ggplot Part II

SBW - Data Visualization Workshop

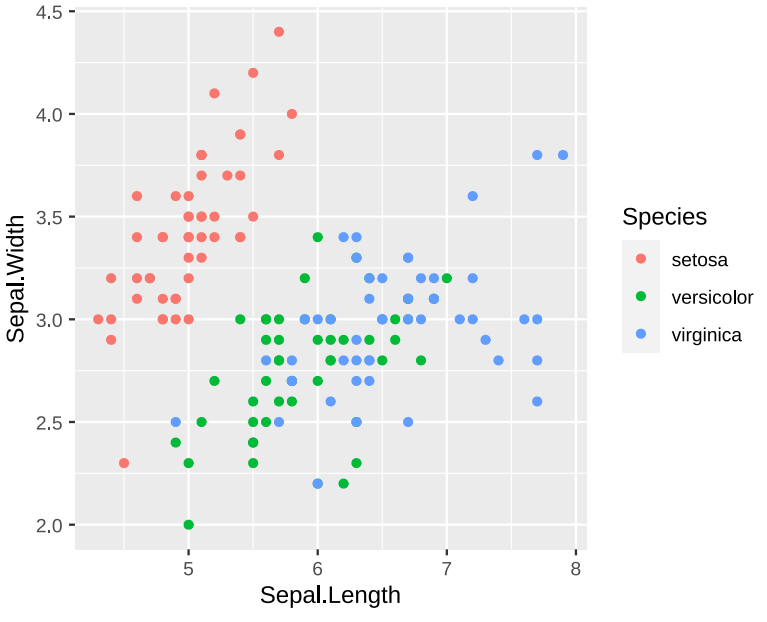
Lokesh Mano • 19-Oct-2022
NBIS, SciLifeLab

Contents

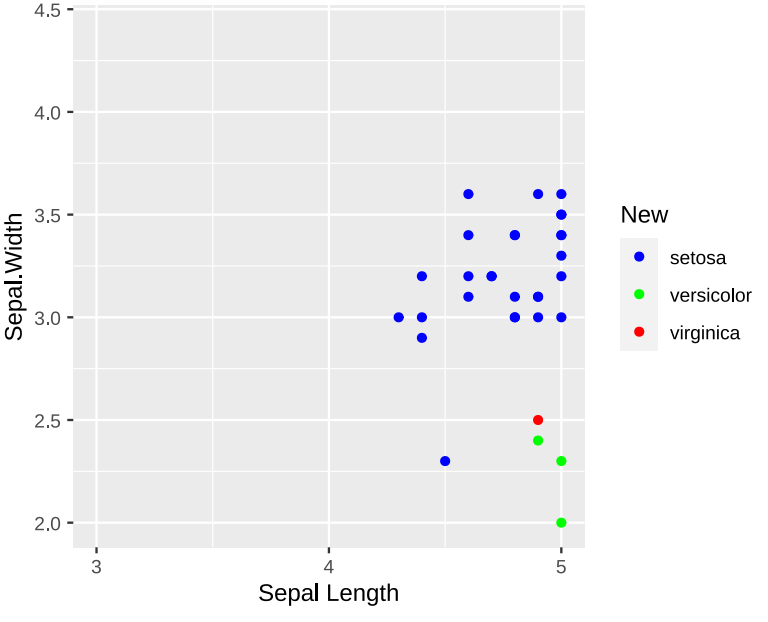
Scales • Axes

- scales: x, y
- syntax: `scale_<axis>_<type>`
- arguments: name, limits, breaks, labels

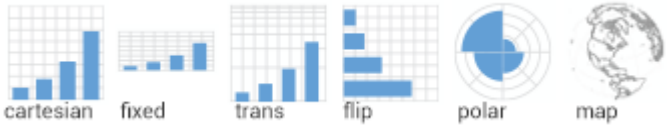
```
p <- ggplot(iris)+
      geom_point(aes(x=Sepal.Length,
                     y=Sepal.Width,
                     color=Species))
p
```




```
p + scale_color_manual(name="New",
   values=c("blue","green","red"))+
scale_x_continuous(name="Sepal Length",
        breaks=seq(1,8),limits=c(3,5))
```



Coordinate Systems



- `coord_cartesian(xlim=c(2,8))` for zooming in
- `coord_map` for controlling limits on maps
- `coord_polar`

```
p <- ggplot(iris,aes(x="",y=Petal.Length,fi
  geom_bar(stat="identity")
p
```

```
p+coord_polar("y",start=0)
```

Theme

- Modify non-data plot elements/appearance
- Axis labels, panel colors, legend appearance etc
- Save a particular appearance for reuse
- `?theme`

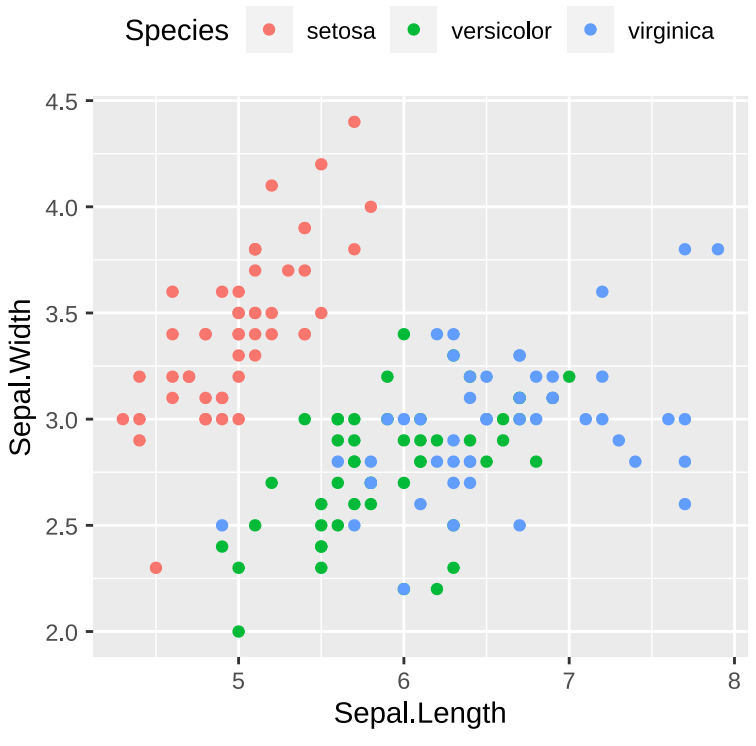
```
ggplot(iris,aes(Petal.Length))+
  geom_histogram()+
  facet_wrap(~Species,nrow=2)+
  theme_grey()
```

```
ggplot(iris,aes(Petal.Length))+
  geom_histogram()+
  facet_wrap(~Species,nrow=2)+
  theme_bw()
```

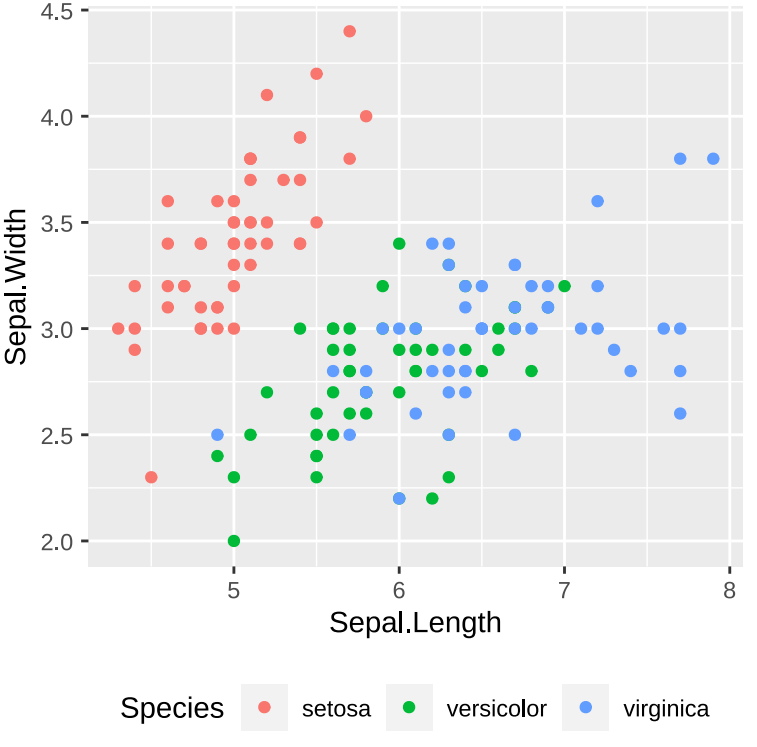
Theme • Legend

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

```
p + theme(legend.position="top")
```




```
p + theme(legend.position="bottom")
```



Theme • Text

```
element_text(family=NULL,face=NULL,color=NULL,size=NULL,hjust=NULL,
             vjust=NULL, angle=NULL,lineheight=NULL,margin = NULL)
```

```
p <- p + theme(
    axis.title=element_text(color="#e41a1c"),
    axis.text=element_text(color="#377eb8"),
    plot.title=element_text(color="#4daf4a"),
    plot.subtitle=element_text(color="#984ea3"),
    legend.text=element_text(color="#ff7f00"),
    legend.title=element_text(color="#ffff33"),
    strip.text=element_text(color="#a65628")
)
```

Theme • Rect

```
element_rect(fill=NULL,color=NULL,size=NULL,linetype=NULL)
```

```
p <- p + theme(
    plot.background=element_rect(fill="#b3e2cd"),
    panel.background=element_rect(fill="#fdcdac"),
    panel.border=element_rect(fill=NA,color="#cbd5e8",size=3),
    legend.background=element_rect(fill="#f4cae4"),
    legend.box.background=element_rect(fill="#e6f5c9"),
    strip.background=element_rect(fill="#fff2ae")
)
```

Theme • Reuse

```
newtheme <- theme_bw() + theme(
  axis.ticks=element_blank(),
  panel.background=element_rect(fill="white"),
  panel.grid.minor=element_blank(),
  panel.grid.major.x=element_blank(),
  panel.grid.major.y=element_line(size=0.3,color="grey90"),
  panel.border=element_blank(),
  legend.position="top",
  legend.justification="right"
)
```

```
p
```

```
p + newtheme
```

Saving plots

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

- `ggplot2` plots can be saved just like base plots

```
png("plot.png",height=5,width=7,units="cm",res=200)
print(p)
dev.off()
```

- `ggplot2` package offers a convenient function

```
ggsave("plot.png",p,height=5,width=7,units="cm",dpi=200,type="cairo")
```

- Use `type="cairo"` for nicer anti-aliasing
- Note that default units in `png` is pixels while in `ggsave` it's inches

Extensions

- **gridExtra**: Extends grid graphics functionality
- **ggpubr**: Useful functions to prepare plots for publication
- **cowplot**: Combining plots
- **ggthemes**: Set of extra themes
- **ggthemr**: More themes
- **ggsci**: Color palettes for scales
- **ggrepel**: Advanced text labels including overlap control
- **ggmap**: Dedicated to mapping
- **ggraph**: Network graphs
- **ggiraph**: Converting ggplot2 to interactive graphics

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 : 19-Oct-2022 at 12:15:33