484
##  2  1881 192696
##  3  1882 221533
##  4  1883 216946
##  5  1884 243462
##  6  1885 240854
##  7  1886 255317
##  8  1887 247394
##  9  1888 299473
## 10  1889 288946
## # ... with 128 more rows
```

c. How many distinct names in each year? group by year and ask number o

```
    babynames %>%
        group_by(year) %>%
        summarise(name_count = n_distinct(name))
```

```
## # A tibble: 138 x 2
##     year name_count
##    <dbl>      <int>
##  1  1880       1889
##  2  1881       1830
##  3  1882       2012
##  4  1883       1962
##  5  1884       2158
##  6  1885       2139
##  7  1886       2225
##  8  1887       2215
##  9  1888       2454
## 10  1889       2390
## # ... with 128 more rows
```

d. How many distinct names of each sex in each year?

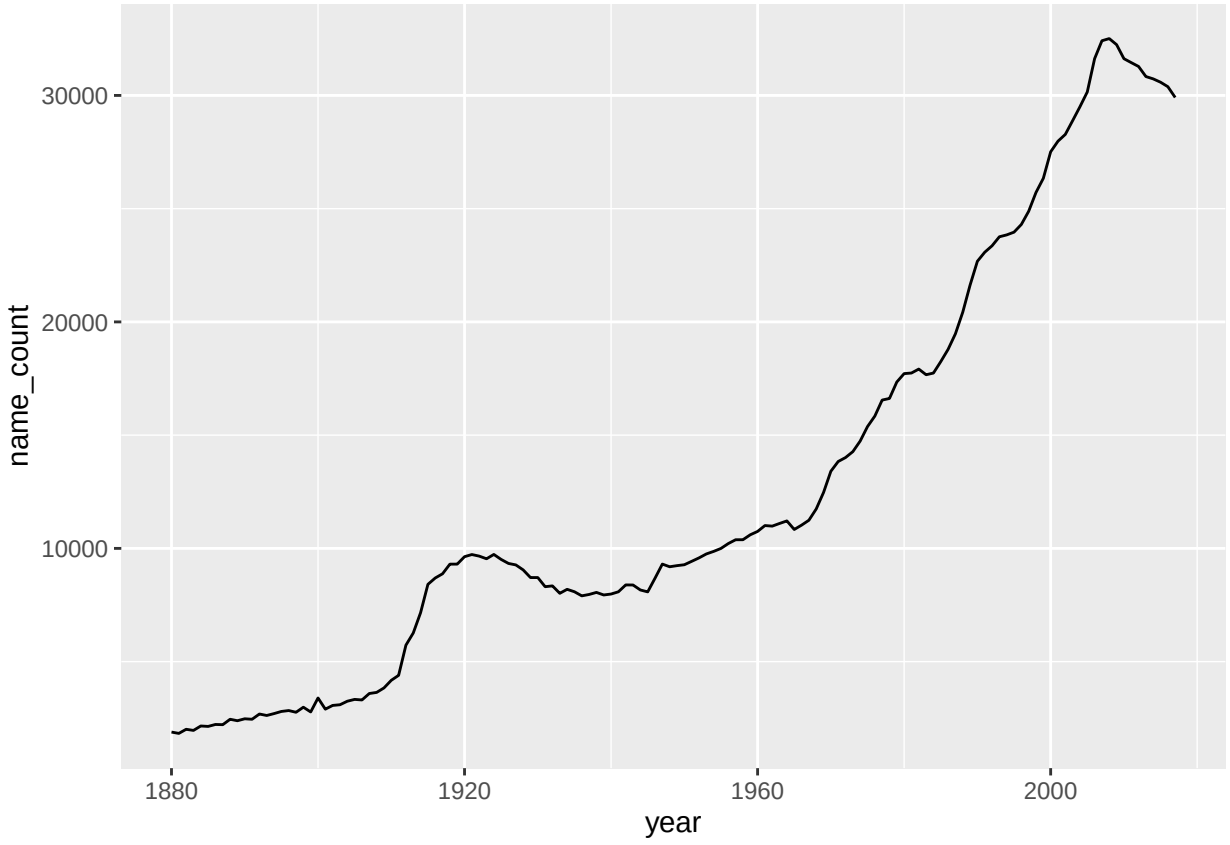
```
babynames %>%
         group_by(sex, year) %>%
         summarise(name_count = n_distinct(name))
```

```
## # A tibble: 276 x 3
## # Groups:   sex [2]
##    sex    year name_count
##    <chr> <dbl>      <int>
##  1 F      1880        942
```

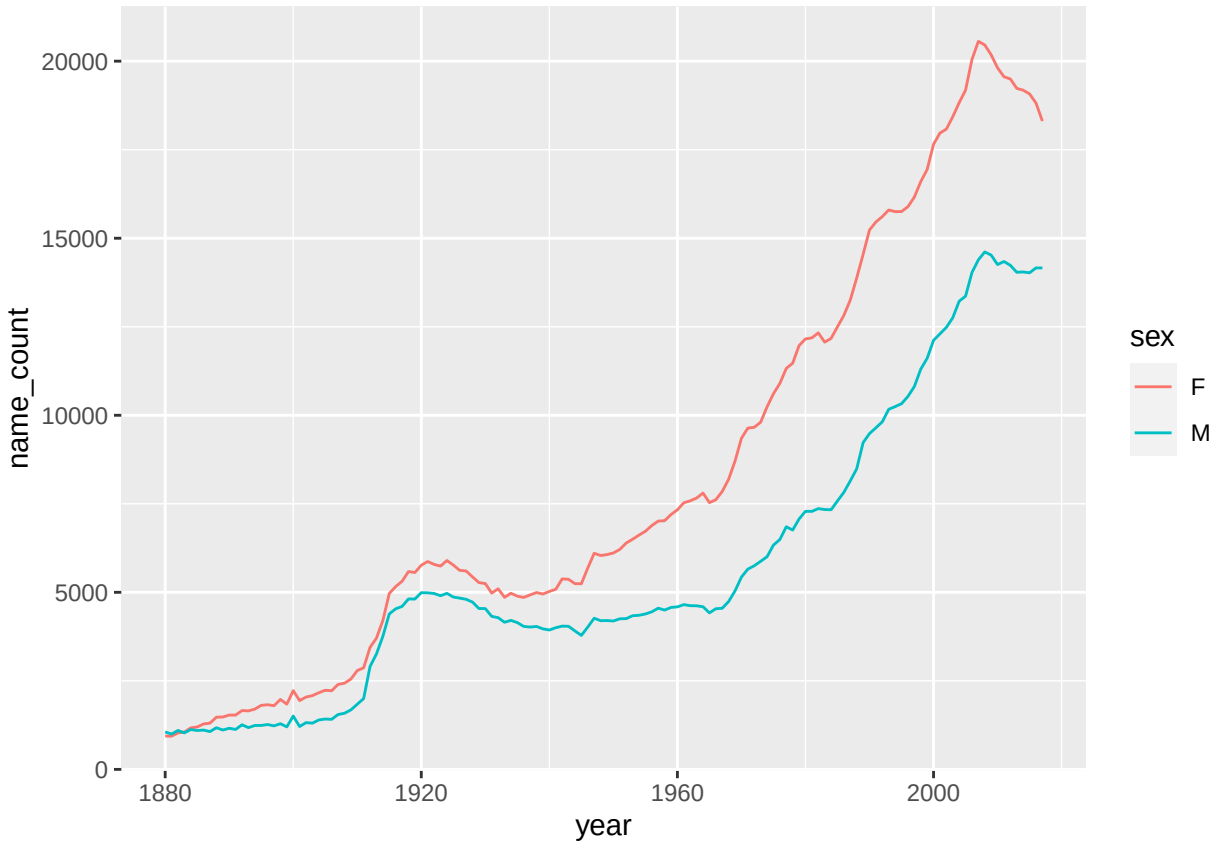
```
##  2 F      1881        938
##  3 F      1882       1028
##  4 F      1883       1054
##  5 F      1884       1172
##  6 F      1885       1197
##  7 F      1886       1282
##  8 F      1887       1306
##  9 F      1888       1474
## 10 F      1889       1479
## # ... with 266 more rows
```

e. Graphically summarize the previous two commands (two separate plots: (1) number of distinct names over time, (2) number of distinct names over time broken down by gender), because they are too long to look at as a table.

```
    babynames %>%
        group_by(year) %>%
        summarise(name_count = n_distinct(name)) %>%
ggplot(mapping = aes(x = year, y = name_count))+ geom_line()
```

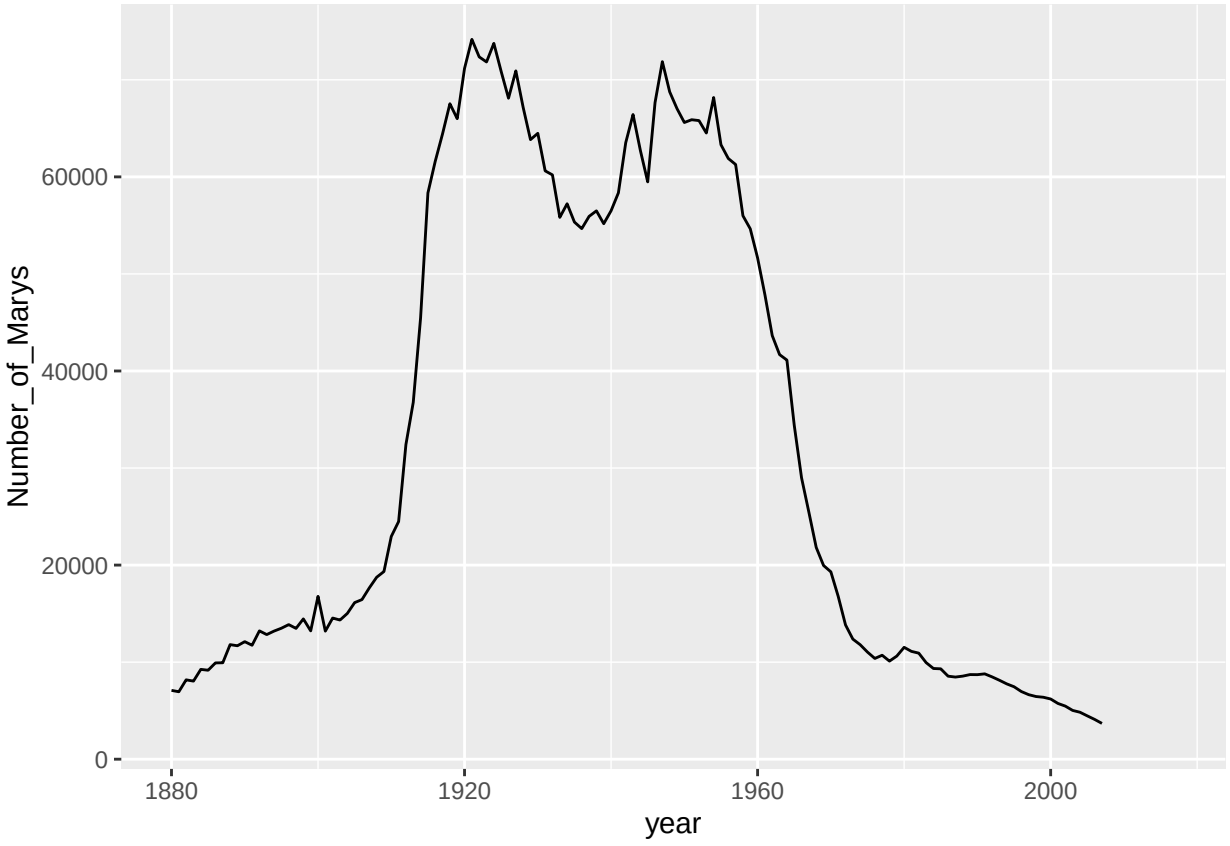



```
    babynames %>%
        group_by(sex, year) %>%
        summarise(name_count = n_distinct(name))%>%
ggplot(mapping = aes(x = year, y = name_count, color = sex))+ geom_line()
```



f. Pick out a name (or names) of interest to you. Plot out its popularity over time.

```
babynames %>% select(-prop)  %>% filter(name == "Mary") %>% pivot_wider(names_from = sex, values_from =
```



5. **verbs** Each of the tasks below can be performed using a single data verb. For each task, identify the appropriate verb (problem taken from MDSR):

a. Add a new column that is the sum of two variables.

mutate

b. Sort the cases in descending order of a particular variable.

arrange

c. Create a new dataframe that includes only those cases that meet a criterion.

filter

d. From a dataframe with three categorical variables A, B, & C, and a quantitative variable X, produce an output that has the same cases but only the variables A and X.

select

e. Find the average of one of the variables.

mean