MATH 118: Notes E

Making plots with ggplot2: barplots

```
library(tidyverse)

#Import the can_lang dataset from last class

#can_lang <- read.csv("can_lang.csv")

#OR
can_lang <- read.csv("https://raw.githubusercontent.com/ttimbers/canlang/master/inst/extdata/can_lang.c
```
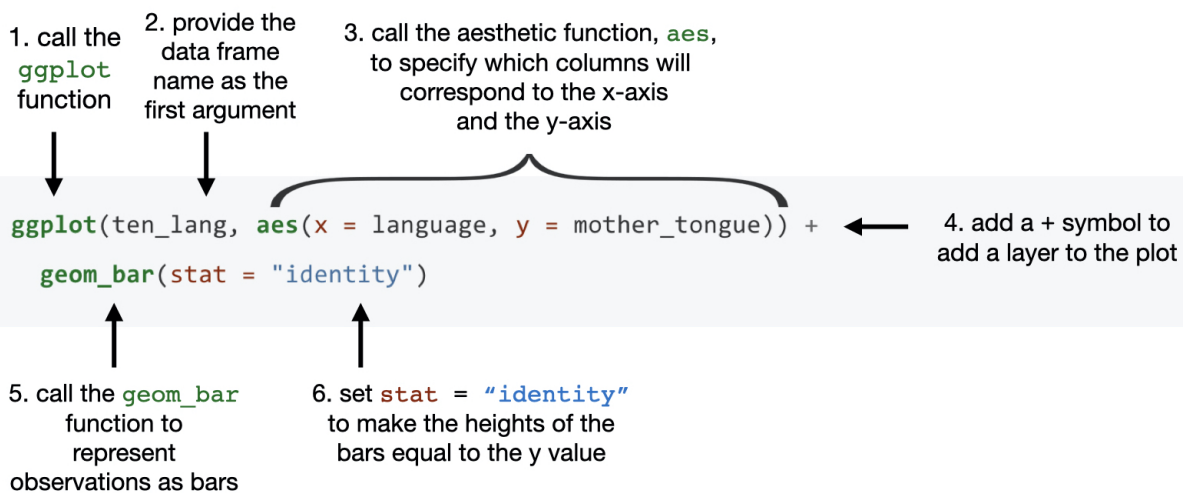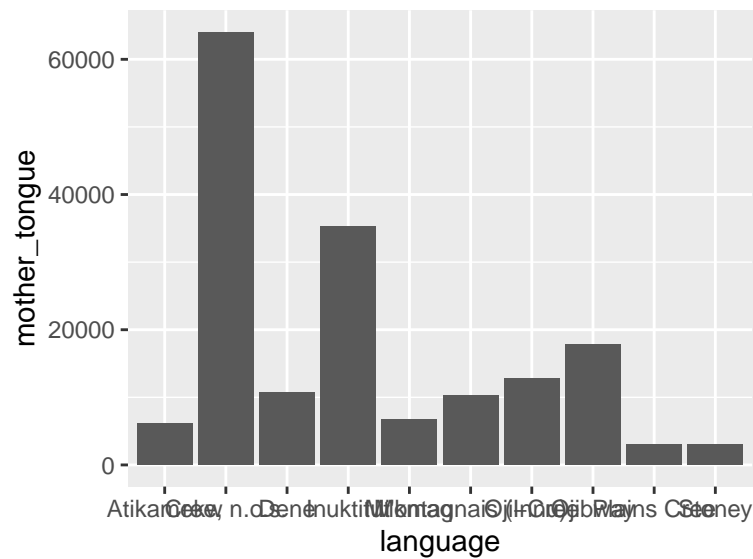
Recall our last example:

```
ten_lang <- can_lang %>%
  filter(category == "Aboriginal languages") %>%
  arrange(by=desc(mother_tongue)) %>%
  select(language, mother_tongue) %>%
  slice(1:10)
```

ggplot: barplots with `geom_bar`

1. call the `ggplot` function

2. provide the data frame name as the first argument

3. call the aesthetic function, `aes`, to specify which columns will correspond to the x-axis and the y-axis

```
ggplot(ten_lang, aes(x = language, y = mother_tongue)) +

  geom_bar(stat = "identity")
```

4. add a + symbol to add a layer to the plot

5. call the `geom_bar` function to represent observations as bars

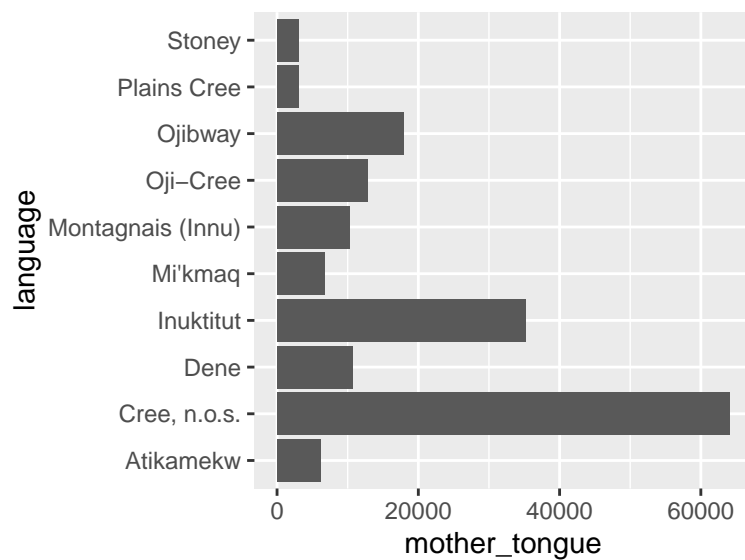6. set `stat = "identity"` to make the heights of the bars equal to the y value

```
ggplot(ten_lang, aes(x = language, y = mother_tongue)) +
  geom_bar(stat = "identity")
```
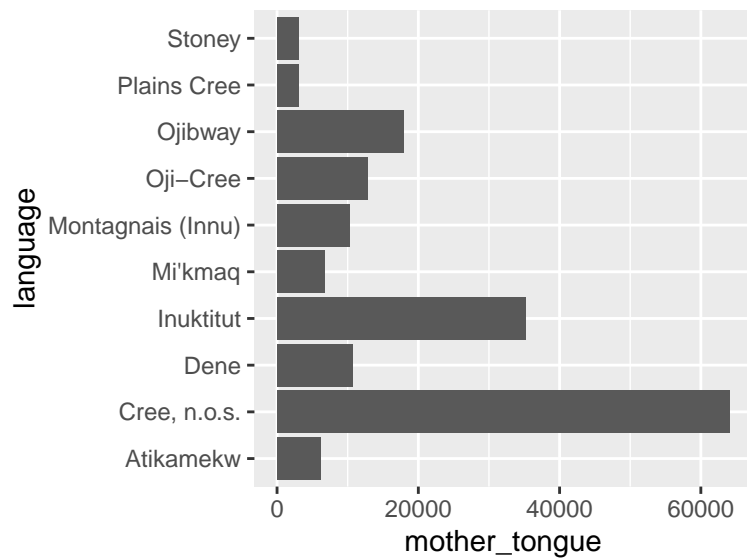
Is there any improvements we could make to this graph?

To better view text

```
ggplot(ten_lang, aes(x = language, y = mother_tongue)) +
  geom_bar(stat = "identity") +
  coord_flip()
```



```
#OR
```

```
ggplot(ten_lang, aes(x = mother_tongue, y = language)) +
  geom_bar(stat = "identity")
```

Labels, Colors, and Themes

```
ggplot(ten_lang, aes(x = mother_tongue, y = reorder(language, mother_tongue))) +
  geom_bar(fill="lightblue", stat = "identity") +
  ylab("Language") +
  xlab("Mother Tongue (Number of Canadian Residents)") +
  ggtitle("Ten Aboriginal Languages Most Often \n Reported by Canadian Residents \n as Their Mother Ton
  theme_minimal()
```
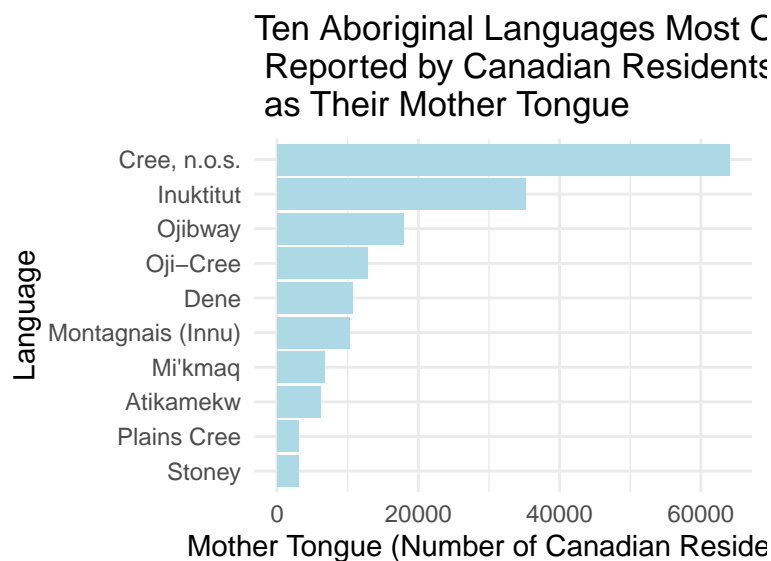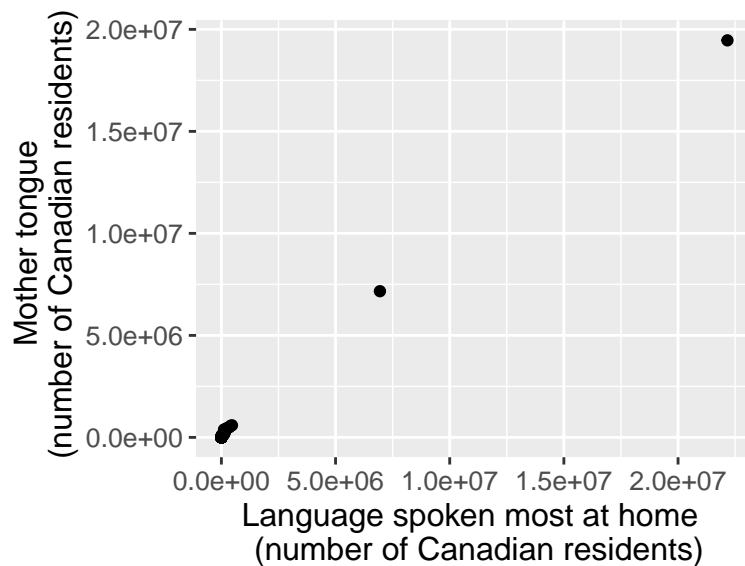

ggplot: scatterplot with `geom_point`

```
ggplot(can_lang, aes(x=most_at_home, y=mother_tongue)) +
  geom_point()
```
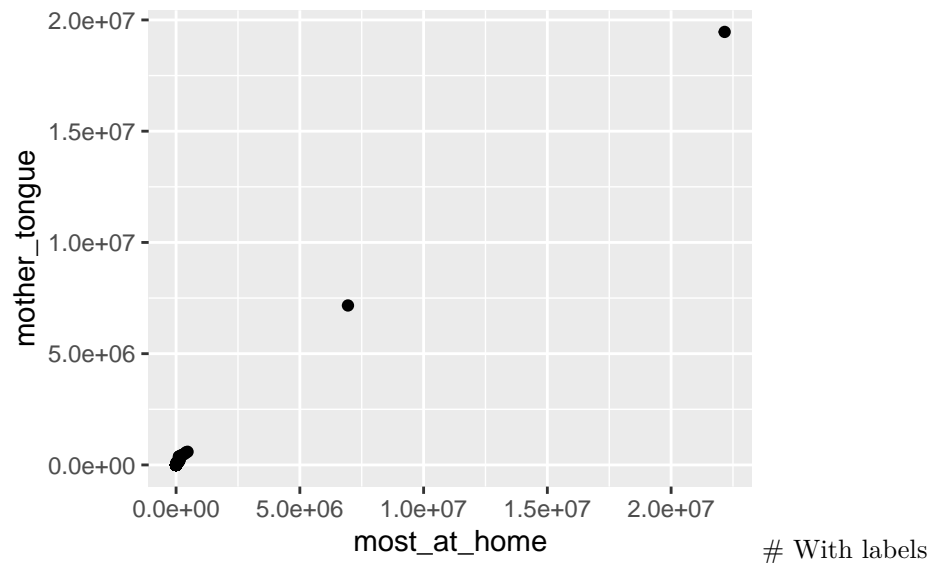
With labels

```
ggplot(can_lang, aes(x = most_at_home, y = mother_tongue)) +
  geom_point() +
  xlab("Language spoken most at home \n (number of Canadian residents)") +
  ylab("Mother tongue \n (number of Canadian residents)") +
  theme(text = element_text(size = 12))
```


Axis transformations
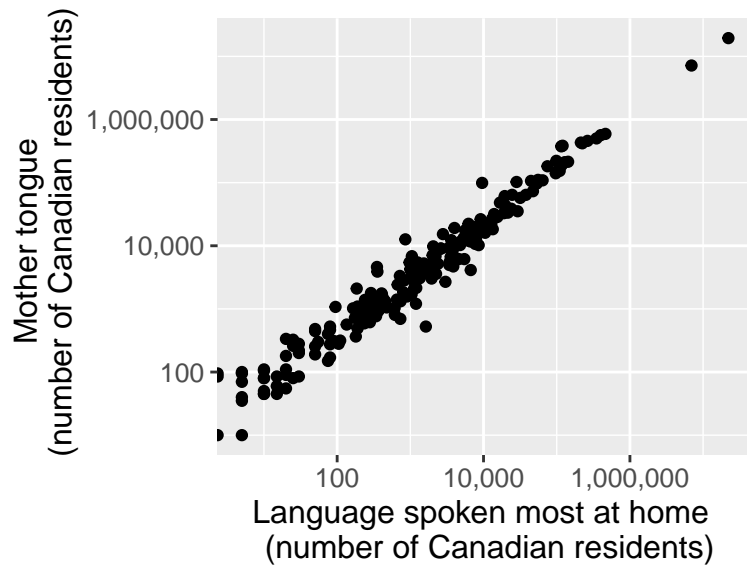
```
library(scales)
```

```
ggplot(can_lang, aes(x = most_at_home, y = mother_tongue)) +
  geom_point() +
  xlab("Language spoken most at home \n (number of Canadian residents)") +
  ylab("Mother tongue \n (number of Canadian residents)") +
  theme(text = element_text(size = 12)) +
  scale_x_log10(labels = label_comma()) +
  scale_y_log10(labels = label_comma())
```

mutate to create new columns

```r
can_lang <- can_lang %>%
  mutate(
    mother_tongue_percent = (mother_tongue / 35151728) * 100,
    most_at_home_percent = (most_at_home / 35151728) * 100
  )
```

Scatterplot with percents

```r
ggplot(can_lang, aes(x = most_at_home_percent, y = mother_tongue_percent)) +
  geom_point() +
  xlab("Language spoken most at home \n (percentage of Canadian residents)") +
  ylab("Mother tongue \n (percentage of Canadian residents)") +
  theme(text = element_text(size = 12)) +
  scale_x_log10(labels = comma) +
  scale_y_log10(labels = comma)
```
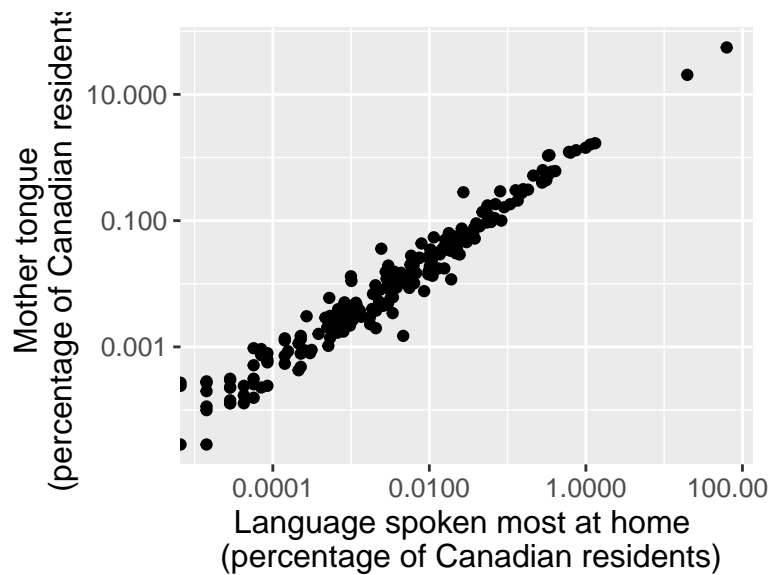
Scatterplot with Percents and Colors

```
ggplot(can_lang, aes(x = most_at_home_percent,
                     y = mother_tongue_percent,
                     color = category, shape=category)) +
  geom_point() +
  xlab("Language spoken most at home \n (percentage of Canadian residents)") +
  ylab("Mother tongue \n (percentage of Canadian residents)") +
  theme(text = element_text(size = 12),
        legend.position = "top",
        legend.direction = "vertical") +
  scale_x_log10(labels = comma) +
  scale_y_log10(labels = comma)
```