Session 7: More data visualisation and ggplot

Heiner Salomon

Bremen University

2026-06-08

Universität Bremen

Overview I

1 Introduction

2 Additional variable in ggplot()

3 Wrapping up

4 Working on projects

Universität Bremen

Introduction

Learning goals of today

- Short repetition from last week

Learning goals of today

- Short repetition from last week
- Some more data viz examples

Learning goals of today

- Short repetition from last week
- Some more data viz examples
- Some more ggplot options

Universität Bremen

Repetition: Basic components of ggplot() function

- call function: ggplot(...

Repetition: Basic components of ggplot() function

- call function: ggplot(...
- add data: data = ...

Universität Bremen

Repetition: Basic components of ggplot() function

- call function: ggplot(...
- add data: data = ...
- add aesthetics aes(x = ... , y = ... , color = ...)

Universität Bremen

Repetition: Basic components of ggplot() function

- call function: ggplot(...
- add data: data = ...
- add aesthetics aes(x = ... , y = ... , color = ...)
- add layer that you want to visualise, like points + geom_point()

Universität Bremen

Repitition: Simple example of ggplot() - Code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width,
           colour = Species)) +
  geom_point()
```

Universität Bremen

Other important things to remember

- ggpplot() usually requires long, tidy data

Other important things to remember

- ggpplot() usually requires long, tidy data
- Meaning information you want to diplay need to be stored in one separate variables, not many separate variables

Universität Bremen

What additional things today?

- Hihglighting additional variable using colour and shapes

Universität Bremen

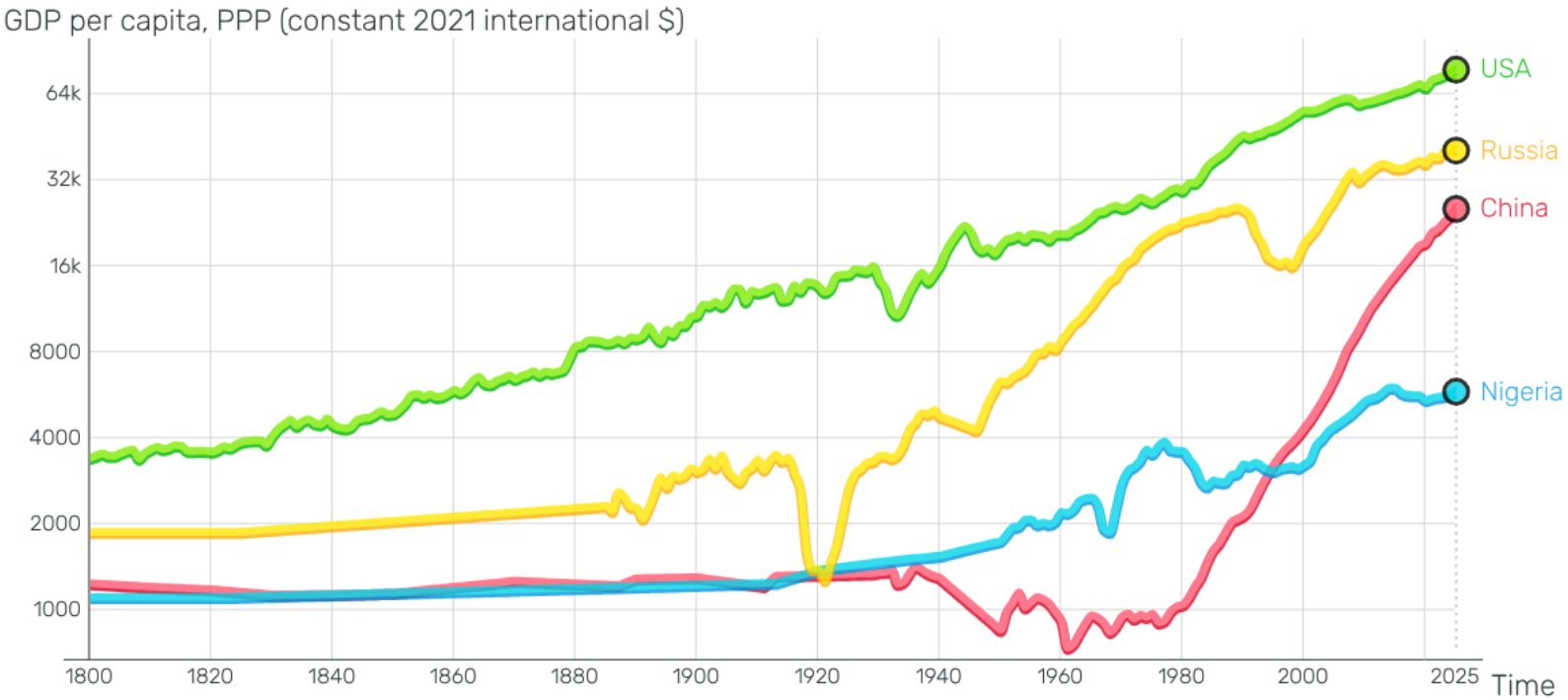
What additional things today?

- Hihglighting additional variable using colour and shapes
- Separate out by additional variable using faceting

Universität Bremen

Additional variable in ggplot()

Great example: Colour



Source: www.gapminder.org

Universität Bremen

Options for additional variable

- There are many ways to include additional variable into graph

Options for additional variable

- There are many ways to include additional variable into graph
- *colour* / *fill*

Universität Bremen

Options for additional variable

- There are many ways to include additional variable into graph
- *colour* / *fill*
- *size*

Options for additional variable

- There are many ways to include additional variable into graph
- *colour* / *fill*
- *size*
- *shape*

Universität Bremen

Options for additional variable

- There are many ways to include additional variable into graph
- *colour* / *fill*
- *size*
- *shape*
- *alpha*

Universität Bremen

Options for additional variable

- There are many ways to include additional variable into graph
- *colour* / *fill*
- *size*
- *shape*
- *alpha*
- *facet_wrap()* / *facet_grid()*

Universität Bremen

Options for additional variable

- There are many ways to include additional variable into graph
- *colour* / *fill*
- *size*
- *shape*
- *alpha*
- *facet_wrap()* / *facet_grid()*

Universität Bremen

colour / *fill*

- Adding *colour* or *fill* adds colour to graph

colour / *fill*

- Adding *colour* or *fill* adds colour to graph
- lines and strokes (e.g. geom_point()/ geom_line()) –> *colour*

Universität Bremen

colour / *fill*

- Adding *colour* or *fill* adds colour to graph
- lines and strokes (e.g. geom_point()/ geom_line()) –> *colour*
- inside areas (e.g. geom_bar()/ geom_hist()) –> *fill*

Universität Bremen

colour / *fill*

- Adding *colour* or *fill* adds colour to graph
- lines and strokes (e.g. geom_point()/ geom_line()) –> *colour*
- inside areas (e.g. geom_bar()/ geom_hist()) –> *fill*
- Colours depend on input variable (factor vs. continuous)

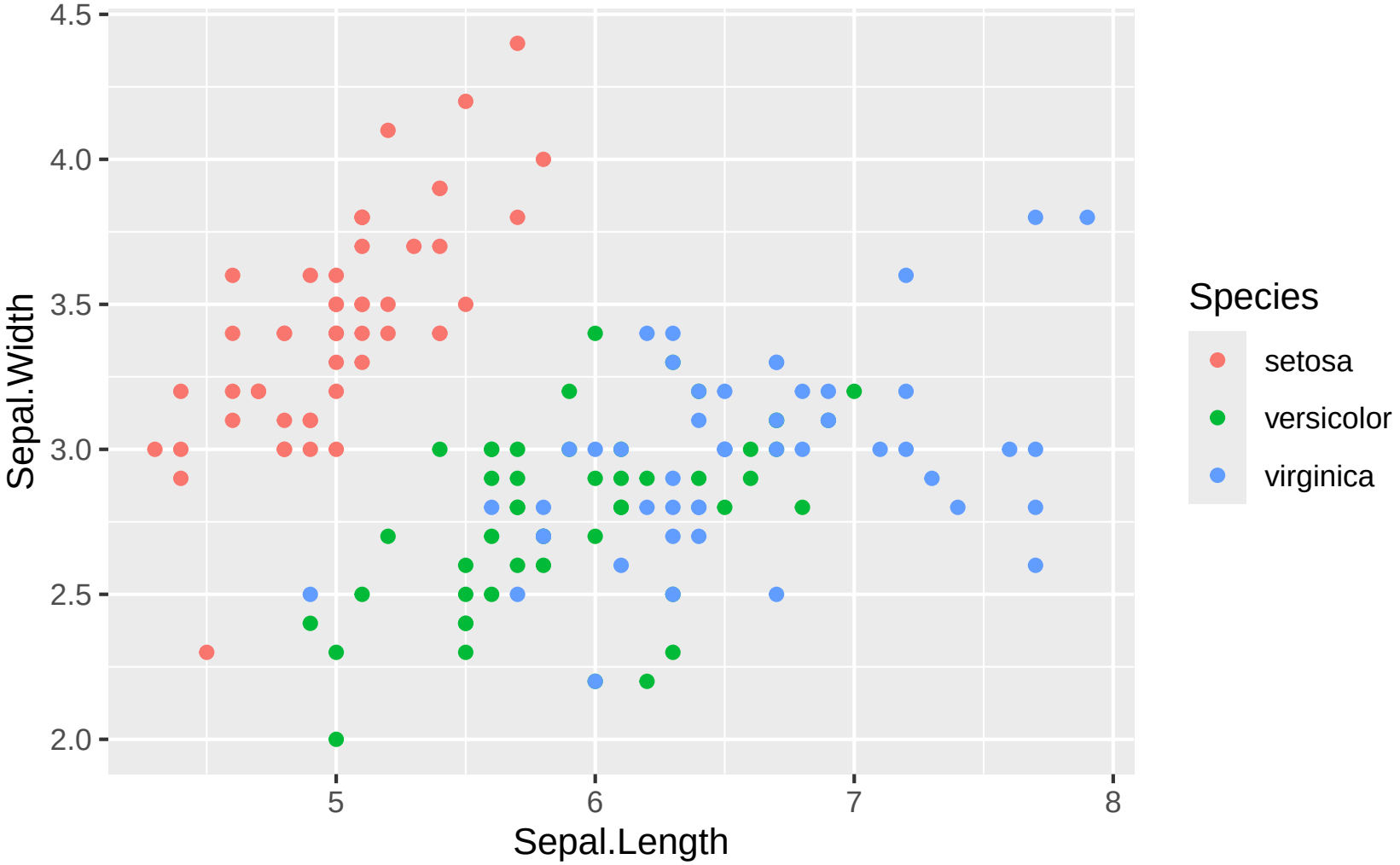
Universität Bremen

colour (factor): Example code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set with colour
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width,
           colour = Species)) +
  geom_point()
```

Universität Bremen

colour (factor): Example image

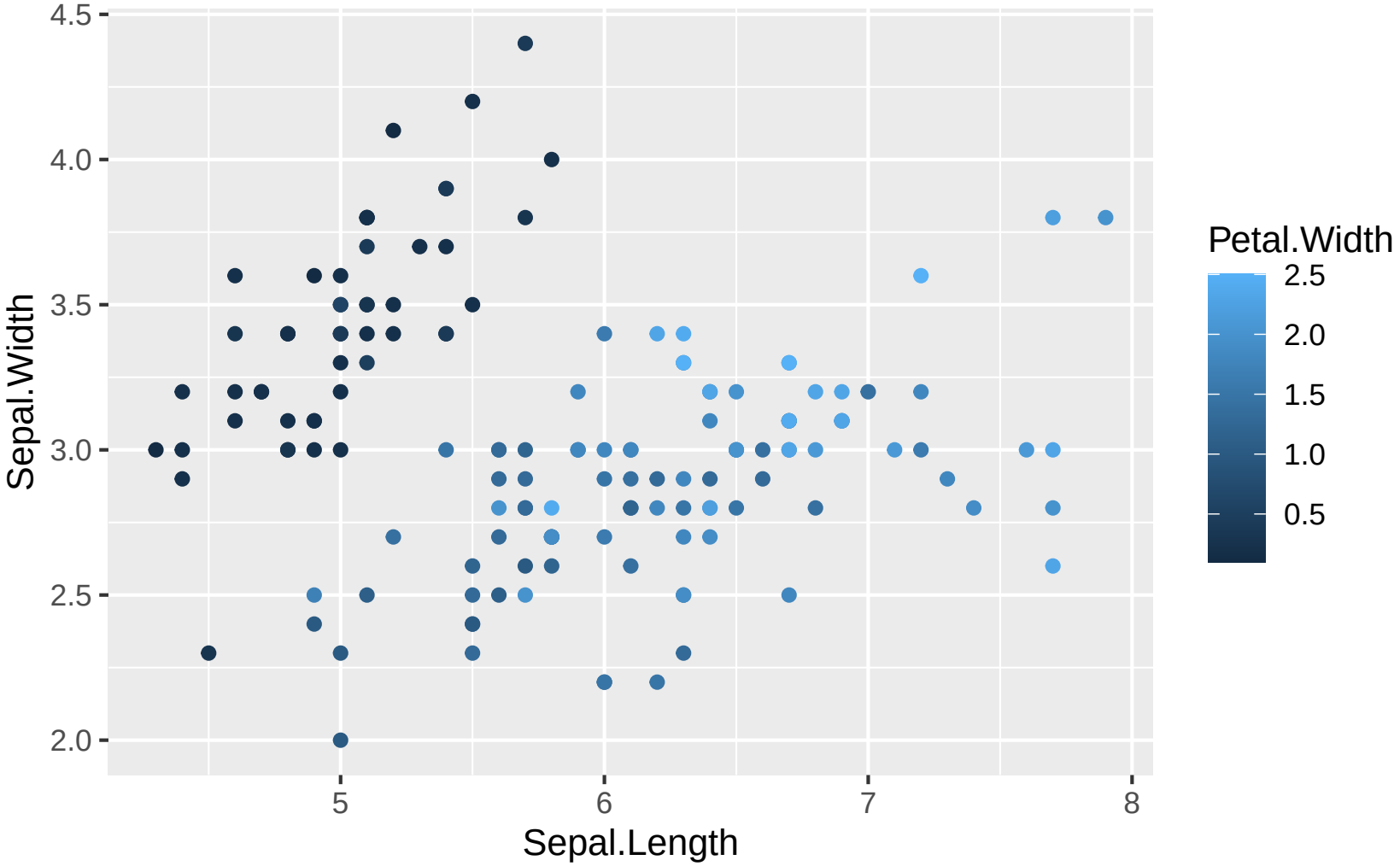


colour (continuous): Example code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set with colour (continuous)
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width,
           colour = Petal.Width)) +
  geom_point()
```

Universität Bremen

colour (continuous): Example image

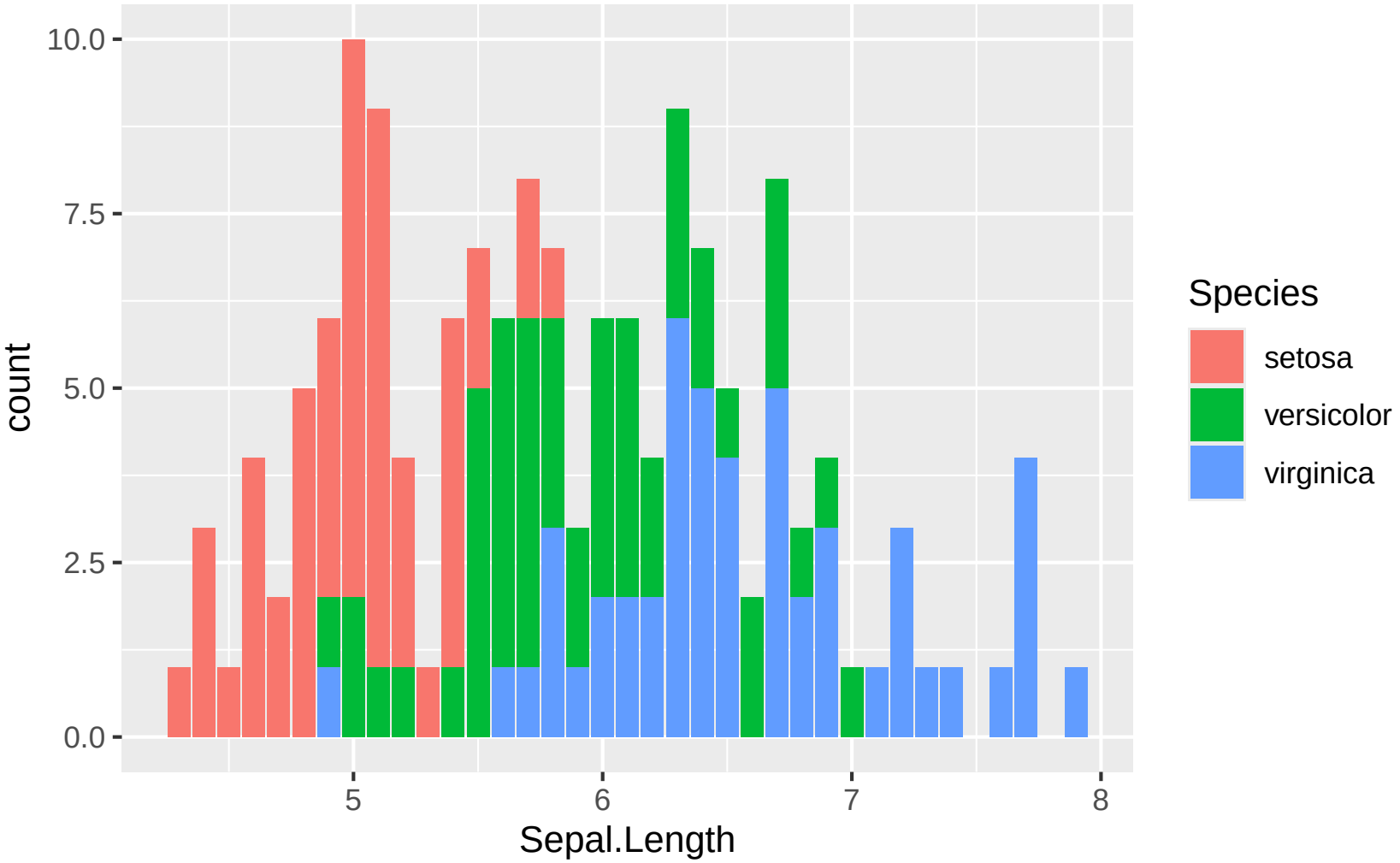


fill (factor): Example code

```
# Load ggplot2
library(ggplot2)

# Bar plot using iris data set with fill
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           fill = Species)) +
  geom_bar()
```

Universität Bremen

fill (factor): Example image



Exercise: Create three colour graphs with your data

- Create one graph using fill (factor)

Exercise: Create three colour graphs with your data

- Create one graph using fill (factor)
- Create one graph using colour (factor)

Universität Bremen

Exercise: Create three colour graphs with your data

- Create one graph using fill (factor)
- Create one graph using colour (factor)
- Create one graph using fill or colour (continuous)

Universität Bremen

size

- for geom_point(), adding *size* changes size of points

size

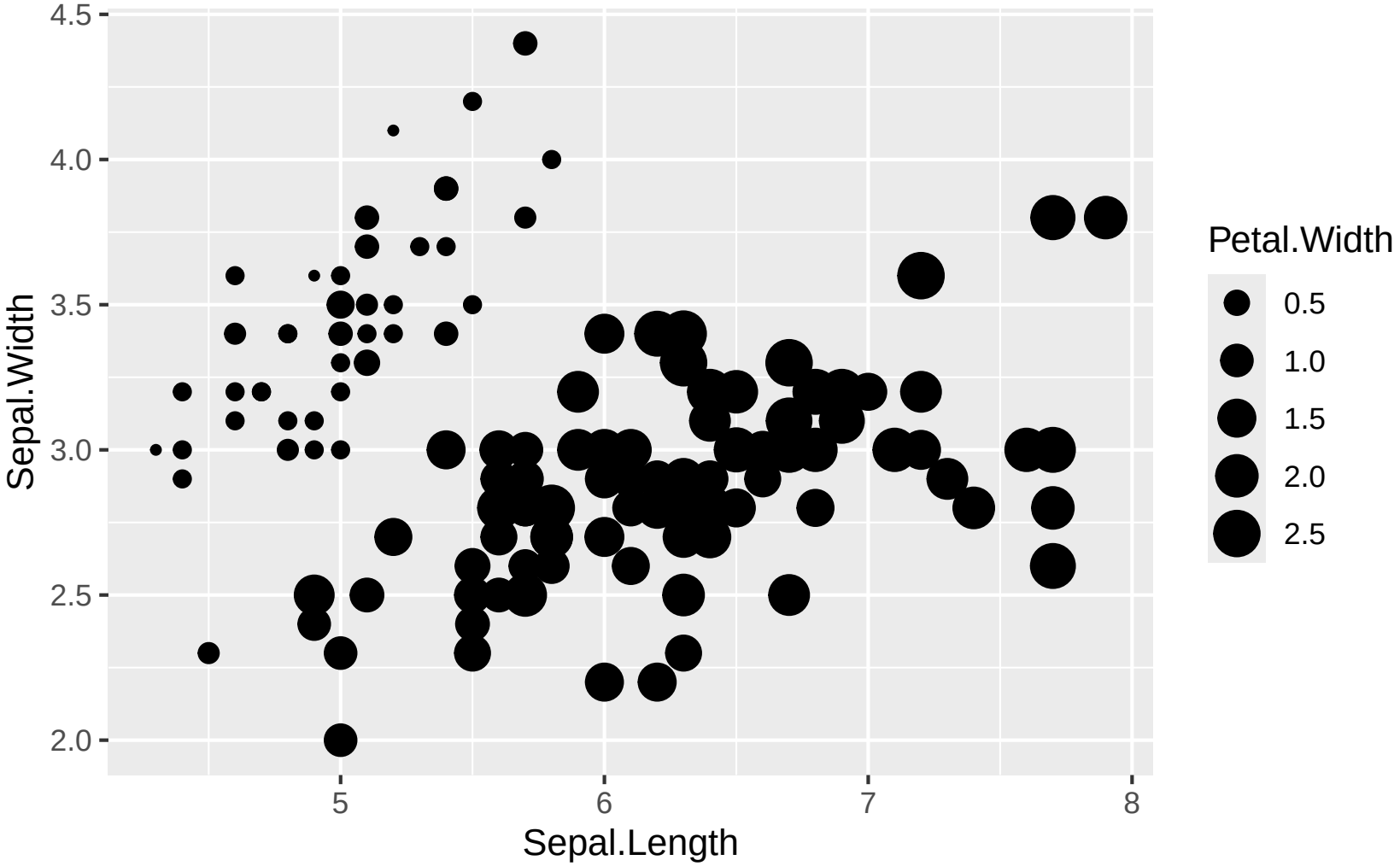
- for geom_point(), adding *size* changes size of points
- Can be useful to highlight larger weights of individual points (e.g. country population)

size: Example code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set with size
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width,
           size = Petal.Width)) +
  geom_point()
```

Universität Bremen

size: Example image



shape

- for geom_point(), adding *shape* changes shape of points

shape

- for geom_point(), adding *shape* changes shape of points
- Can be useful to visually highlight different sub-divisions in data

Universität Bremen

shape

- for geom_point(), adding *shape* changes shape of points
- Can be useful to visually highlight different sub-divisions in data
- Originally more useful for black and white applications, nowadays colours used more regularly

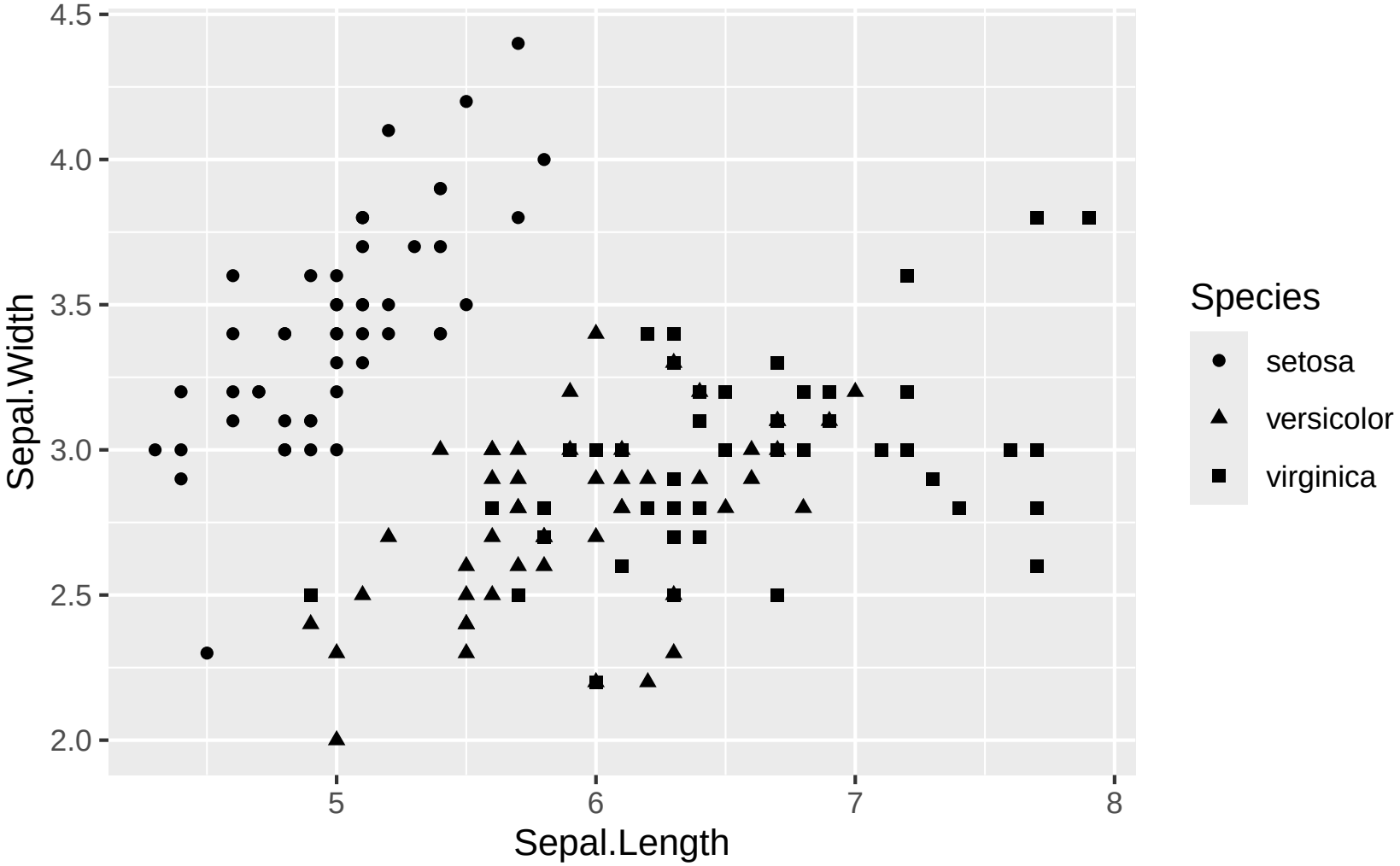
Universität Bremen

shape: Example code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set with shapes
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width,
           shape = Species)) +
  geom_point()
```

Universität Bremen

shape: Example image



Combining options *colour* and *shape*

- Of course, one can also combine options

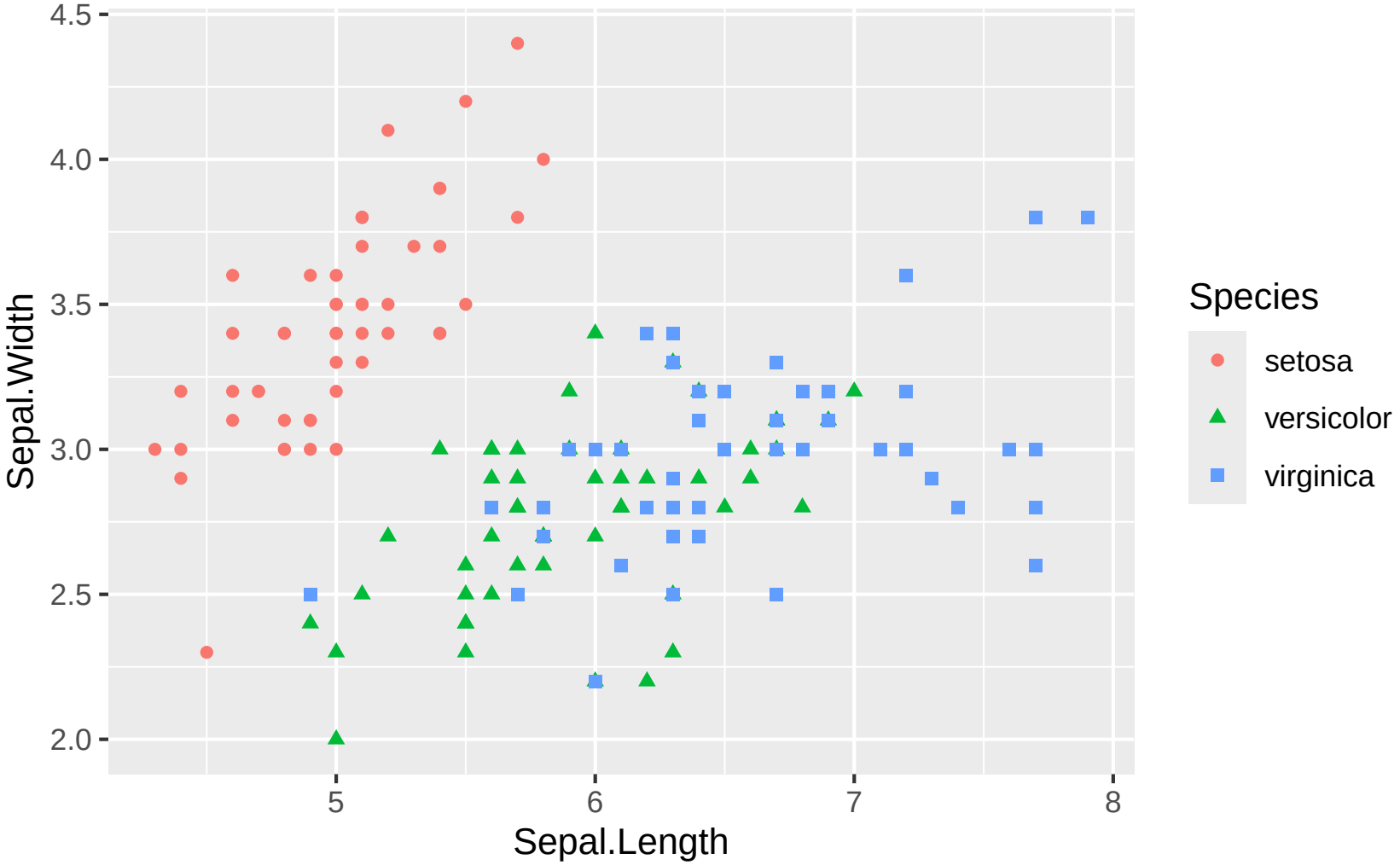
Combining options *colour* and *shape*

- Of course, one can also combine options
- may increase visibility of differences even more

colour + *shape*: Example code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set with colour and shapes
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width,
           colour = Species,
           shape = Species)) +
  geom_point()
```

colour + *shape*: Example image



alpha

- *alpha* sets the opacity of information

alpha

- *alpha* sets the opacity of information
- Can be useful to visually highlight different sub-divisions in data

alpha

- *alpha* sets the opacity of information
- Can be useful to visually highlight different sub-divisions in data
- Originally more useful for black and white applications, nowadays colours used more regularly

Universität Bremen

alpha

- *alpha* sets the opacity of information
- Can be useful to visually highlight different sub-divisions in data
- Originally more useful for black and white applications, nowadays colours used more regularly
- Remember the bar plot from before?

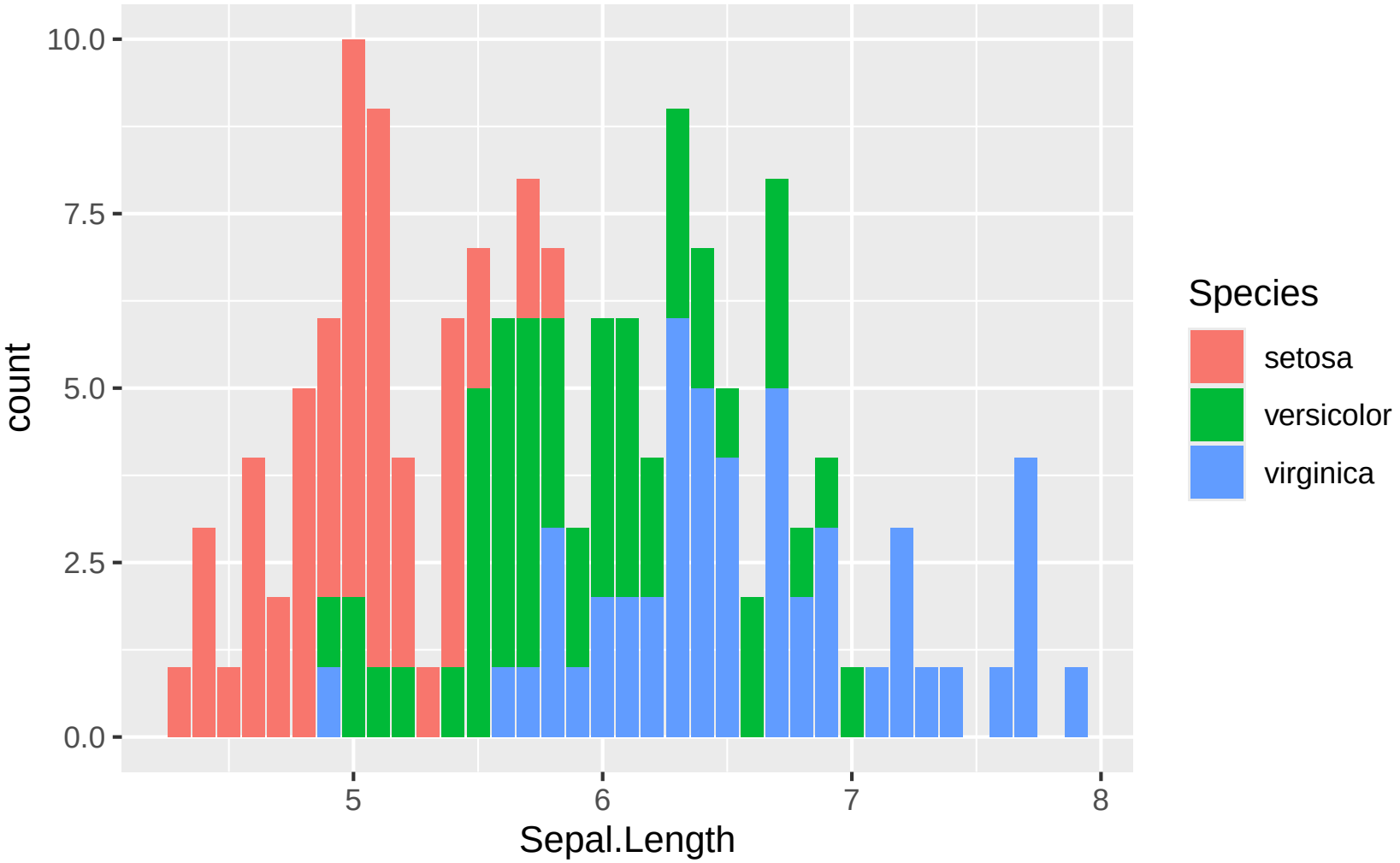
Universität Bremen

fill (factor): Example code

```
# Load ggplot2
library(ggplot2)

# Bar plot using iris data set with fill
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           fill = Species)) +
  geom_bar()
```

Universität Bremen

fill (factor): Example image

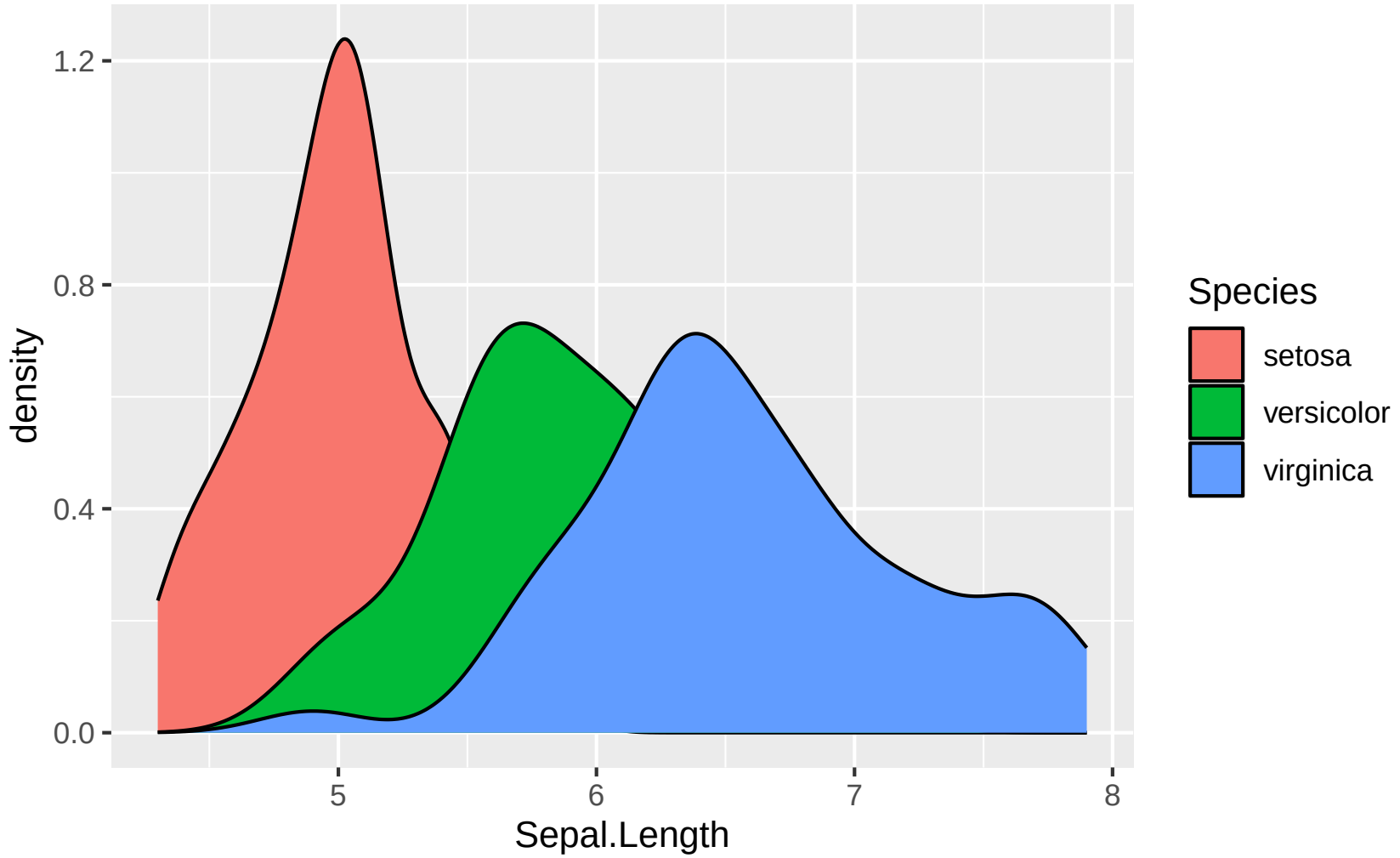


fill (density): Example code

```
# Load ggplot2
library(ggplot2)

# Density plot using iris data set with fill
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           fill = Species)) +
  geom_density()
```

Universität Bremen

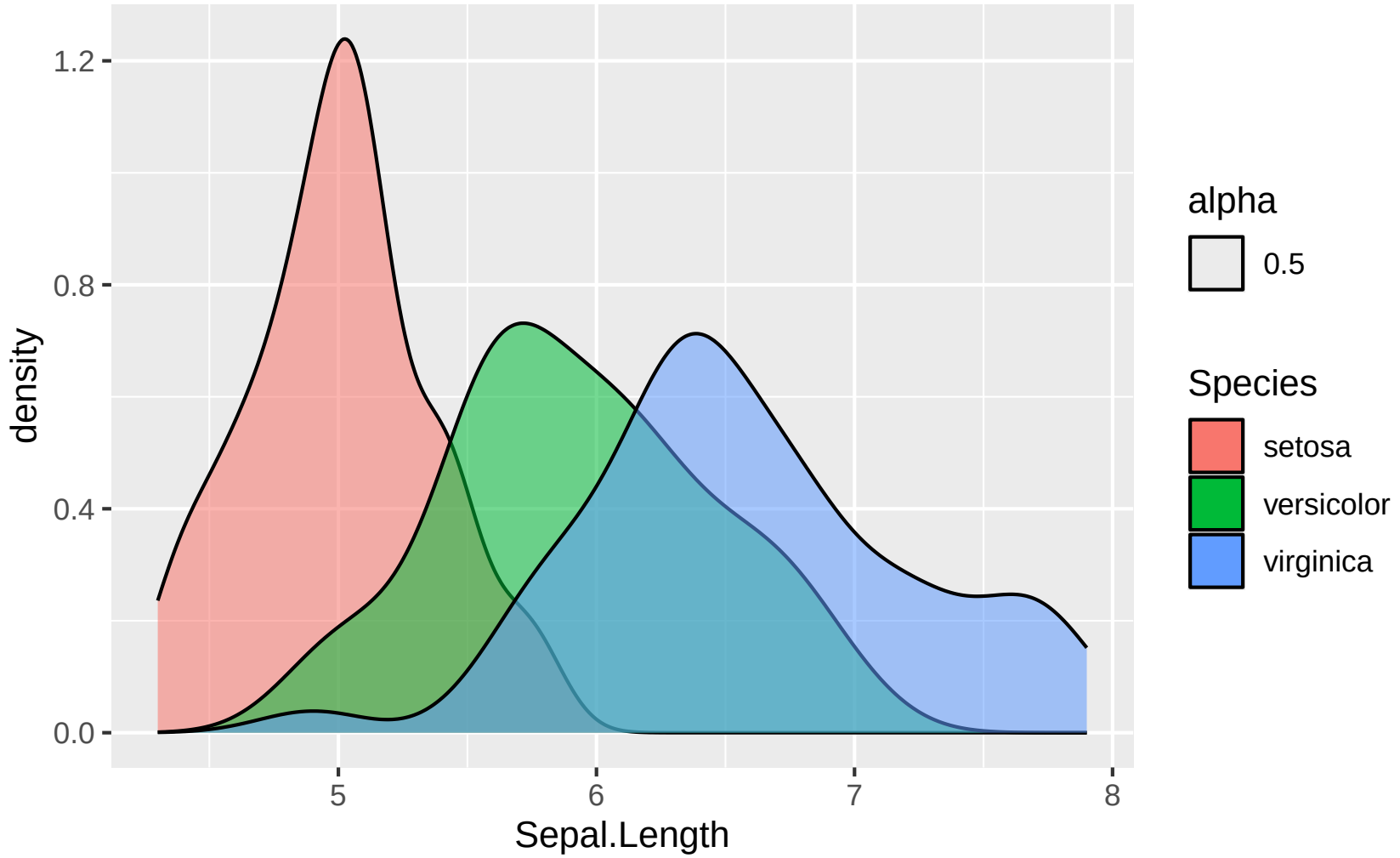
fill (density): Example image



fill (density) + *alpha*: Example code

```
# Load ggplot2
library(ggplot2)

# Density plot using iris data set with fill and alpha
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           fill = Species,
           position = "identity",
           alpha = 0.5)) +
  geom_density()
```

fill (density) + *alpha*: Example image



alpha as variable

- Of course, *alpha* can also be used with variable

alpha as variable

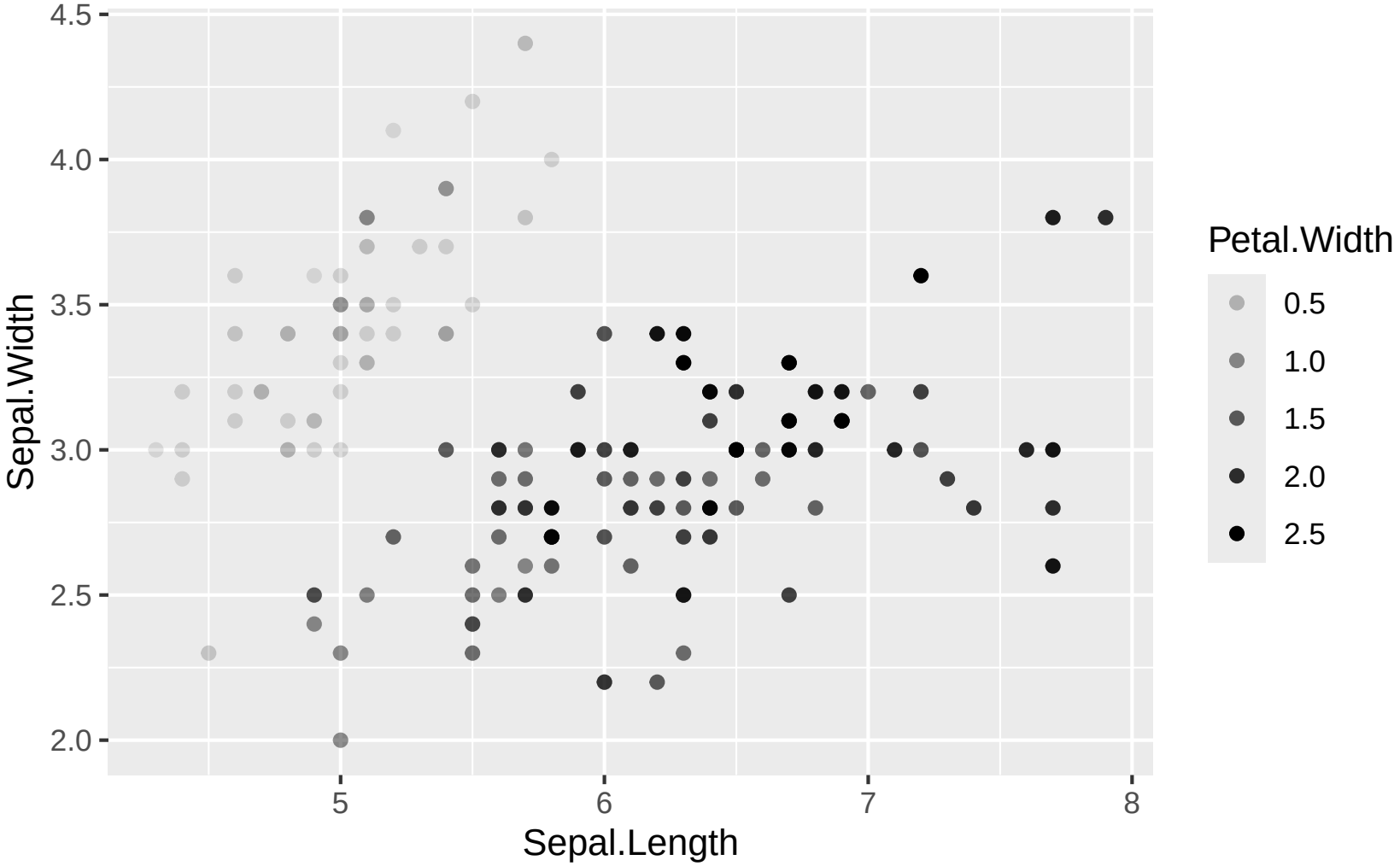
- Of course, *alpha* can also be used with variable
- Though use case less obvious

alpha: Example code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set with shapes
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width,
           alpha = Petal.Width)) +
  geom_point()
```

Universität Bremen

alpha: Example image



Exercise: Create three graphs with your data

- Create one graph using *size*

Exercise: Create three graphs with your data

- Create one graph using *size*
- Create one graph using *shape*and *colour*

Universität Bremen

Exercise: Create three graphs with your data

- Create one graph using *size*
- Create one graph using *shape*and *colour*
- Create a density plot with *fill* and *alpha*

Universität Bremen

Exercise: Create three graphs with your data

- Create one graph using *size*
- Create one graph using *shape*and *colour*
- Create a density plot with *fill* and *alpha*
- Consider using different variables than in the last exercise

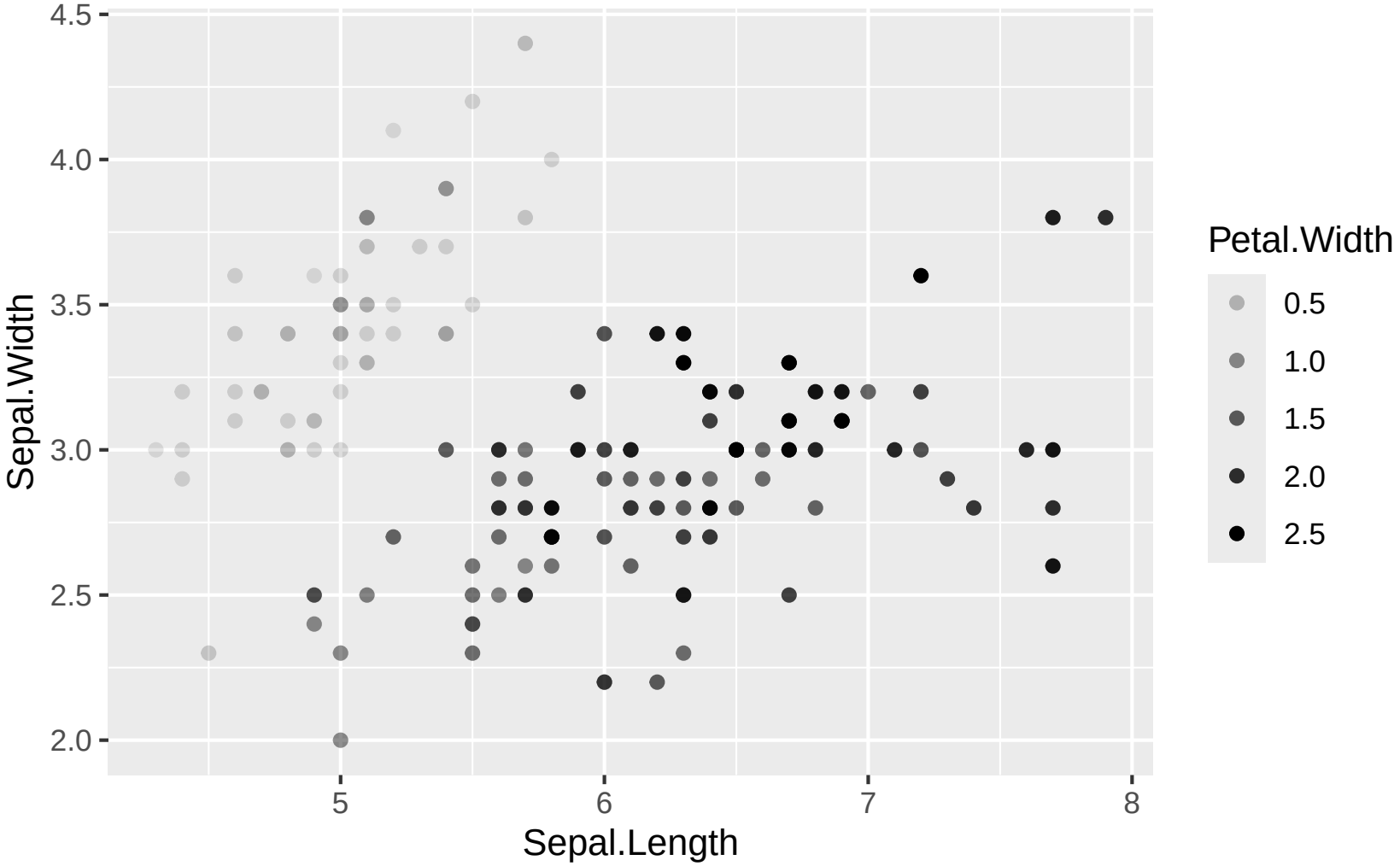
Universität Bremen

alpha: Example code

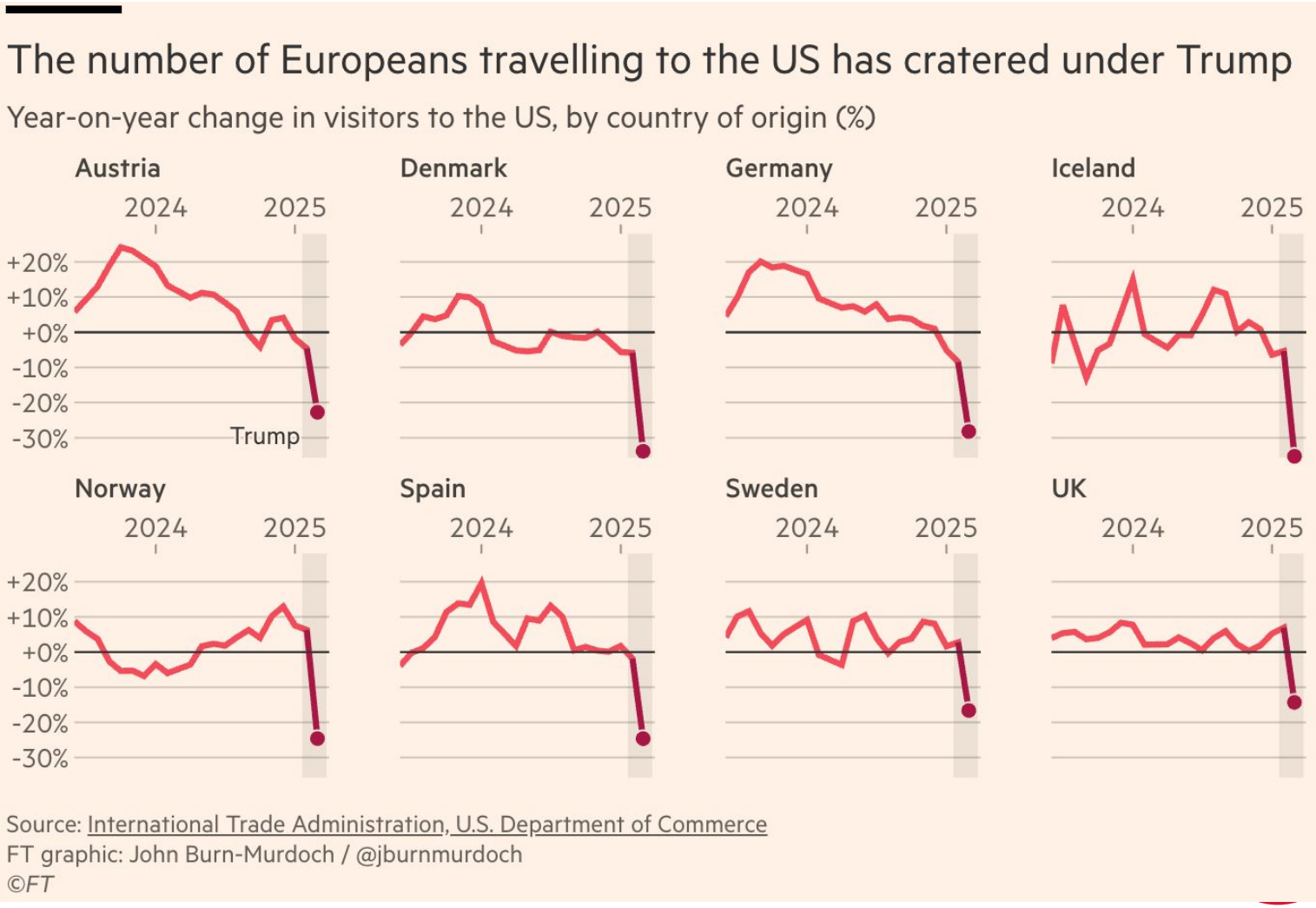
```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set with size
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width,
           alpha = Petal.Width)) +
  geom_point()
```

Universität Bremen

alpha: Example image



Great example: Faceting



Universität Bremen

facet_wrap() / face_grid()

- faceting is super useful, especially with large data sets

facet_wrap() / face_grid()

- faceting is super useful, especially with large data sets
- Creates separate graph per sub-group (hence requires factor variable)

facet_wrap() / face_grid()

- faceting is super useful, especially with large data sets
- Creates separate graph per sub-group (hence requires factor variable)
- *facet_wrap(~ x)* <- best to use with one factor variable (x), plots separate graphs per factor

Universität Bremen

facet_wrap() / face_grid()

- faceting is super useful, especially with large data sets
- Creates separate graph per sub-group (hence requires factor variable)
- *facet_wrap(~ x)* <- best to use with one factor variable (x), plots separate graphs per factor
- *facet_grid(x ~ y)* <- best to use with two factor variables (x, y), plots grid of all combinations of x and y

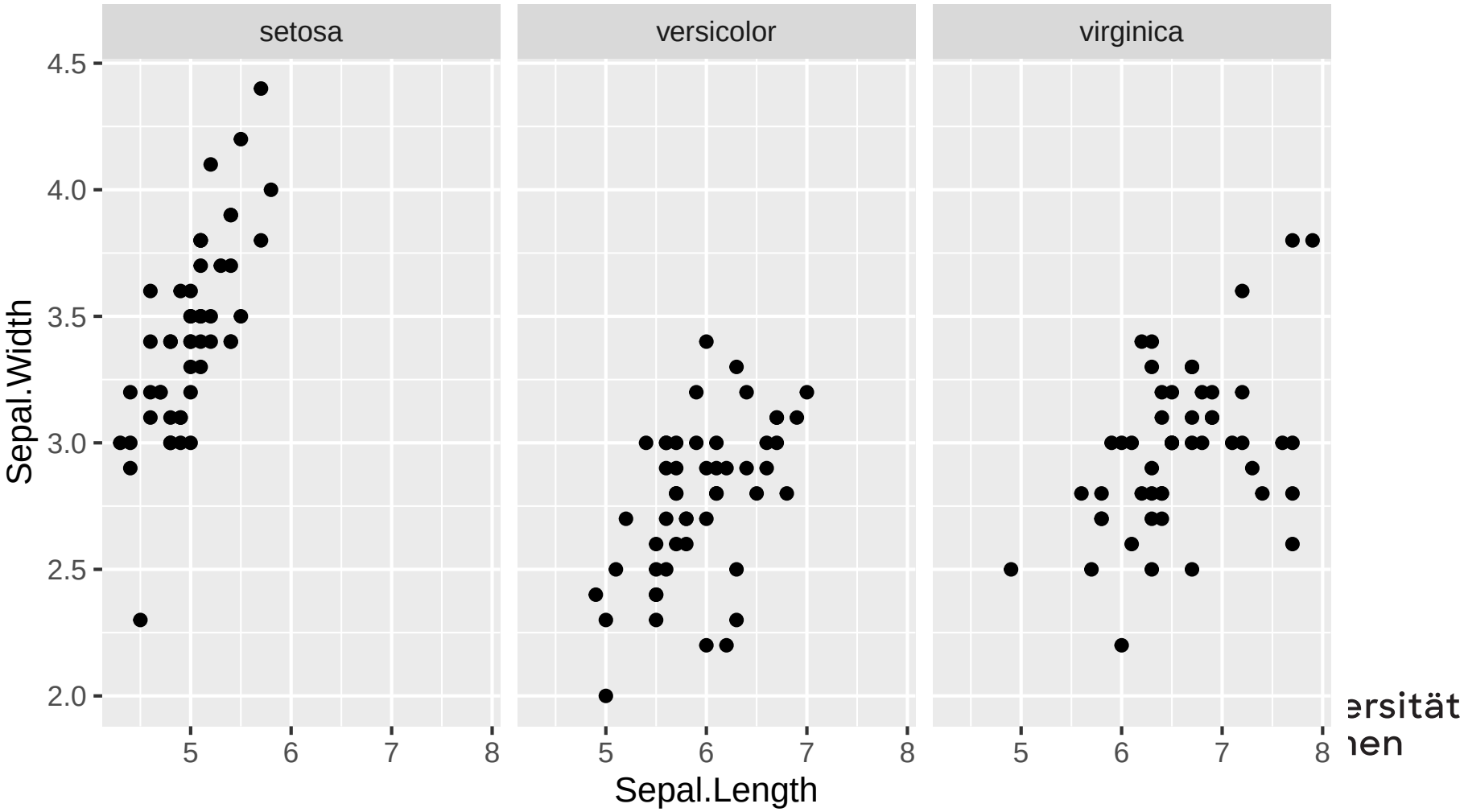
Universität Bremen

facet_wrap(): Example code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using iris data set with facet_wrap()
ggplot(data = iris,
       mapping =
         aes(x = Sepal.Length,
           y = Sepal.Width)) +
  geom_point() +
  facet_wrap(~ Species)
```

Universität Bremen

facet_wrap(): Example image

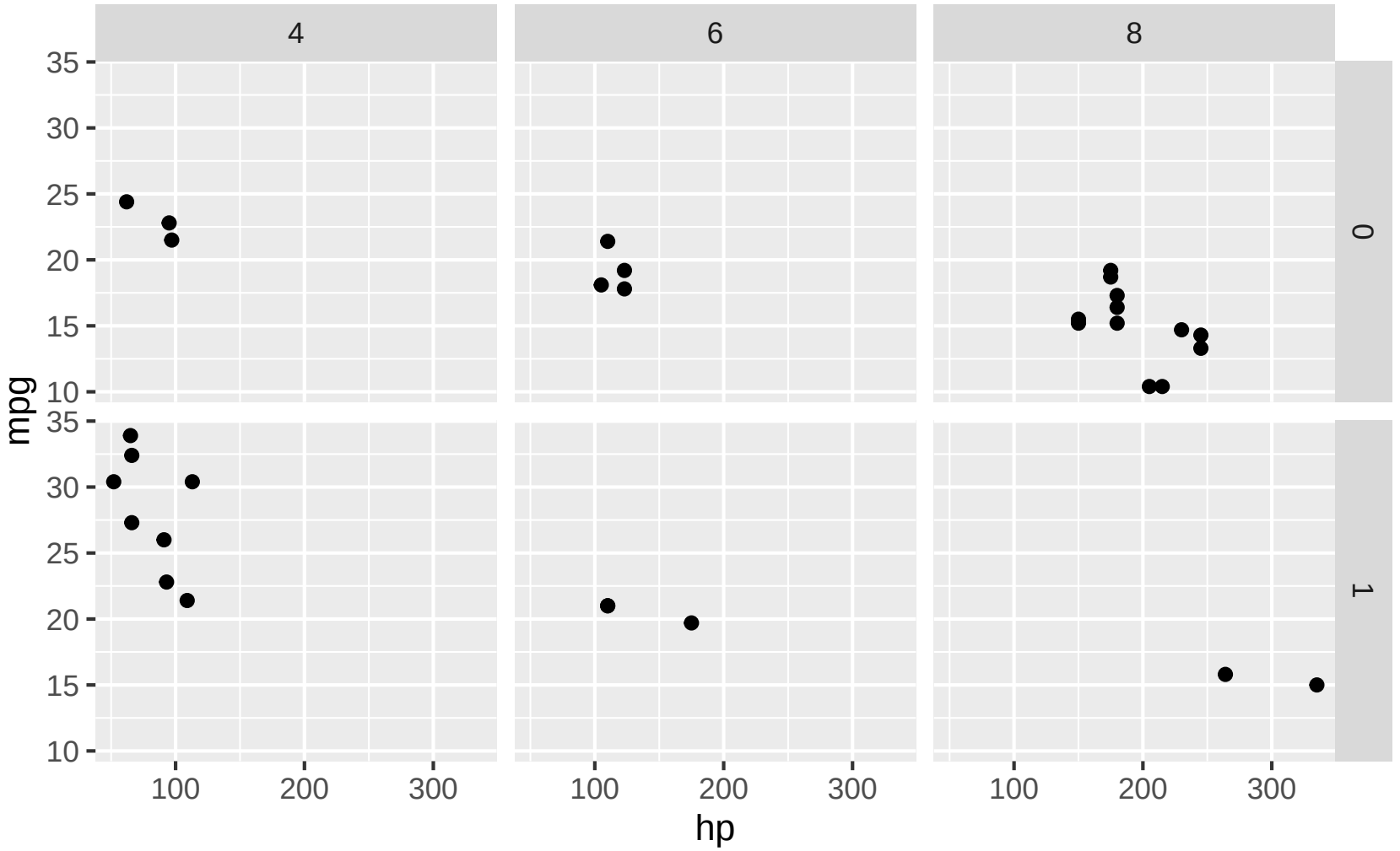


facet_grid(): Example code

```
# Load ggplot2
library(ggplot2)

# Scatterplot using mtcars data set with facet_wrap()
ggplot(data = mtcars,
       mapping =
         aes(x = hp,
           y = mpg)) +
  geom_point() +
  facet_grid(am ~ cyl)
```

Universität Bremen

facet_grid(): Example image



Exercise: Create two graphs with your data

- Create one graph using *facet_wrap*

Exercise: Create two graphs with your data

- Create one graph using *facet_wrap*
- Create one graph using *facet_grid()*

Universität Bremen

Things to keep in mind

- Not all ways to integrate third variable work with every geom

Things to keep in mind

- Not all ways to integrate third variable work with every geom
- *colour* and *fill* work with nearly all geoms

Universität Bremen

Things to keep in mind

- Not all ways to integrate third variable work with every geom
- *colour* and *fill* work with nearly all geoms
- Otherwise, just try out and see what works (for you)

Universität Bremen

Wrapping up

Which topics to cover?

- Which topics would you like me to still cover (more)?

Universität Bremen

Which topics to cover?

- Which topics would you like me to still cover (more)?
- One suggestions could be Git

Universität Bremen

Working on projects

Universität Bremen