

ggplot Part I

Workshop on Data Visualization in R

Lokesh Mano • 18-Oct-2022

NBIS, SciLifeLab

Contents

- Grammar of Graphics
- Building a graph
- Build-Demo
- Geoms
- Aesthetics
- Multiple Geoms
- Scales - color
- Scales - shape
- facet_wrap

Grammar of Graphics



- **Data:** Input data
- **Geom:** A geometry representing data. Points, Lines etc
- **Aesthetics:** Visual characteristics of the geometry. Size, Color, Shape etc
- **Scale:** How visual characteristics are converted to display values
- **Statistics:** Statistical transformations. Counts, Means etc
- **Coordinates:** Numeric system to determine position of geometry. Cartesian, Polar etc
- **Facets:** Split data into subsets

Building a graph

```
ggplot (data = <DATA>) +
```

```
<GEOM_FUNCTION> (mapping = aes(<MAPPINGS>),
```

```
stat = <STAT> , position = <POSITION>) +
```

```
<COORDINATE_FUNCTION> +
```

```
<FACET_FUNCTION> +
```

```
<SCALE_FUNCTION> +
```

```
<THEME_FUNCTION>
```

required

Not
required,
sensible
defaults
supplied

Build-Demo

```
ggplot(iris)
```



Build-Demo

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



Build-Demo

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



Build-Demo

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

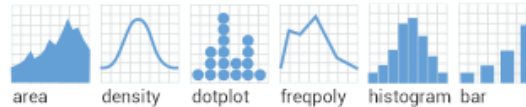


Geoms

Basic



One variable



Two variables



Error



Three variables



Map



```
p <- ggplot(iris)
# scatterplot
p+geom_point(aes(x=Sepal.Length,y=Sepal.Width))
# barplot
p+geom_bar(aes(x=Sepal.Length))
# boxplot
p+geom_boxplot(aes(x=Species,y=Sepal.Width))
# search
help.search("^geom_",package="ggplot2")
```

Aesthetics

- Aesthetic mapping vs aesthetic parameter

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



```
ggplot(iris)+  
  geom_point(aes(x=Sepal.Length,  
                 y=Sepal.Width,  
                 size=2,  
                 alpha=0.8,  
                 shape=15,  
                 color="steelblue"))
```



Multiple geoms

```
ggplot(iris,aes(x=Sepal.Length,y=Sepal.Width))+  
  geom_point()+  
  geom_line()+  
  geom_smooth()+  
  geom_rug()+  
  geom_step()+  
  geom_text(data=subset(iris,iris$Species=="setosa"),aes(label=Species))
```



Scales • Discrete Colors

- scales: position, color, fill, size, shape, alpha, linetype
- syntax: `scale_aesthetic>_<type>`



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



```
p + scale_color_manual(
  name="Manual",
  values=c("#5BC0EB", "#FDE74C", "#9BC53D")
)
```



Scales • Continuous Colors

- In RStudio, type `scale_`, then press TAB

```
p <- ggplot(iris)+  
  geom_point(aes(x=Sepal.Length,  
                 y=Sepal.Width,  
                 shape=Species,color=Petal.Length))  
p
```



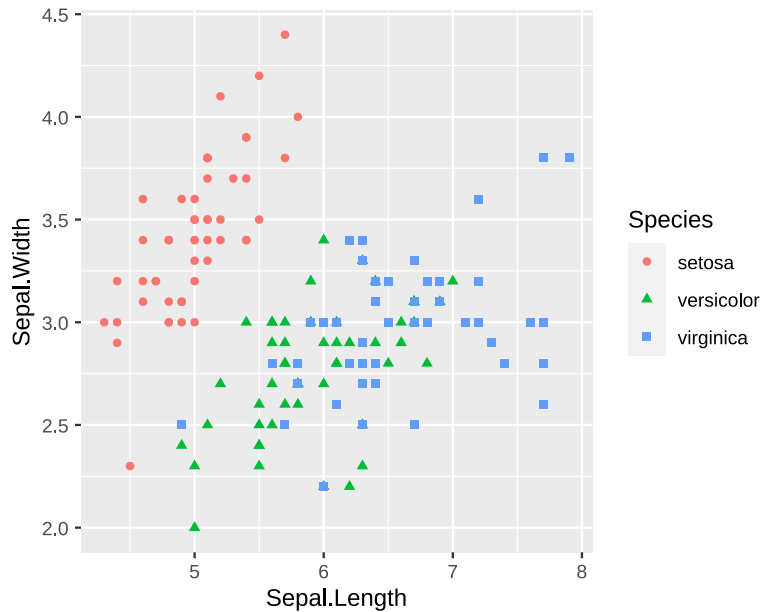
```
p +  
  scale_color_gradient(name="Pet Len",  
                       breaks=range(iris$Petal.Length),  
                       labels=c("Min", "Max"),  
                       low="black", high="red")
```



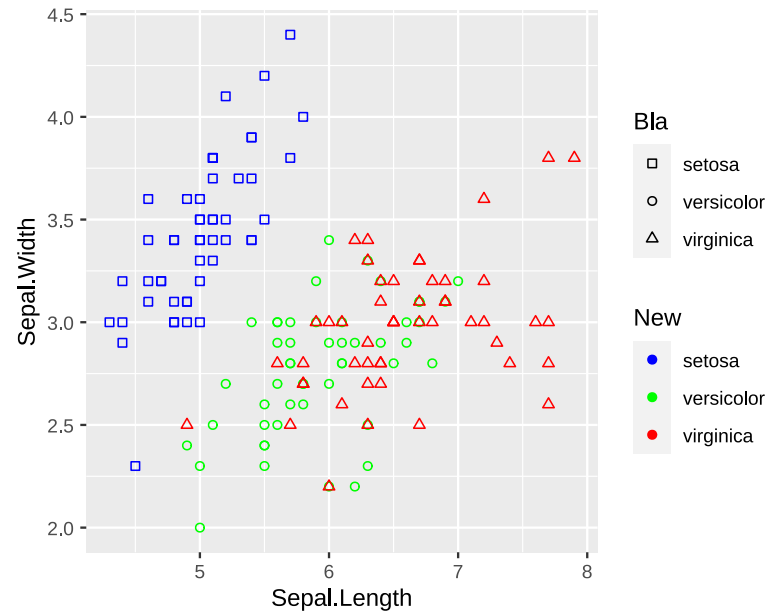
Scales • Shape

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

p



```
p +  
  scale_color_manual(name="New",  
                    values=c("blue","green","red"))+  
  scale_shape_manual(name="Bla",values=c(0,1,
```



Facets • **facet_wrap**

- Split to subplots based on variable(s)
- Facetting in one dimension

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

p



```
p + facet_wrap(~Species)
```



```
p + facet_wrap(~Species, nrow=3)
```





Thank you. Questions?

R version 4.1.3 (2022-03-10)

Platform: x86_64-pc-linux-gnu (64-bit)

OS: Ubuntu 18.04.6 LTS

Built on : 📅 18-Oct-2022 at 🕒 09:56:28

2022 • SciLifeLab • NBIS