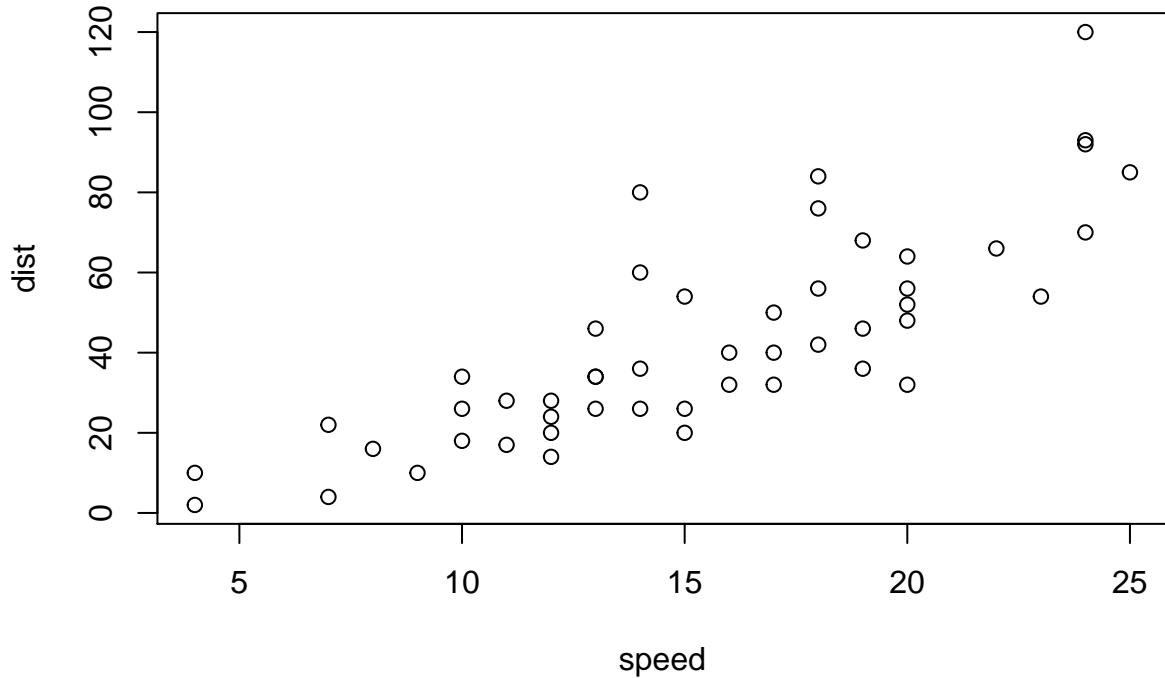


R Notebook

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

```
plot(cars)
```



Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.

```
library(tidyverse)
```

```
## Registered S3 methods overwritten by 'ggplot2':  
##   method      from  
##   [.quosures    rlang  
##   c.quosures    rlang  
##   print.quosures rlang  
  
## -- Attaching packages ----- tidyverse 1.2.1 --  
  
## v ggplot2 3.1.1     v purrr    0.3.3  
## v tibble  2.1.3     v dplyr    0.8.4
```

```

## v tidyverse 0.8.3      v stringr 1.4.0
## v readr    1.3.1      v forcats 0.4.0

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()

```

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Cmd+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.

```
surveys <- read_csv('data_raw/portal_data_joined.csv')
```

```

## Parsed with column specification:
## cols(
##   record_id = col_double(),
##   month = col_double(),
##   day = col_double(),
##   year = col_double(),
##   plot_id = col_double(),
##   species_id = col_character(),
##   sex = col_character(),
##   hindfoot_length = col_double(),
##   weight = col_double(),
##   genus = col_character(),
##   species = col_character(),
##   taxa = col_character(),
##   plot_type = col_character()
## )

```

```

surveys_no_na <- surveys %>%
  filter(!is.na(weight),
         !is.na(hindfoot_length),
         !is.na(sex))
head(surveys_no_na)

```

```

## # A tibble: 6 x 13
##   record_id month   day   year plot_id species_id sex   hindfoot_length weight
##       <dbl>   <dbl> <dbl> <dbl>   <dbl>   <chr>   <chr>       <dbl>   <dbl>
## 1        845     5     6  1978      2    NL      M          32     204
## 2       1164     8     5  1978      2    NL      M          34     199
## 3       1261     9     4  1978      2    NL      M          32     197
## 4       1756     4    29  1979      2    NL      M          33     166
## 5       1818     5    30  1979      2    NL      M          32     184
## 6       1882     7     4  1979      2    NL      M          32     206
## # ... with 4 more variables: genus <chr>, species <chr>, taxa <chr>,
## #   plot_type <chr>

```

```

species_counts <- surveys_no_na %>%
  count(species_id) %>%
  filter(n >= 50)
head(species_counts)

## # A tibble: 6 x 2
##   species_id     n
##   <chr>       <int>
## 1 DM           9727
## 2 DO           2790
## 3 DS           2023
## 4 NL           1045
## 5 OL            905
## 6 OT           2081

surveys_complete <- surveys_no_na %>%
  filter(species_id %in% species_counts$species_id)

```

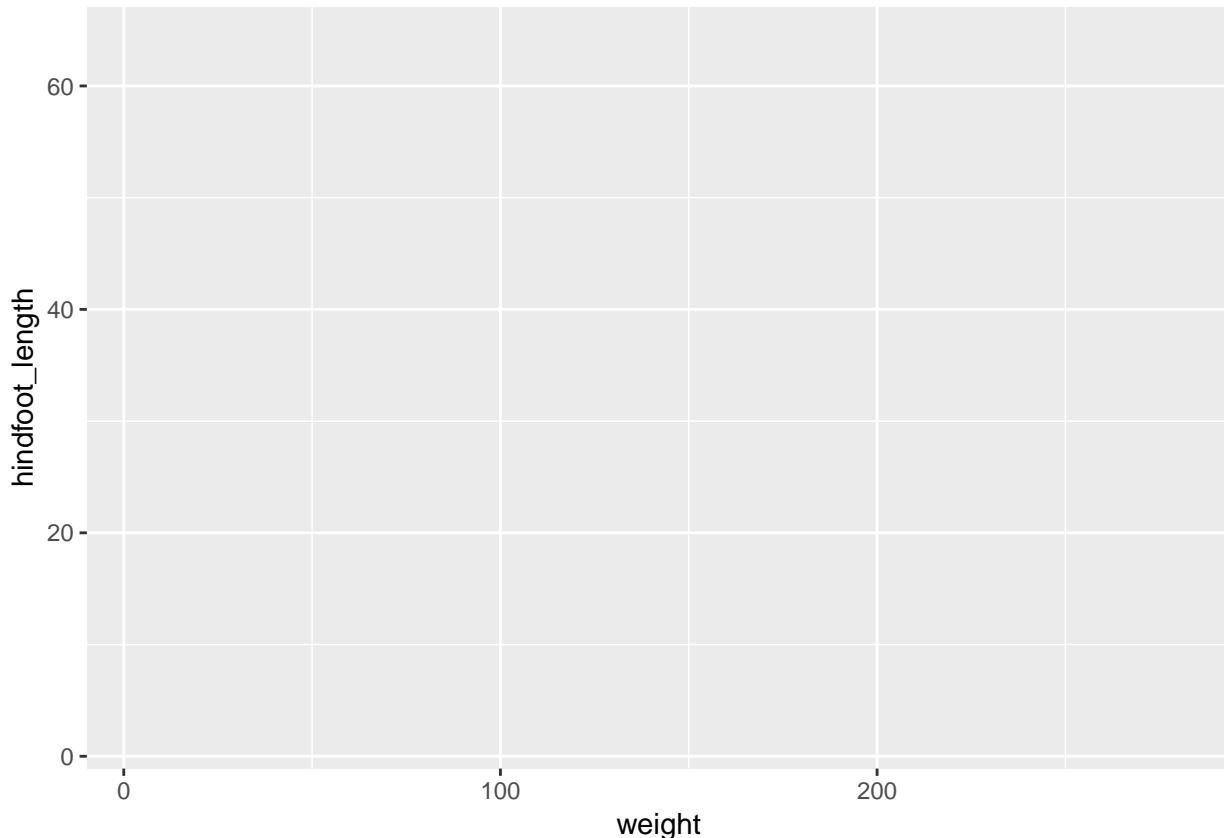
Now, we're ready to make some graphs! (Taken from <https://datacarpentry.org/R-ecology-lesson/04-visualization-ggplot2.html>)

GGPLOT uses the basic template: `ggplot(data = <DATA>, mapping = aes(<MAPPINGS>)) + <GEOM_FUNCTION>()`

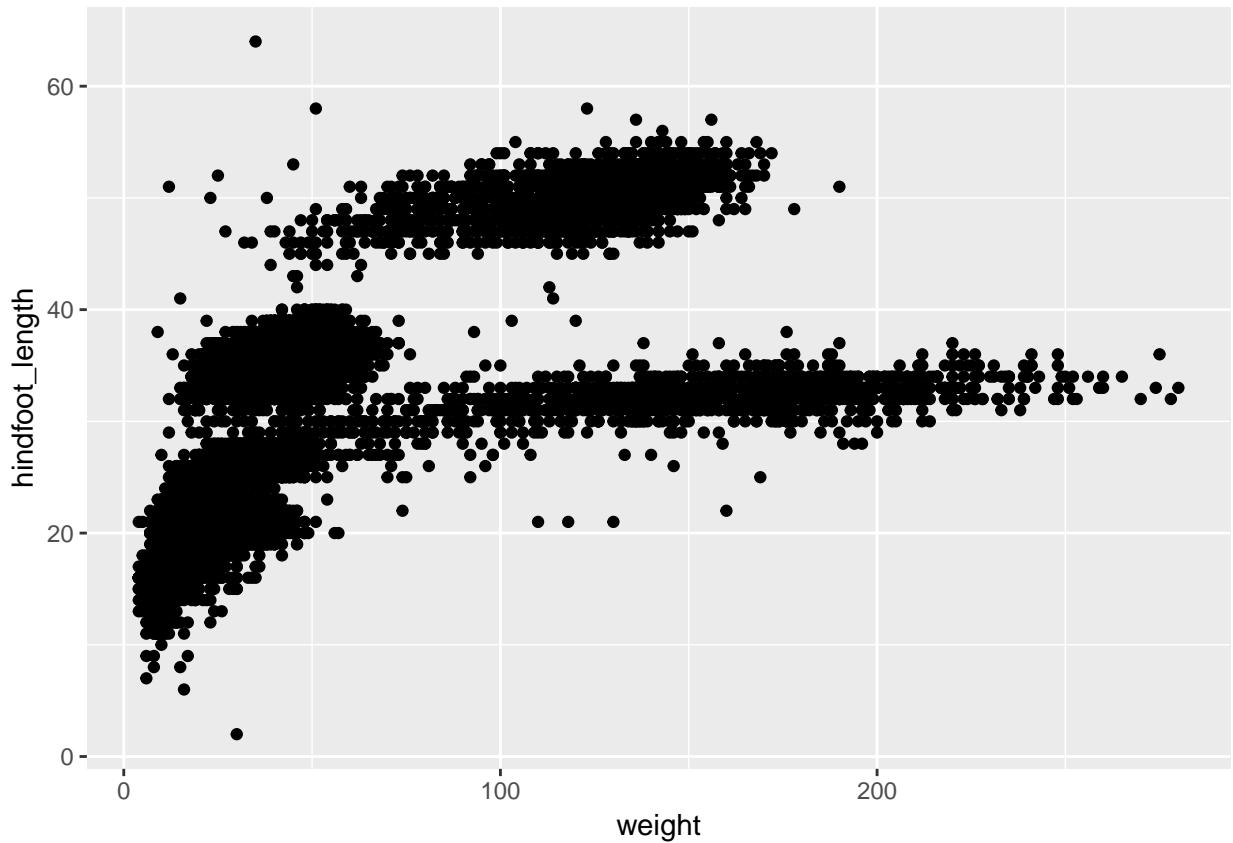
```

ggplot(data = surveys_complete,
       mapping = aes(x = weight,
                      y = hindfoot_length))

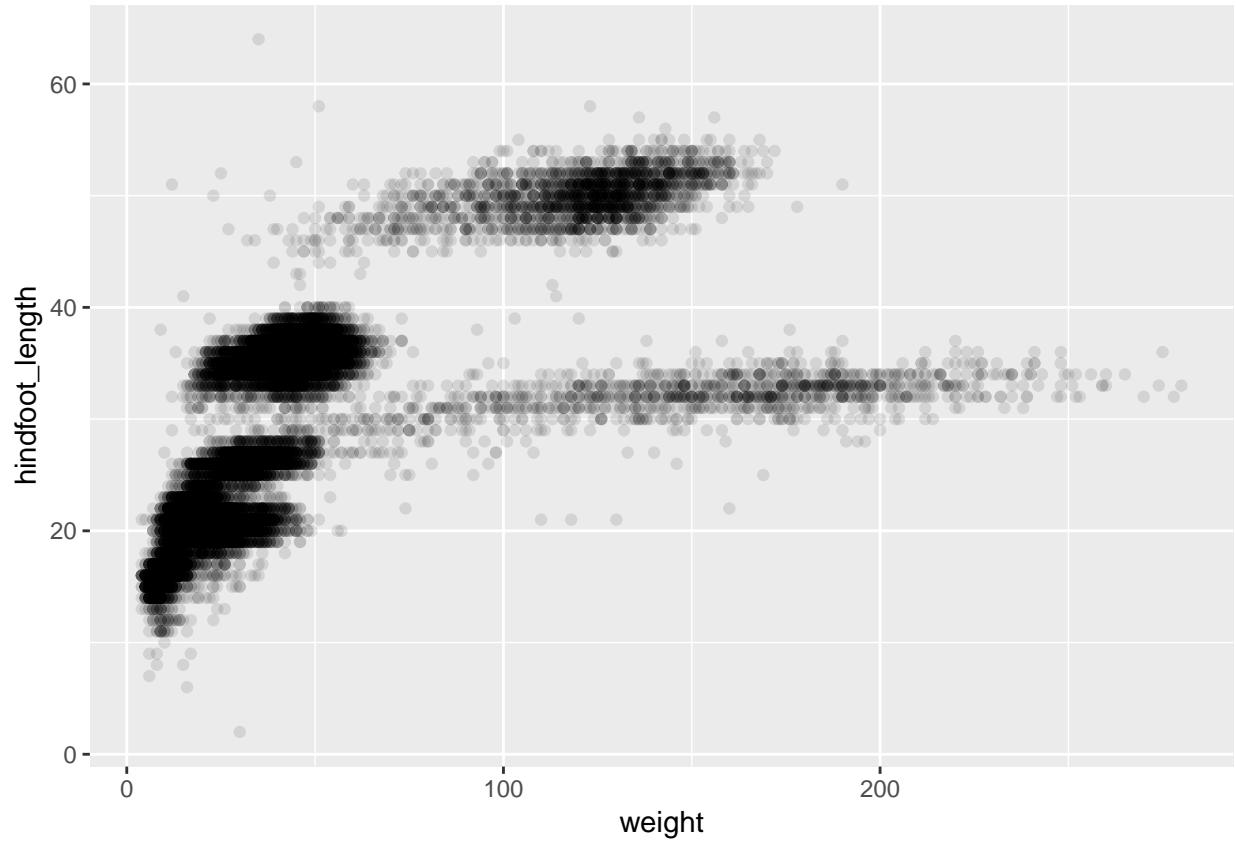
```



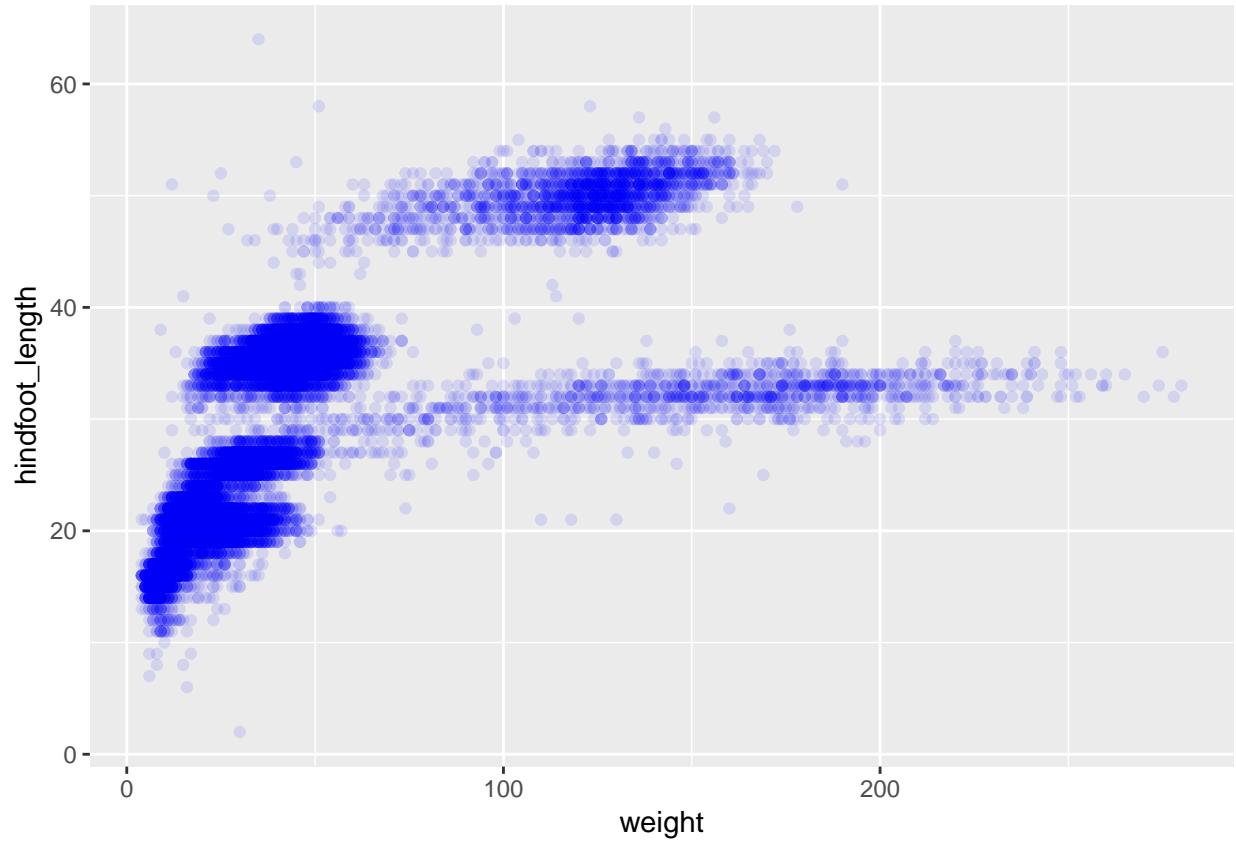
```
ggplot(data = surveys_complete,  
       mapping = aes(x = weight,  
                      y = hindfoot_length)) +  
  geom_point()
```



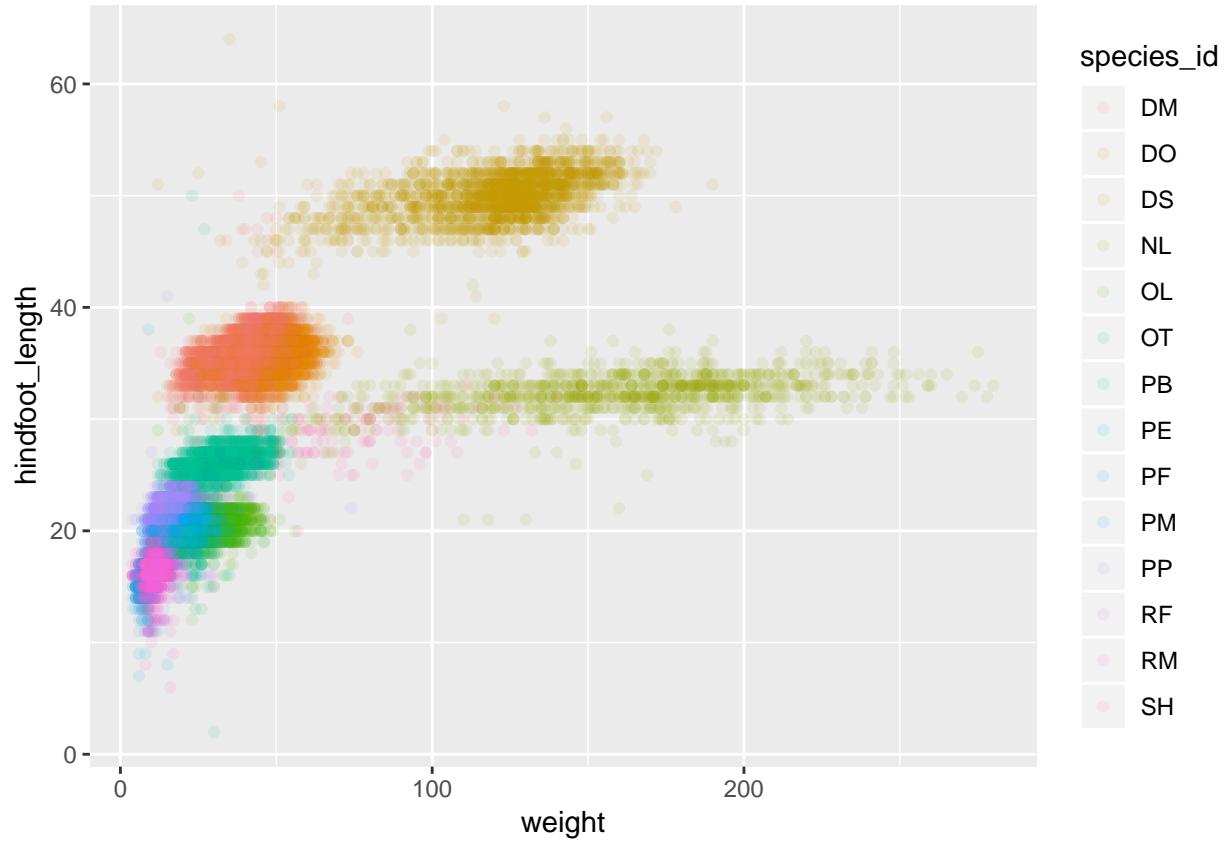
```
ggplot(data = surveys_complete,  
       mapping = aes(x = weight,  
                      y = hindfoot_length)) +  
  geom_point(alpha = 0.1)
```



```
ggplot(data = surveys_complete,  
       mapping = aes(x = weight,  
                      y = hindfoot_length)) +  
  geom_point(alpha = 0.1, colour = "blue")
```

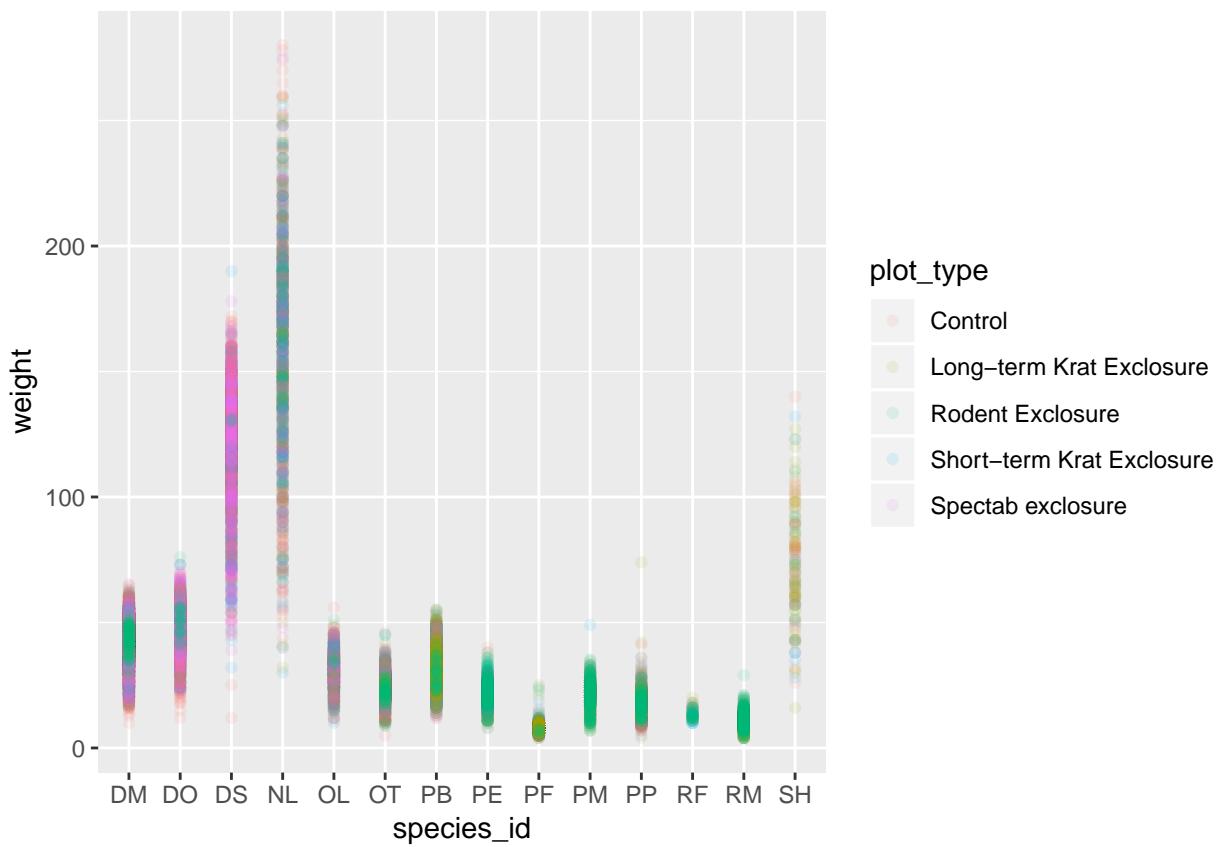


```
ggplot(data = surveys_complete,
       mapping = aes(x = weight,
                      y = hindfoot_length,
                      color = species_id)) +
  geom_point(alpha = 0.1)
```

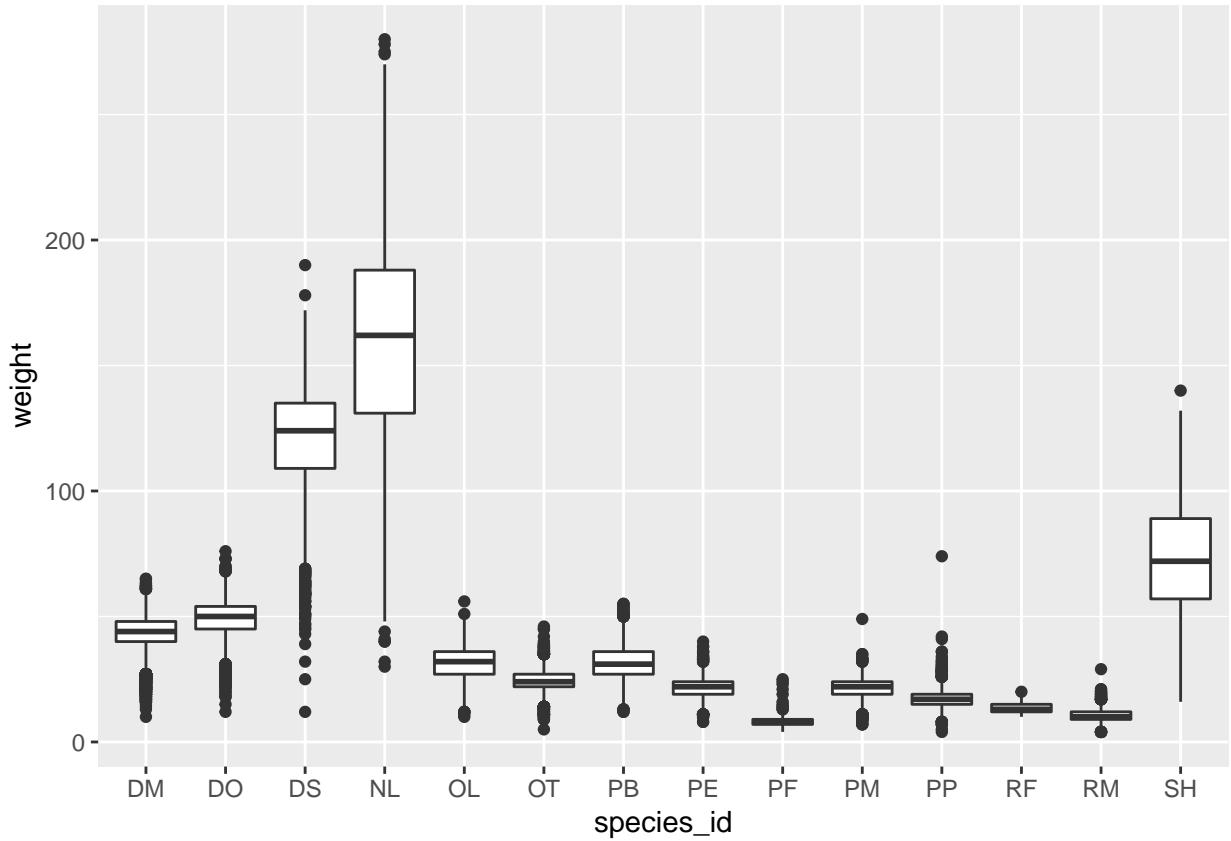


Use what you just learned to create a scatter plot of weight over species_id with the plot types showing in different colors.

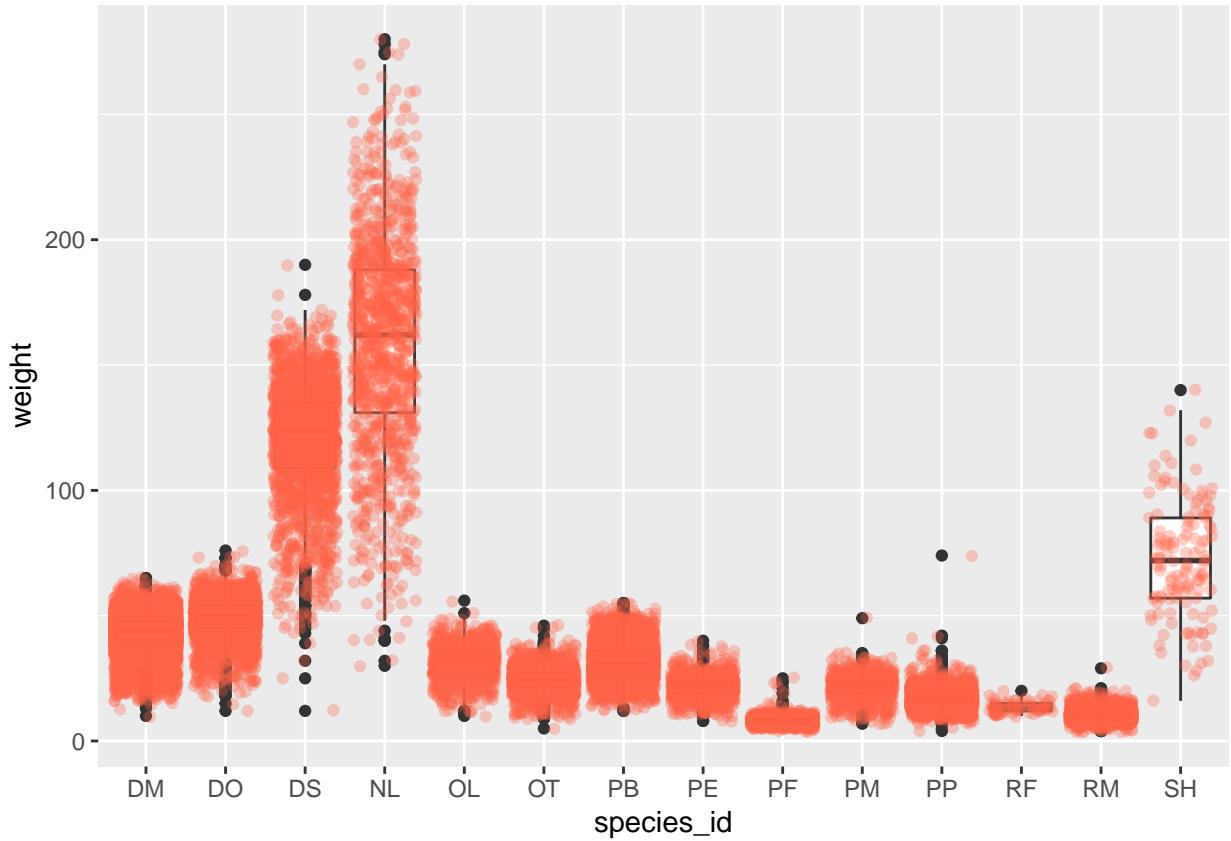
```
ggplot(data = surveys_complete,
       mapping = aes(x = species_id,
                      y = weight,
                      color = plot_type)) +
  geom_point(alpha = 0.1)
```



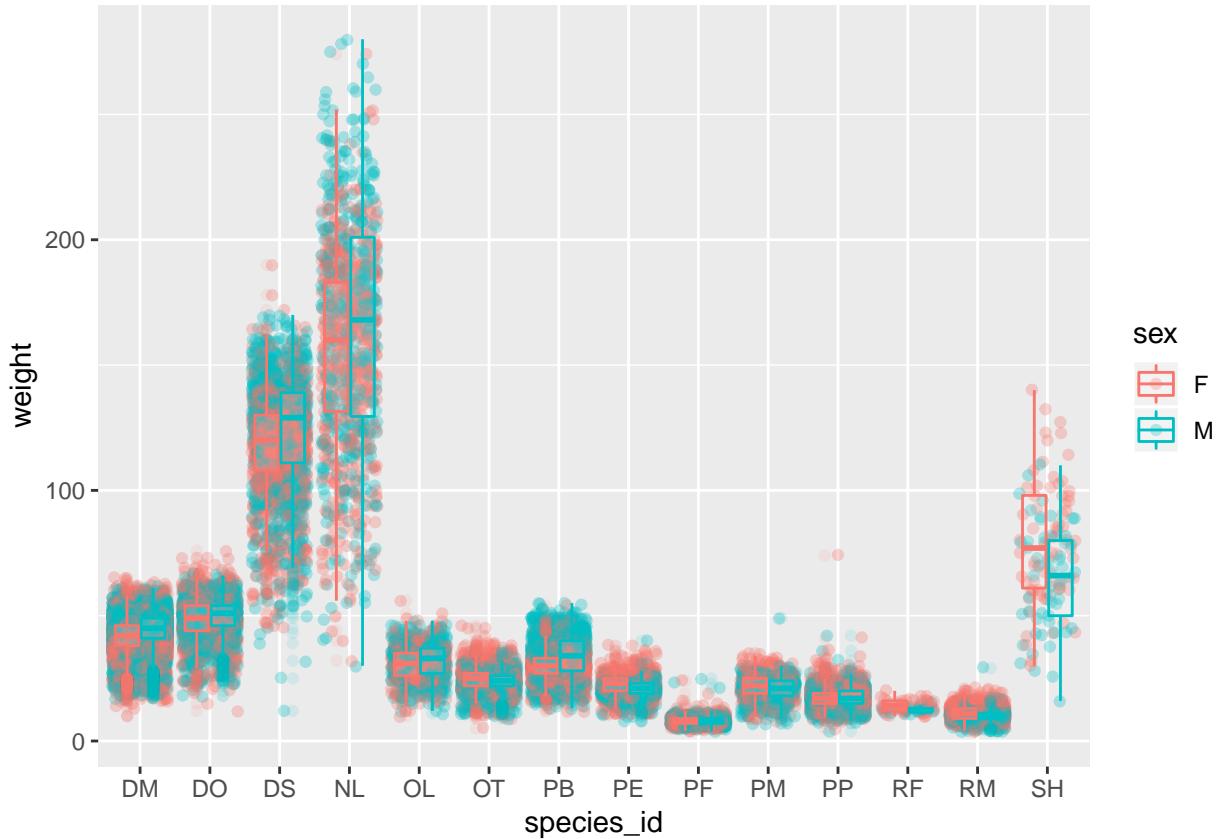
```
ggplot(data = surveys_complete,
       mapping = aes(x = species_id,
                      y = weight)) +
  geom_boxplot()
```



```
ggplot(data = surveys_complete,
       mapping = aes(x = species_id,
                      y = weight)) +
  geom_boxplot() +
  geom_jitter(alpha=0.3, color = "tomato")
```



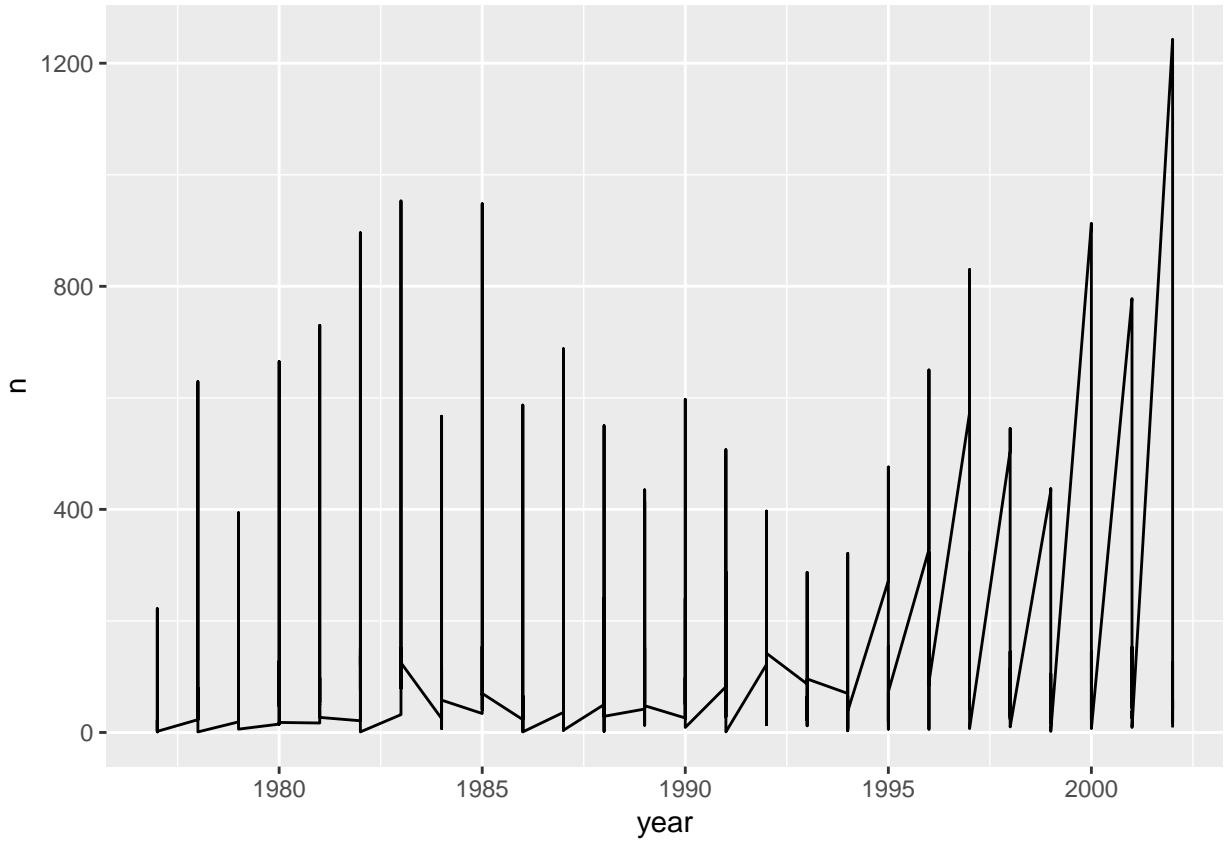
```
ggplot(data = surveys_complete,
       mapping = aes(x = species_id,
                      y = weight,
                      color = sex)) +
  geom_jitter(alpha=0.3) + geom_boxplot(alpha=0.1)
```



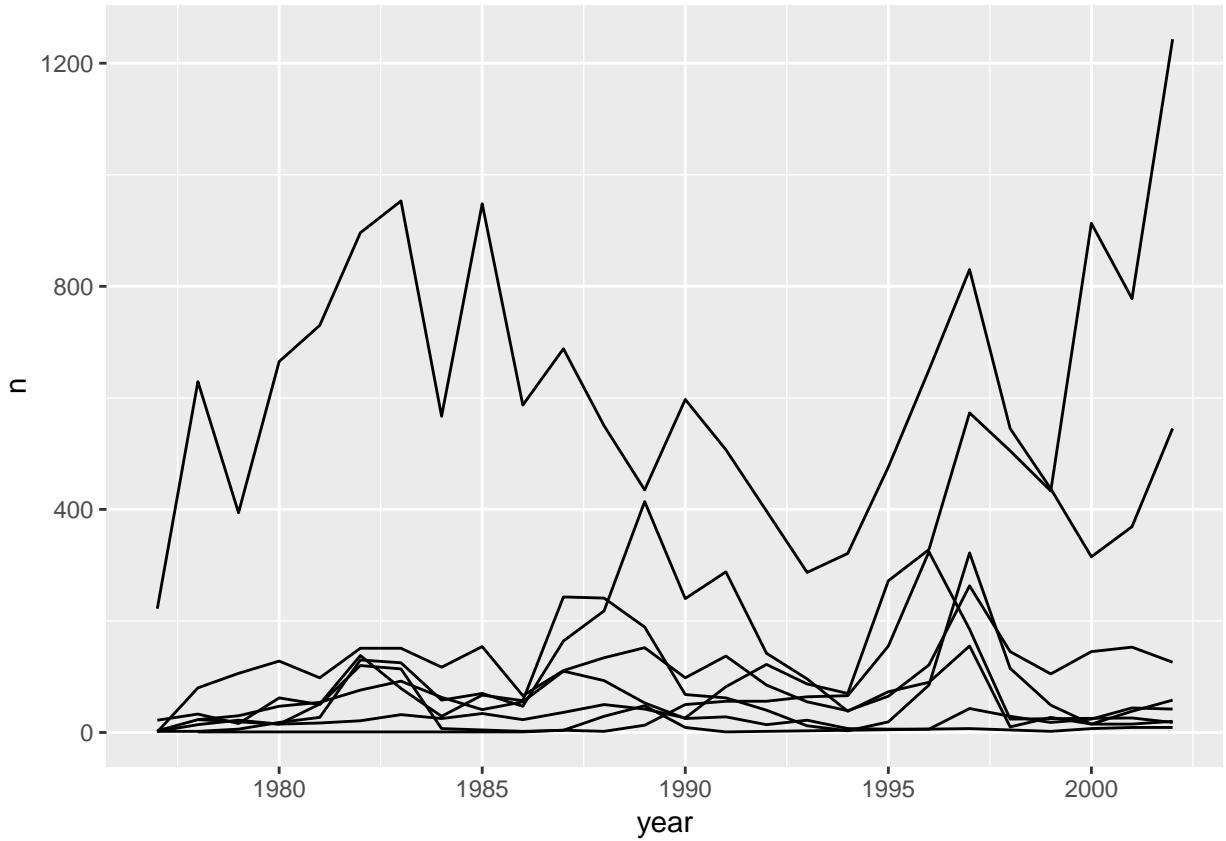
```
yearly_counts <- surveys_complete %>%
  count(year, genus)
head(yearly_counts)
```

```
## # A tibble: 6 x 3
##   year   genus     n
##   <dbl> <chr>    <int>
## 1 1977 Chaetodipus      3
## 2 1977 Dipodomys     222
## 3 1977 Onychomys      1
## 4 1977 Perognathus    22
## 5 1977 Peromyscus     2
## 6 1977 Reithrodontomys 2
```

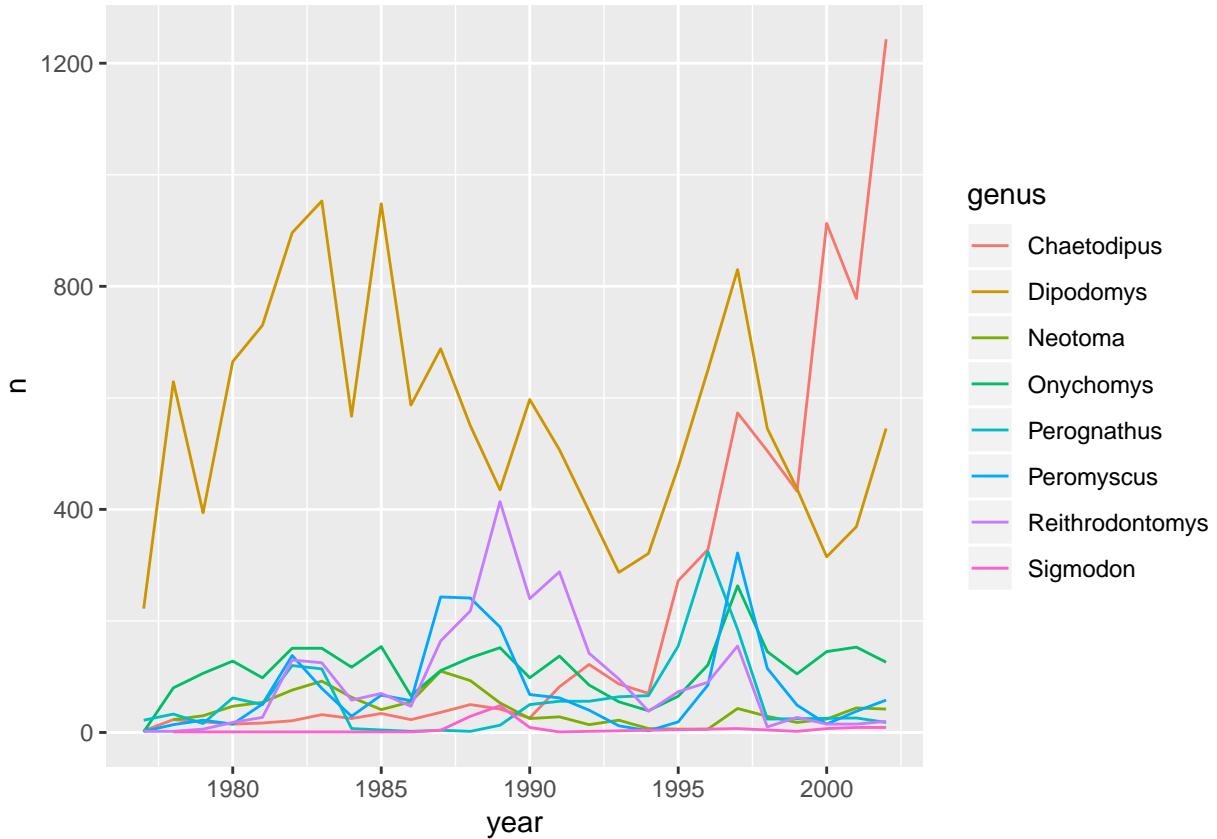
```
ggplot(data = yearly_counts,
       mapping = aes(x = year,
                      y = n)) +
  geom_line()
```



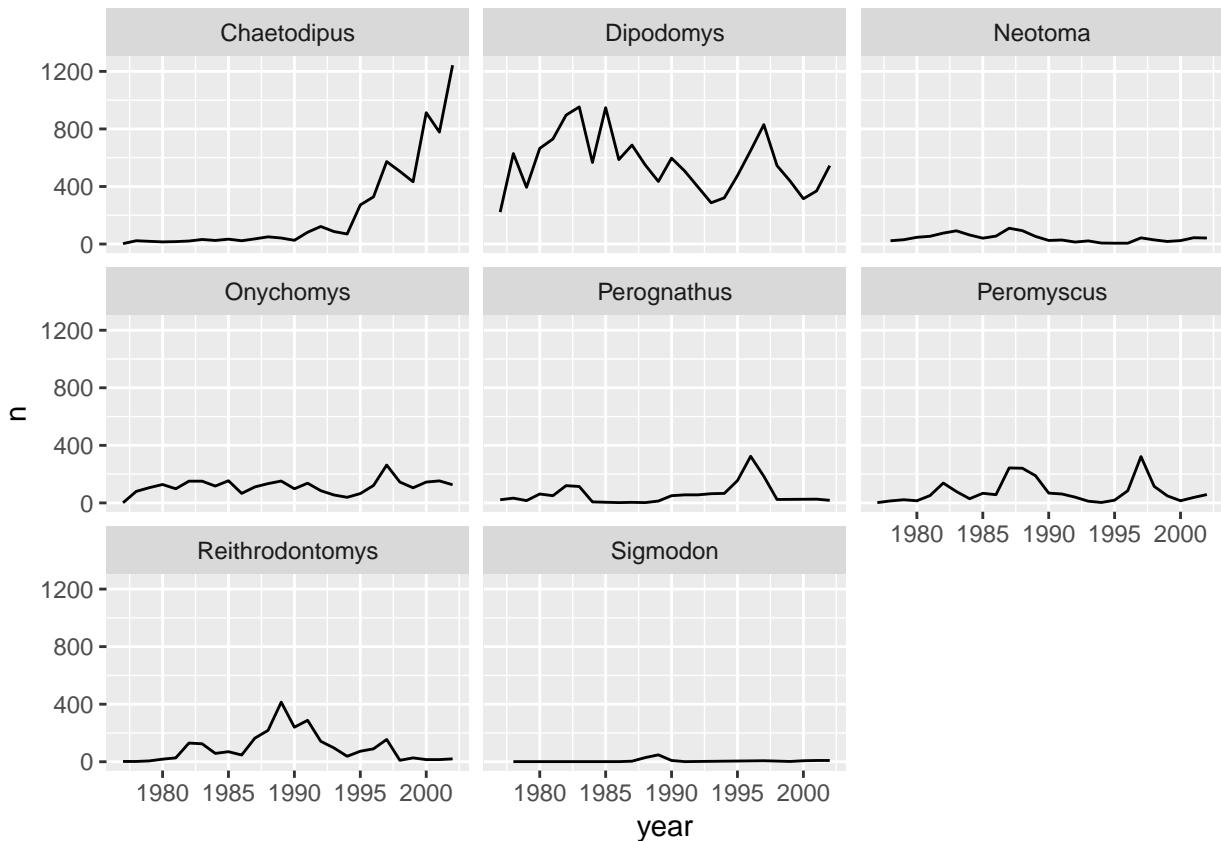
```
ggplot(data = yearly_counts,
       mapping = aes(x = year,
                      y = n,
                      group = genus)) +
  geom_line()
```



```
ggplot(data = yearly_counts,
       mapping = aes(x = year,
                      y = n,
                      color = genus)) +
  geom_line()
```



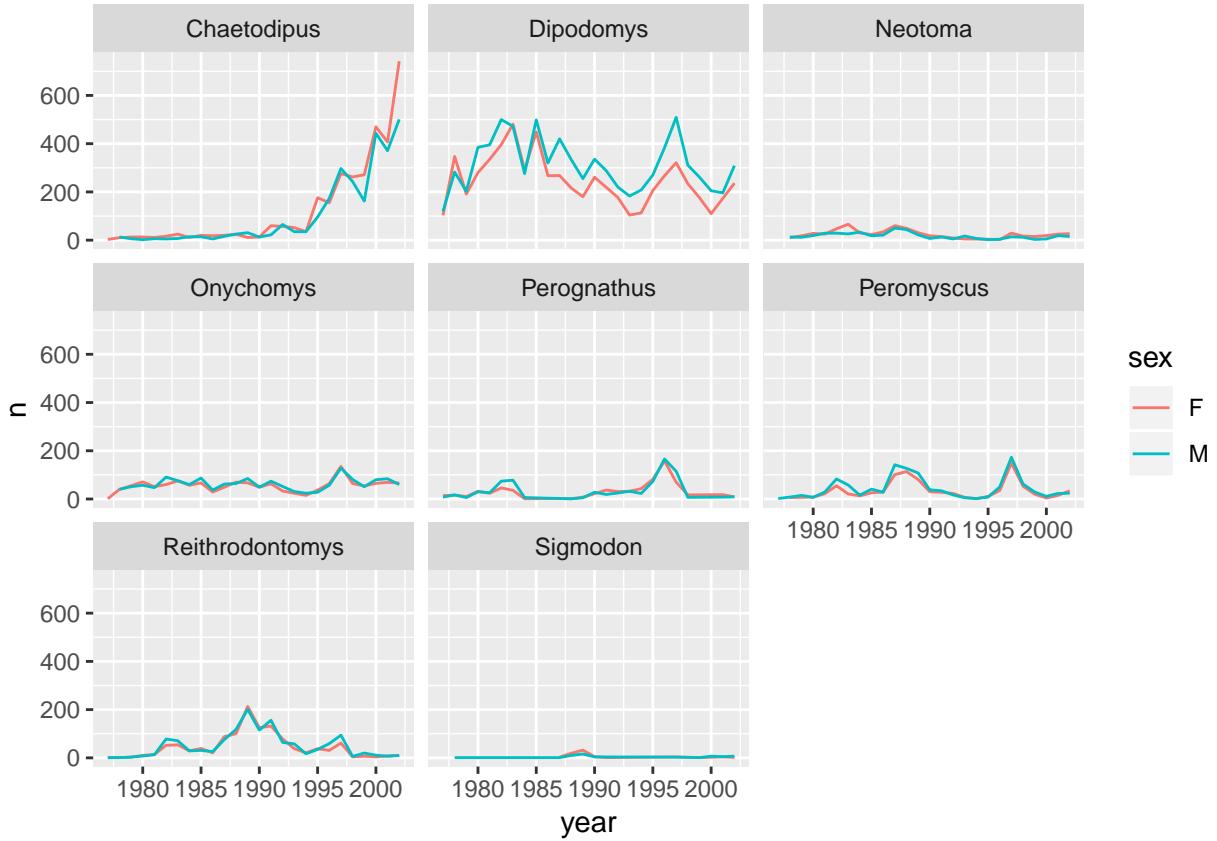
```
ggplot(data = yearly_counts,
       mapping = aes(x = year,
                      y = n)) +
  geom_line() +
  facet_wrap(facets = vars(genus))
```



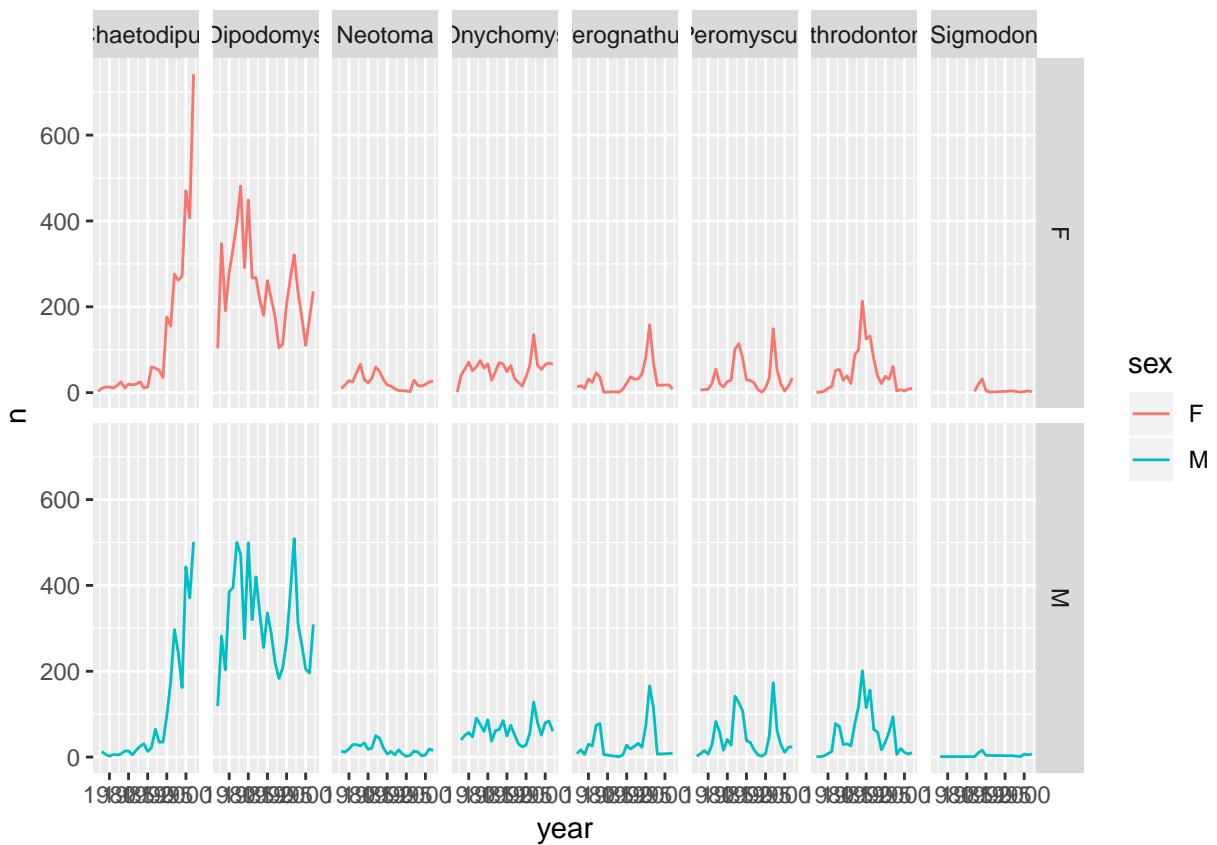
```
yearly_sex_counts <- surveys_complete %>%
  count(year, genus, sex)
head(yearly_sex_counts)
```

```
## # A tibble: 6 x 4
##   year   genus   sex     n
##   <dbl> <chr>    <chr> <int>
## 1 1977 Chaetodipus F        3
## 2 1977 Dipodomys  F       103
## 3 1977 Dipodomys  M       119
## 4 1977 Onychomys  F        1
## 5 1977 Perognathus F       14
## 6 1977 Perognathus M        8
```

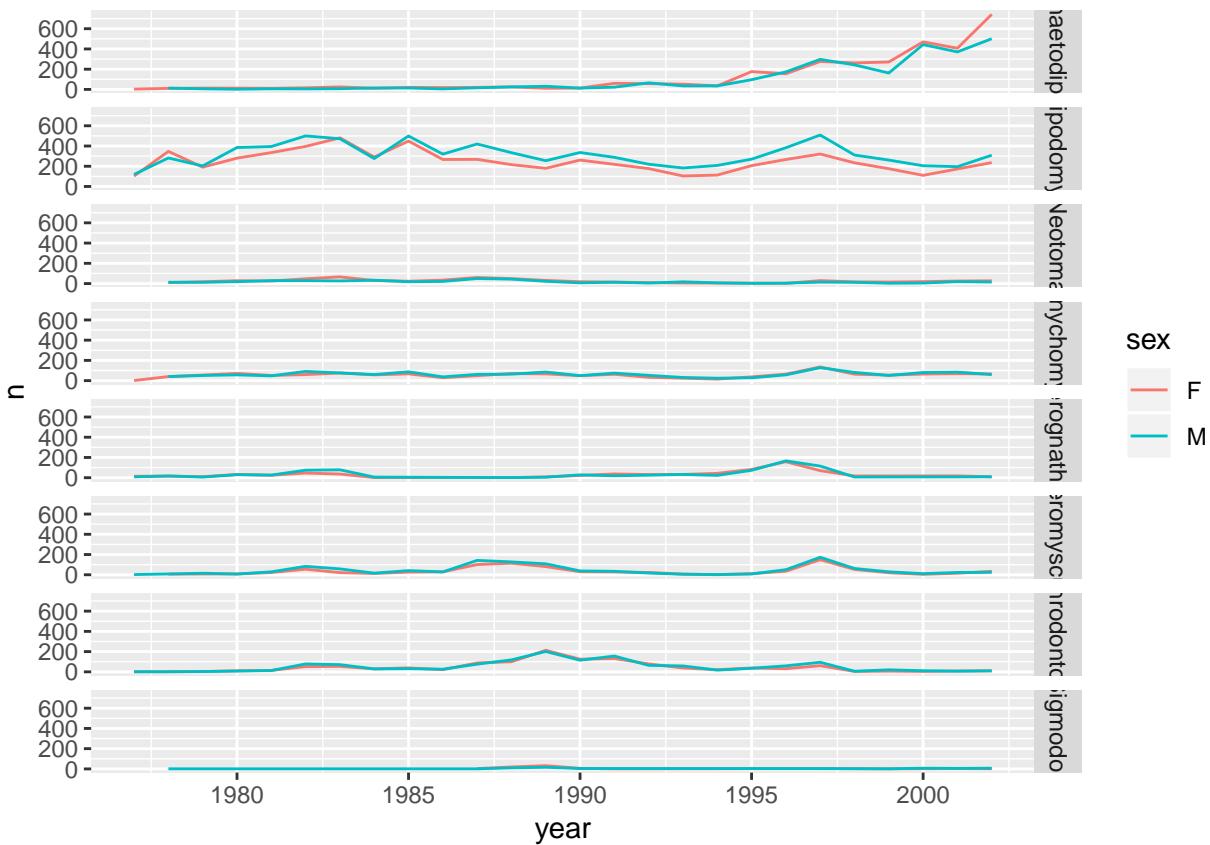
```
ggplot(data = yearly_sex_counts,
       mapping = aes(x = year,
                     y = n,
                     color = sex)) +
  geom_line() +
  facet_wrap(facets = vars(genus))
```



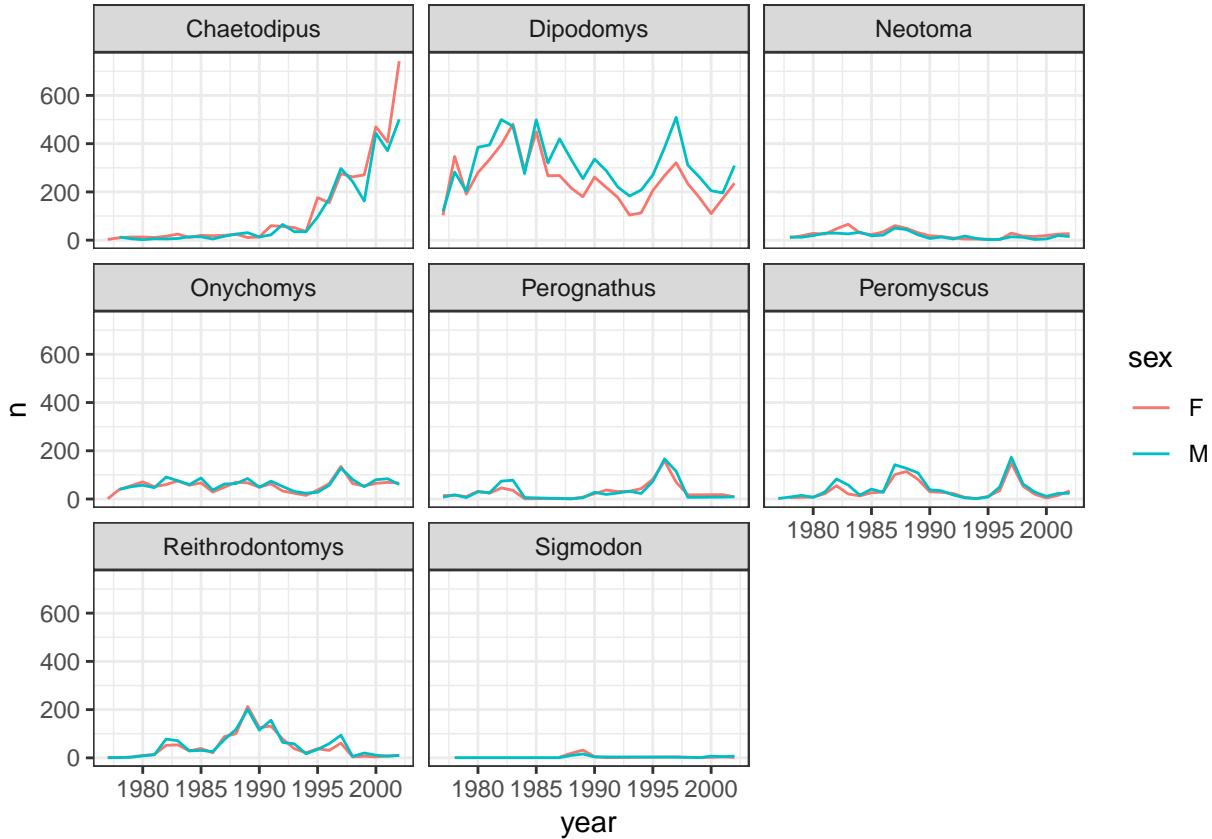
```
ggplot(data = yearly_sex_counts,
       mapping = aes(x = year,
                      y = n,
                      color = sex)) +
  geom_line() +
  facet_grid(rows = vars(sex),
             cols = vars(genus))
```



```
ggplot(data = yearly_sex_counts,
       mapping = aes(x = year,
                      y = n,
                      color = sex)) +
  geom_line() +
  facet_grid(rows = vars(genus))
```



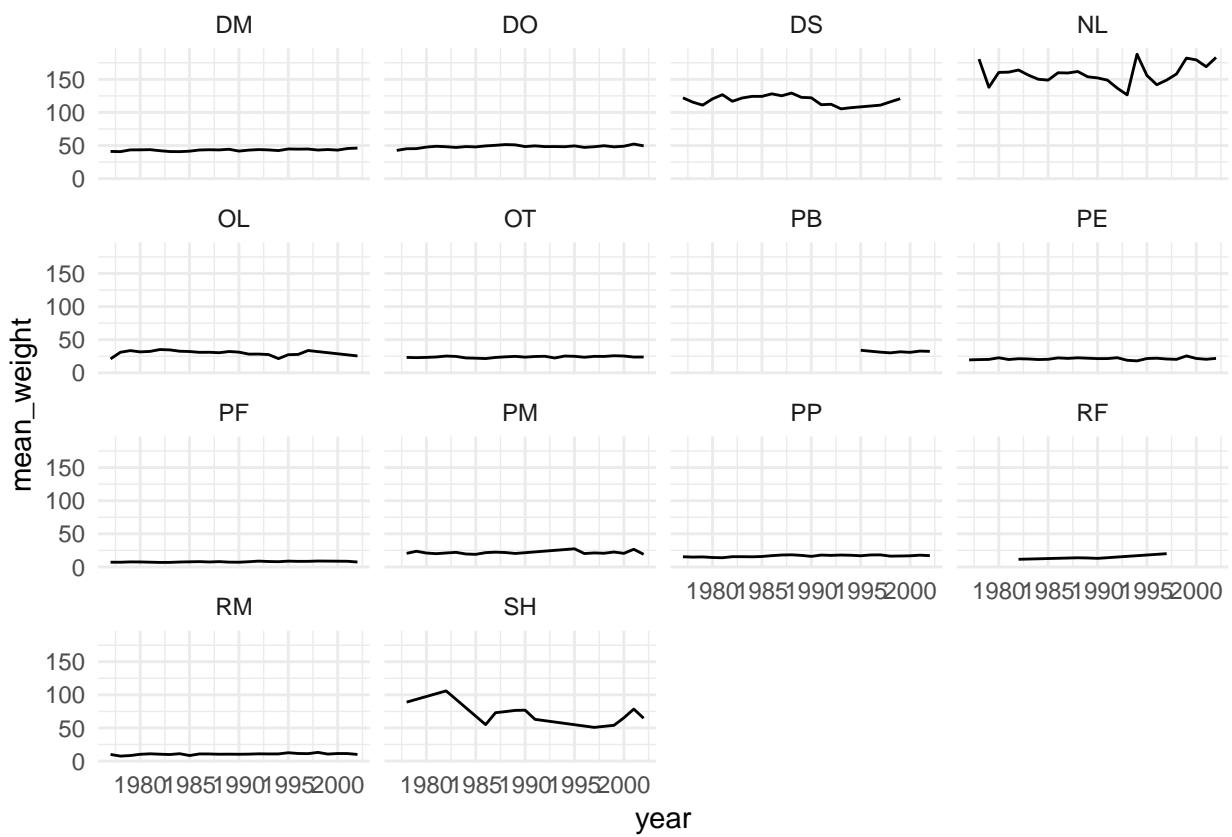
```
ggplot(data = yearly_sex_counts,
       mapping = aes(x = year,
                      y = n,
                      color = sex)) +
  geom_line() +
  facet_wrap(facets = vars(genus)) +
  theme_bw()
```



```
yearly_weights <- surveys_complete %>%
  group_by(year, species_id) %>%
  summarize(mean_weight = mean(weight))
head(yearly_weights)
```

```
## # A tibble: 6 x 3
## # Groups:   year [1]
##   year species_id mean_weight
##   <dbl> <chr>        <dbl>
## 1 1977 DM          41.1
## 2 1977 DO          42.7
## 3 1977 DS         122.
## 4 1977 OL           21
## 5 1977 PE          19.5
## 6 1977 PF          7.14
```

```
ggplot(data = yearly_weights,
       mapping = aes(x = year,
                      y = mean_weight)) +
  geom_line() +
  facet_wrap(facets = vars(species_id)) +
  theme_minimal()
```



```

weight_graph <- ggplot(data = yearly_weights,
  mapping = aes(x = year,
    y = mean_weight)) +
  geom_line() +
  facet_wrap(facets = vars(species_id)) +
  theme_minimal()
ggsave('weight_graph.png', plot = weight_graph, width = 10, dpi=500)

```

```
## Saving 10 x 4.5 in image
```