Scatter plots

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

Rick Scavetta
Founder, Scavetta Academy

48 geometries

geom_*						
abline	contour	dotplot	jitter	pointrange	ribbon	spoke
area	count	errorbar	label	polygon	rug	step
bar	crossbar	errorbarh	line	qq	segment	text
bin2d	curve	freqpoly	linerange	qq_line	sf	tile
blank	density	hex	map	quantile	sf_label	violin
boxplot	density2d	histogram	path	raster	sf_text	vline
col	density_2d	hline	point	rect	smooth	

Common plot types

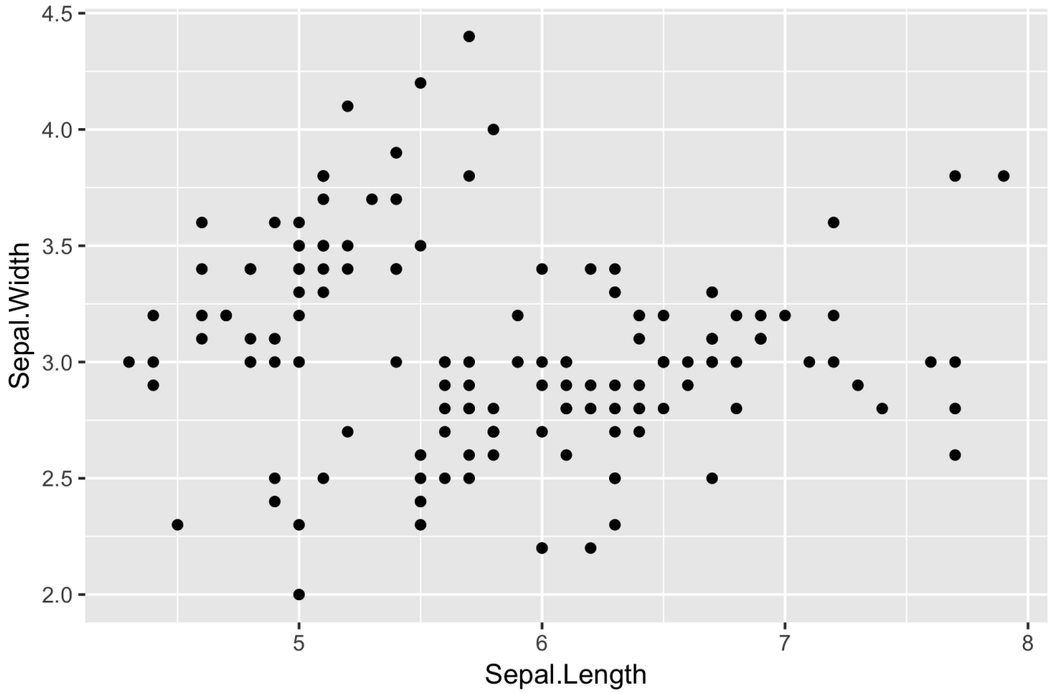
Plot type	Possible Geoms
Scatter plots	points, jitter, abline, smooth, count

Scatter plots

- Each geom can accept specific aesthetic mappings, e.g. geom_point():

Essential
x,y

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width)) +
  geom_point()
```

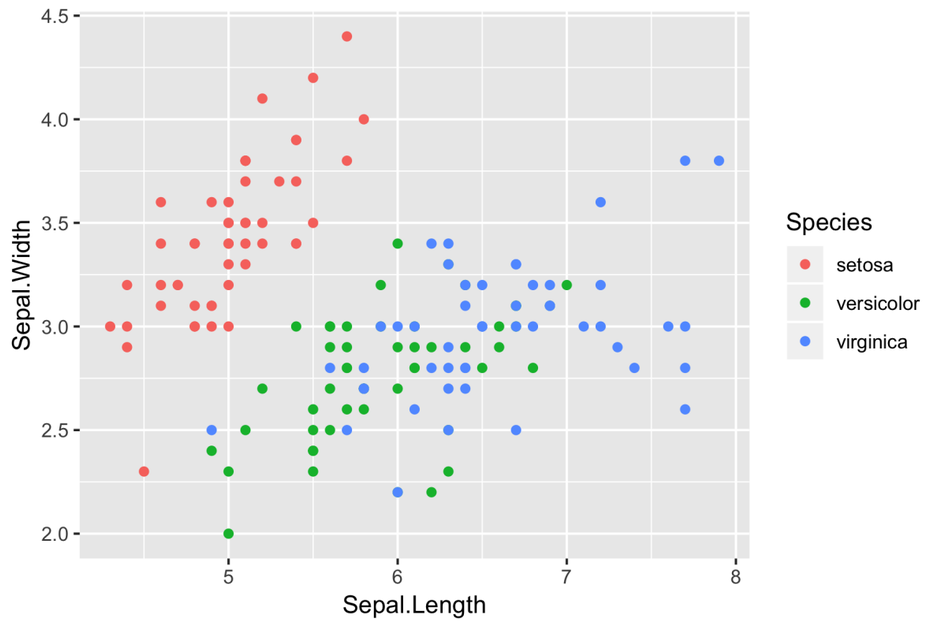


Scatter plots

- Each geom can accept specific aesthetic mappings, e.g. geom_point():

Essential	Optional
x,y	alpha, color, fill, shape, size, stroke

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width,
                 col = Species)) +
  geom_point()
```

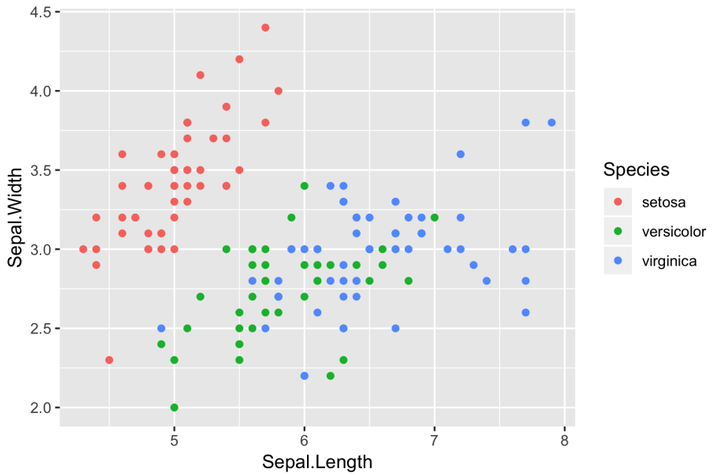


Geom-specific aesthetic mappings

```
# These result in the same plot!
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, col = Species)) +
  geom_point()

ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width)) +
  geom_point(aes(col = Species))
```

Control aesthetic mappings of each layer independently:



```
head(iris, 3) # Raw data
```

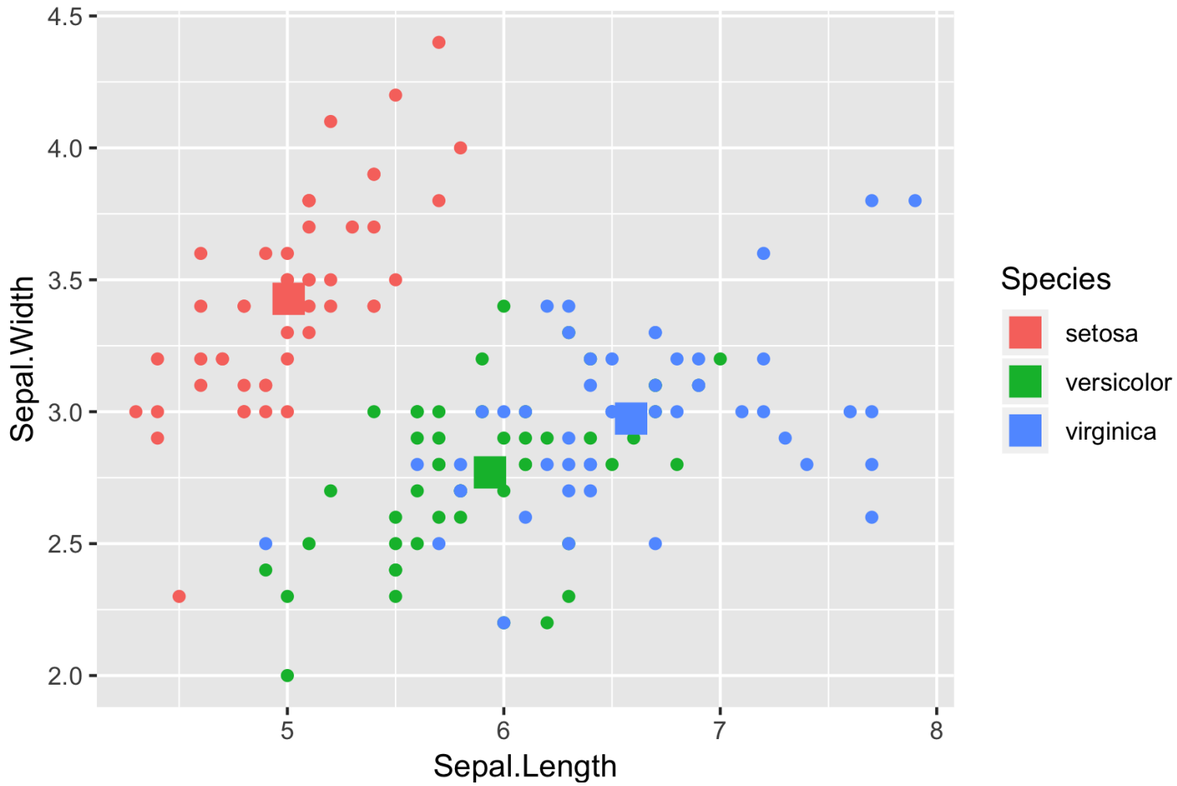
```
Species Sepal.Length Sepal.Width Petal.Length Petal.Width
1  setosa          5.1         3.5          1.4         0.2
2  setosa          4.9         3.0          1.4         0.2
3  setosa          4.7         3.2          1.3         0.2
```

```
iris %>%
  group_by(Species) %>%
  summarise_all(mean) -> iris.summary

iris.summary # Summary statistics
```

```
# A tibble: 3 x 5
  Species    Sepal.Length Sepal.Width Petal.Length Petal.Width
  <fct>             <dbl>       <dbl>        <dbl>       <dbl>
1 setosa             5.01        3.43         1.46       0.246
2 versicolor         5.94        2.77         4.26       1.33
3 virginica          6.59        2.97         5.55       2.03
```

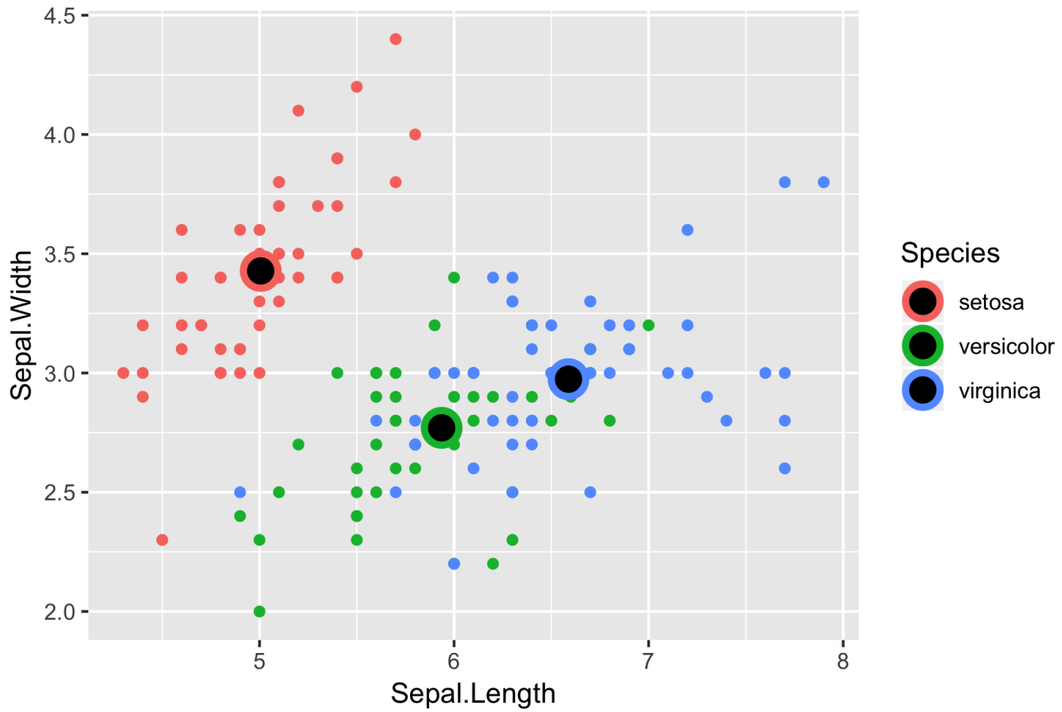
```
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, col = Species)) +
  # Inherits both data and aes from ggplot()
  geom_point() +
  # Different data, but inherited aes
  geom_point(data = iris.summary, shape = 15, size = 5)
```



Shape attribute values

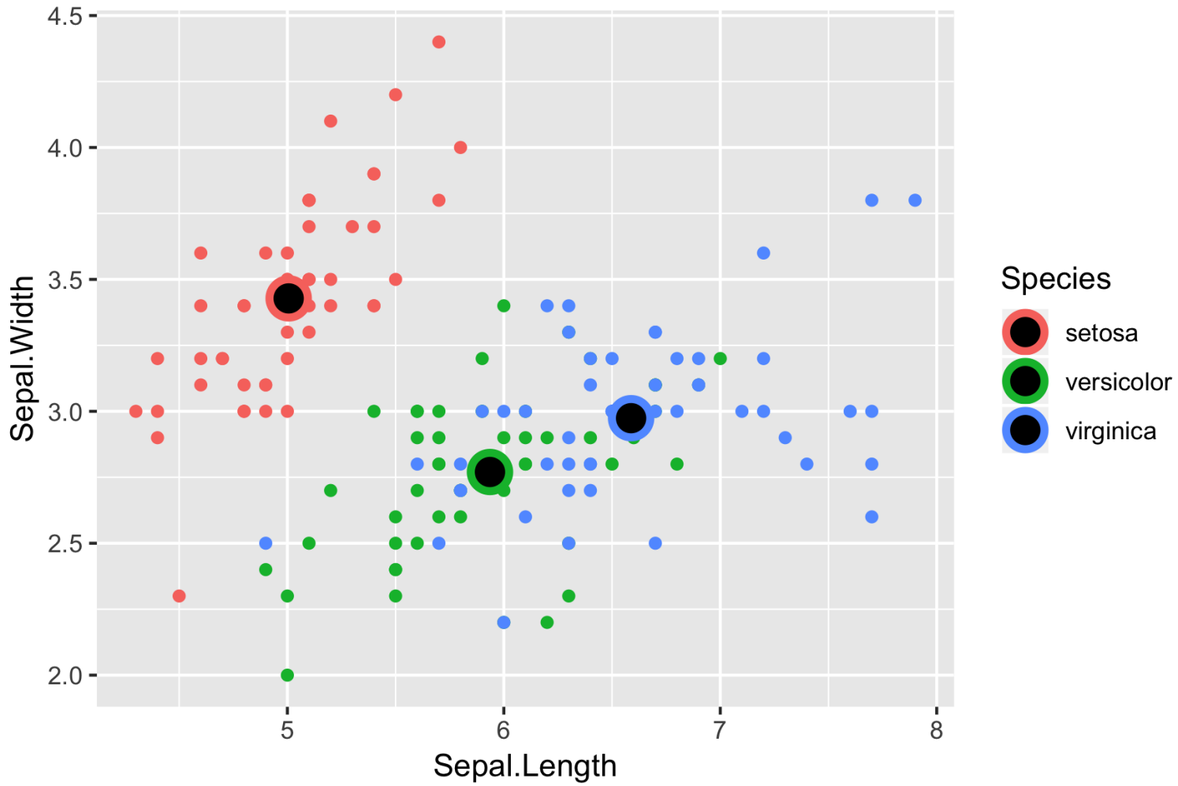
Example

```
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, col = Species)) +
  geom_point() +
  geom_point(data = iris.summary, shape = 21, size = 5,
             fill = "black", stroke = 2)
```



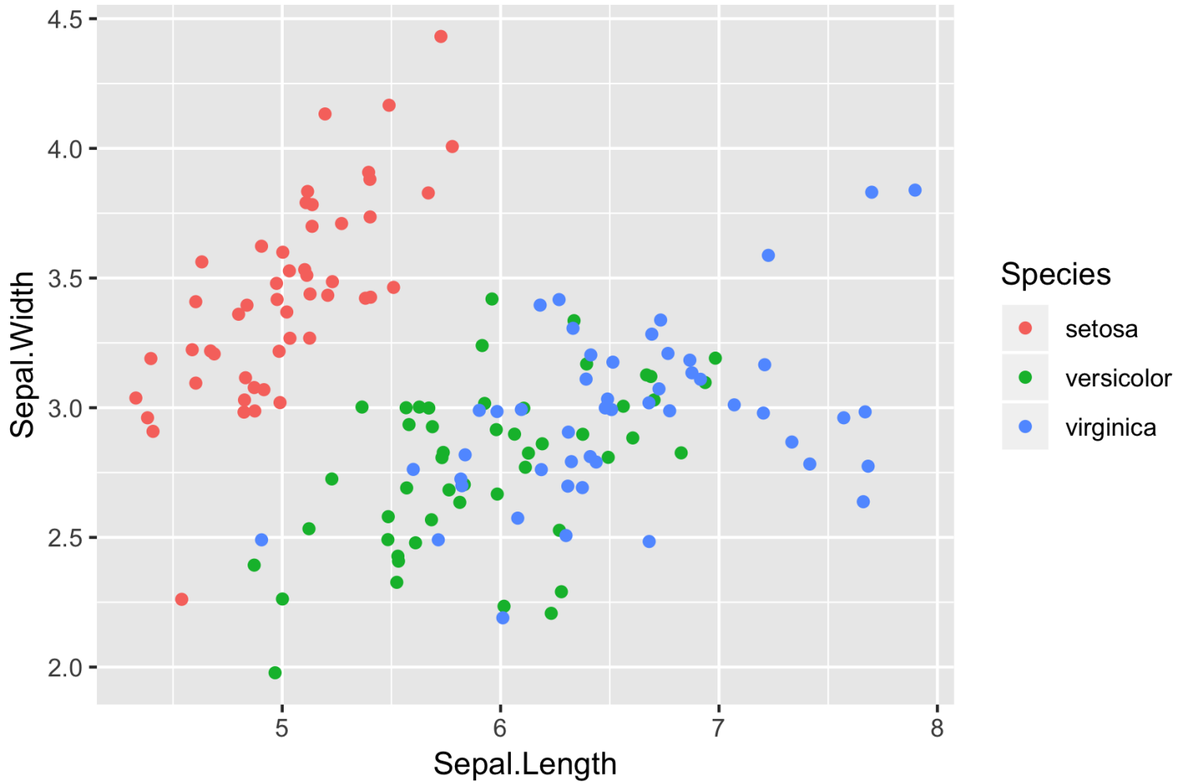
On-the-fly stats by ggplot2

- See the second course for the stats layer.
- Note: Avoid plotting only the mean without a measure of spread, e.g. the standard deviation.



position = "jitter"

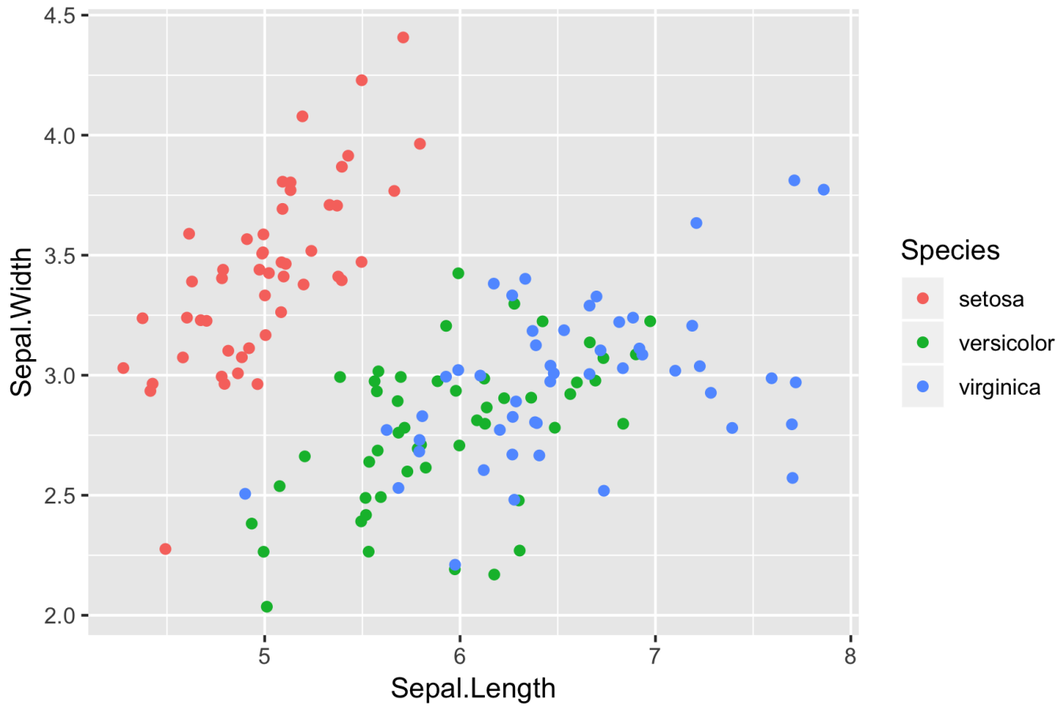
```
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, col = Species)) +
  geom_point(position = "jitter")
```



geom_jitter()

A short-cut to `geom_point(position = "jitter")`

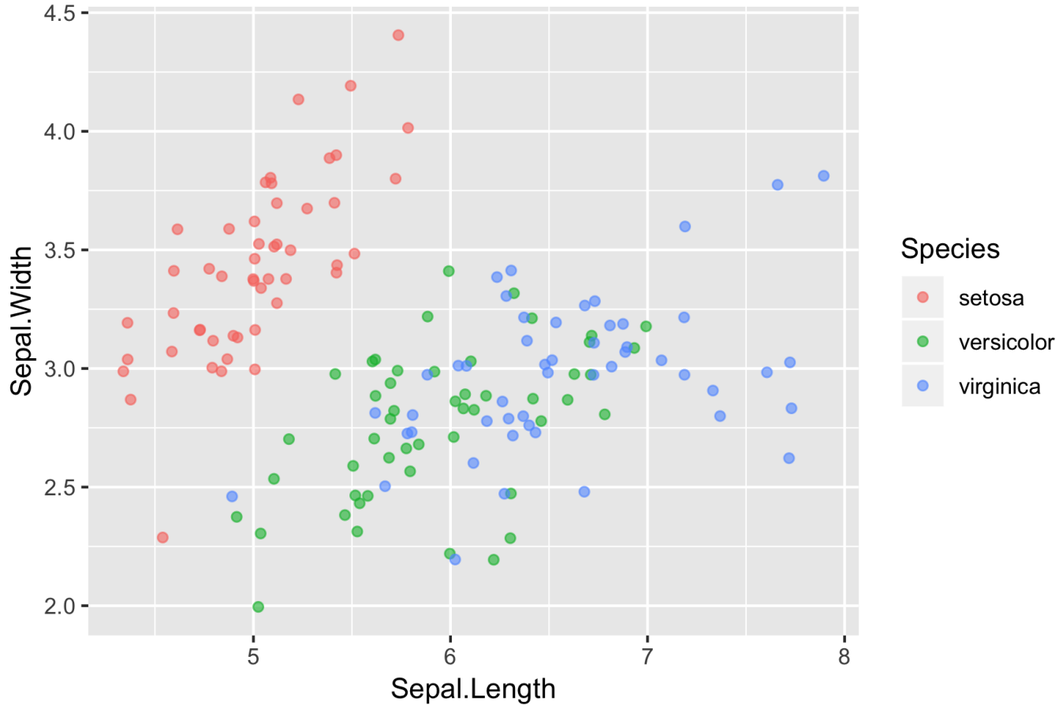
```
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, col = Species)) +
  geom_jitter()
```



Don't forget to adjust alpha

- Combine jittering with alpha-blending if necessary

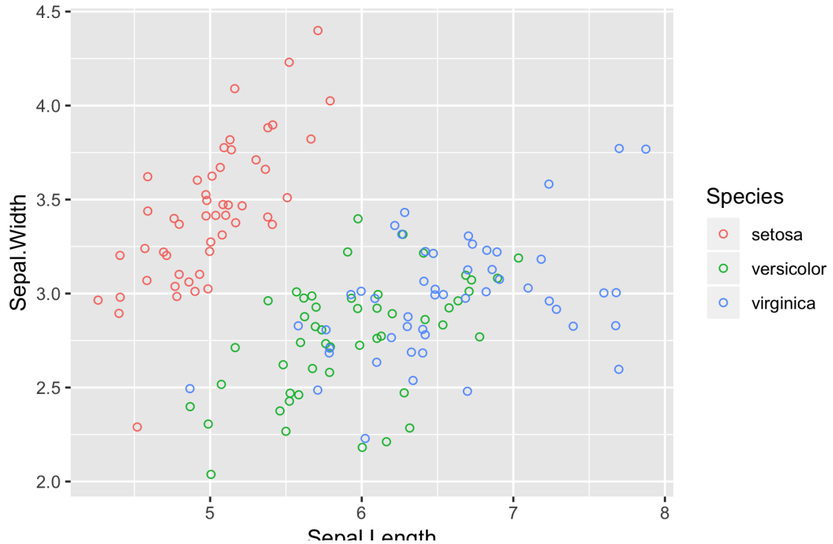
```
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, col = Species)) +
  geom_jitter(alpha = 0.6)
```



Hollow circles also help

- `shape = 1` is a. hollow circle.
- Not necessary to also use alpha-blending.

```
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, col = Species)) +
  geom_jitter(shape = 1)
```



Let's practice!

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

Histograms

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

Rick Scavetta
Founder, Scavetta Academy

Common plot types

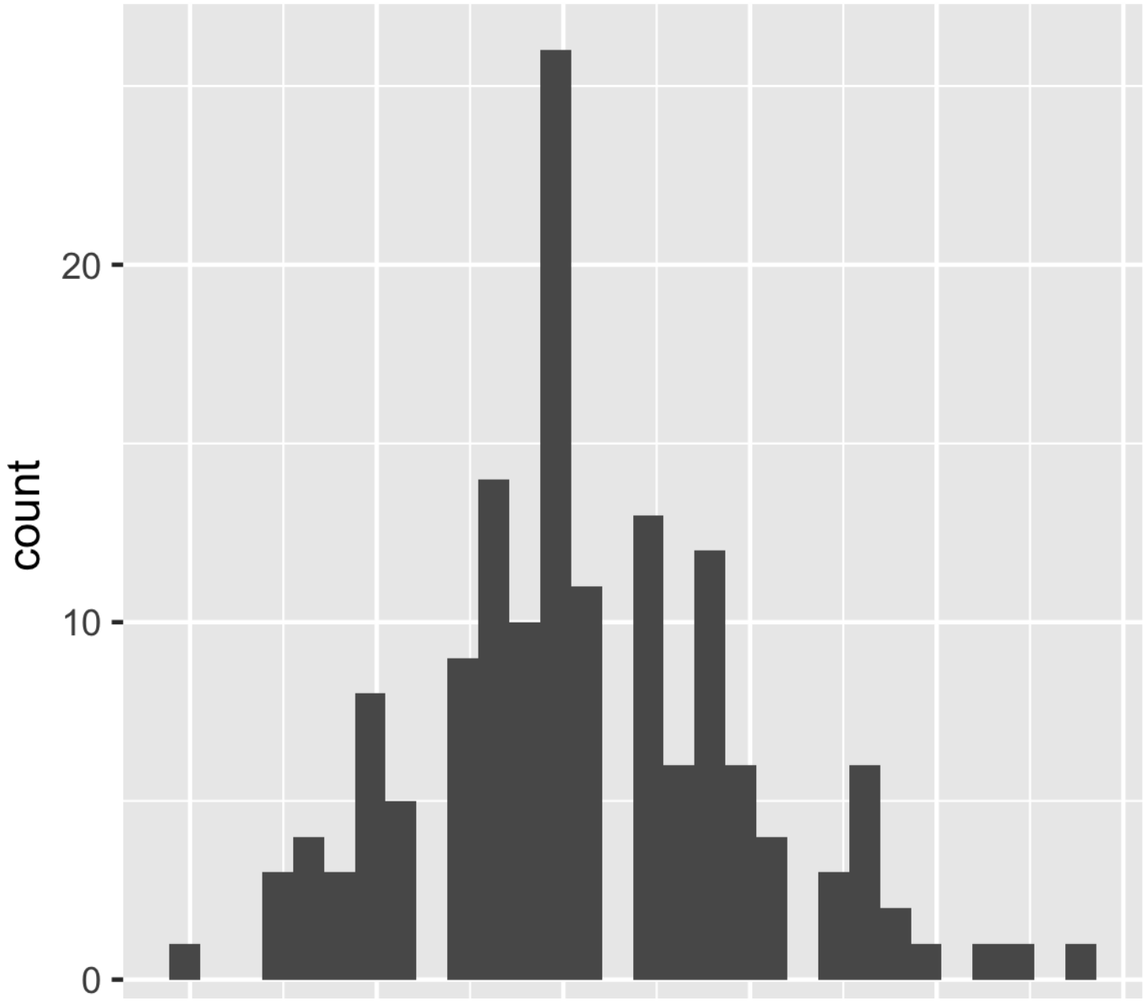
Plot type	Possible Geoms
Scatter plots	points, jitter, abline, smooth, count
Bar plots	**histogram, bar, col, errorbar**
Line plots	line, path

Histograms

```
ggplot(iris, aes(x = Sepal.Width)) +
  geom_histogram()
```

- A plot of binned values
  - i.e. a statistical function

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



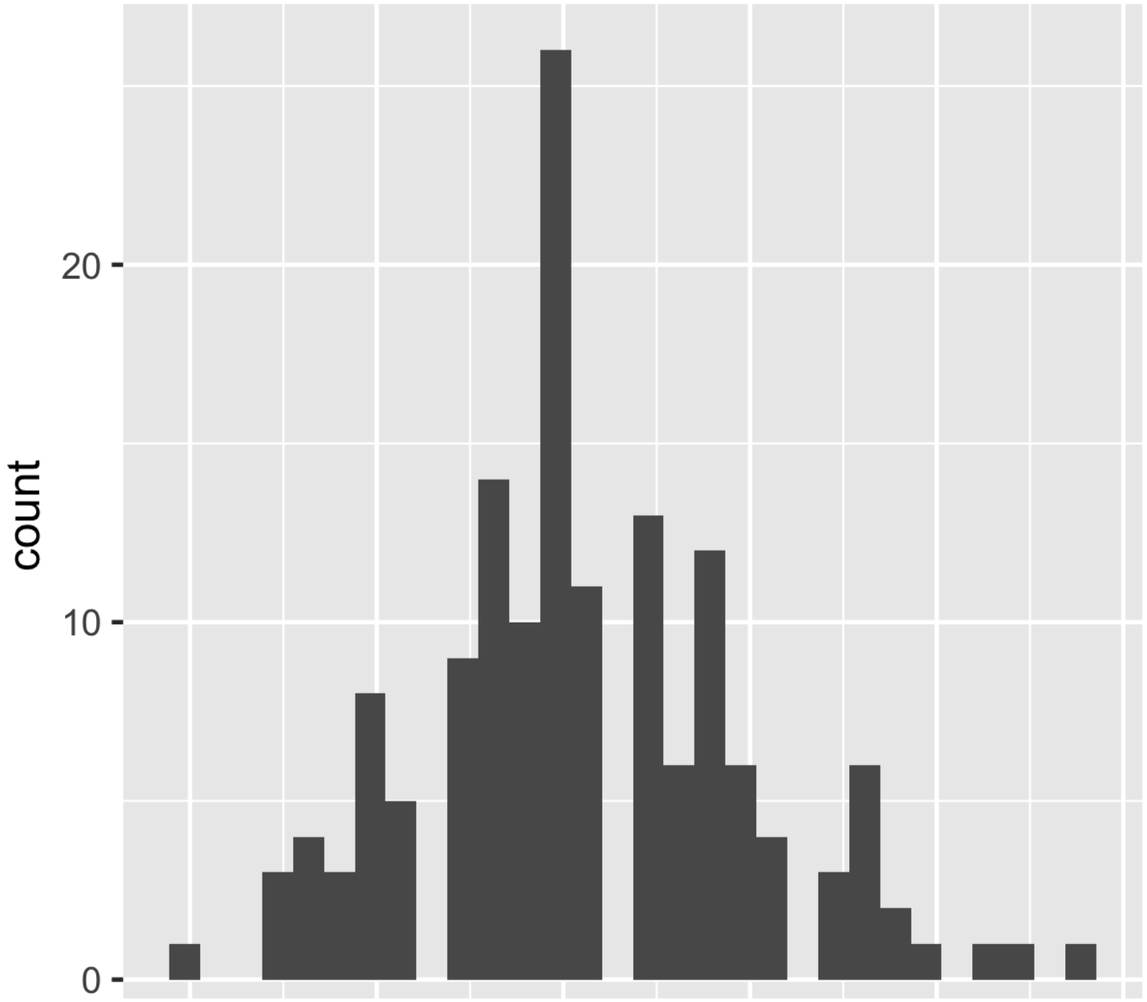
Default of 30 even bins

```
ggplot(iris, aes(x = Sepal.Width)) +
  geom_histogram()
```

- A plot of binned values
  - i.e. a statistical function

```
# Default bin width:
diff(range(iris$Sepal.Width))/30
```

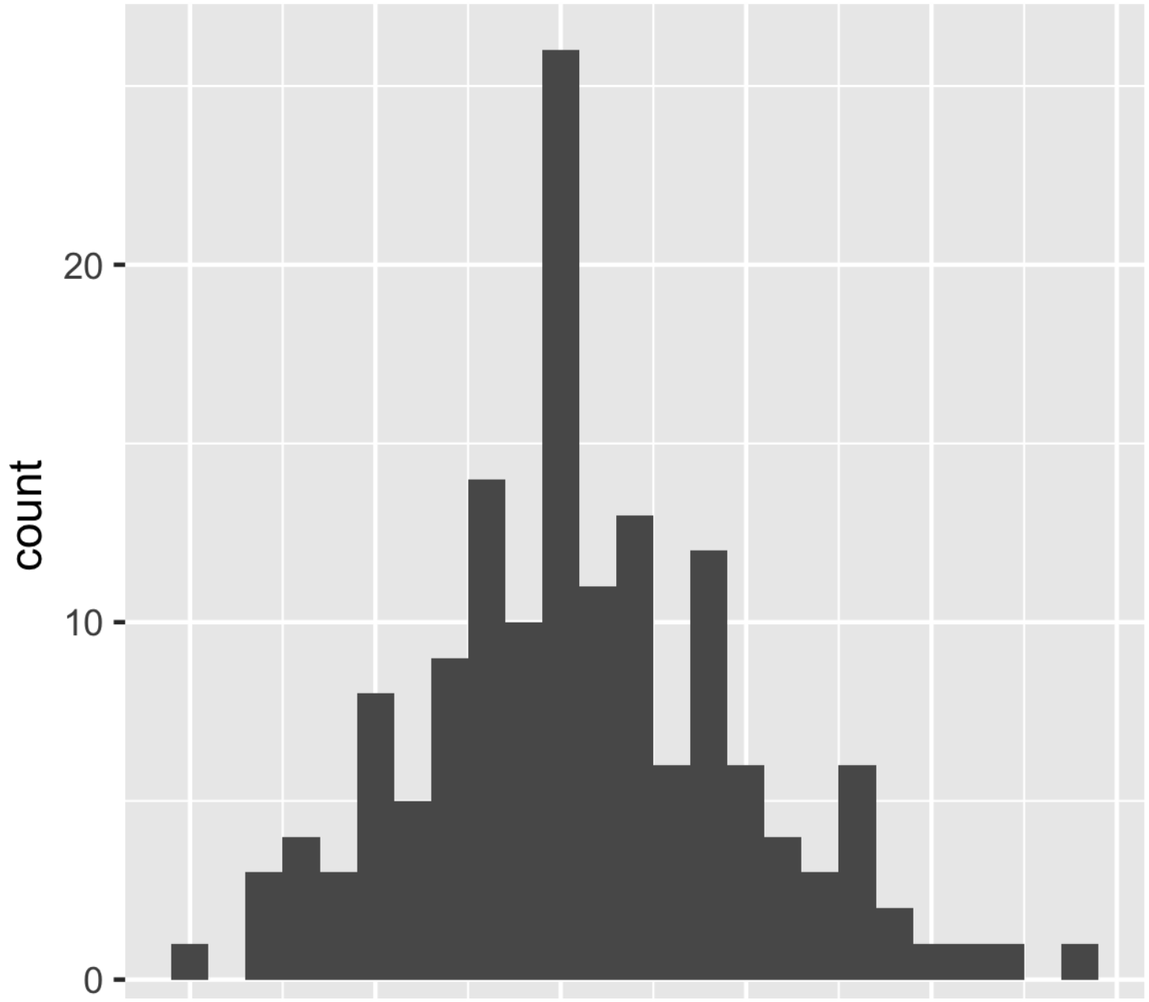
```
[1] 0.08
```



Intuitive and meaningful bin widths

```
ggplot(iris, aes(x = Sepal.Width)) +
  geom_histogram(binwidth = 0.1)
```

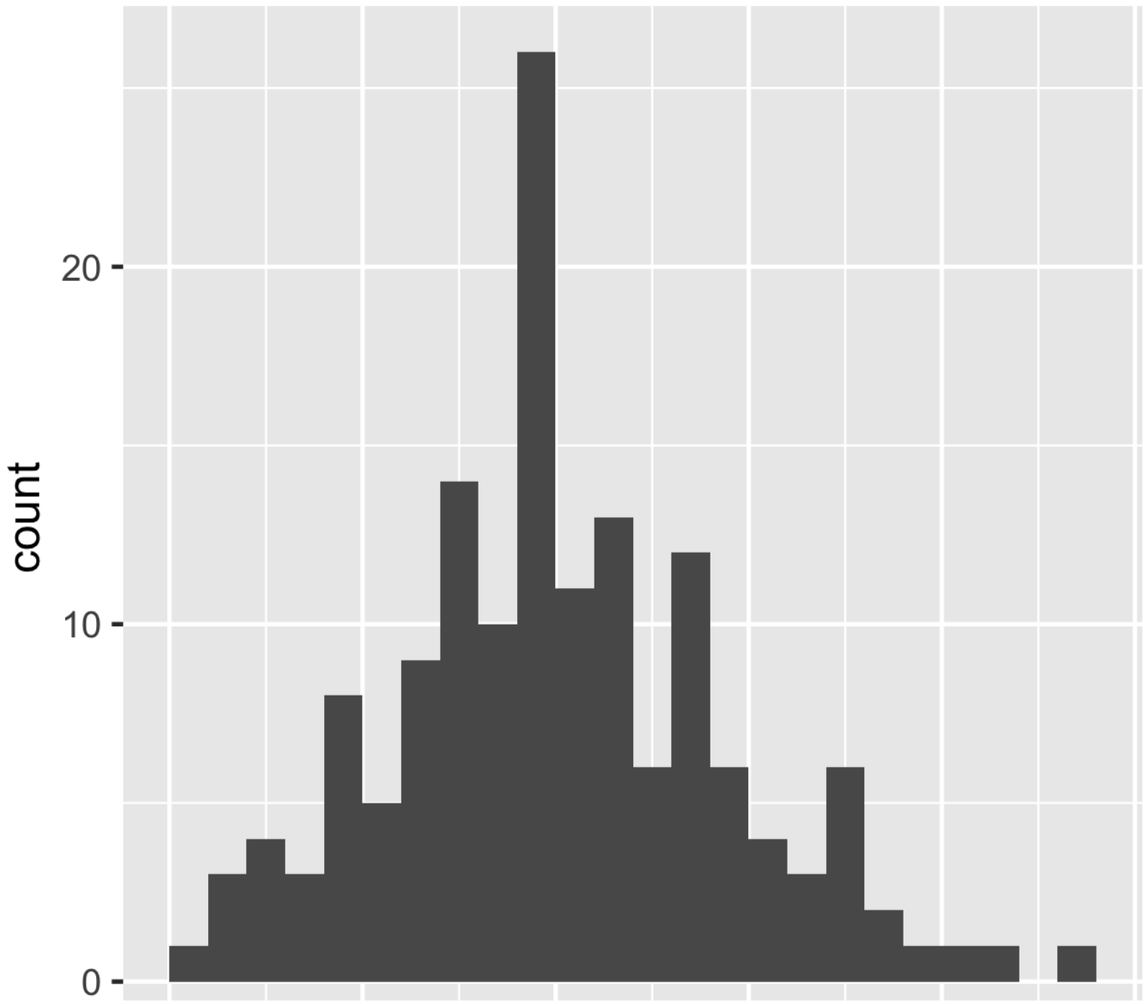
- Always set a meaningful bin widths for your data.
- No spaces between bars.



Re-position tick marks

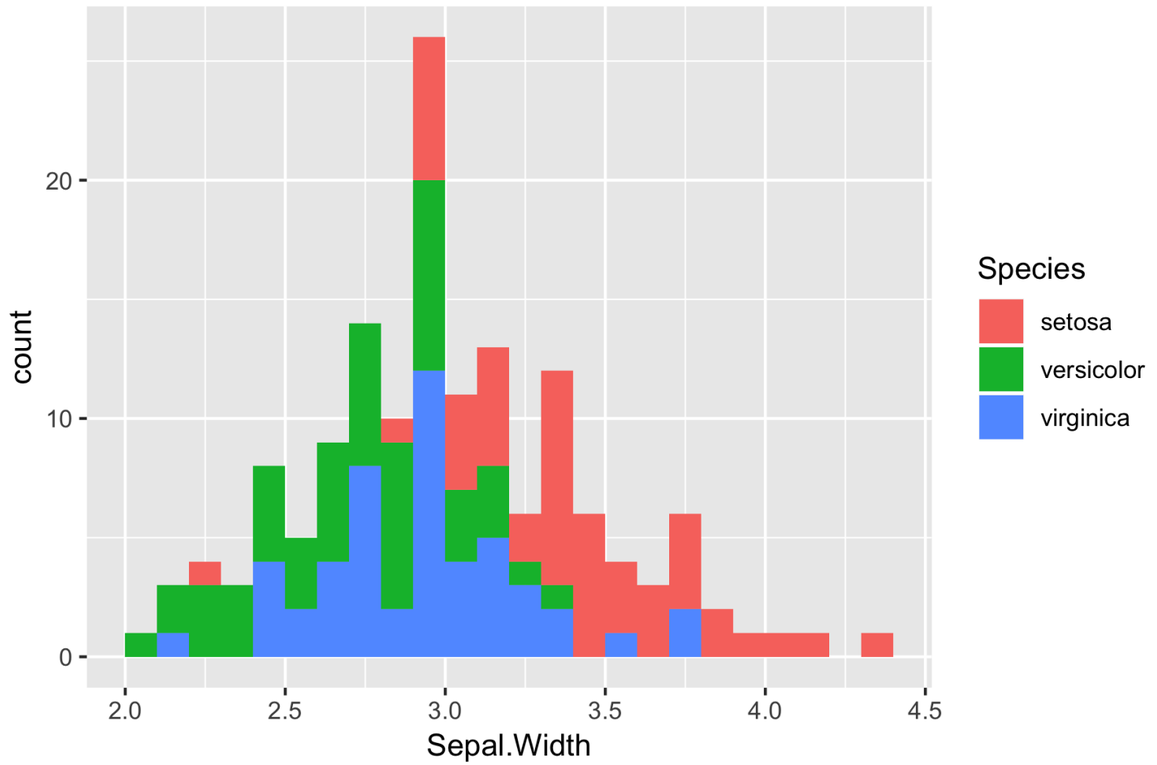
```
ggplot(iris, aes(x = Sepal.Width)) +
  geom_histogram(binwidth = 0.1,
                 center = 0.05)
```

- Always set a meaningful bin widths for your data.
- No spaces between bars.
- X axis labels are between bars.



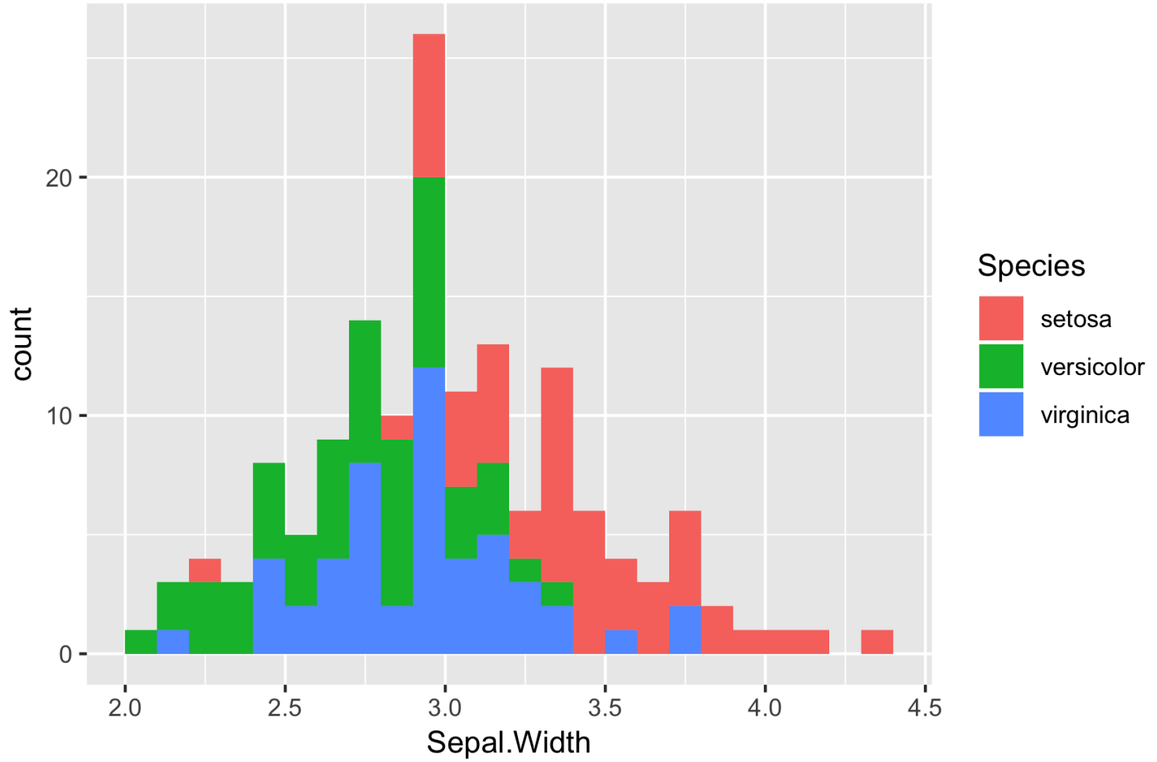
Different Species

```
ggplot(iris, aes(x = Sepal.Width,
                 fill = Species)) +
  geom_histogram(binwidth = .1,
                 center = 0.05)
```



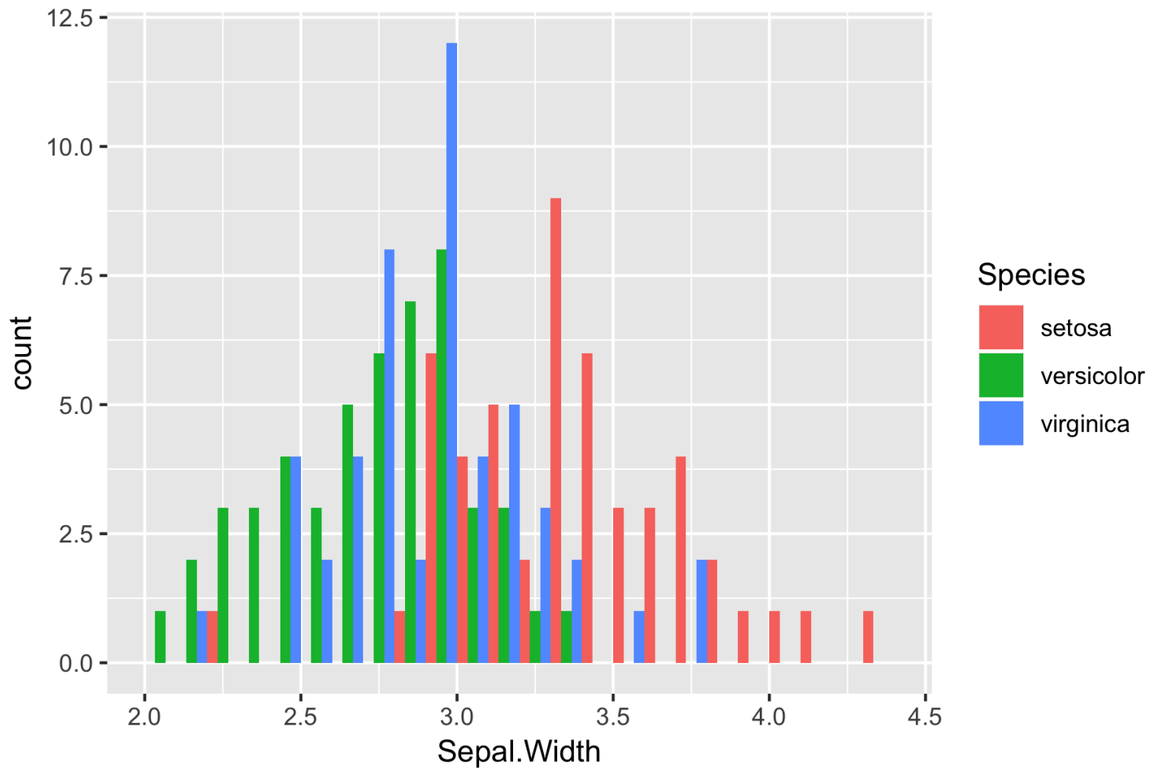
Default position is "stack"

```
ggplot(iris, aes(x = Sepal.Width,
                 fill = Species)) +
  geom_histogram(binwidth = .1,
                 center = 0.05,
                 position = "stack")
```



position = "dodge"

```
ggplot(iris, aes(x = Sepal.Width,
                 fill = Species)) +
  geom_histogram(binwidth = .1,
                 center = 0.05,
                 position = "dodge")
```



position = "fill"

```
ggplot(iris, aes(x = Sepal.Width, fill = S
  geom_histogram(binwidth = .1, center = 0
```

Final Slide

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

Bar plots

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

Rick Scavetta
Founder, Scavetta Academy

Bar Plots, with a categorical X-axis

- Use geom_bar() or geom_col()

Geom	Stat	Action
`geom_bar()`	"count"	Counts the number of cases at each x position
`geom_col()`	"identity"	Plot actual values

  - All positions from before are available
  - Two types
    - Absolute counts
    - Distributions

Bar Plots, with a categorical X-axis

- Use geom_bar() or geom_col()

Geom	Stat	Action
`geom_bar()`	"count"	Counts the number of cases at each x position
`geom_col()`	"identity"	Plot actual values

Bar Plots, with a categorical X-axis

- Use geom_bar() or geom_col()

Geom	Stat	Action
`geom_bar()`	"count"	Counts the number of cases at each x position
`geom_col()`	"identity"	Plot actual values

  - All positions from before are available
  - Two types
    - Absolute counts
    - Distributions

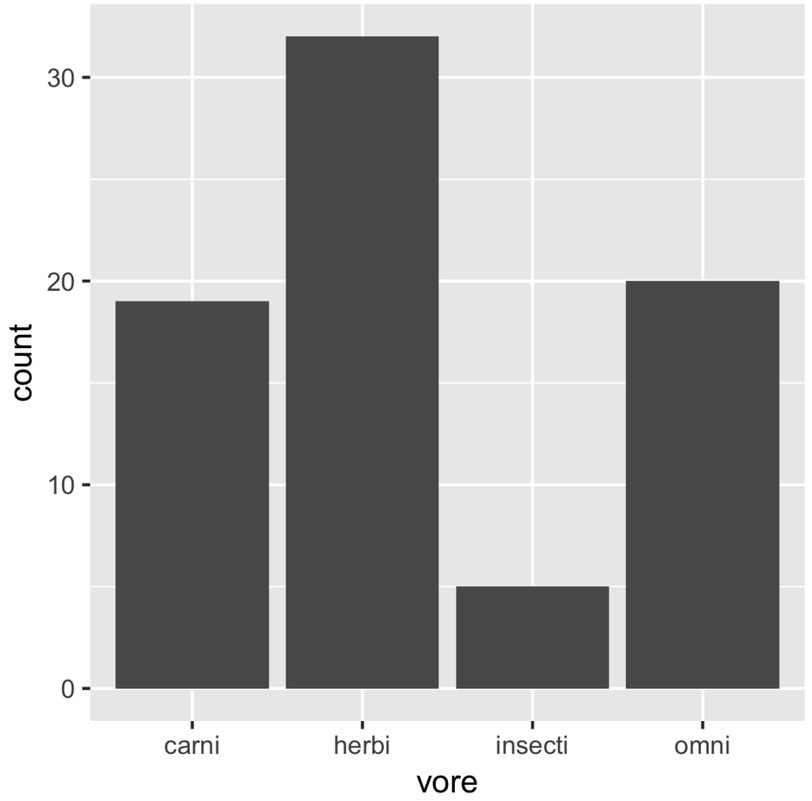
Habits of mammals

```
str(sleep)
```

```
'data.frame':    76 obs. of  3 variables:
 $ vore : Factor w/ 4 levels "carni","herbi",..: 1 4 2 4 2 2 1 1 2 2 ...
 $ total: num  12.1 17 14.4 14.9 4 14.4 8.7 10.1 3 5.3 ...
 $ rem  : num  NA 1.8 2.4 2.3 0.7 2.2 1.4 2.9 NA 0.6 ...
```

Bar plot

```
ggplot(sleep, aes(vore)) +
  geom_bar()
```



Plotting distributions instead of absolute counts

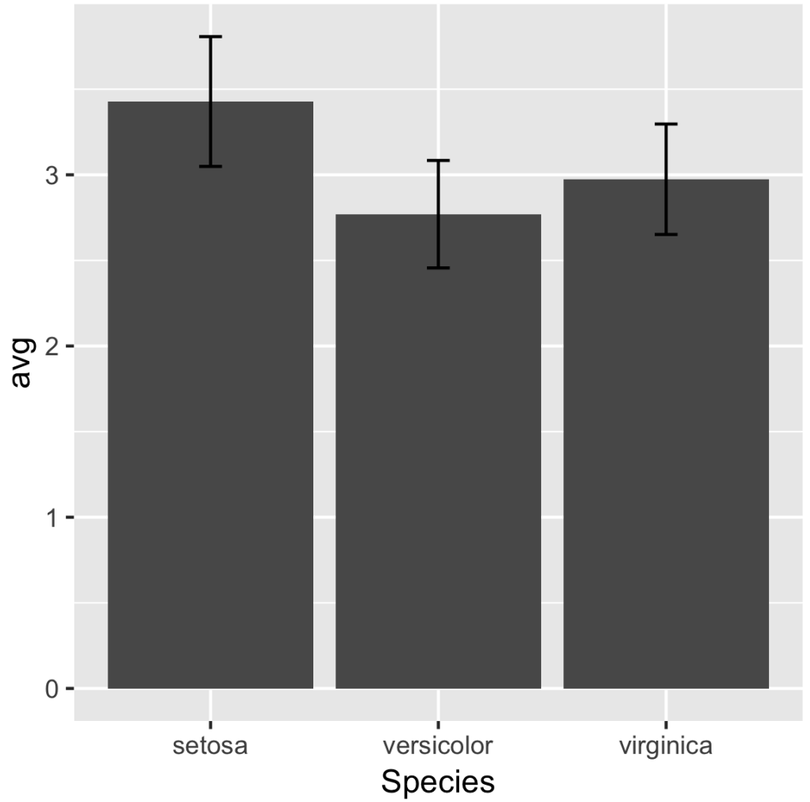
```
# Calculate Descriptive Statistics:
iris %>%
  select(Species, Sepal.Width) %>%
  gather(key, value, -Species) %>%
  group_by(Species) %>%
  summarise(avg = mean(value),
            stdev = sd(value))
   -> iris_summ_long
```

```
iris_summ_long
```

Species	avg	stdev
setosa	3.43	0.38
versicolor	2.77	0.31
virginica	2.97	0.32

Plotting distributions

```
ggplot(iris_summ_long, aes(x = Species,
                           y = avg)) +
  geom_col() +
  geom_errorbar(aes(ymin = avg - stdev,
                    ymax = avg + stdev),
                width = 0.1)
```



Let's practice!

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

Line plots

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

Rick Scavetta
Founder, Scavetta Academy

Common plot types

Plot type	Possible Geoms
Scatter plots	points, jitter, abline, smooth, count
Bar plots	histogram, bar, col, errorbar
Line plots	**line, path**

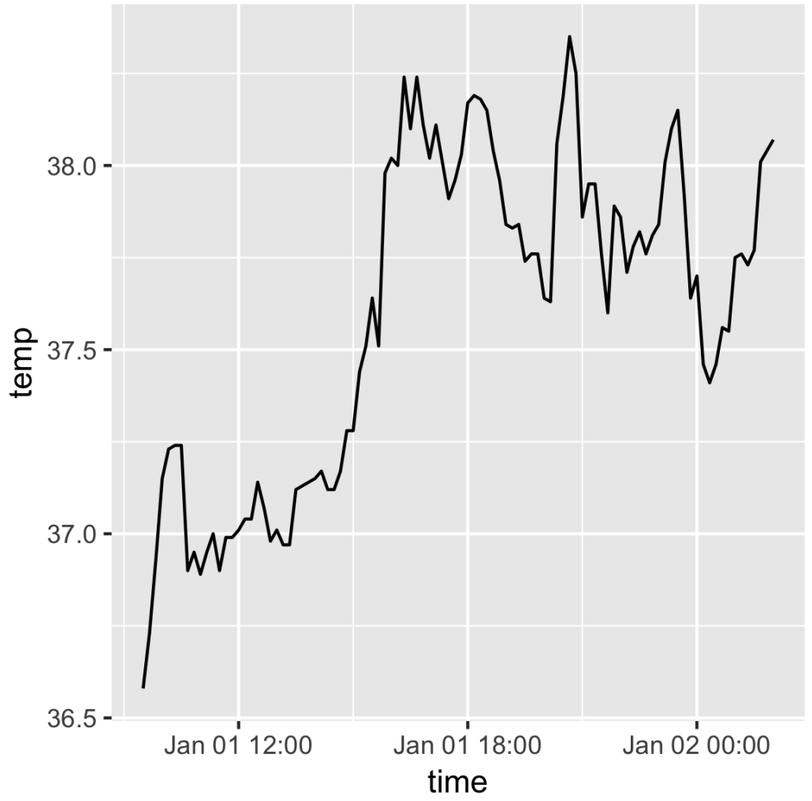
Beaver

```
str(beaver)
```

```
'data.frame':    101 obs. of  3 variables:
 $ time  : POSIXct, format: "2000-01-01 09:30:00" "2000-01-01 09:40:00" "2000-01-01 09:5
 $ temp  : num  36.6 36.7 36.9 37.1 37.2 ...
 $ active: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

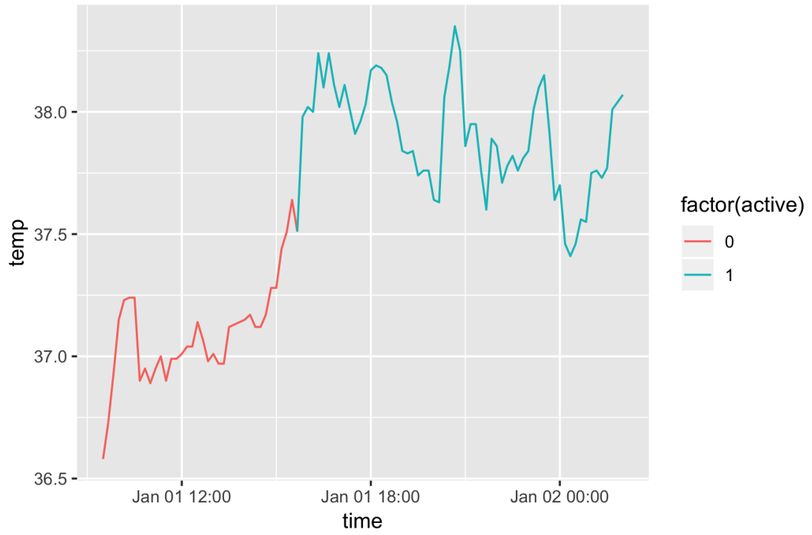
Beaver

```
ggplot(beaver, aes(x = time, y = temp)) +
  geom_line()
```



Beaver

```
ggplot(beaver, aes(x = time, y = temp,
                   color = factor(active))
                   ) +
  geom_line()
```



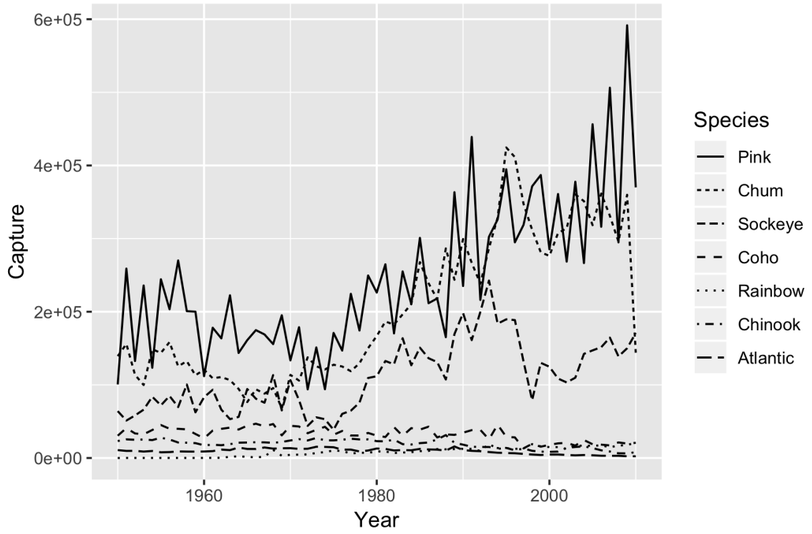
The fish catch dataset

```
str(fish)
```

```
'data.frame':    427 obs. of  3 variables:
 $ Species: Factor w/ 7 levels "Pink","Chum",..: 1 2 3 4 5 6 7 1 2 3 ...
 $ Year   : int  1950 1950 1950 1950 1950 1950 1950 1951 1951 1951 ...
 $ Capture: int  100600 139300 64100 30500 0 23200 10800 259000 155900 51200 ...
```

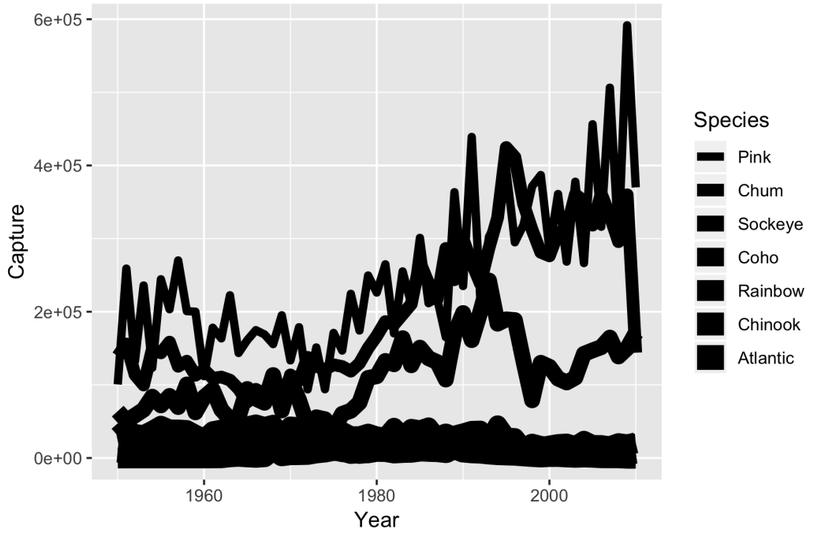
Linetype aesthetic

```
ggplot(fish, aes(x = Year,
                 y = Capture,
                 linetype = Species)) +
  geom_line()
```



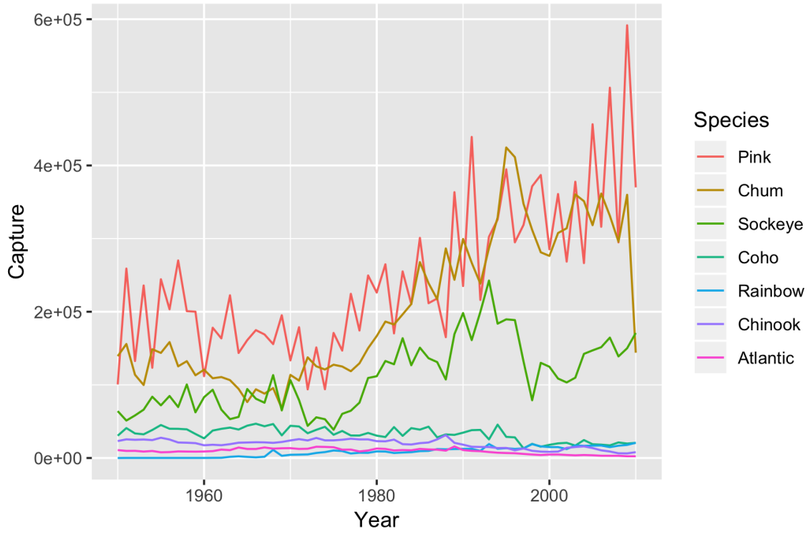
Size aesthetic

```
ggplot(fish, aes(x = Year,
                 y = Capture,
                 size = Species)) +
  geom_line()
```

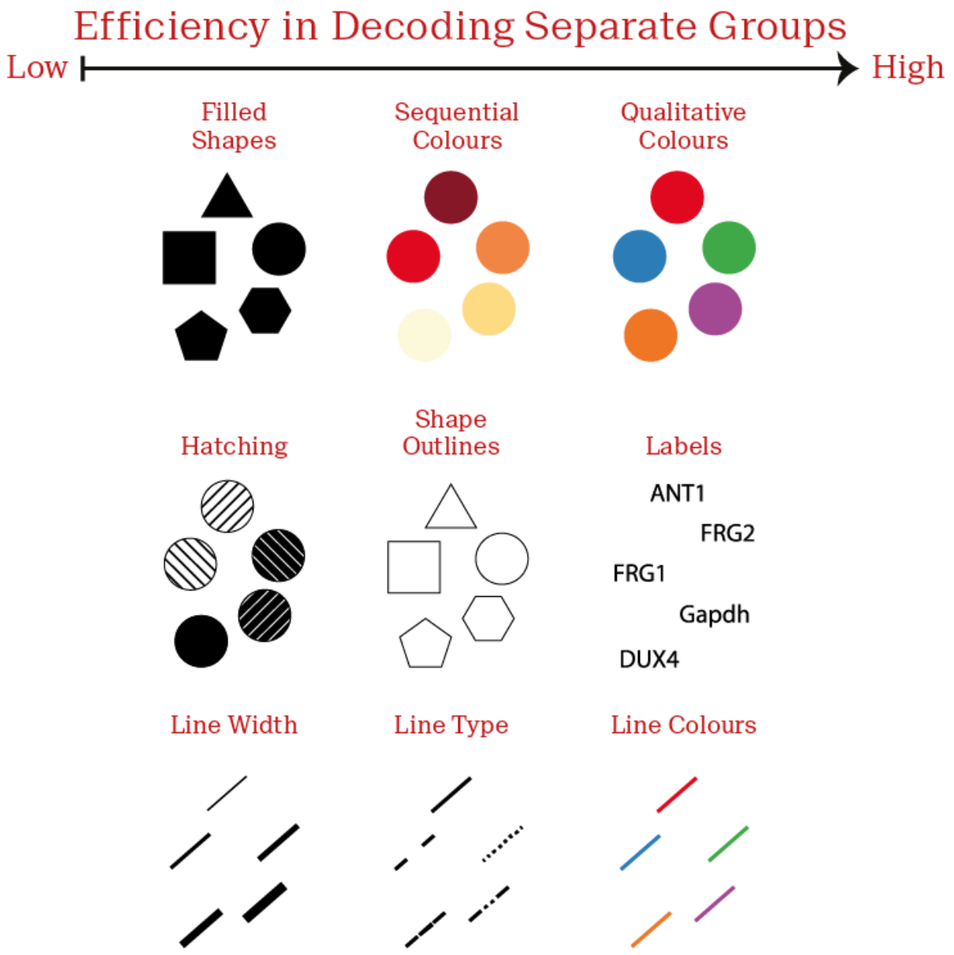


Color aesthetic

```
ggplot(fish, aes(x = Year,
                 y = Capture,
                 color = Species)) +
  geom_line()
```

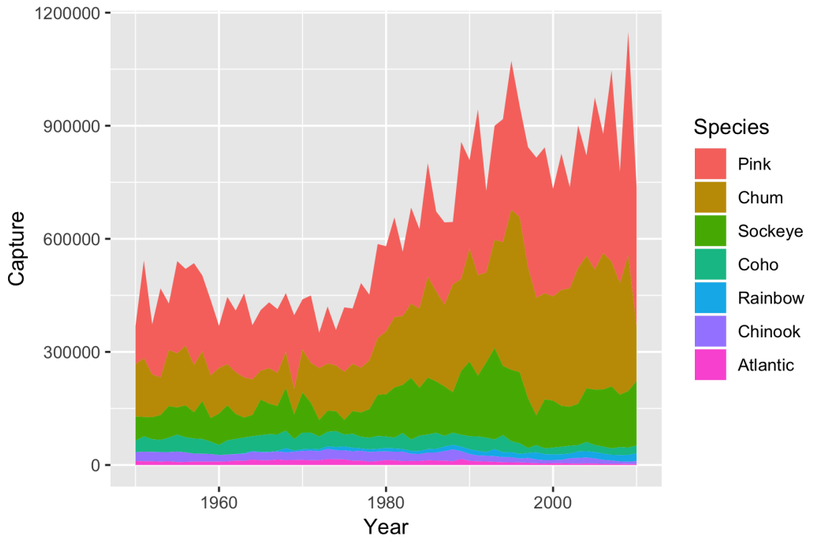


Aesthetics for categorical variables



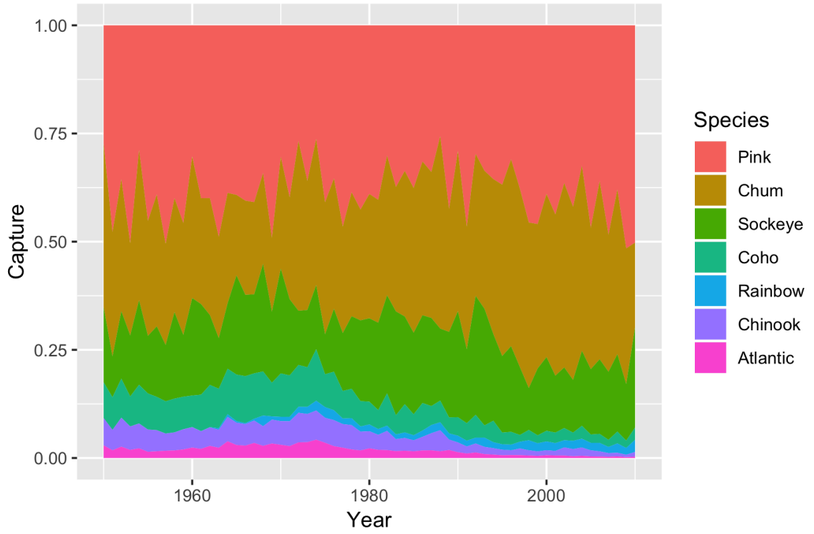
Fill aesthetic with geom_area()

```
ggplot(fish, aes(x = Year,
                 y = Capture,
                 fill = Species)) +
  geom_area()
```



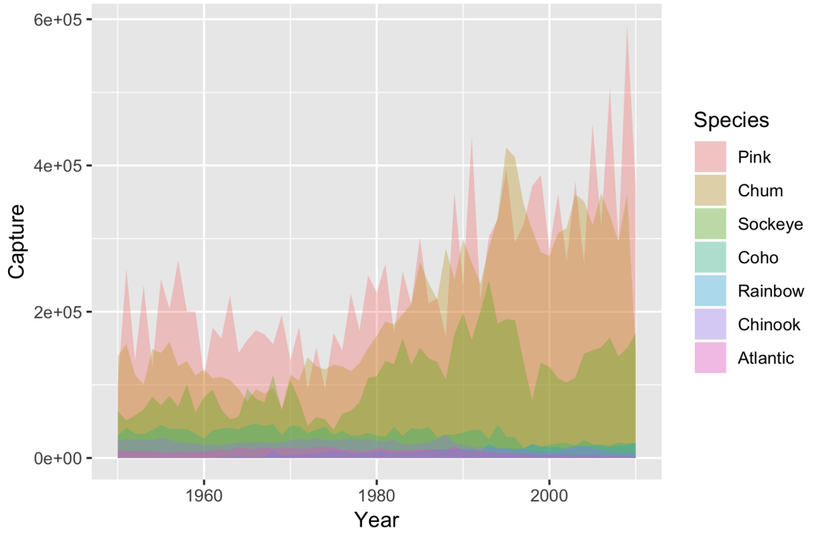
Using position = "fill"

```
ggplot(fish, aes(x = Year,
                 y = Capture,
                 fill = Species)) +
  geom_area(position = "fill")
```



geom_ribbon()

```
ggplot(fish, aes(x = Year,
                 y = Capture,
                 fill = Species)) +
  geom_ribbon(aes(ymax = Capture,
                  ymin = 0),
              alpha = 0.3)
```



Let's practice!

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2