



# Data Visualization using R

**C. Tobin Magle, PhD**

02-15-2018

10:00-11:30 a.m.  
Morgan Library

Computer Classroom 175

Based on <http://www.datacarpentry.org/R-ecology-lesson/>





# Outline

- **Why ggplot?**
- **How to use it**
  - **Basic elements** (data, aesthetics, geoms)
  - **Modifications** (transparency, color, grouping)
  - **Themes** (modifying default, using premade, saving your own)
- **Exporting plots** (ggsave)



# Why ggplot2?

- **Reproducibility**
- Part of the **tidyverse**
- **Pretty** by default



<http://varianceexplained.org/r/why-i-use-ggplot2/>



# Data set: survey of small animals

- Stored in a data frame
- **Rows:** observations of individual animals
- **Columns:** Variables that describe the animals
  - Species, sex, date, location, etc





# Setup

- Install R and R studio

[http://www.datacarpentry.org/R-ecology-lesson/index.html#setup\\_instructions](http://www.datacarpentry.org/R-ecology-lesson/index.html#setup_instructions)

- Download the quickstart files: <http://tinyurl.com/kp6bxt4>

- See the Basic Analysis with R lesson if you're unfamiliar with R or R studio

<http://libguides.colostate.edu/data-and-donuts/r-analysis>



# Load data into R

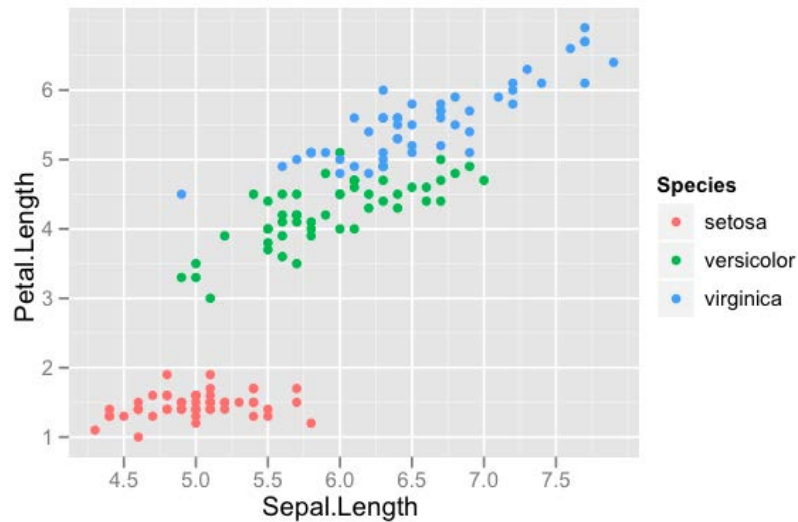
- Import data using **read\_csv** function
- **Arguments:** a csv file
- **Output:** a data frame

**Example:** `surveys_complete <- read_csv('data/surveys_complete.csv')`



# Graphics with ggplot2

- **data**: tibble
- **aesthetics**: looks
- **geoms**: type of plot
  - Ex: points, lines, bars





# ggplot2 functions

- **ggplot()**: initializes ggplot object
- **aes()**: draws axes based on arguments
- **geom\_XXX()**: draws points/lines etc.
- **+** operator: adds components to plot
  - Modular structure

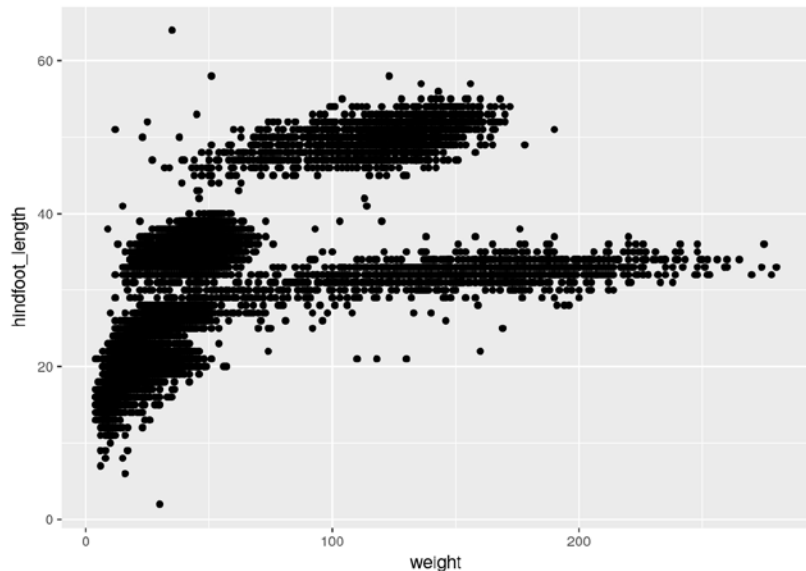


# Simplest ggplot

Need data, aesthetics and a geom to create a plot.

## Example:

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



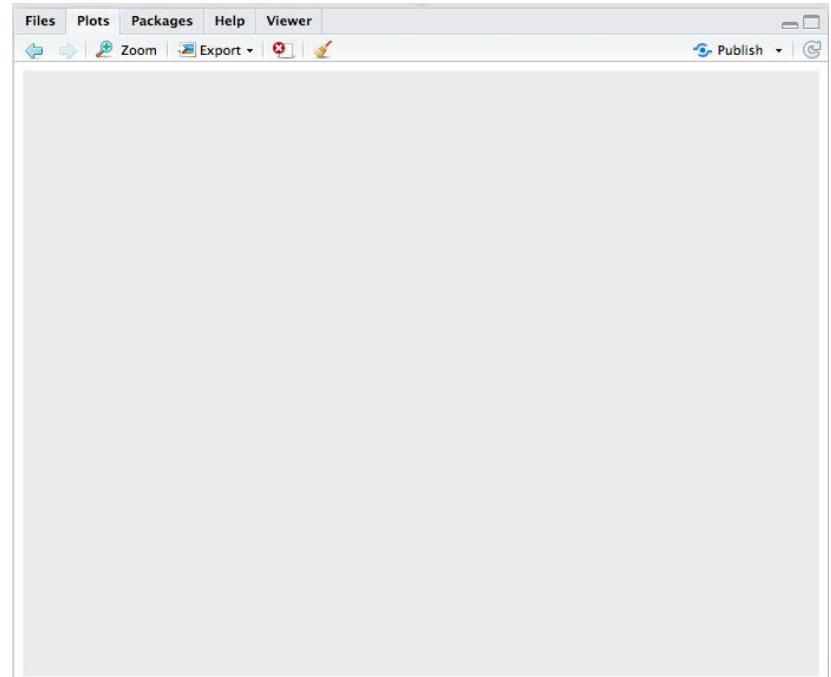


# ggplot()

```
ggplot(data = surveys_complete)
```

**Argument:** data frame

**Output:** blank plot area



# ggplot() + aes()

```
ggplot(data = surveys_complete,
```

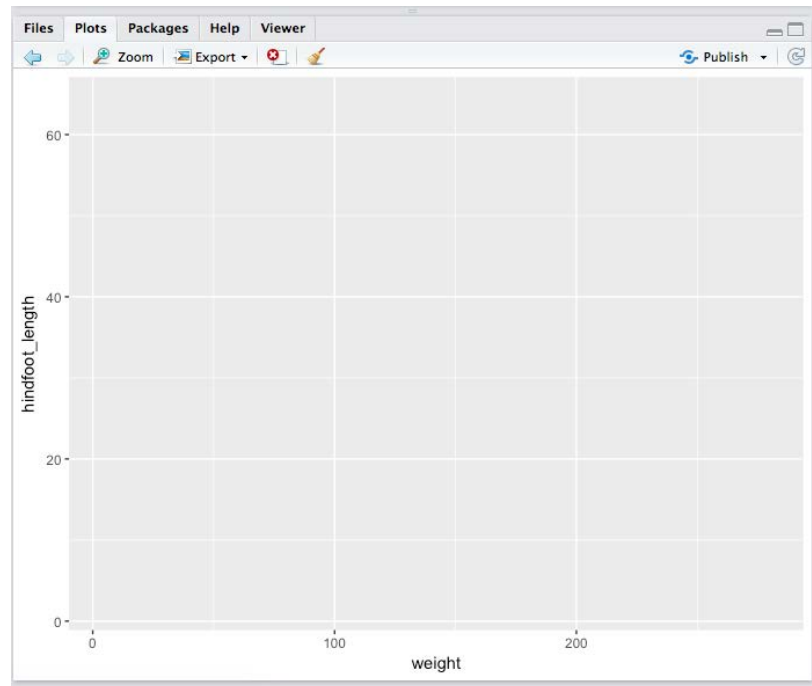
```
  aes(x = weight,  
      y = hindfoot_length))
```

## ggplot arguments:

data frame + aes()

## aes arguments:

- x = x axis variable
- y = y axis variable
- **Output:** draws axes

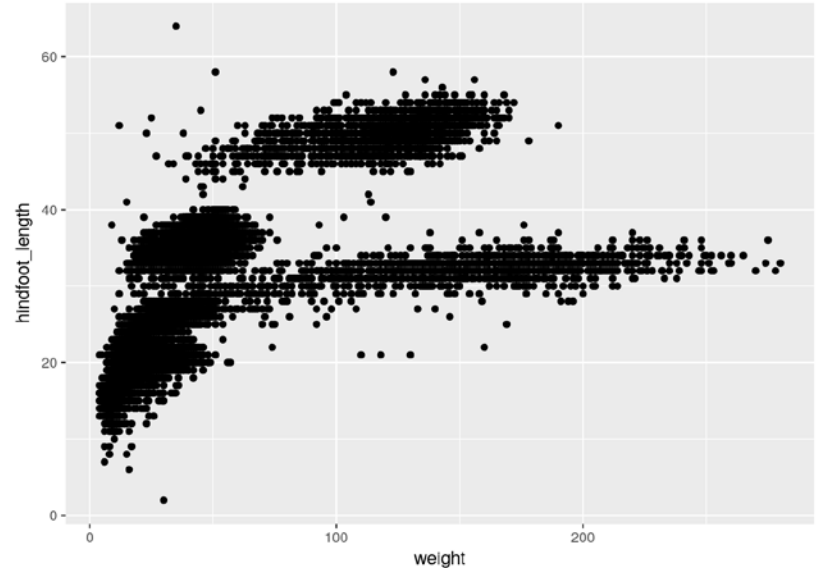


# ggplot + aes + geom\_point

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

**+ operator:** adds point to the specified plot area

**Output:** scatterplot of weigh vs. hindfood length



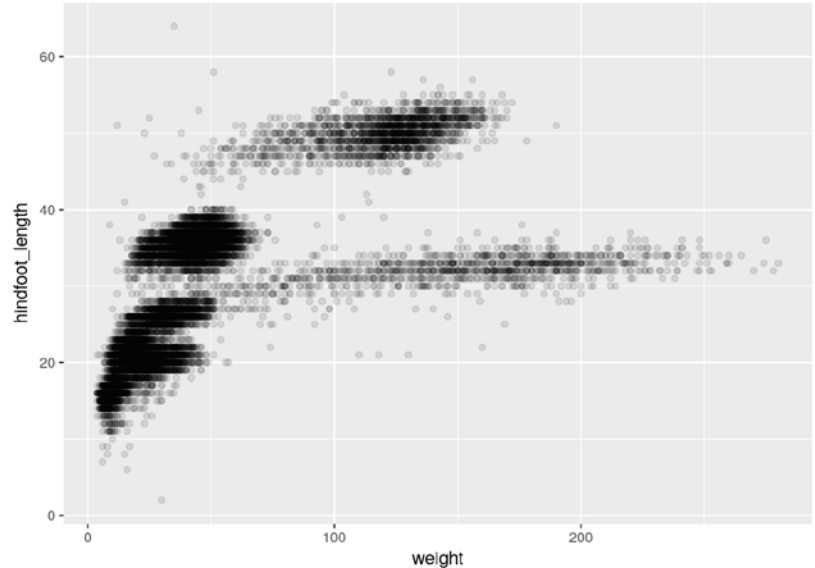


# Add transparency

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

**Argument:** alpha = 0.1

- 1/10 opacity
- Range: 0-1

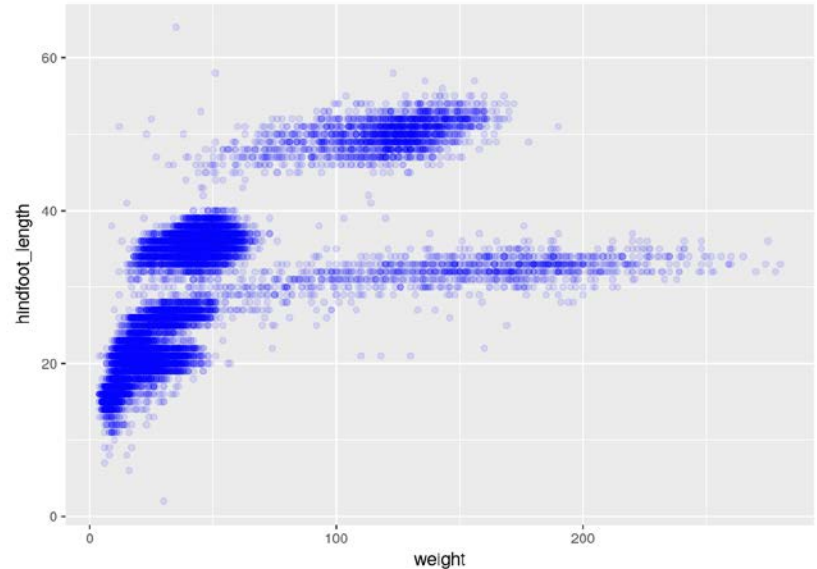


# Add color

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

**Argument:** color; makes all points blue

Ref chart: <http://sape.inf.usi.ch/quick-reference/ggplot2/colour>



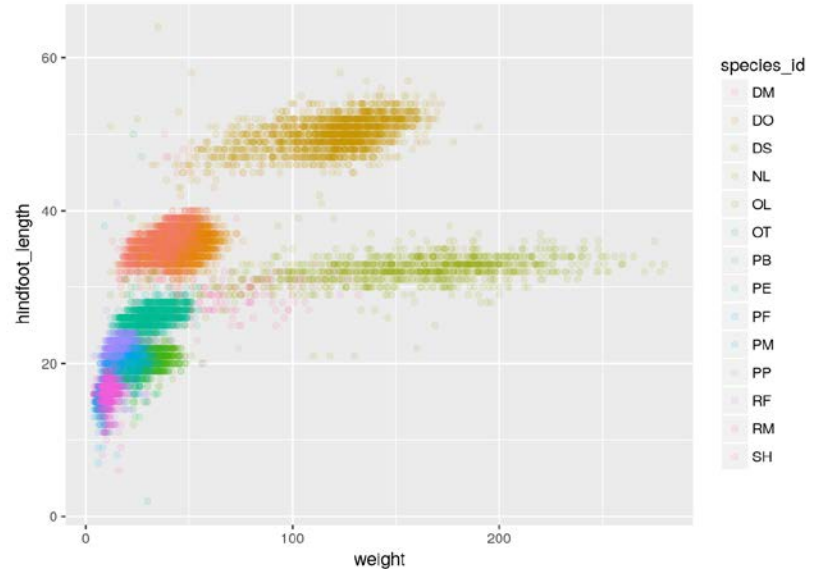


# Add color by species

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

Argument: color = <factor variable>

- Must be inside aes()





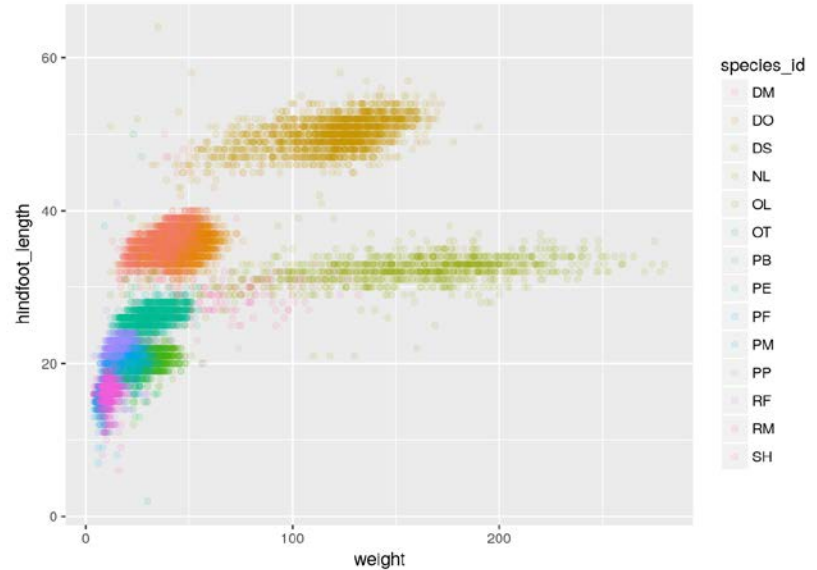
# Add color by species

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

Argument: color = <factor variable>

- Must be inside aes()

Whole plot



Points only





# Exercise 1

- Use the previous example as a starting point.
- Add color to the data points according to the plot from which the sample was taken (`plot_id`).
- **Hint:** Check the class for `plot_id`. Consider changing the class of `plot_id` from integer to factor. Why does this change how R makes the graph?

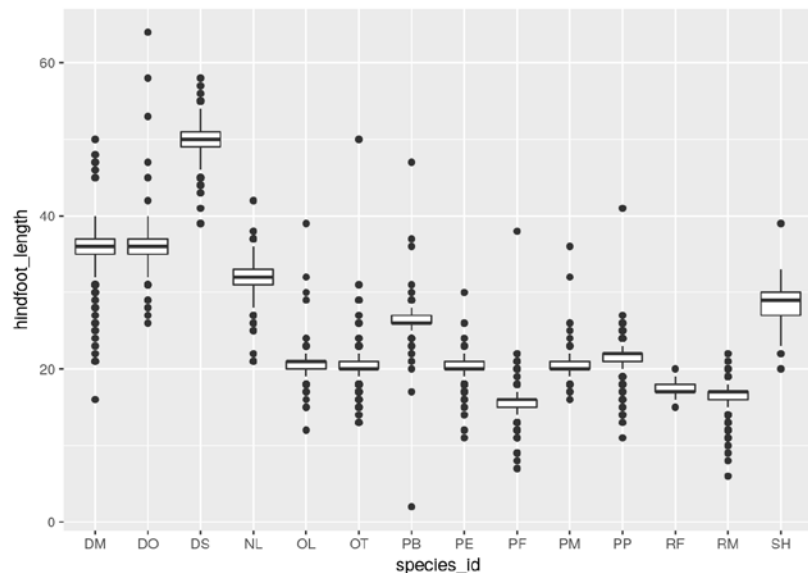
# Plot factor variables with box plot

```
ggplot(data = surveys_complete,  
       aes(x = species_id,  
           y = hindfoot_length)) +
```

```
  geom_boxplot()
```

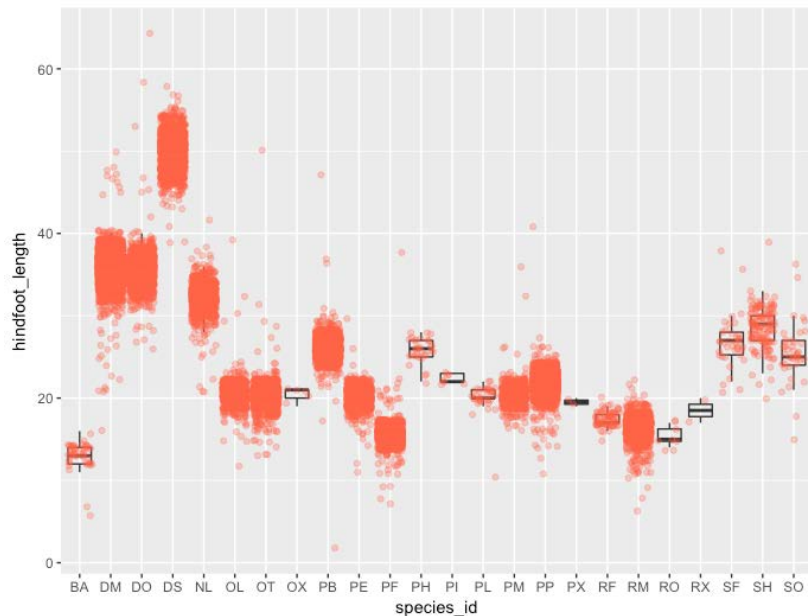
aes arguments:

- x: species id (factor)
- y: hinfot length (numeric)



# Overlay points on a box plot

```
ggplot(data = surveys_complete,  
       aes(x = species_id,  
           y = hindfoot_length)) +  
  geom_boxplot(alpha = 0) +  
  geom_jitter(alpha = 0.3,  
             color = "tomato")
```





## Exercise 2: Violin plot

- Plot the same data as in the previous example, but as a Violin plot
  - Hint: see `geom_violin()`.
- What information does this give you about the data that a box plot does?



# Time series data

## Reshape data:

```
yearly_counts <- surveys_complete %>%  
  group_by(year, species_id)  
  tally
```

%>%

	year	species_id	n
1	1977	DM	181
2	1977	DO	12
3	1977	DS	29
4	1977	OL	1
5	1977	OX	2
6	1977	PE	2
7	1977	PF	22
8	1977	PP	3
9	1977	RM	2
10	1978	DM	336
11	1978	DO	21
12	1978	DS	272
13	1978	NL	23
14	1978	OL	35
15	1978	OT	45
16	1978	PE	12
17	1978	PF	33
18	1978	PM	2
19	1978	PP	23

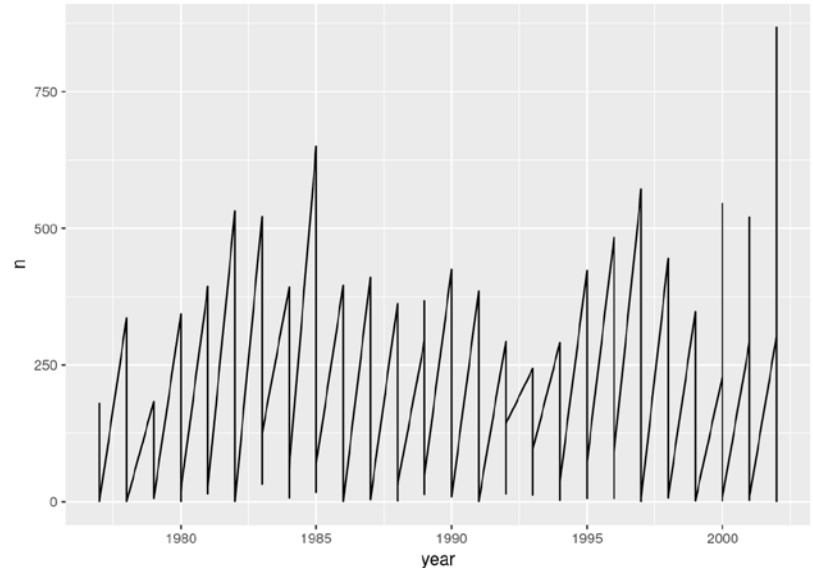


# Time series data

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

Arguments:

- Data = yearly counts
- X = year
- Y = n (# observations)

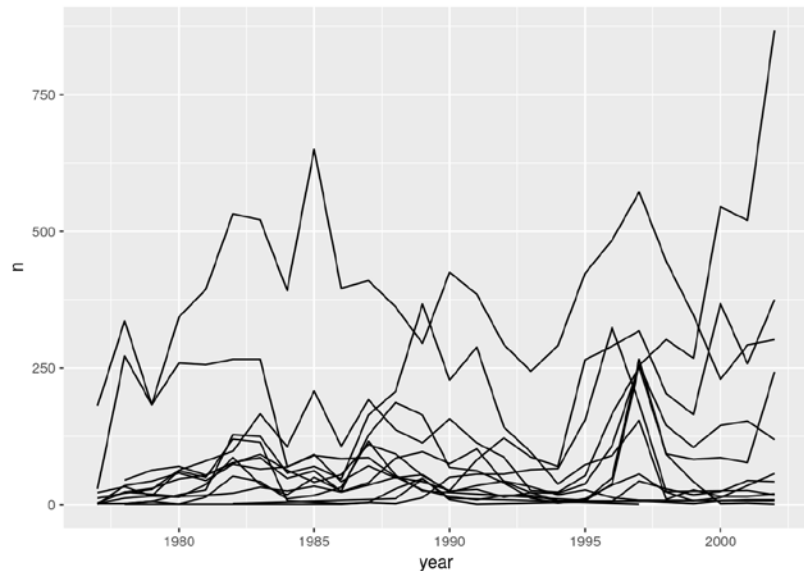


# Separate by species

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

New aes argument: **group**

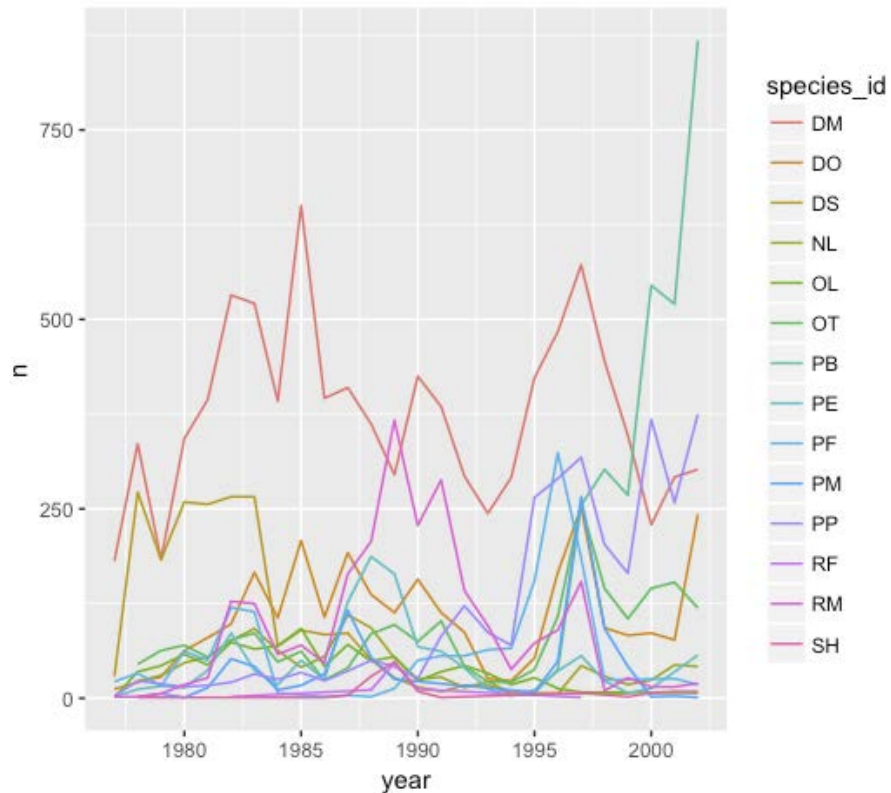
- Makes a line for each species id



# Color by species

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

Combine group and color to create  
species\_id legend





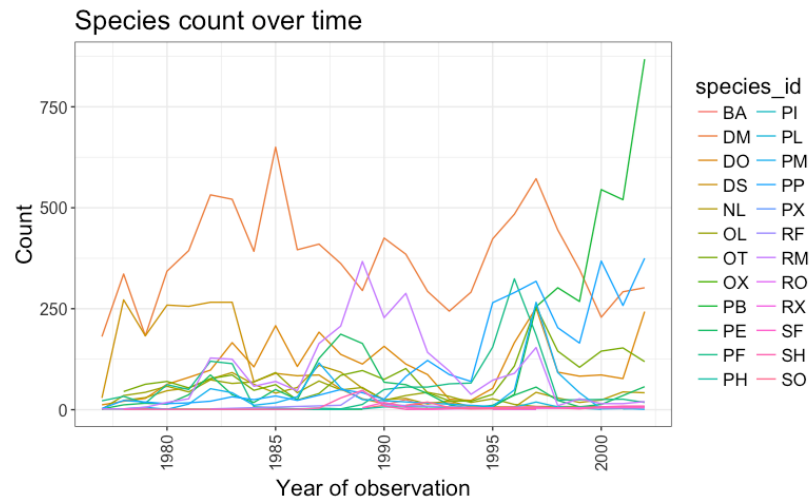
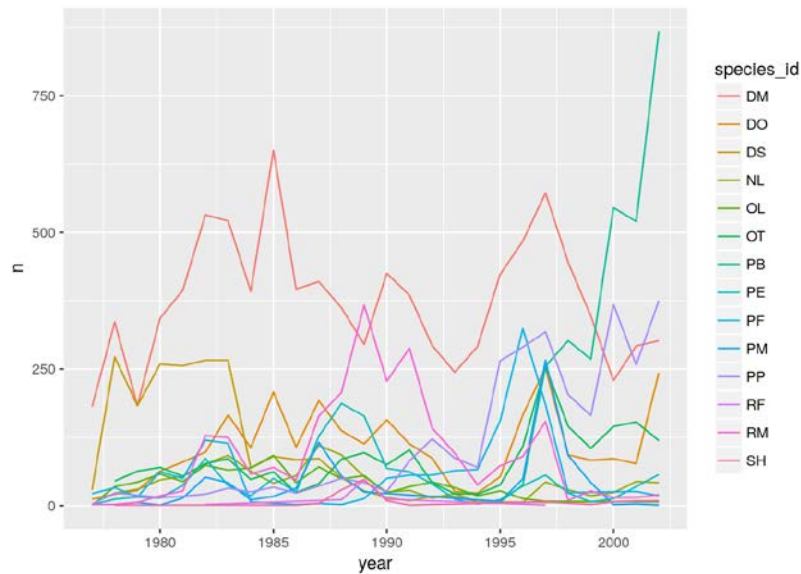


## Exercise #3

- Use what you just learned to create a plot that depicts how the average weight of each species changes through the years.
- Hint: reshape the data using the following code

```
yearly_weight <- surveys_complete %>%  
  group_by(year, species_id) %>%  
  summarize(avg_weight = mean(weight))
```

# Publication quality graph



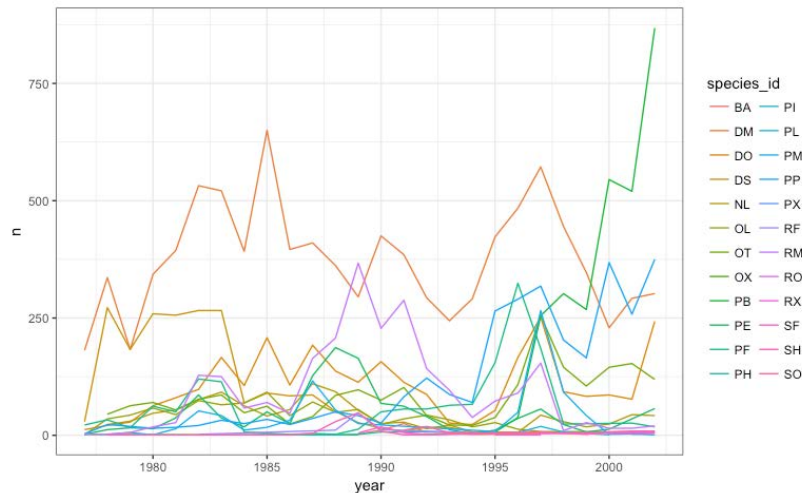


# Applying a premade theme

```
ggplot(data = yearly_counts,  
       aes(x = year, y = n,  
           color = sex,  
           group = sex)) +  
geom_line() +  
theme_bw()
```

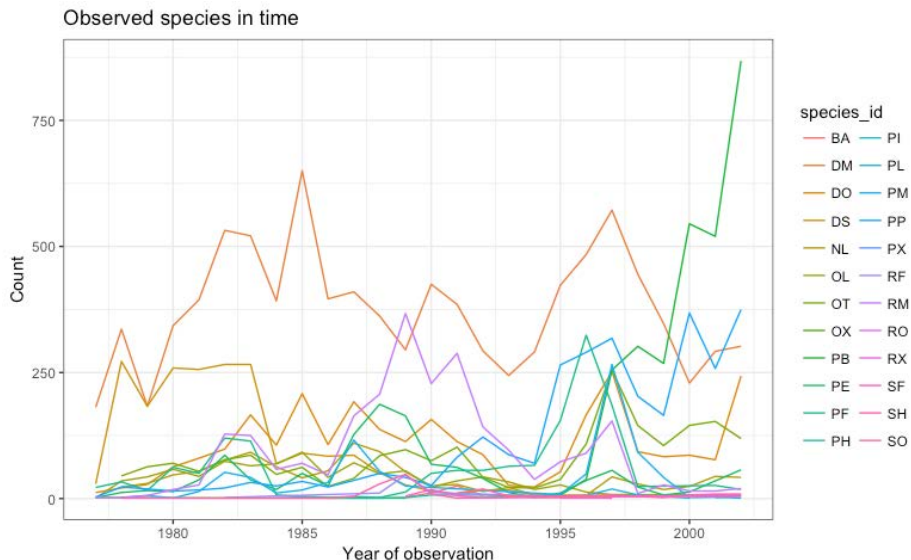


- See `?theme_bw()` to see descriptions of all ggplot themes



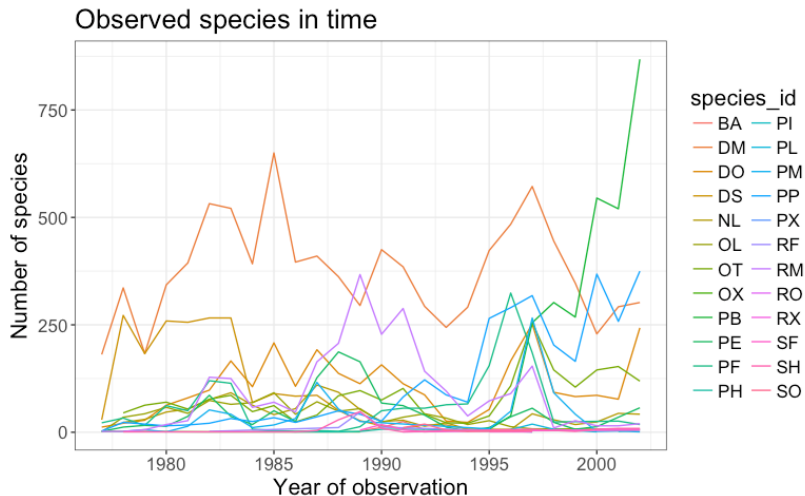
# Customize axis labels with **labs()**

```
ggplot(data = yearly_counts,  
       aes(x = year,  
           y = n,  
           color = species_id)) +  
geom_line() +  
labs(title = 'Observed Species in time',  
     x = 'Year of observation',  
     y = 'Count') +  
theme_bw()
```



# Customize font size with `element_text()`

```
ggplot(data = yearly_counts,  
       aes(x = year,  
           y = n,  
           color = species_id)) +  
geom_line() +  
labs(title = 'Observed Species in Time',  
      x = 'Year of observation',  
      y = 'Count') +  
theme_bw() +  
theme(text=element_text(size=16,  
                          family="Arial"))
```



See `?theme` for more theme arguments  
See `?margin` for more ggplot theme elements



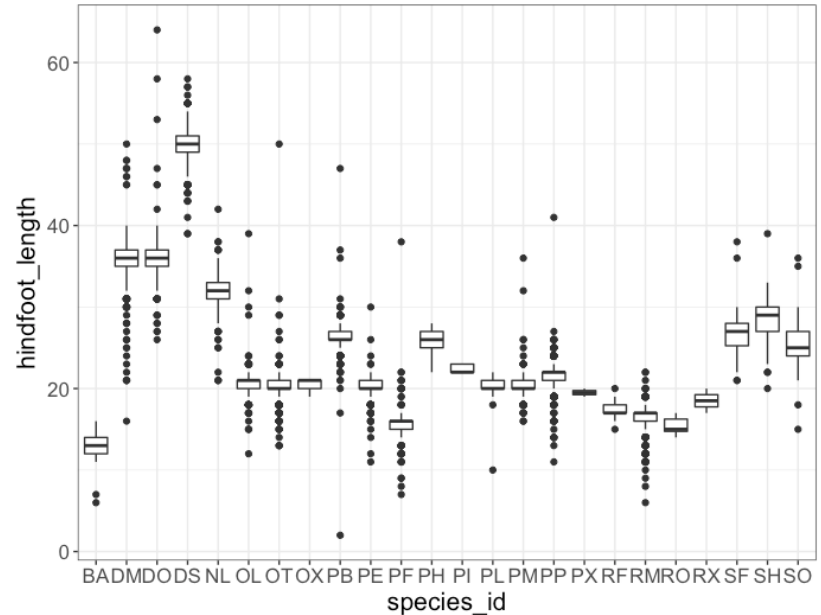
# Create your own theme

```
arial_theme <- theme_bw()+  
  theme(text = element_text(size=16,  
                             family="Arial"))
```



# Apply your theme

```
ggplot(surveys_complete,  
       aes(x = species_id,  
           y = hindfoot_length)) +  
geom_boxplot() +  
arial_theme
```





# Save your plot with **ggsave()**

- Save a plot to a variable
- **ggsave**: saves plot to a file
  - Arguments: name of file, ggplot variable, width + height
  - Output: a png file

## Example:

```
ggsave("name_of_file.png", my_plot, width=15, height=10)
```





# Need help?

- Email: [tobin.magle@colostate.edu](mailto:tobin.magle@colostate.edu)
- Data Management Services website:  
<http://lib.colostate.edu/services/data-management>
- Data Carpentry: <http://www.datacarpentry.org/>
  - R Ecology Lesson:  
<http://www.datacarpentry.org/R-ecology-lesson/04-visualization-ggplot2.html>
- Ggplot2 Cheat Sheets:
  - <https://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf>