# Visible aesthetics

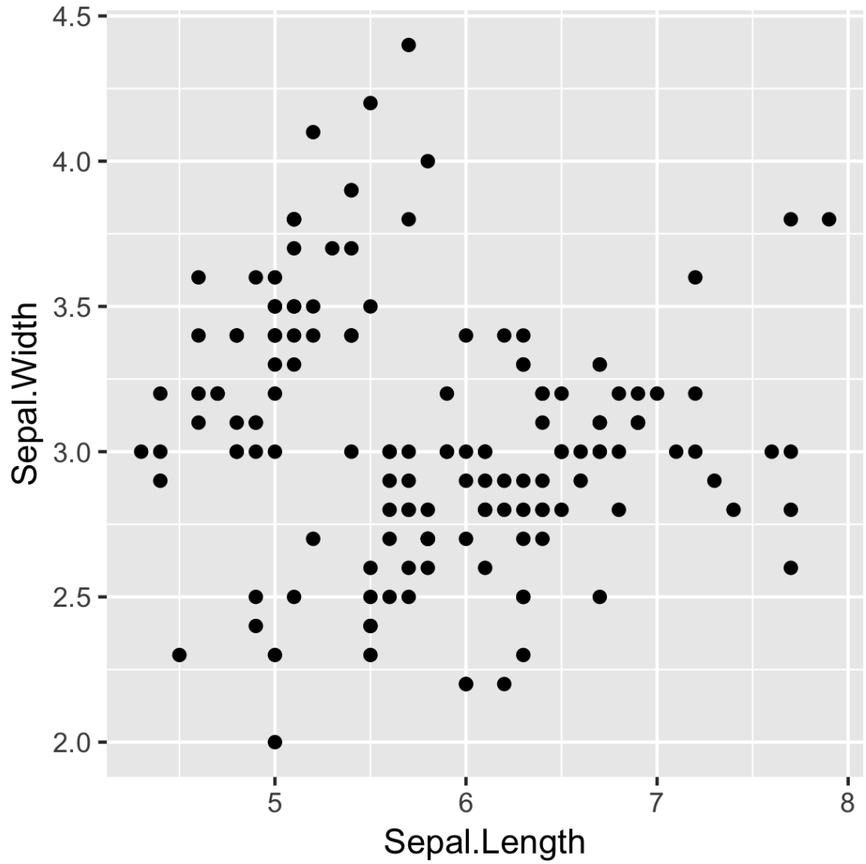
INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

**Rick Scavetta**
Founder, Scavetta Academy

DataCamp

# Mapping onto the X and Y axes

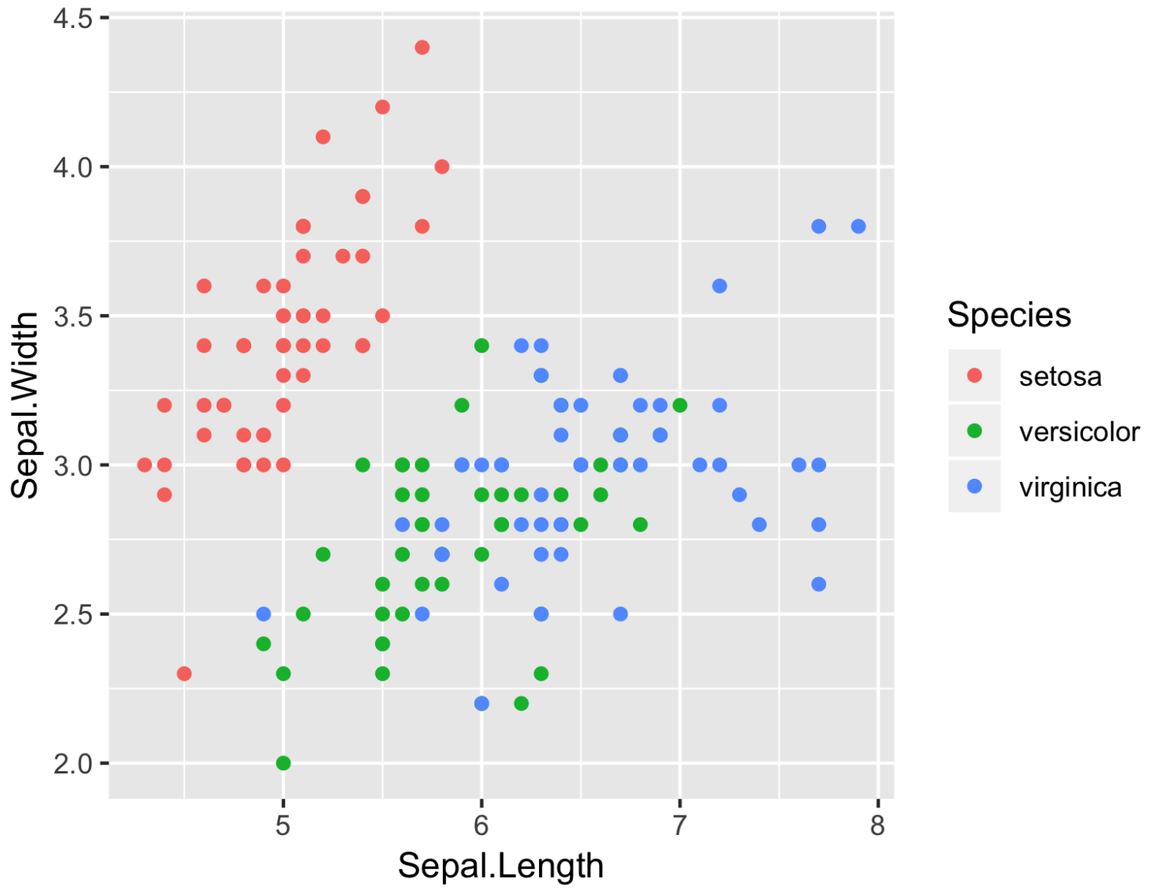
```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width)) +
  geom_point()
```



# Mapping onto color

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

| Type | Variable |
| --- | --- |
| Color | Species |

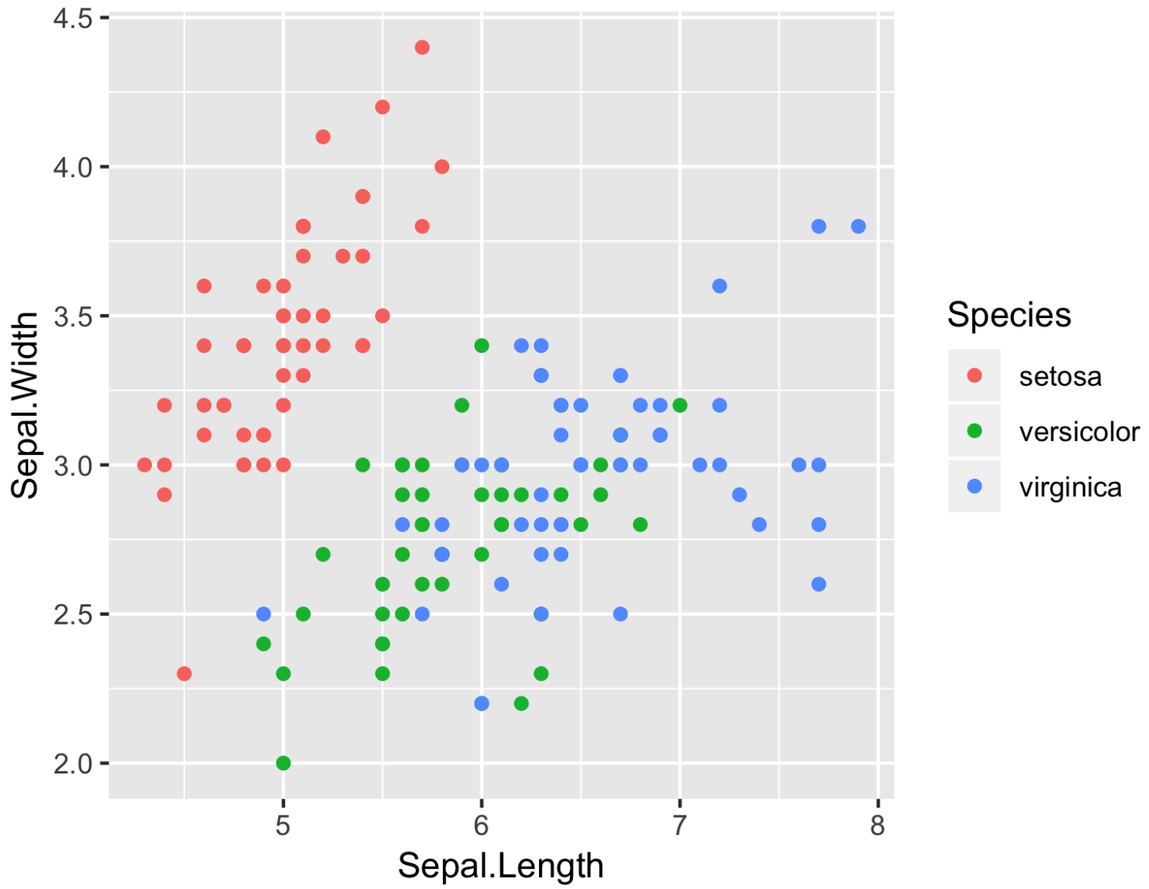


# Mapping onto the color aesthetic

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

| Type | Variable |
|---|---|
| Color | Species |

**Species**, a dataframe column, is *mapped onto* **color**, a visible aesthetic.



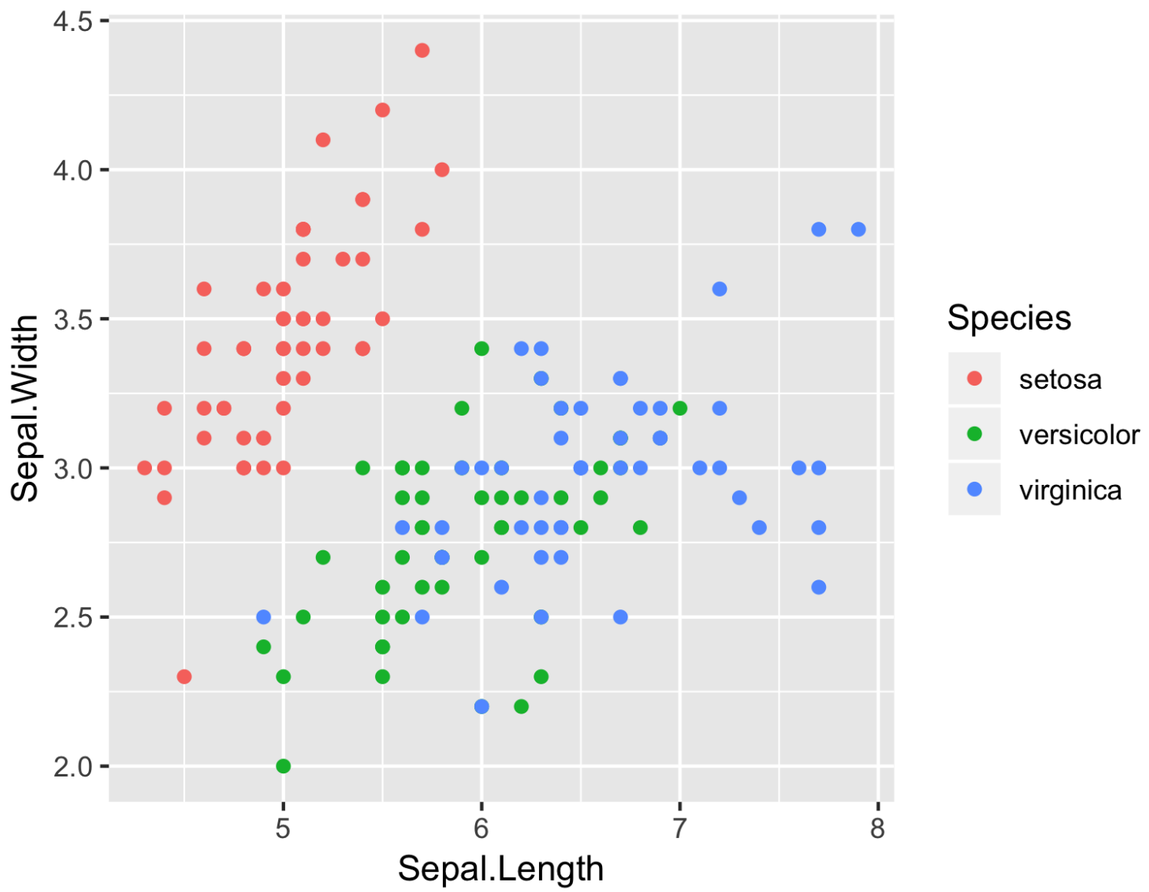
# Mapping onto the color aesthetic

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

| Type | Variable |
|---|---|
| Color | Species |

**Species**, a dataframe column, is *mapped onto* **color**, a visible aesthetic.

Map aesthetics in `aes()` .

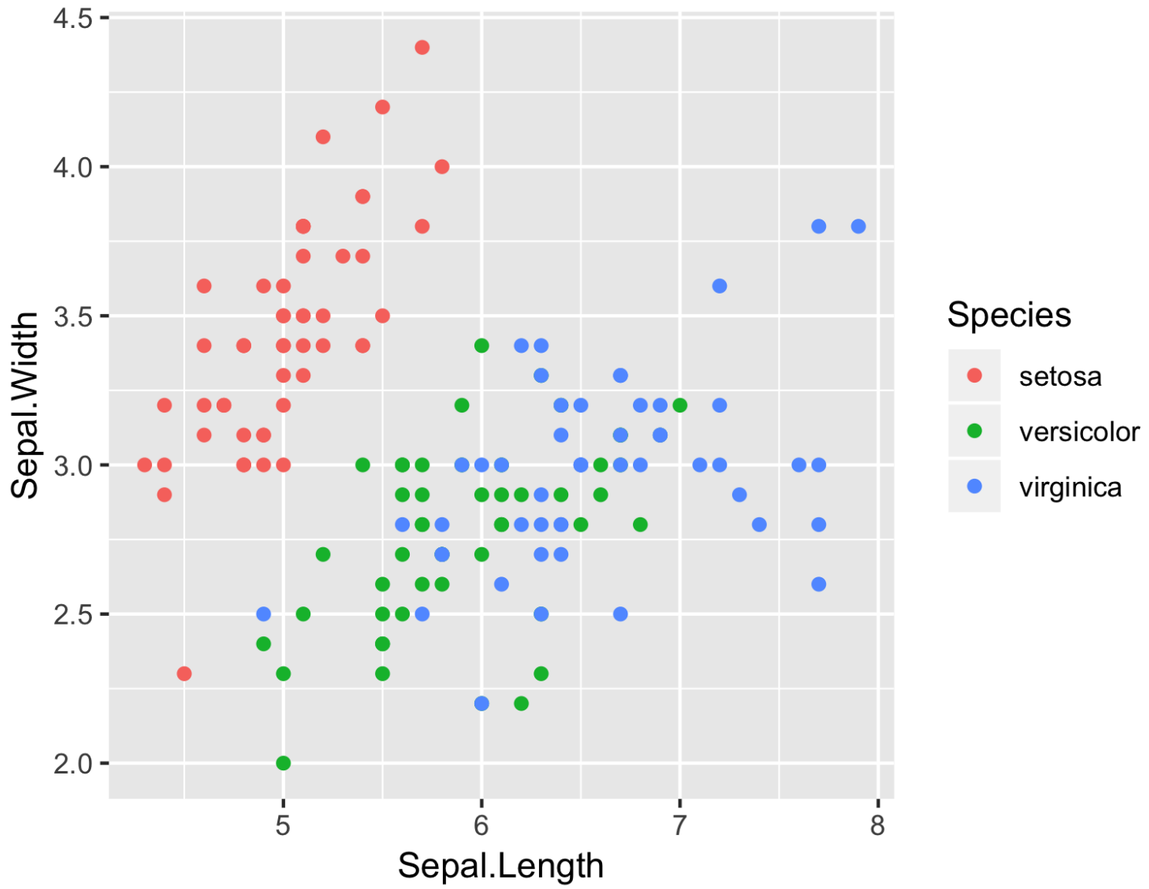


# Mapping onto the color aesthetic in geom

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

Only necessary if:

- All layers should *not* inherit the same aesthetics
- Mixing different data sources



# Typical visible aesthetics

| Aesthetic | Description |
| --- | --- |
| x | X axis position |
| y | Y axis position |

# Typical visible aesthetics

| Aesthetic | Description |
|---|---|
| x | X axis position |
| y | Y axis position |
| fill | Fill color |

# Typical visible aesthetics

| Aesthetic | Description |
|---|---|
| x | X axis position |
| y | Y axis position |
| fill | Fill color |
| color | Color of points, outlines of other geoms |

# Typical visible aesthetics

| Aesthetic | Description |
|---|---|
| x | X axis position |
| y | Y axis position |
| fill | Fill color |
| color | Color of points, outlines of other geoms |
| size | Area or radius of points, thickness of lines |

# Typical visible aesthetics

| Aesthetic | Description |
|---|---|
| x | X axis position |
| y | Y axis position |
| fill | Fill color |
| color | Color of points, outlines of other geoms |
| size | Area or radius of points, thickness of lines |

| Aesthetic | Description |
|---|---|
| alpha | Transparency |

# Typical visible aesthetics

| Aesthetic | Description |
|---|---|
| x | X axis position |
| y | Y axis position |
| fill | Fill color |
| color | Color of points, outlines of other geoms |
| size | Area or radius of points, thickness of lines |

| Aesthetic | Description |
|---|---|
| alpha | Transparency |
| linetype | Line dash pattern |

# Typical visible aesthetics

| Aesthetic | Description |
|---|---|
| x | X axis position |
| y | Y axis position |
| fill | Fill color |
| color | Color of points, outlines of other geoms |
| size | Area or radius of points, thickness of lines |

| Aesthetic | Description |
|---|---|
| alpha | Transparency |
| linetype | Line dash pattern |
| labels | Text on a plot or axes |

# Typical visible aesthetics

| Aesthetic | Description |
|---|---|
| x | X axis position |
| y | Y axis position |
| fill | Fill color |
| color | Color of points, outlines of other geoms |
| size | Area or radius of points, thickness of lines |

| Aesthetic | Description |
|---|---|
| alpha | Transparency |
| linetype | line dash pattern |
| labels | Text on a plot or axes |
| shape | Shape |

# Let's Practice

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

DataCamp

# Using attributes

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

**Rick Scavetta**
Founder, Scavetta Academy

DataCamp

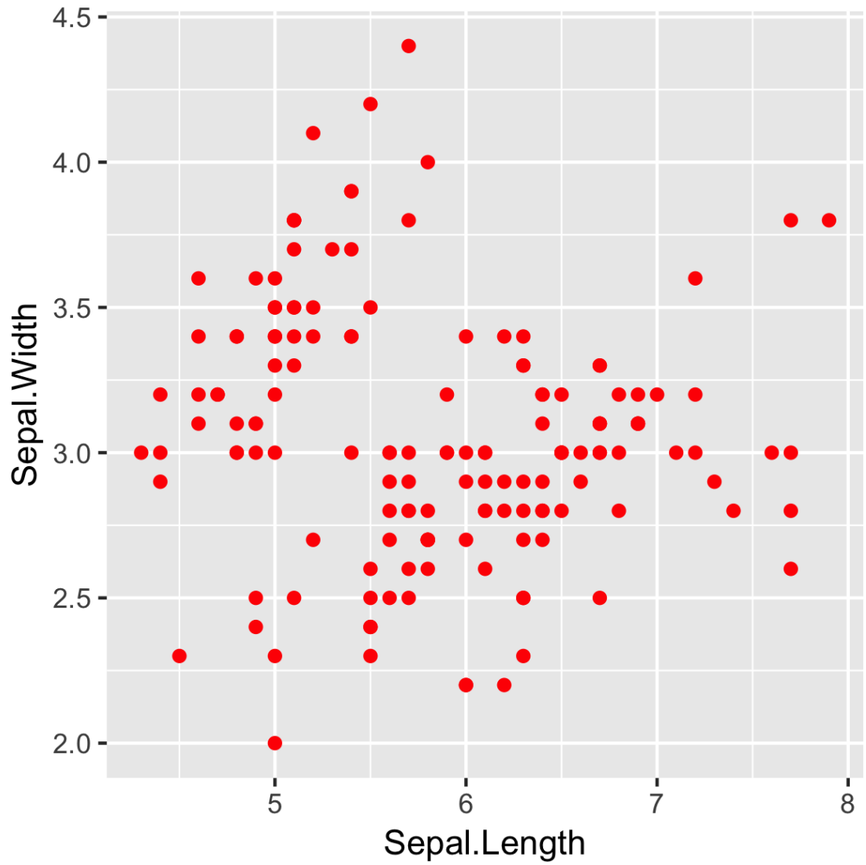
# Aesthetics? Attributes!

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

| Type | Property |
| --- | --- |
| Color | "red" |

Set attributes in `geom_*()` .

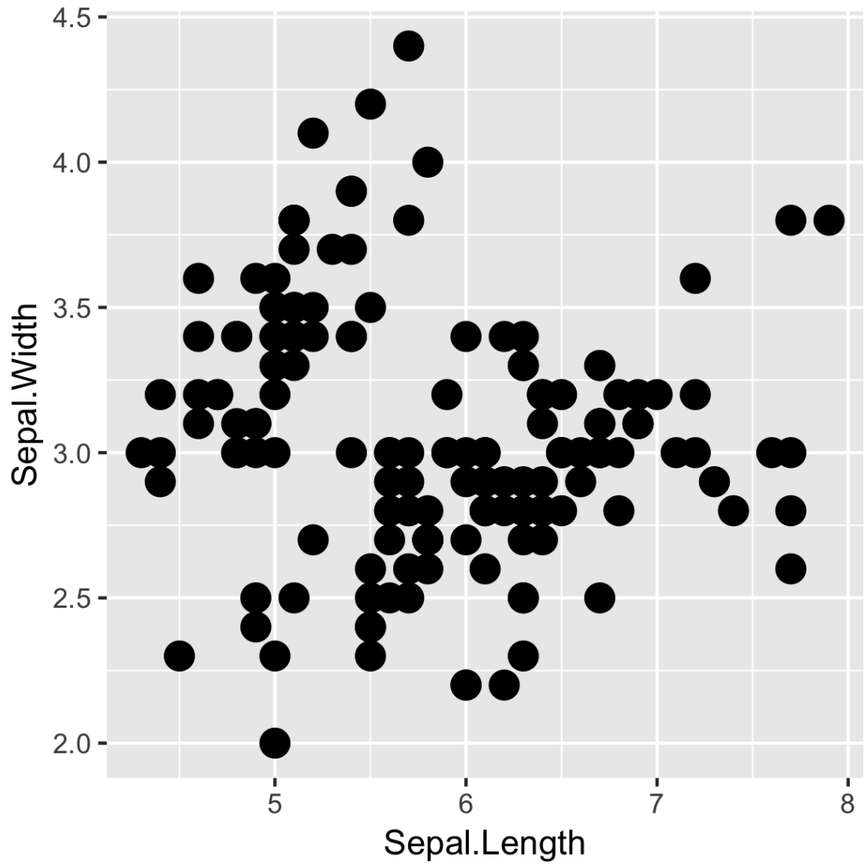
The **color** attribute is *set* to **"red"**.



# Aesthetics? Attributes!

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width)) +
  geom_point(size = 10)
```

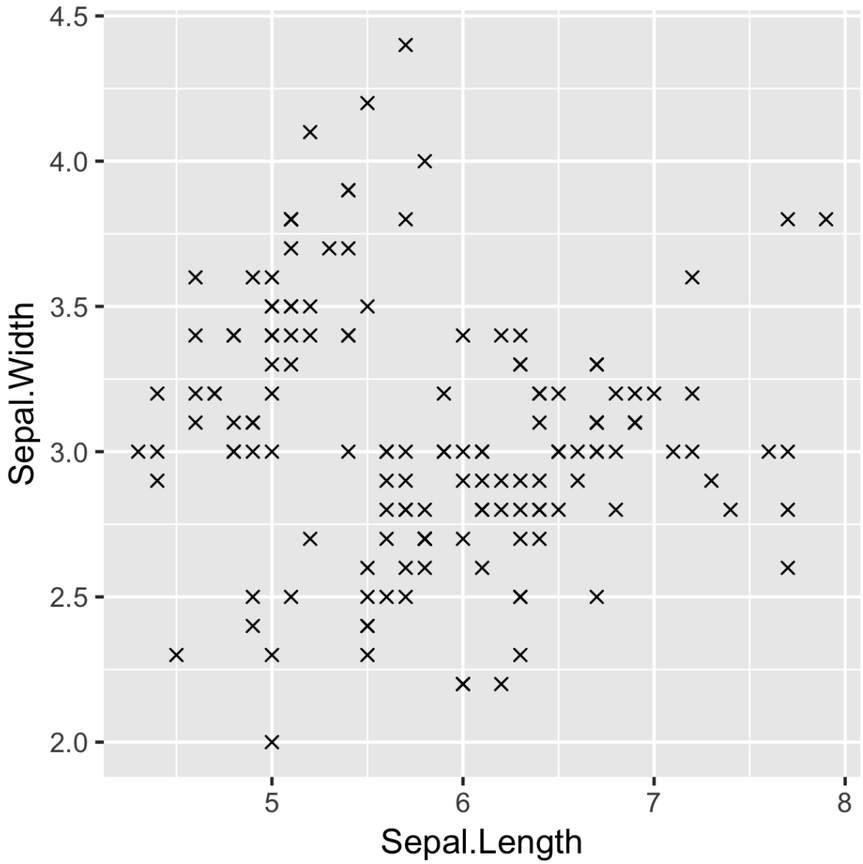
| Type | Property |
| --- | --- |
| Size | 4 |



# Aesthetics? Attributes!

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width)) +
  geom_point(shape = 4)
```

| Type | Property |
| --- | --- |
| Shape | 4 |



# Let's practice!

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

DataCamp

# Modifying Aesthetics

## INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

**Rick Scavetta**
Founder, Scavetta Academy

DataCamp

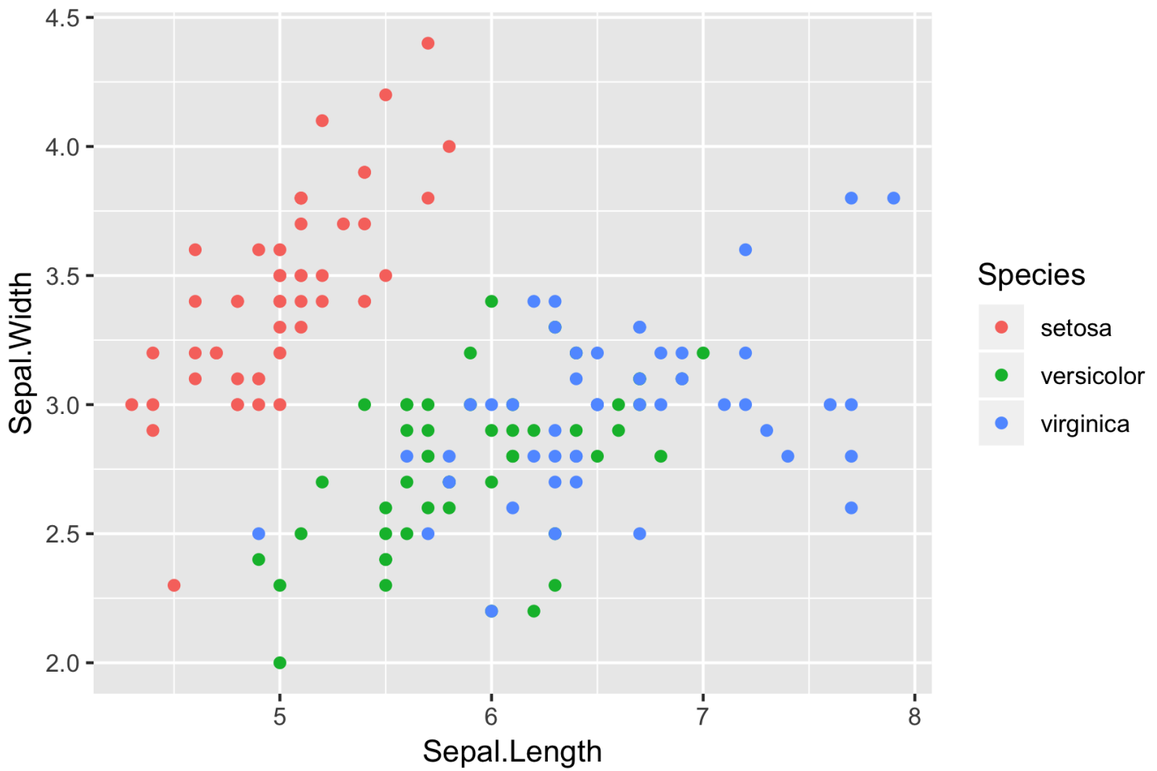
# Positions

Adjustment for overlapping

- identity
- dodge
- stack
- fill
- jitter
- jitterdodge
- nudge

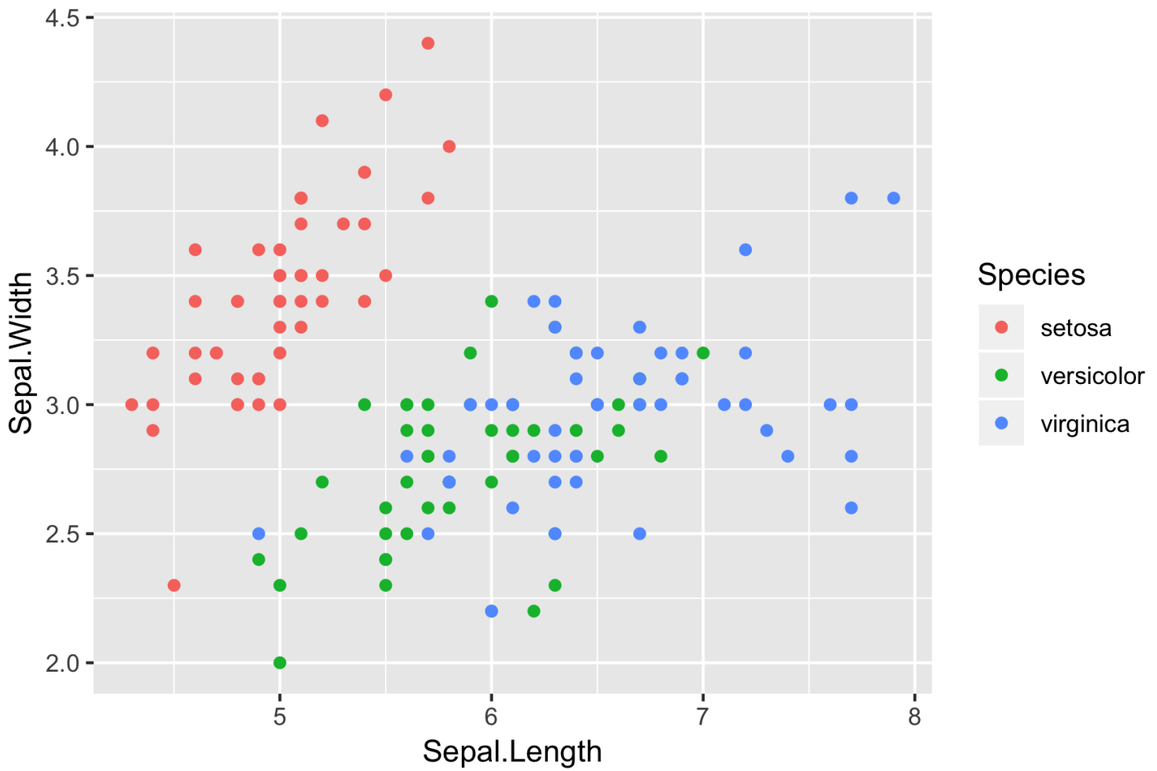
# position = "identity" (default)

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



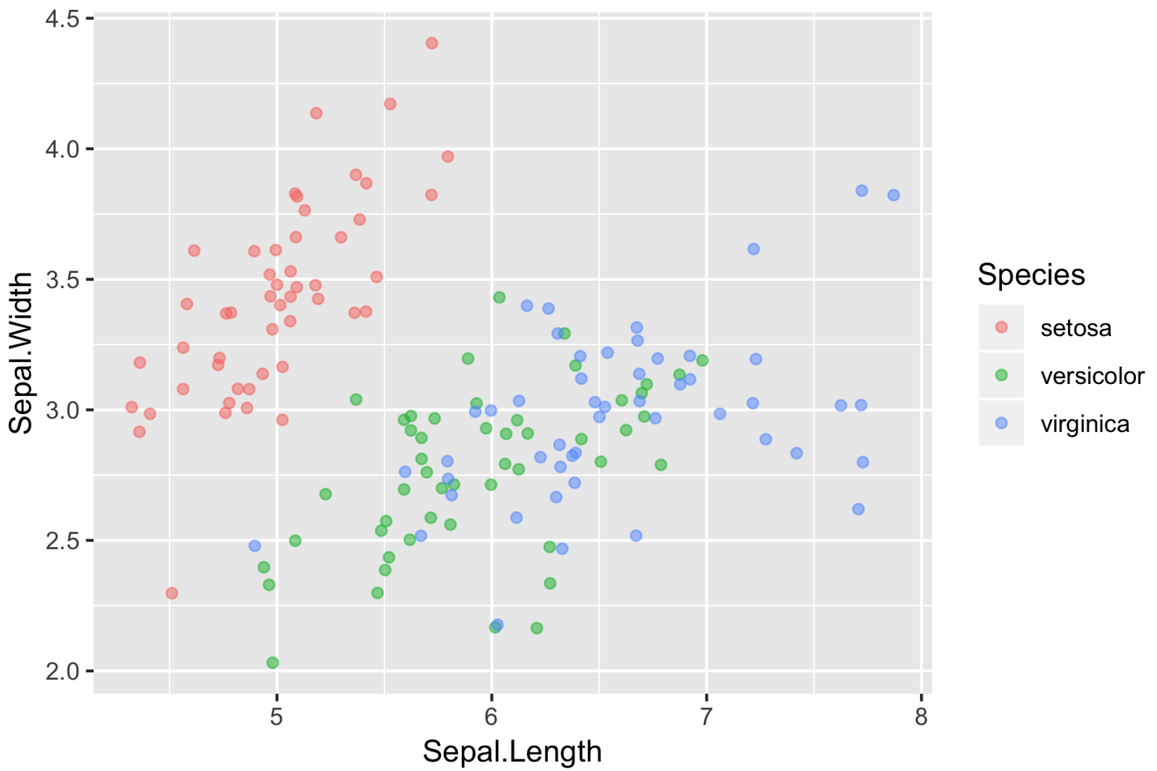
# position = "identity" (default)

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



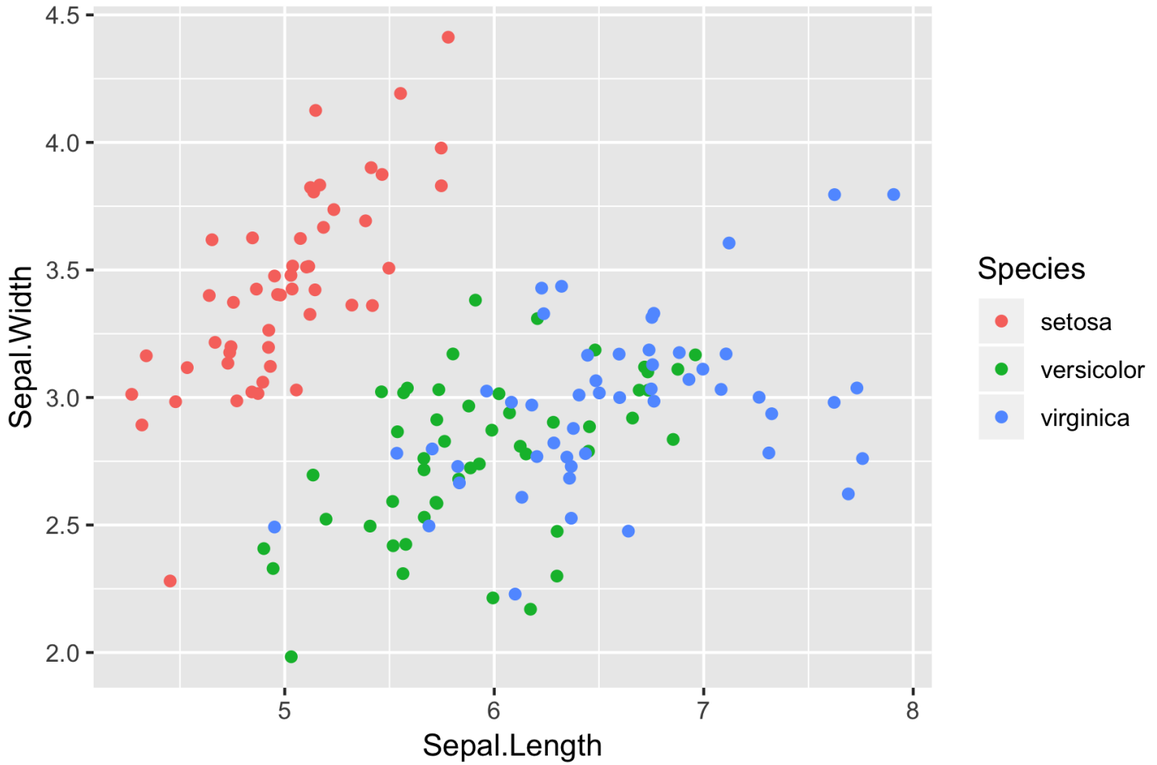
# position = "jitter"

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



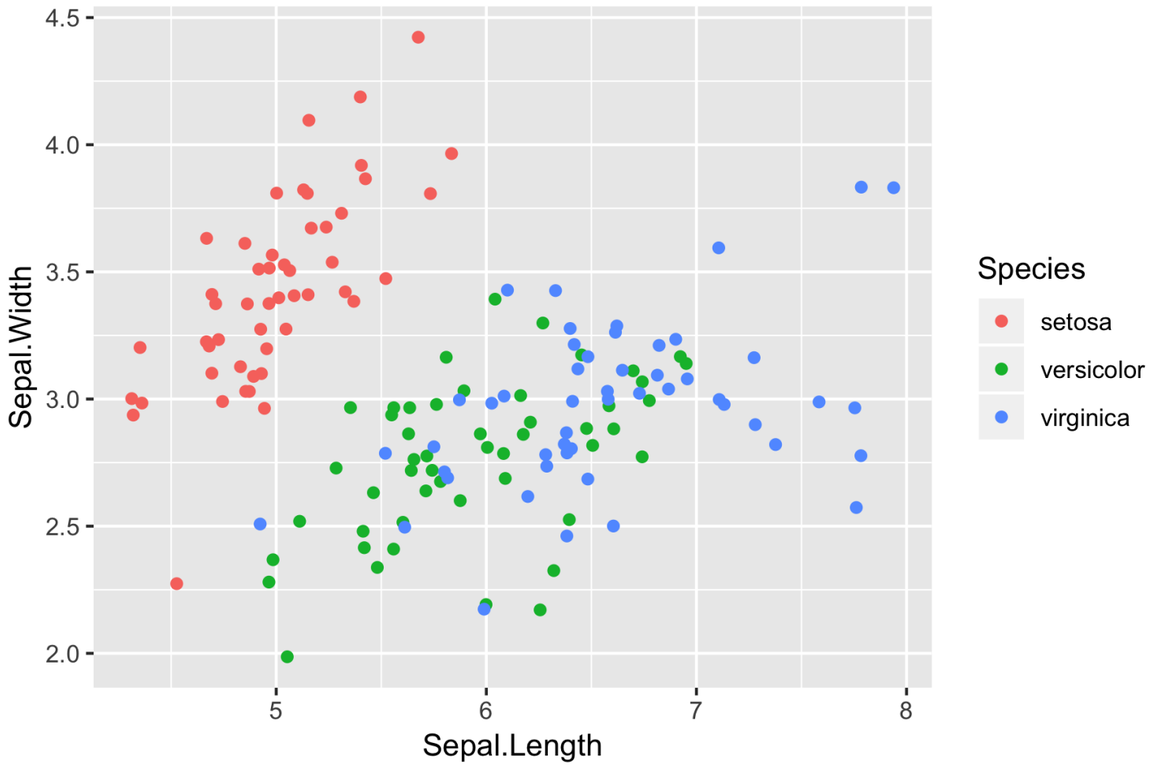
# position_jitter()

```
posn_j <- position_jitter(0.1)

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



# position_jitter()

```
posn_j <- position_jitter(0.1,
                          seed = 136)

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




- Set arguments for the position
- Consistency across plots & layers

# Scale functions

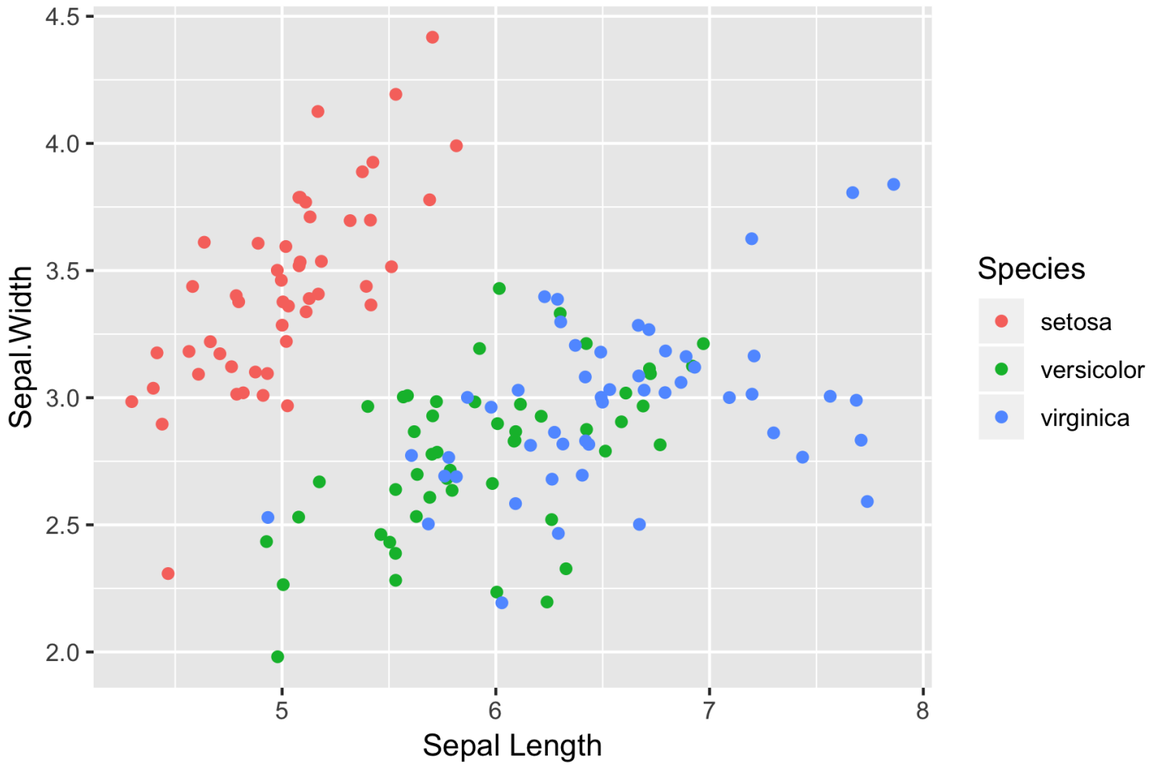
- `scale_x_*()`
- `scale_y_*()`
- `scale_color_*()`
  - Also `scale_colour_*()`
- `scale_fill_*()`
- `scale_shape_*()`
- `scale_linetype_*()`
- `scale_size_*()`

# Scale functions

- **`scale_x_continuous()`**
- `scale_y_*()`
- **`scale_color_discrete()`**
  - Alternatively, `scale_colour_*()`
- `scale_fill_*()`
- `scale_shape_*()`
- `scale_linetype_*()`
- `scale_size_*()`

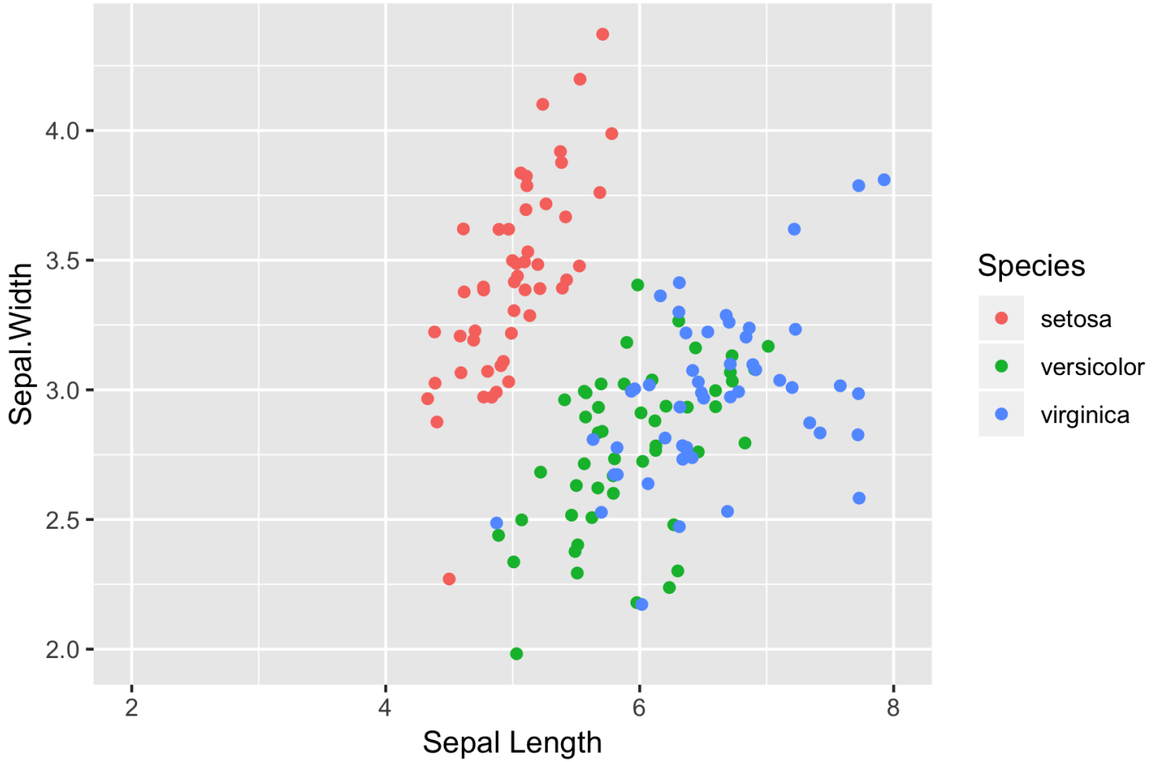
# scale_*_*()

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



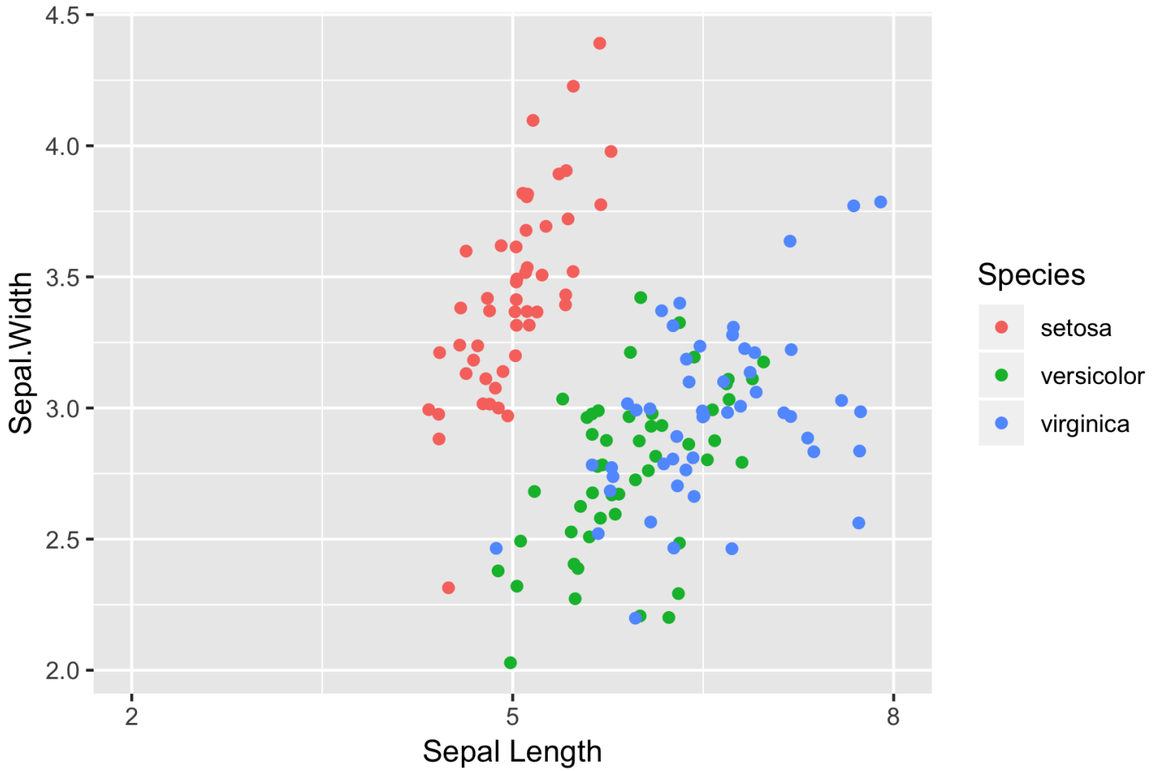
# The limits argument

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width,
                 color = Species)) +
  geom_point(position = "jitter") +
  scale_x_continuous("Sepal Length",
                     limits = c(2,8)) +
  scale_color_discrete("Species")
```



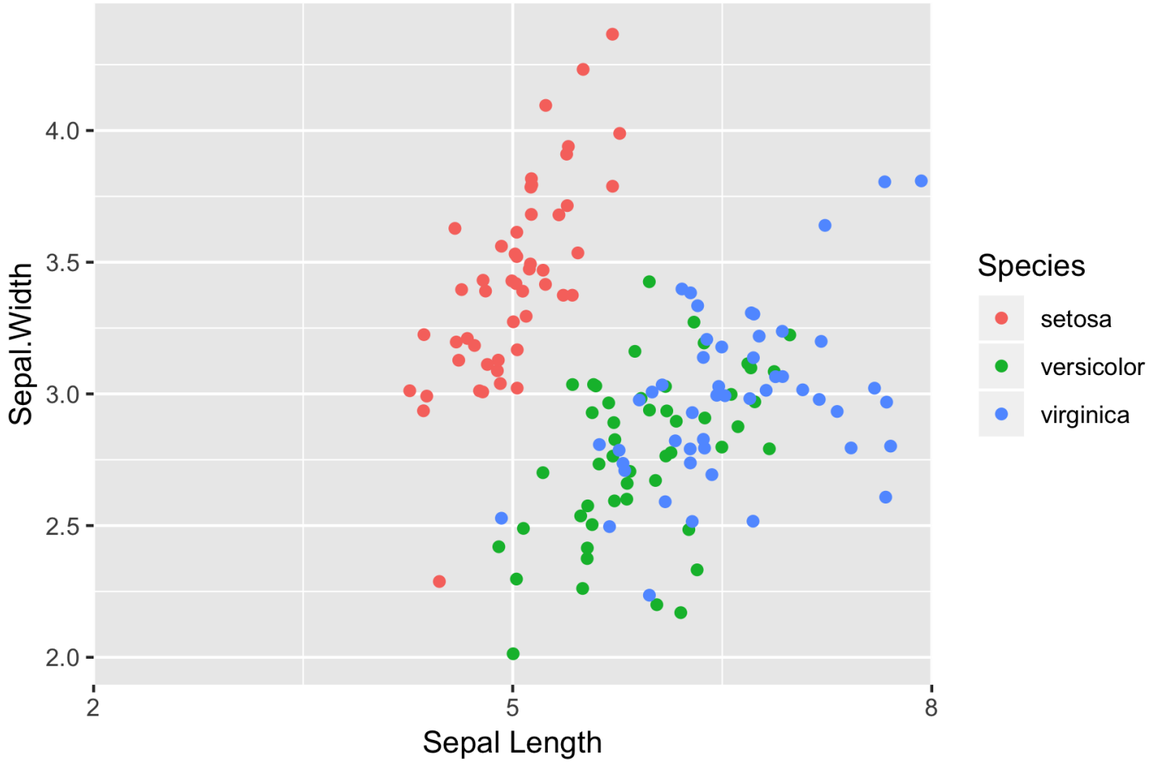
# The breaks argument

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width,
                 color = Species)) +
  geom_point(position = "jitter") +
  scale_x_continuous("Sepal Length",
                     limits = c(2, 8),
                     breaks = seq(2, 8, 3)) +
  scale_color_discrete("Species")
```



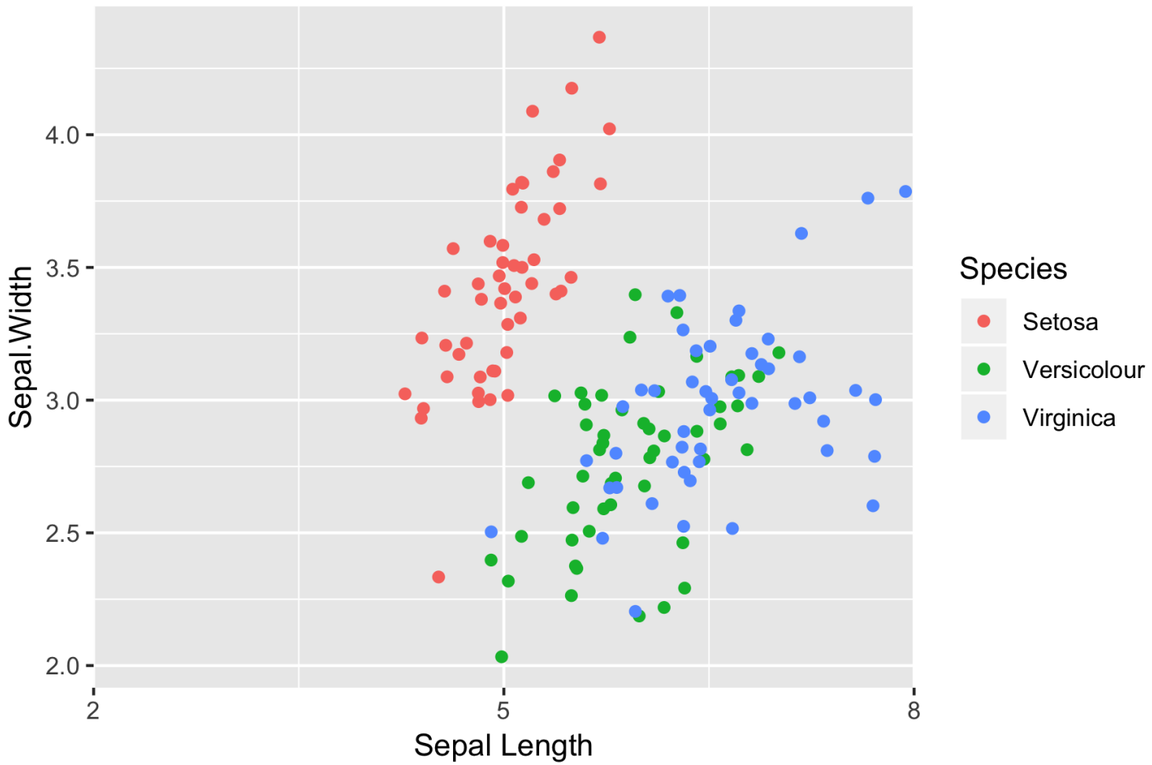
# The expand argument

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width,
                 color = Species)) +
  geom_point(position = "jitter") +
  scale_x_continuous("Sepal Length",
                     limits = c(2, 8),
                     breaks = seq(2, 8, 3),
                     expand = c(0, 0)) +
  scale_color_discrete("Species")
```



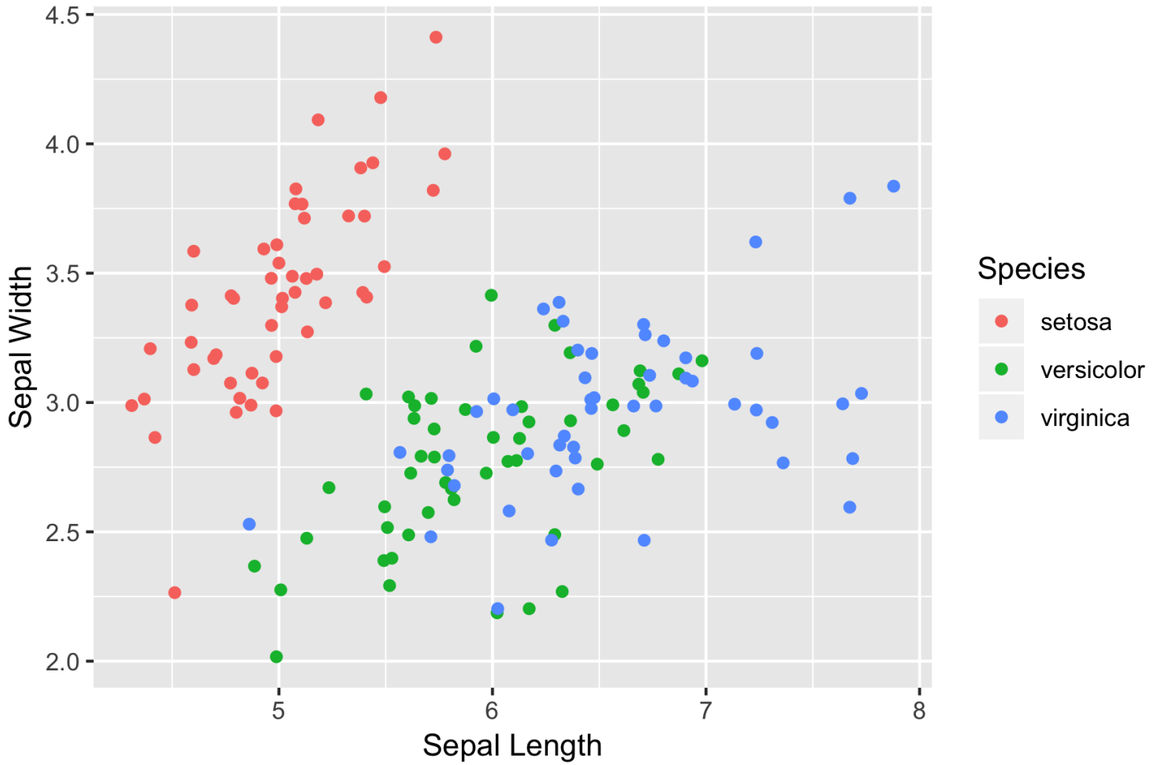
# The labels argument

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width,
                color = Species)) +
  geom_point(position = "jitter") +
  scale_x_continuous("Sepal Length",
                     limits = c(2, 8),
                     breaks = seq(2, 8, 3),
                     expand = c(0, 0),
                     labels = c("Setosa",
                                "Versicolor",
                                "Virginica")) +
  scale_color_discrete("Species")
```



# labs()

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



# Let's try it out!

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

DataCamp

# Aesthetics best practices

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

**Rick Scavetta**
Founder, Scavetta Academy

DataCamp

# Which aesthetics?

- Use your creative know-how, and
- Follow some clear guidelines
- Jacques Bertin
  - The Semiology of Graphics, 1967
- William Cleveland
  - The Elements of Graphing Data, 1985
  - Visualizing Data, 1993

# Form follows function

# Form follows function

## Function

Primary:

- Accurate and efficient representations

Secondary:

- Visually appealing, beautiful plots

## Guiding principles

Never:

- Misrepresent or obscure data
- Confuse viewers with complexity

Always:

- Consider the audience and purpose of every plot

| x | y = f(x) | Group |
|---|---|---|
| 5.1 | 3.5 | A |
| 4.9 | 3.0 | A |
| 4.7 | 3.2 | A |
| 4.6 | 3.1 | A |
| ⋮ | ⋮ | ⋮ |
| 7.0 | 3.2 | B |
| 6.4 | 3.2 | B |
| 6.9 | 3.1 | B |
| 5.5 | 2.3 | B |
| ⋮ | ⋮ | ⋮ |

| x | y = f(x) | Group |
|---|---|---|
| 5.1 | 3.5 | A |
| 4.9 | 3.0 | A |
| 4.7 | 3.2 | A |
| 4.6 | 3.1 | A |
| ⋮ | ⋮ | ⋮ |
| 7.0 | 3.2 | B |
| 6.4 | 3.2 | B |
| 6.9 | 3.1 | B |
| 5.5 | 2.3 | B |
| ⋮ | ⋮ | ⋮ |

Diffucult directly from data

For each group:

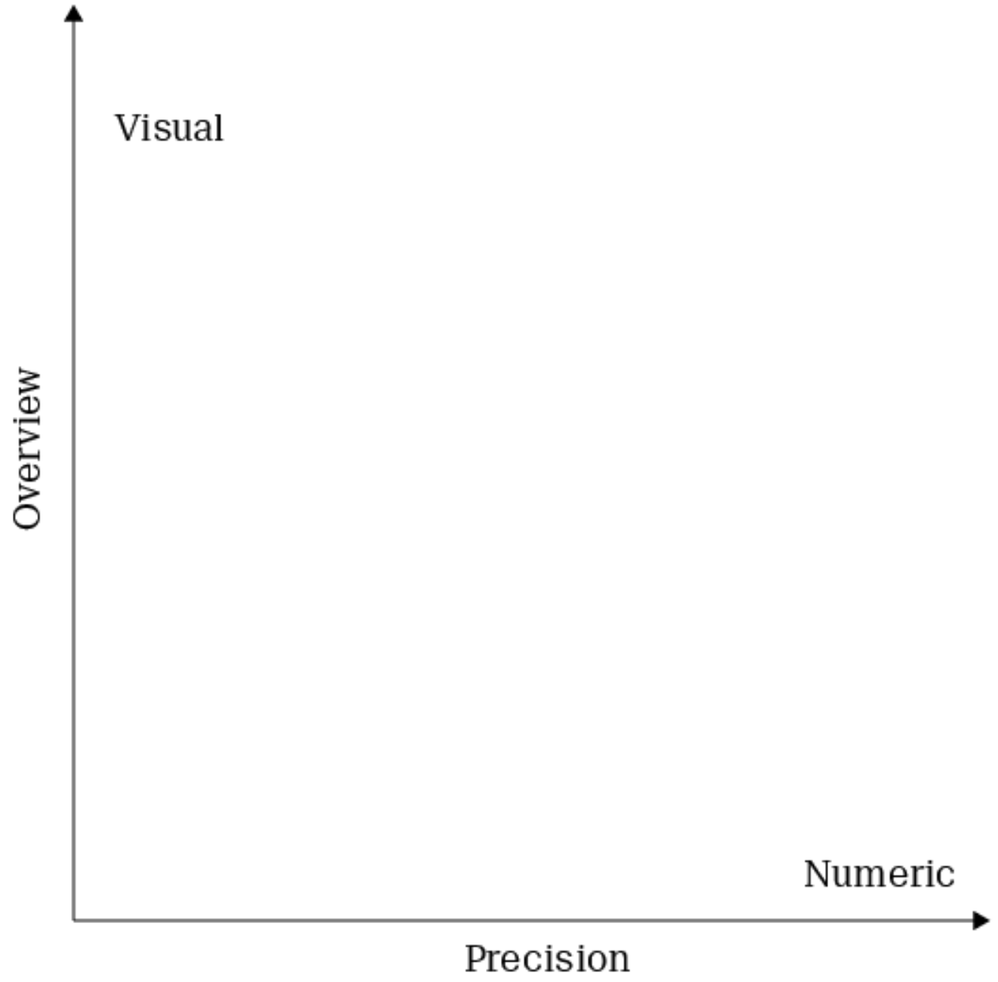
$n$ range outliers

$\bar{x}$ $s_x$ $\bar{y}$ $s_y$

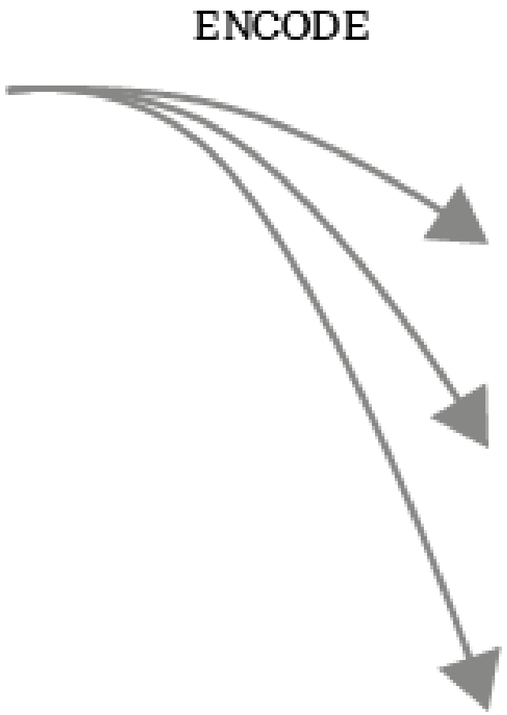
$y = f(x) = \beta_0 + \beta_1 x$

$r$ $R^2$

# Extracting information from Data



| x | y = f(x) | Group |
|---|---|---|
| 5.1 | 3.5 | A |
| 4.9 | 3.0 | A |
| 4.7 | 3.2 | A |
| 4.6 | 3.1 | A |
| ⋮ | ⋮ | ⋮ |
| 7.0 | 3.2 | B |
| 6.4 | 3.2 | B |
| 6.9 | 3.1 | B |
| 5.5 | 2.3 | B |
| ⋮ | ⋮ | ⋮ |




For each group:

$n$ range outliers

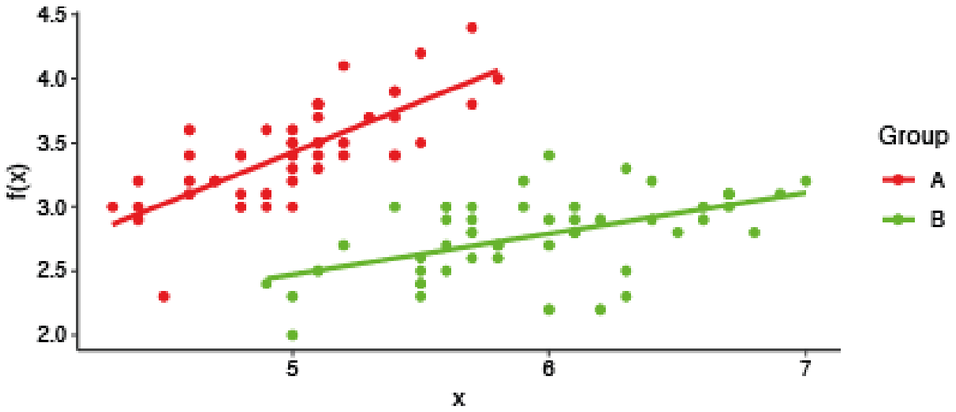
$\bar{x}$ $s_x$ $\bar{y}$ $s_y$

$y = f(x) = \beta_0 + \beta_1 x$

$r$ $R^2$

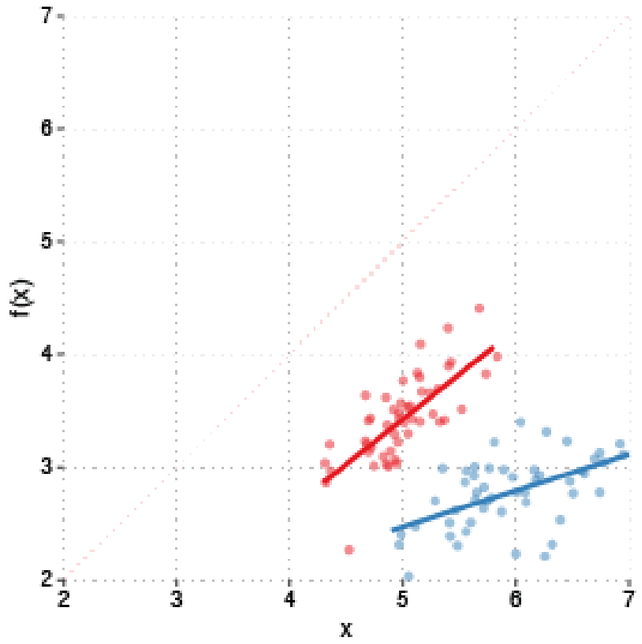
| x | y = f(x) | Group |
|---|---|---|
| 5.1 | 3.5 | A |
| 4.9 | 3.0 | A |
| 4.7 | 3.2 | A |
| 4.6 | 3.1 | A |
| ⋮ | ⋮ | ⋮ |
| 7.0 | 3.2 | B |
| 6.4 | 3.2 | B |
| 6.9 | 3.1 | B |
| 5.5 | 2.3 | B |
| ⋮ | ⋮ | ⋮ |

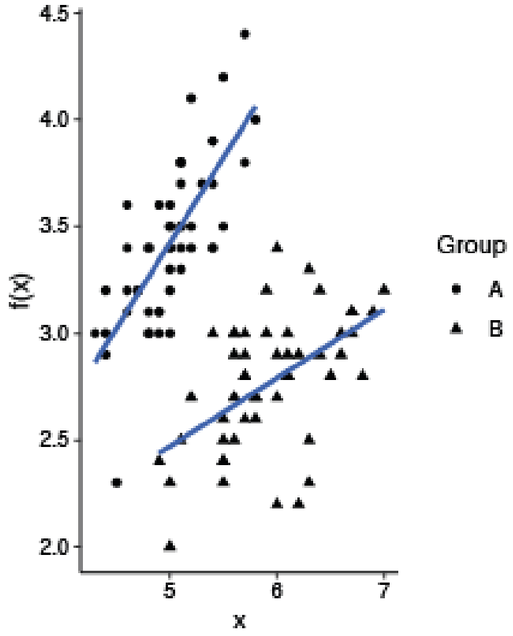
ENCODE




For each group:

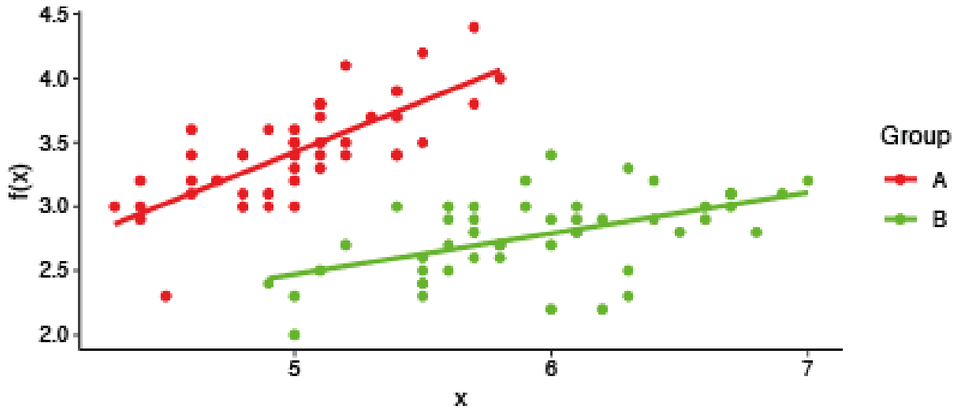
- $n$ range outliers
- $\bar{x}$ $s_x$ $\bar{y}$ $s_y$
- $y = f(x) = \beta_0 + \beta_1 x$
- $r$ $R^2$



| x | y = f(x) | Group |
|---|---|---|
| 5.1 | 3.5 | A |
| 4.9 | 3.0 | A |
| 4.7 | 3.2 | A |
| 4.6 | 3.1 | A |
| ⋮ | ⋮ | ⋮ |
| 7.0 | 3.2 | B |
| 6.4 | 3.2 | B |
| 6.9 | 3.1 | B |
| 5.5 | 2.3 | B |
| ⋮ | ⋮ | ⋮ |

ENCODE




For each group:

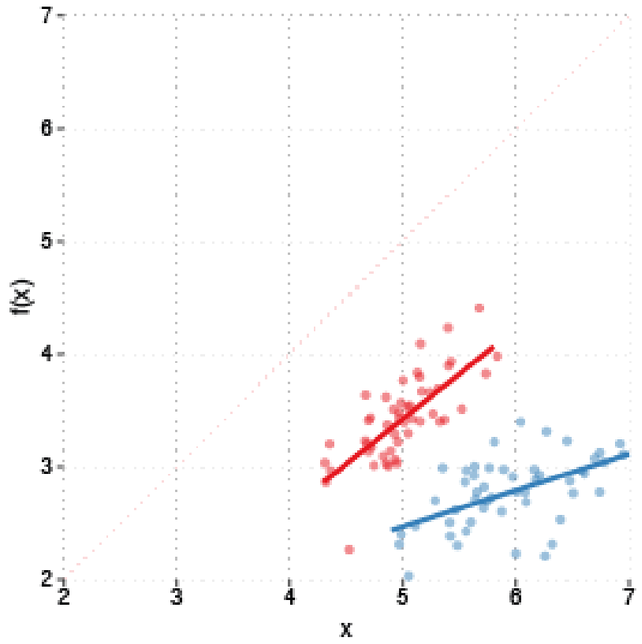
$n$ range outliers

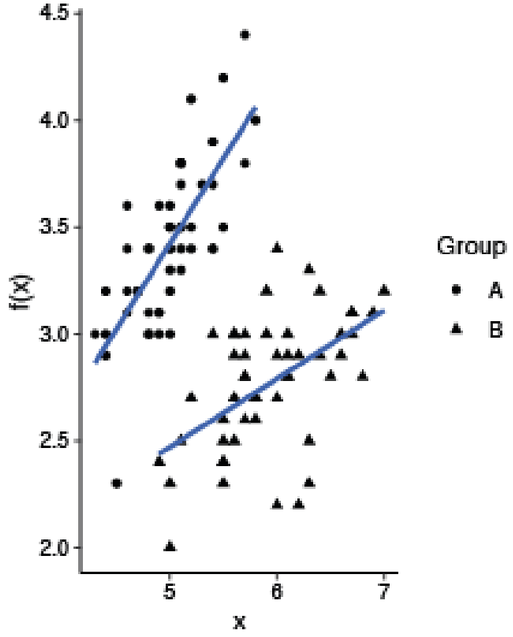
$\bar{x}$ $s_x$ $\bar{y}$ $s_y$

$y = f(x) = \beta_0 + \beta_1 x$

$r$ $R^2$

DECODE



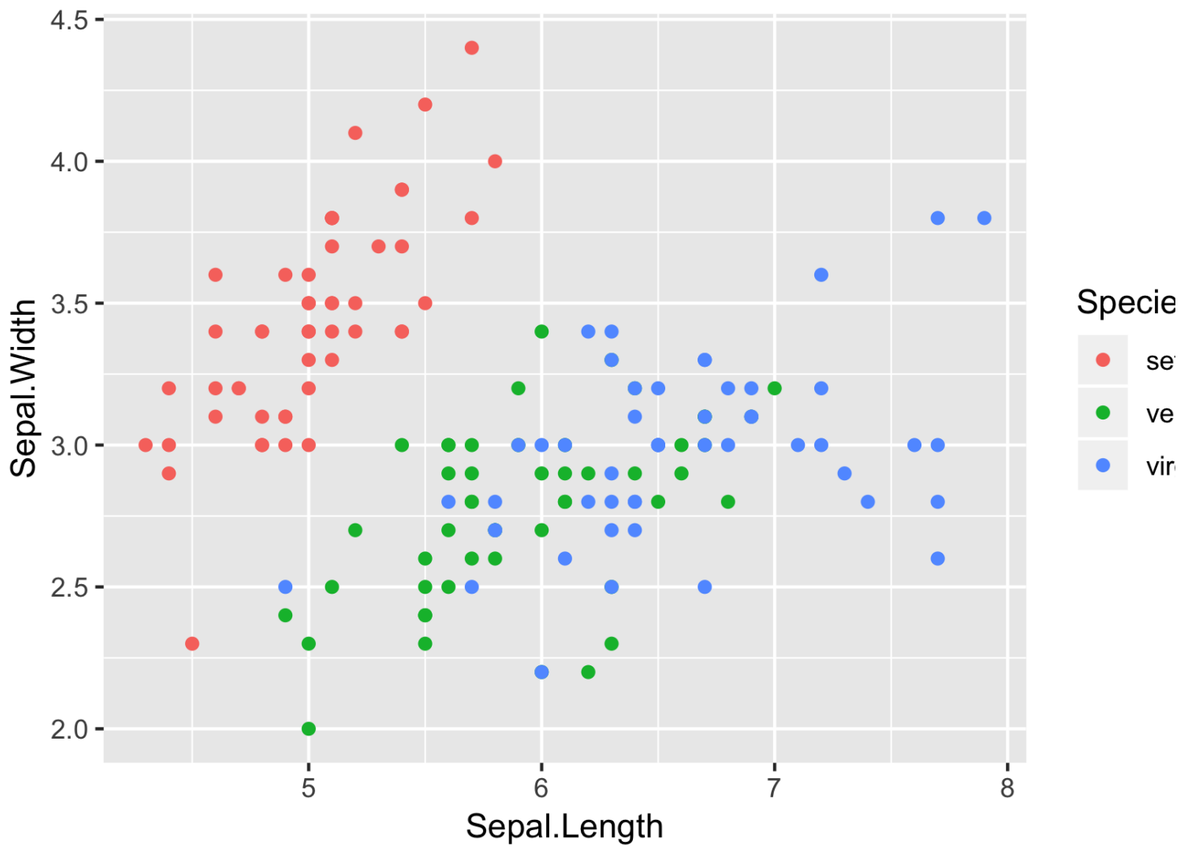



# The best choices for aesthetics

- Efficient
  - Provides a faster overview than numeric summaries
- Accurate
  - Minimizes information loss

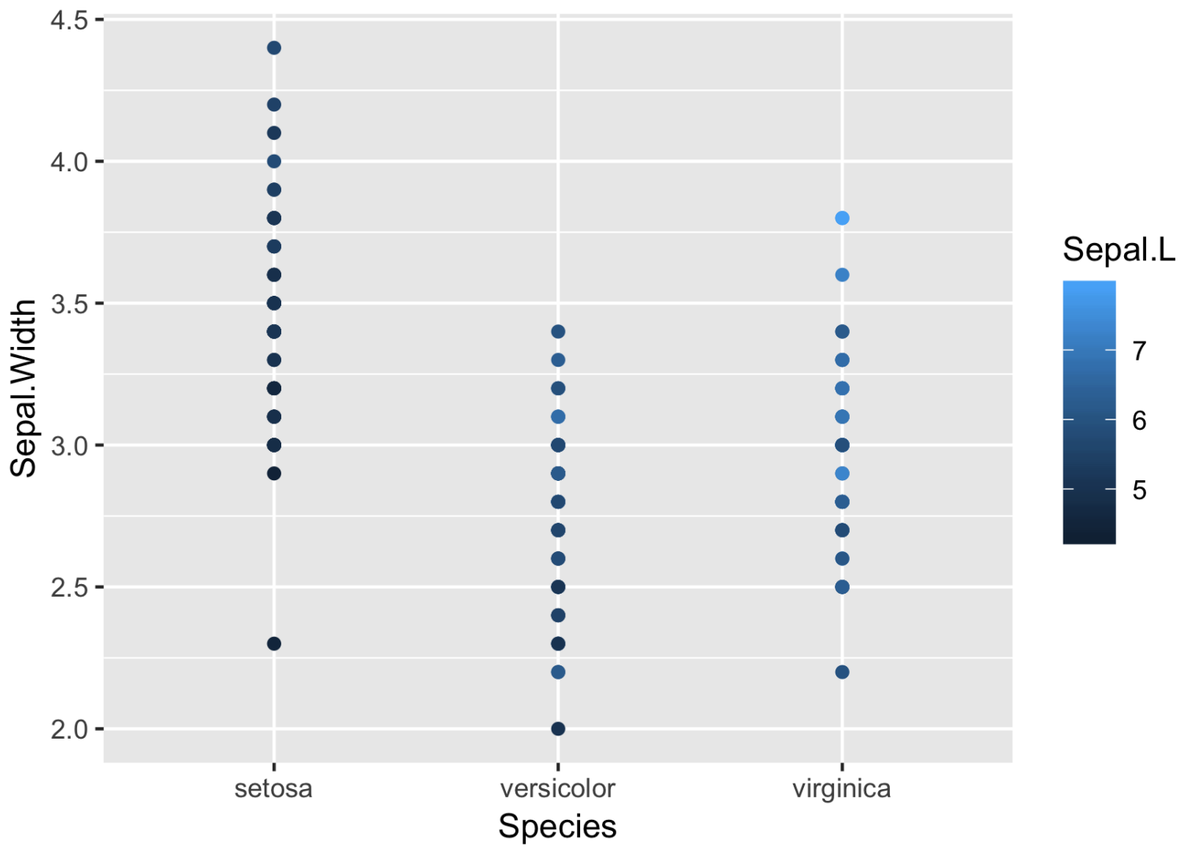
# Aesthetics - continuous variables

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

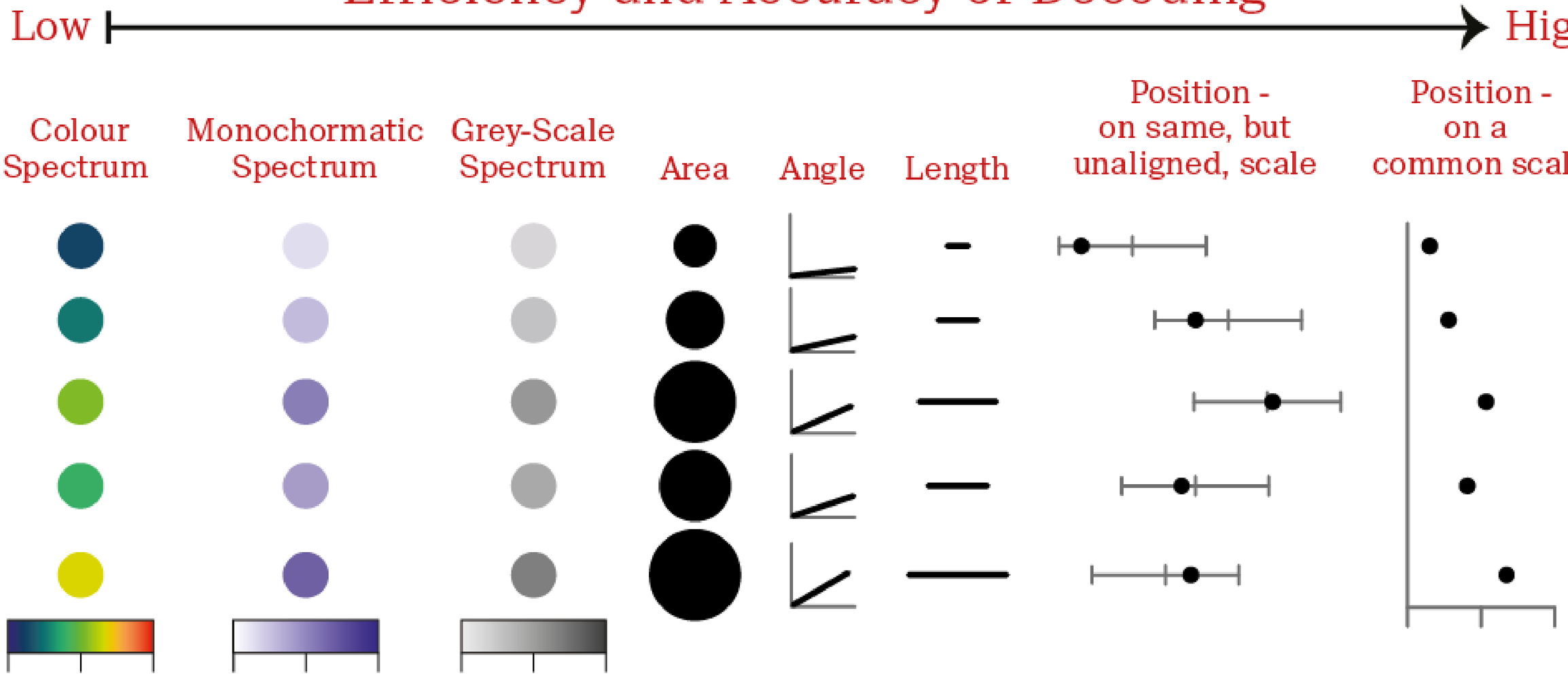


# Aesthetics - continuous variables

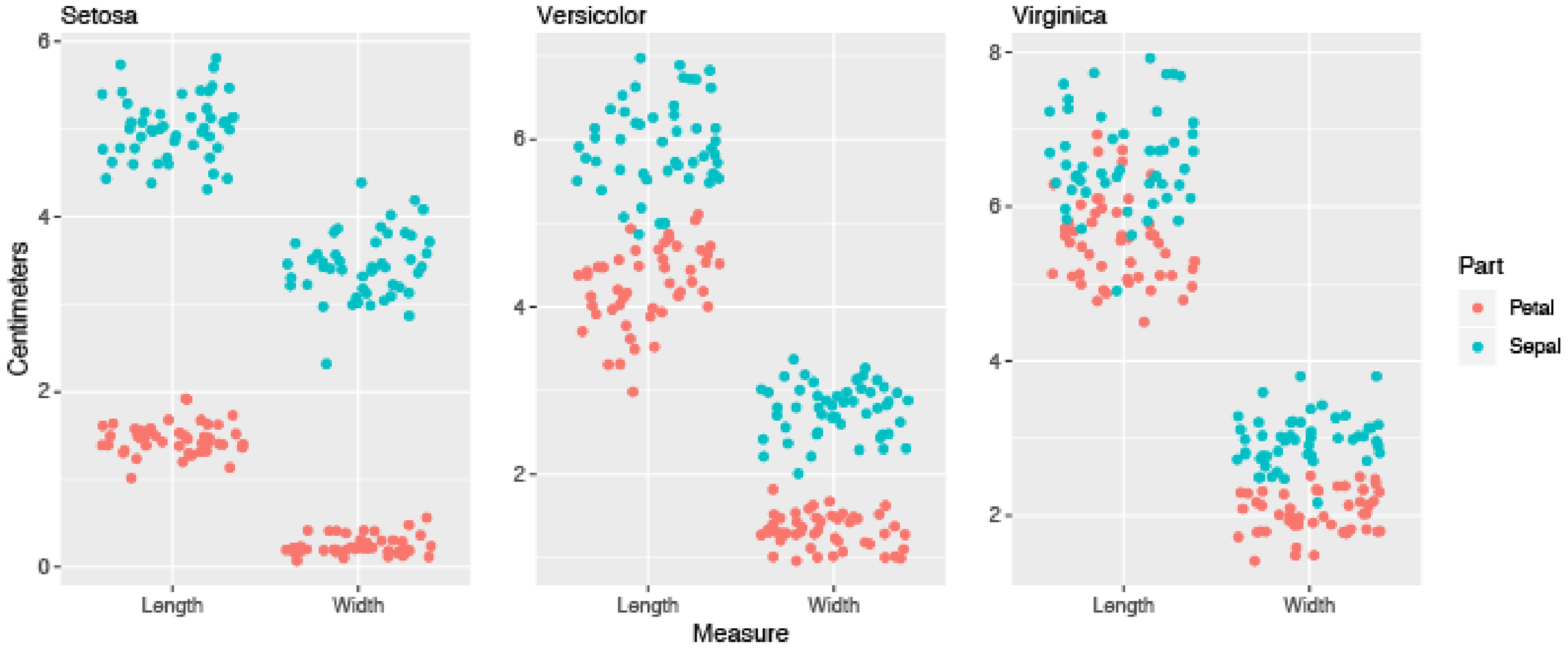
```
ggplot(iris, aes(color = Sepal.Length,
                 y = Sepal.Width,
                 x = Species)) +
  geom_point()
```



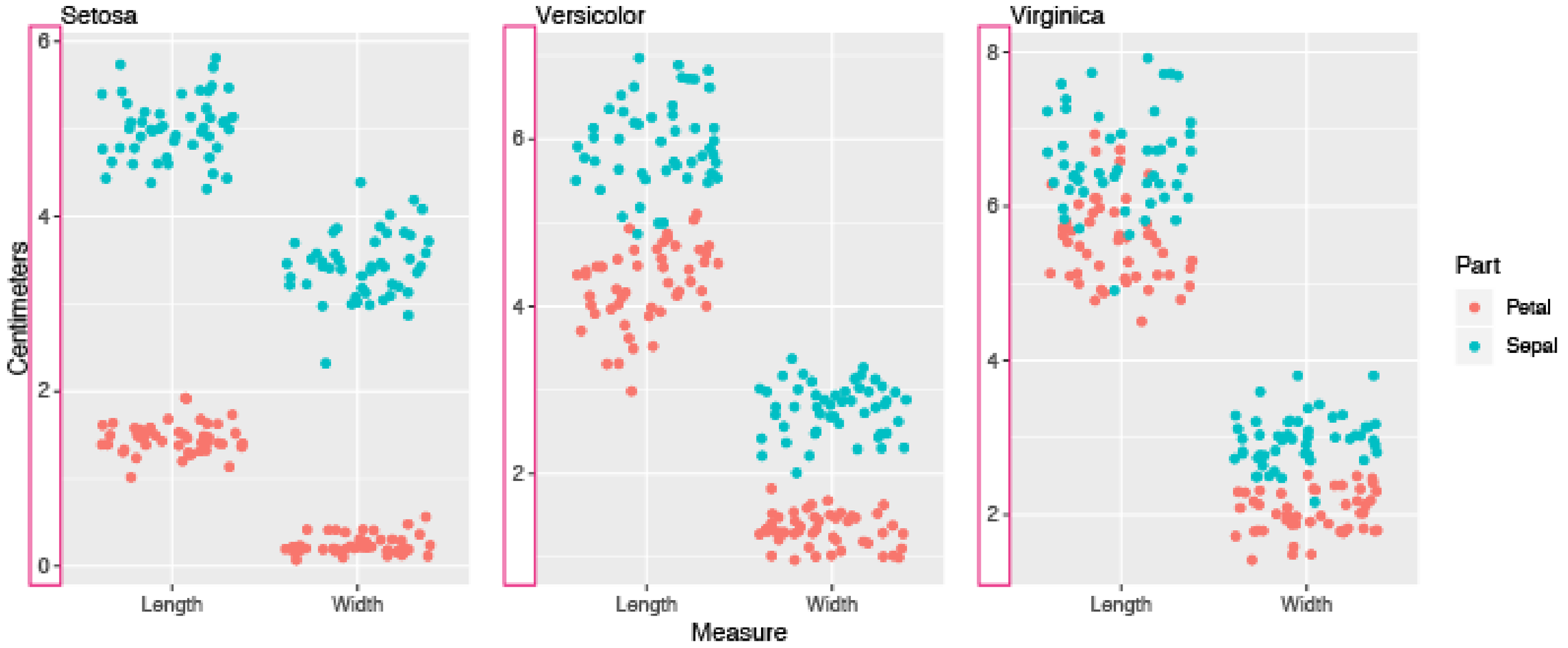
# Efficiency and Accuracy of Decoding



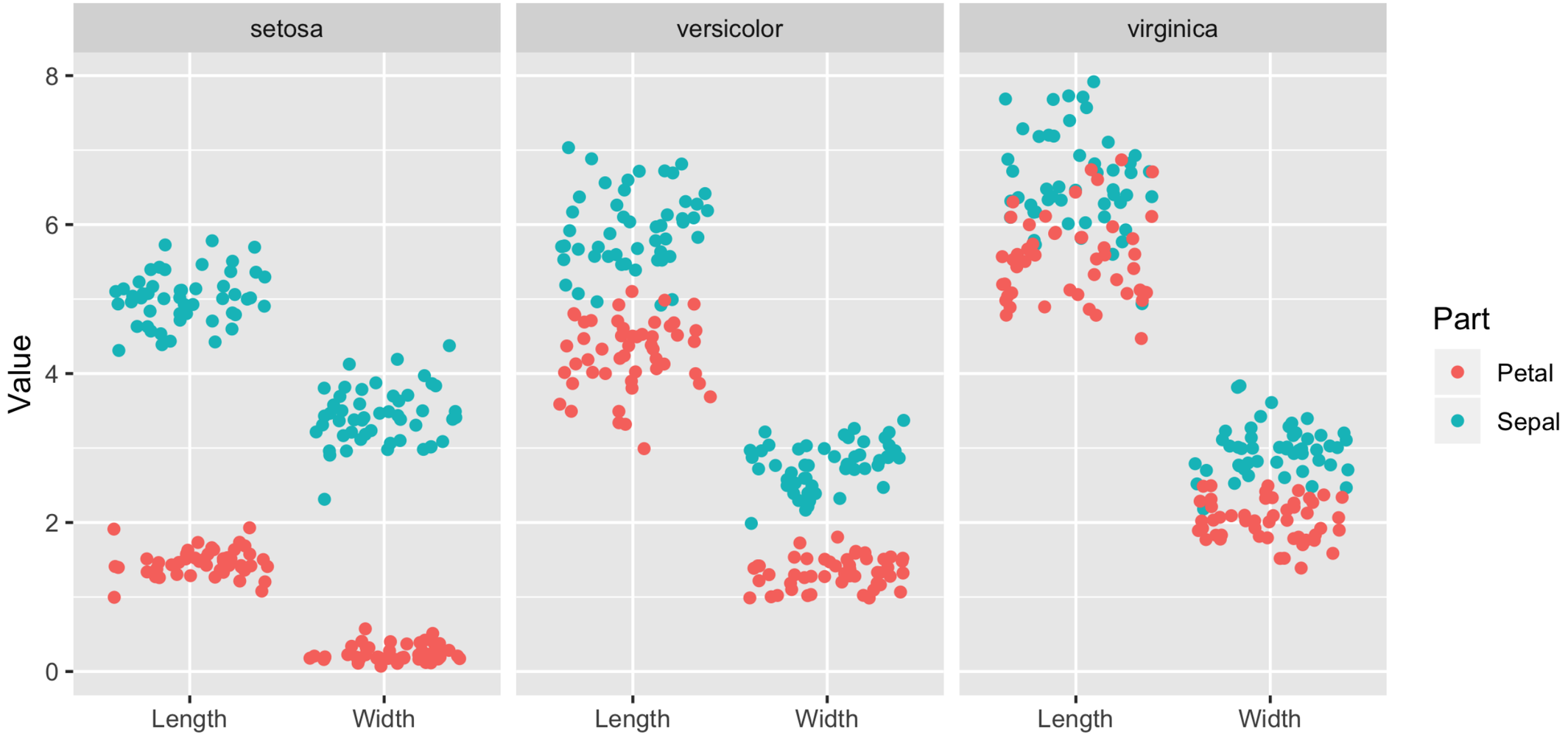
# Three iris scatter plots

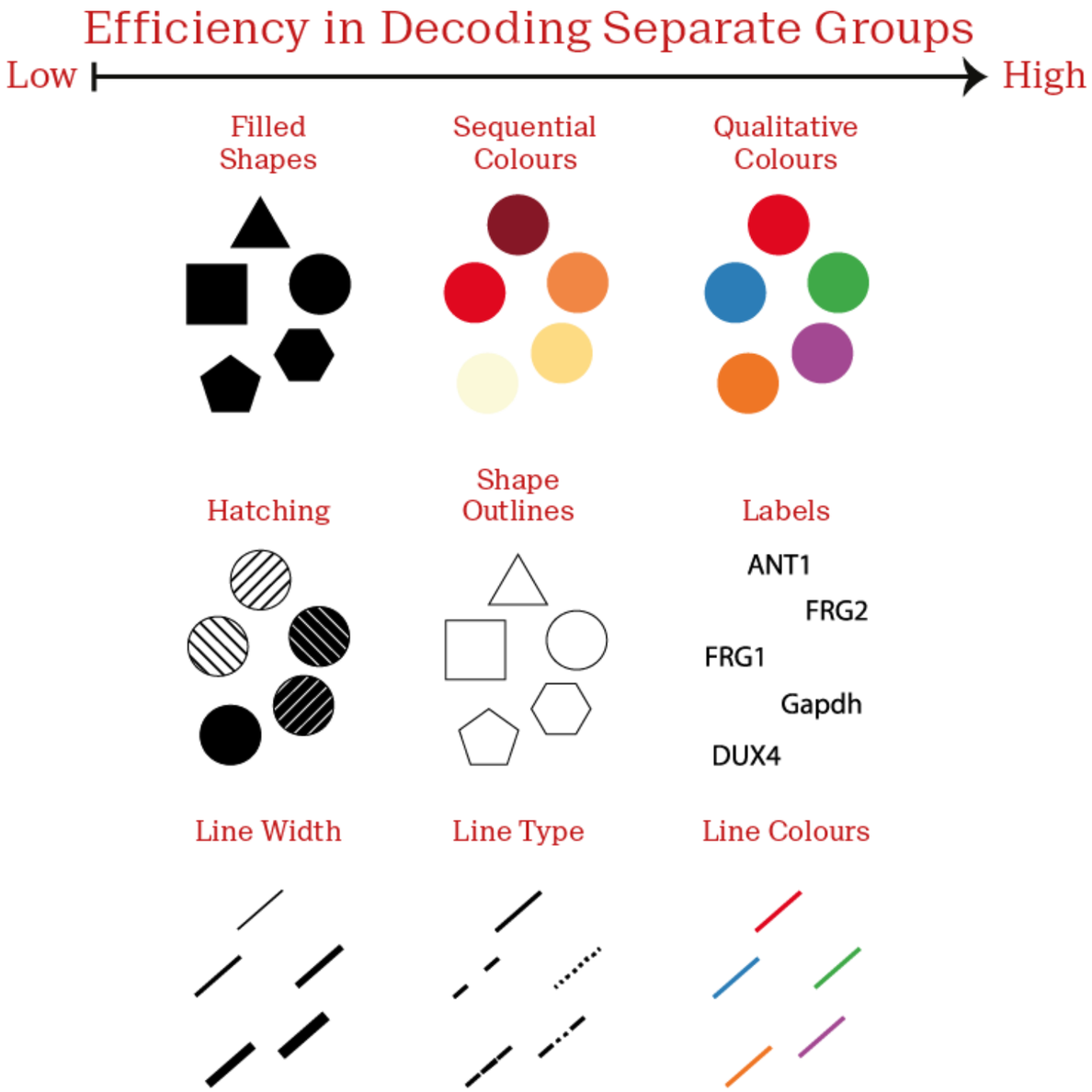


# Three iris scatter plots, unaligned y-axes



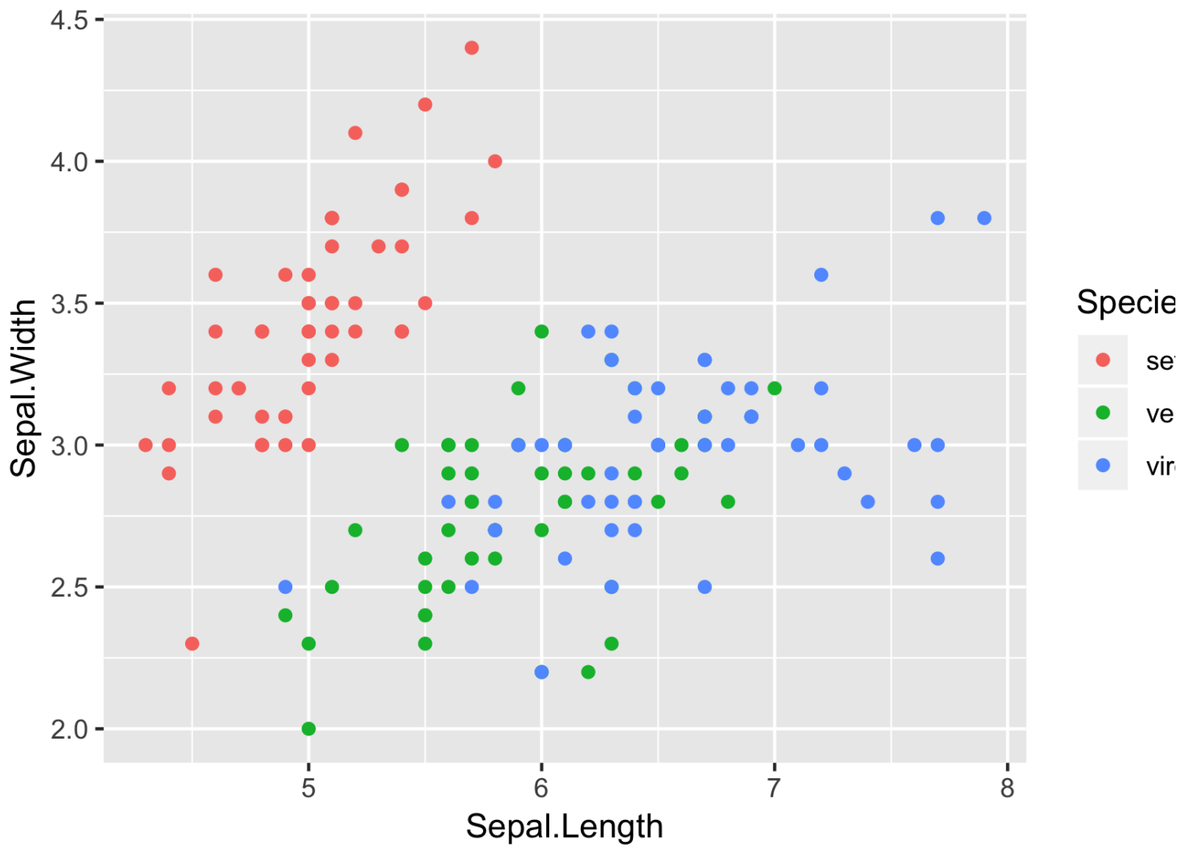
# Single faceted plot, common y-axis




Efficiency in Decoding Separate Groups
Low
High
Filled Shapes
Sequential Colours
Qualitative Colours
Hatching
Shape Outlines
Labels
ANT1
FRG2
FRG1
Gapdh
DUX4
Line Width
Line Type
Line Colours

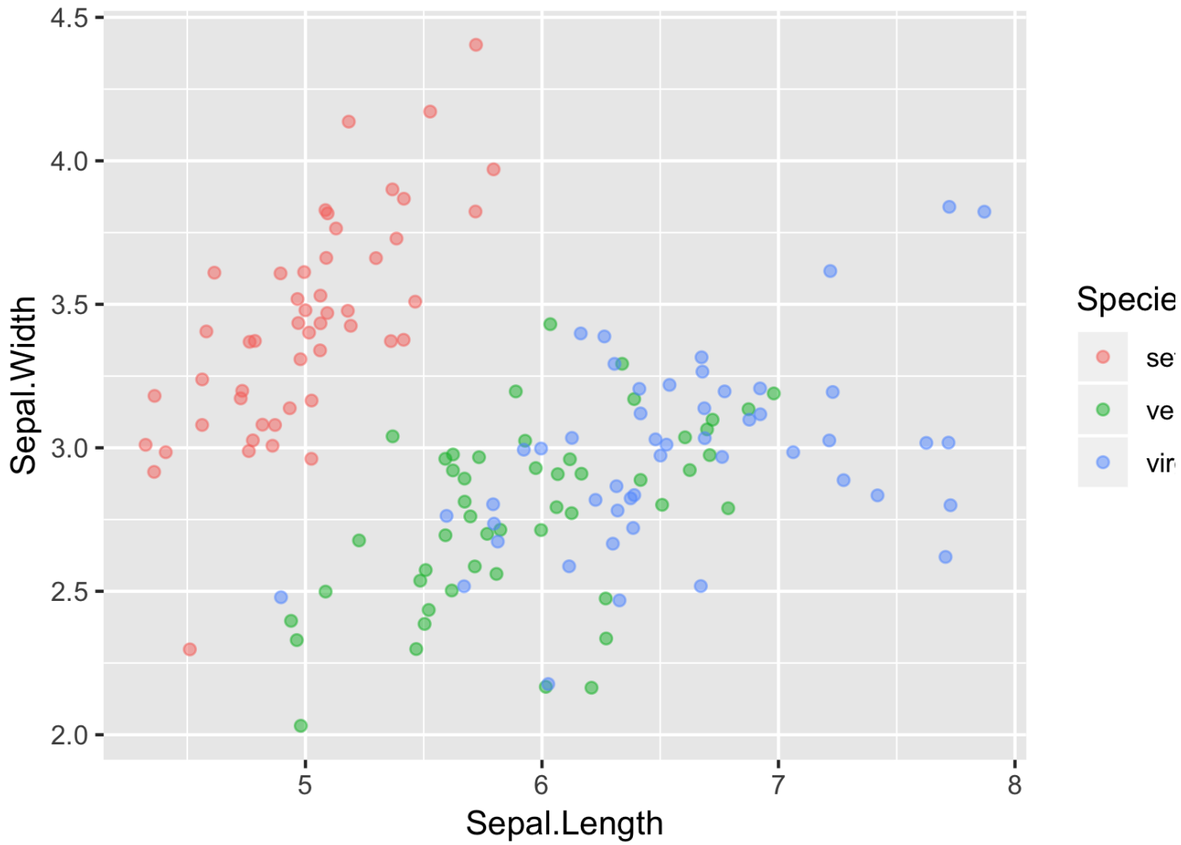
# Aesthetics - categorical variables

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



# Aesthetics - categorical variables

```
ggplot(iris, aes(x = Sepal.Length,
                 y = Sepal.Width,
                 col = Species)) +
  geom_point(position = "jitter",
             alpha = 0.5)
```



# Now it's your turn

INTRODUCTION TO DATA VISUALIZATION WITH GGPLOT2

DataCamp