Data Visualisation with R
Workshop Part 1

Jazzing up your plots

Presented by Emi Tanaka

Department of Econometrics and Business Statistics

MONASH University

6th Dec 2021 @ Statistical Society of Australia NSW Branch | Zoom

5

Publication-ready plots

The `ggplot` object

- **Modifiable**: `ggplot` object can be modified
- **Generalisable**: `ggplot2` uses a cohesive and complex system under the hood to make many kinds of plots
- **Extensible**: the system can be extended to make specialised plots or add more features if the same "grammar" is adopted

Motor Trend Car Road Tests

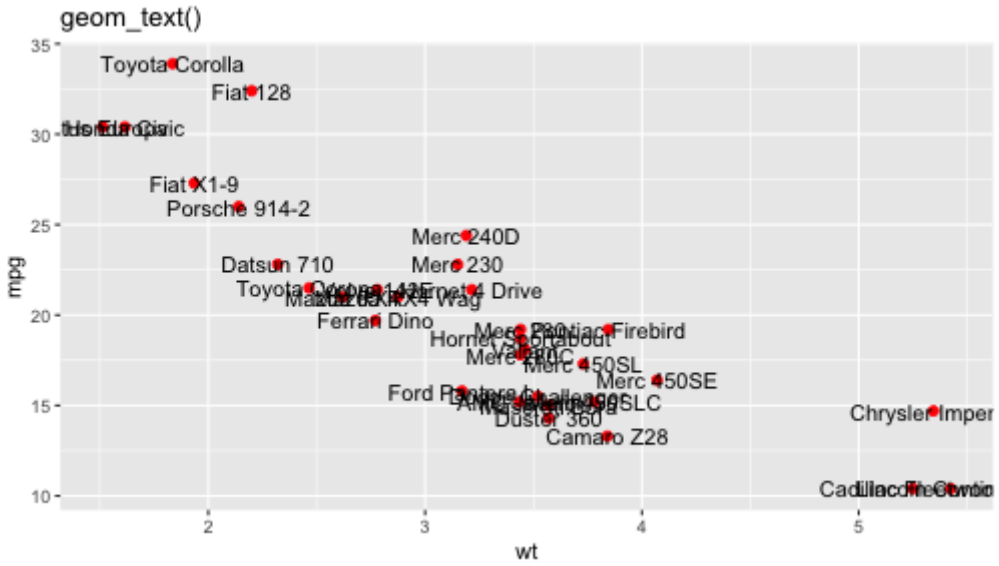
The `mtcars` is from the `datasets` 📦 (loaded by default)

```
mtcars_df <- mtcars %>% rownames_to_column("car")
glimpse(mtcars_df)

## Rows: 32
## Columns: 12
## $ car  <chr> "Mazda RX4", "Mazda RX4 Wag", "Datsun 710", "Hornet 4 Drive", "Ho…
## $ mpg  <dbl> 21.0, 21.0, 22.8, 21.4, 18.7, 18.1, 14.3, 24.4, 22.8, 19.2, 17.8,…
## $ cyl  <dbl> 6, 6, 4, 6, 8, 6, 8, 4, 4, 6, 6, 8, 8, 8, 8, 8, 8, 4, 4, 4, 4, 8,…
## $ disp <dbl> 160.0, 160.0, 108.0, 258.0, 360.0, 225.0, 360.0, 146.7, 140.8, 16…
## $ hp   <dbl> 110, 110, 93, 110, 175, 105, 245, 62, 95, 123, 123, 180, 180, 180…
## $ drat <dbl> 3.90, 3.90, 3.85, 3.08, 3.15, 2.76, 3.21, 3.69, 3.92, 3.92, 3.92,…
## $ wt   <dbl> 2.620, 2.875, 2.320, 3.215, 3.440, 3.460, 3.570, 3.190, 3.150, 3.…
## $ qsec <dbl> 16.46, 17.02, 18.61, 19.44, 17.02, 20.22, 15.84, 20.00, 22.90, 18…
## $ vs   <dbl> 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0,…
## $ am   <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,…
## $ gear <dbl> 4, 4, 4, 3, 3, 3, 3, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 4, 4, 4, 3, 3,…
## $ carb <dbl> 4, 4, 1, 1, 2, 1, 4, 2, 2, 4, 4, 3, 3, 3, 4, 4, 4, 1, 2, 1, 1, 2,…
```

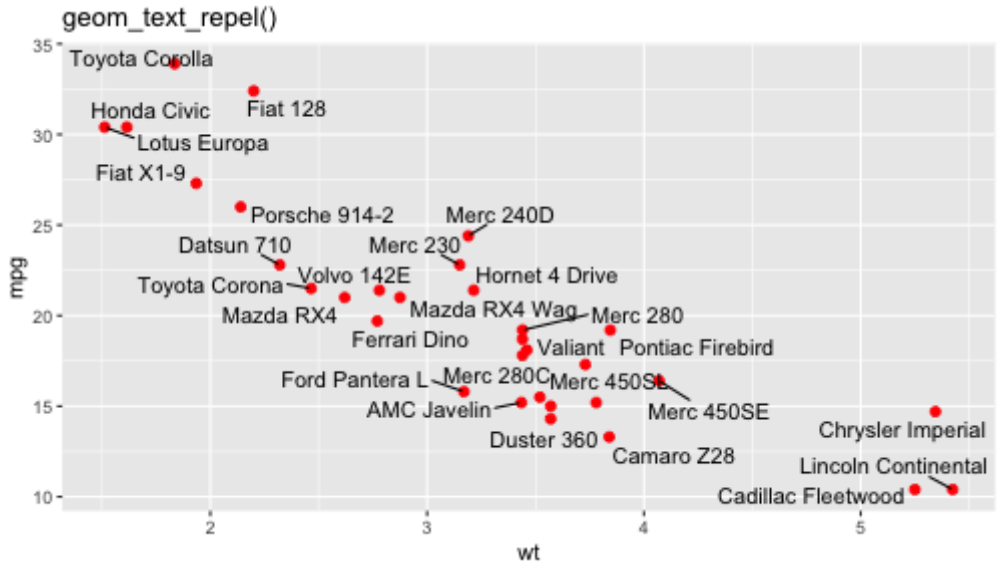
ggrepel one of my favorites!

```
ggplot(mtcars_df, aes(wt, mpg)) +
  geom_point(size = 2, color = "red") +
  geom_text(aes(label = car)) +
  ggtitle("geom_text()")
```



There are many extension packages!

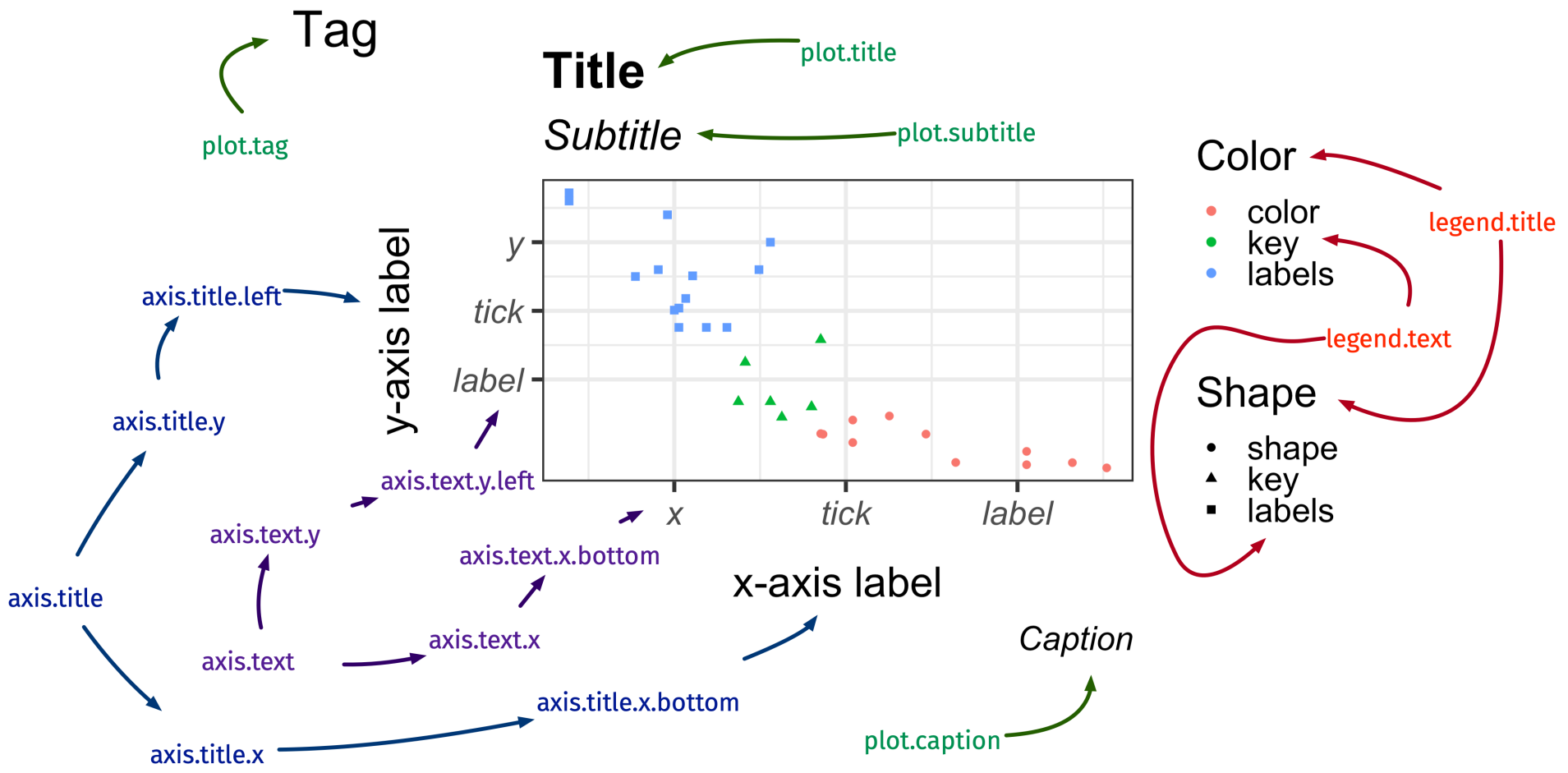
```
library(ggrepel)
set.seed(1)
ggplot(mtcars_df, aes(wt, mpg)) +
  geom_point(size = 2, color = "red") +
  geom_text_repel(aes(label = car))+
  ggtitle("geom_text_repel()")
```



Kamil Slowikowski (2020). ggrepel: Automatically Position Non-Overlapping Text Labels with 'ggplot2'. R package version 0.8.2. https://CRAN.R-project.org/package=ggrepel

Themes

theme: modify the *look* of texts

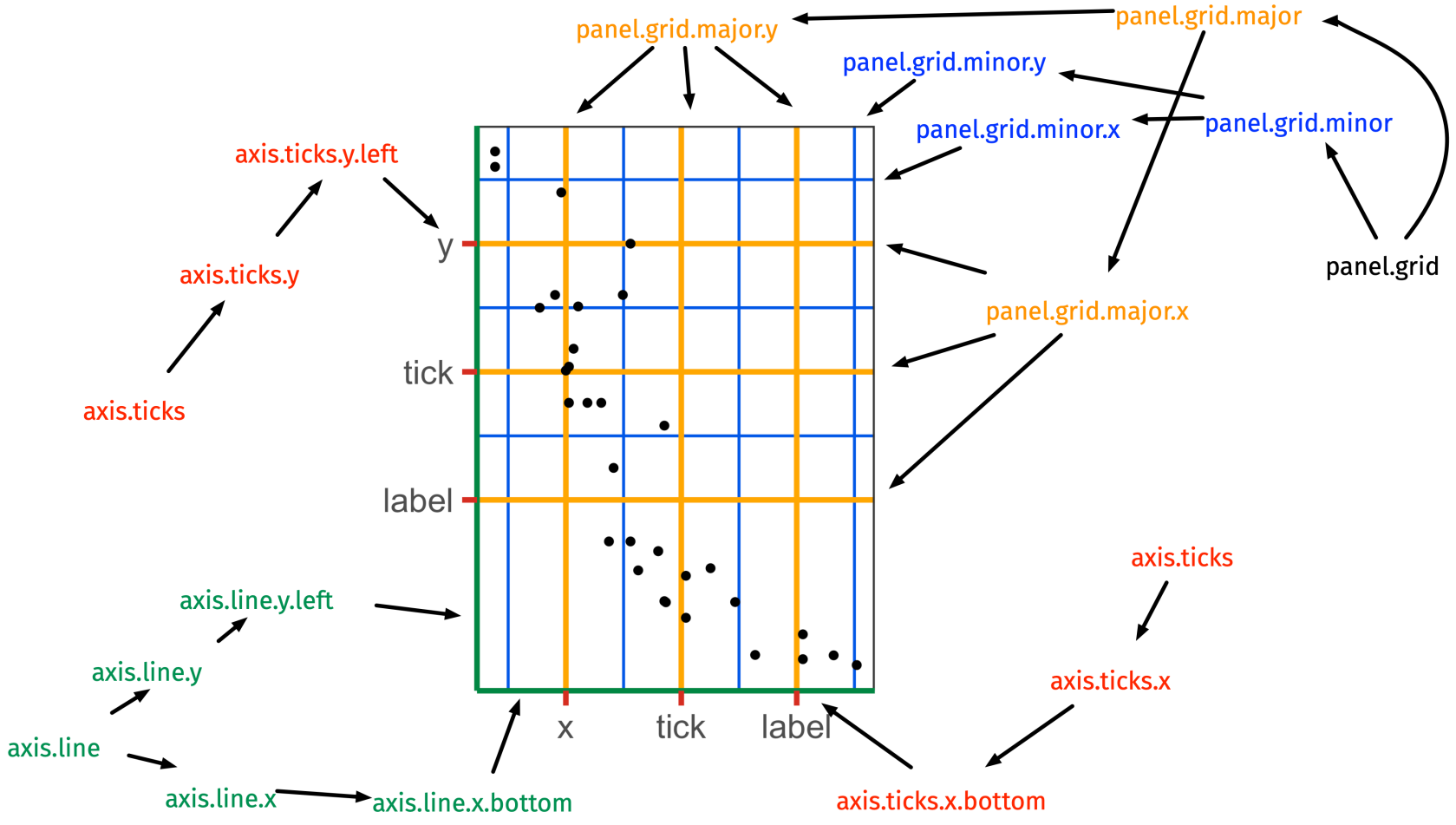


element_text()

element_text()

```
ggplot(diamonds, aes(carat, price)) +
  geom_hex() +
 labs(title = "Diamond") +
 theme(axis.title.x = element_text(size = 30,
                                   color = "red",
                                   face = "bold",
                                   angle = 10,
                                   family = "Fira Code"),
       legend.title = element_text(size = 25,
                                   color = "#ef42eb",
                                   margin = margin(b = 5)),
       plot.title = element_text(size = 35,
                                   face = "bold",
                                   family = "Nunito",
                                   color = "blue"
                                  ))
```

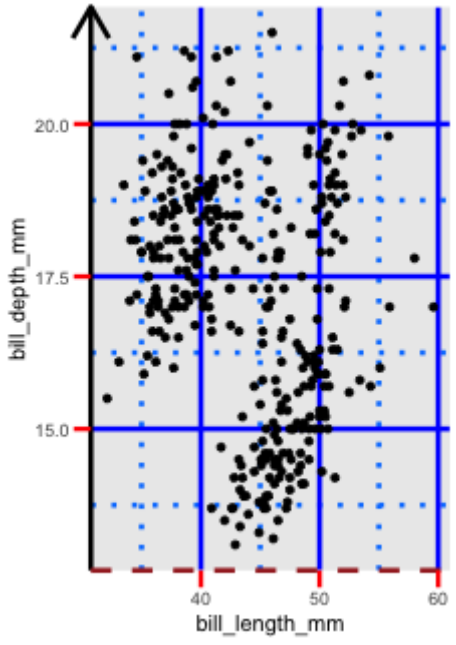
theme: modify the *look* of the lines



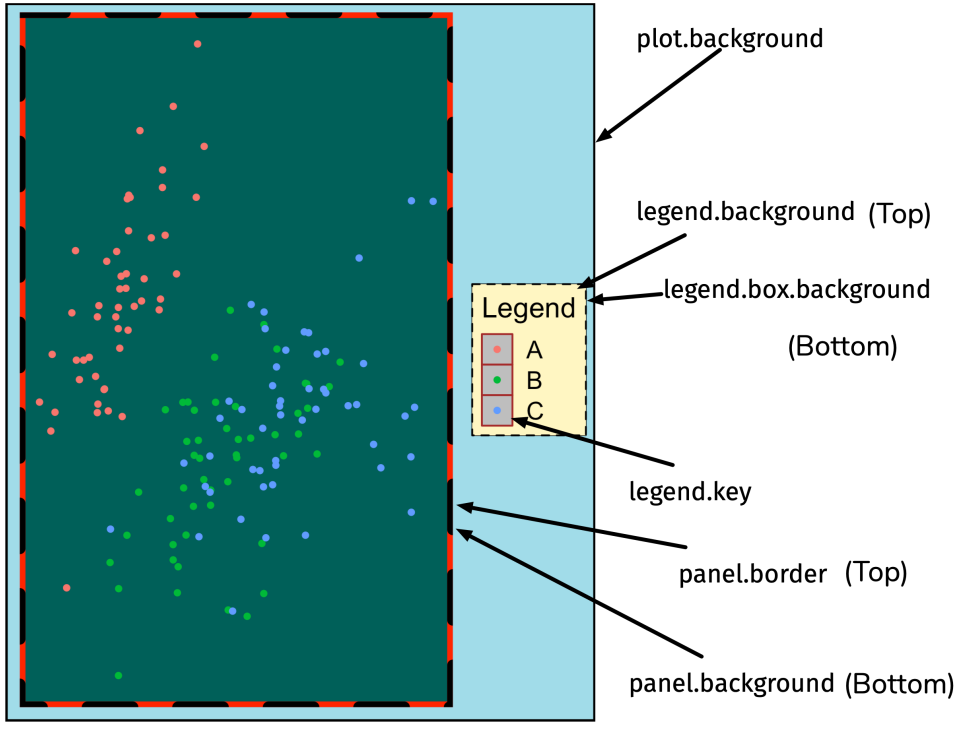
`element_line()`

element_line()

```
ggplot(penguins, aes(bill_length_mm, bill_depth_mm)) +
  geom_point() +
  theme(axis.line.y = element_line(color = "black",
                                   size = 1.2,
                                   arrow = grid::arrow()),
        axis.line.x = element_line(linetype = "dashed",
                                   color = "brown",
                                   size = 1.2),
        axis.ticks = element_line(color = "red", size = 1.1),
        axis.ticks.length = unit(3, "mm"),
        panel.grid.major = element_line(color = "blue",
                                        size = 1.2),
        panel.grid.minor = element_line(color = "#0080ff",
                                        size = 1.2,
                                        linetype = "dotted"))
```



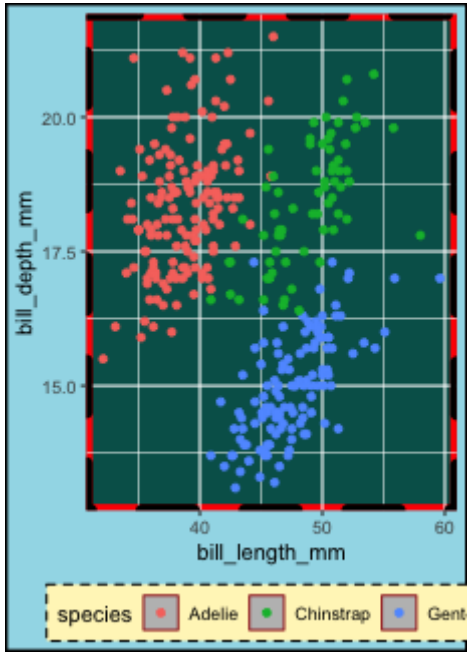
theme: modify the *look* of the rectangular regions



`element_rect()`

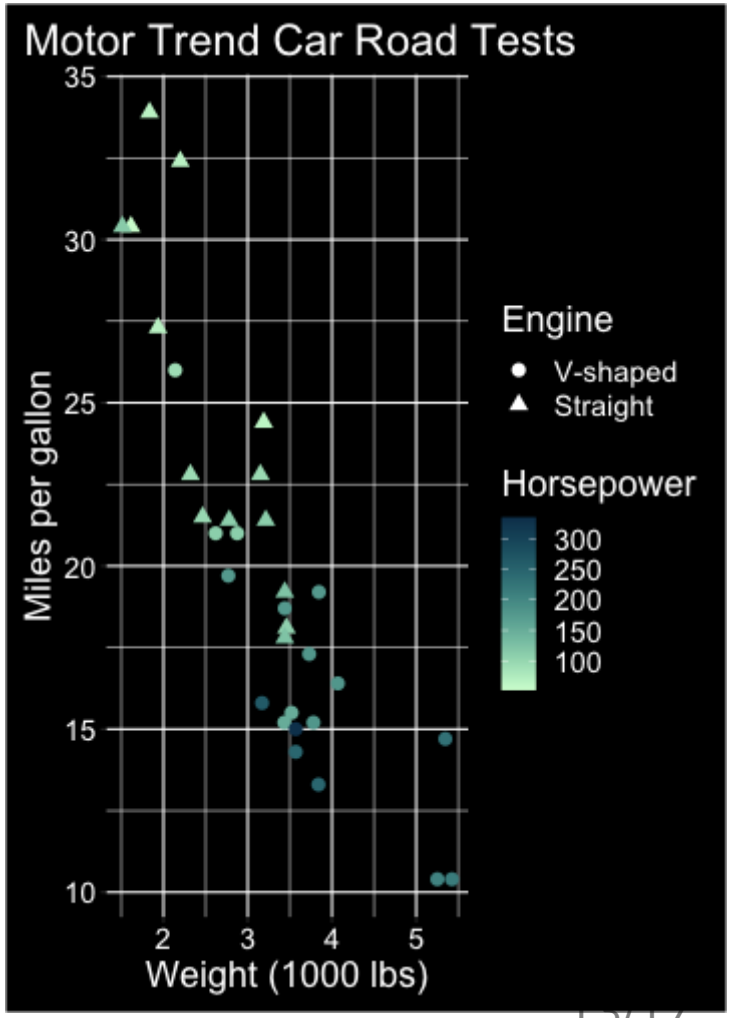
element_rect()

```
ggplot(penguins, aes(bill_length_mm, bill_depth_mm)) +
 geom_point(aes(color = species)) +
 theme(
  legend.background = element_rect(fill = "#fff6c2",
                                   color = "black",
                                  linetype = "dashed"),
  legend.key = element_rect(fill = "grey", color = "brown"),
  panel.background = element_rect(fill = "#005F59",
                                  color = "red", size = 3),
  panel.border = element_rect(color = "black",
                              fill = "transparent",
                     linetype = "dashed", size = 3),
  plot.background = element_rect(fill = "#a1dce9",
                                 color = "black",
                                  size = 1.3),
  legend.position = "bottom")
```



Professional looking plots *without* Adobe Illustrator (or alike)

```
ggplot(mtcars_df,
       aes(wt, mpg, shape = factor(vs), color = hp)) +
 geom_point(size = 3) +
 scale_color_continuous_sequential(palette = "Dark Mint") +
 scale_shape_discrete(labels = c("V-shaped", "Straight")) +
 labs(x = "Weight (1000 lbs)", y = "Miles per gallon",
      title = "Motor Trend Car Road Tests",
      shape = "Engine", color = "Horsepower") +
 theme(text = element_text(size = 18, color = "white"),
       rect = element_rect(fill = "black"),
       panel.background = element_rect(fill = "black"),
       legend.key = element_rect(fill = "black"),
       axis.text = element_text(color = "white"),
       plot.title.position = "plot",
       plot.margin = margin(10, 10, 10, 10)) +
 guides(shape =
      guide_legend(override.aes = list(color = "white")))
```



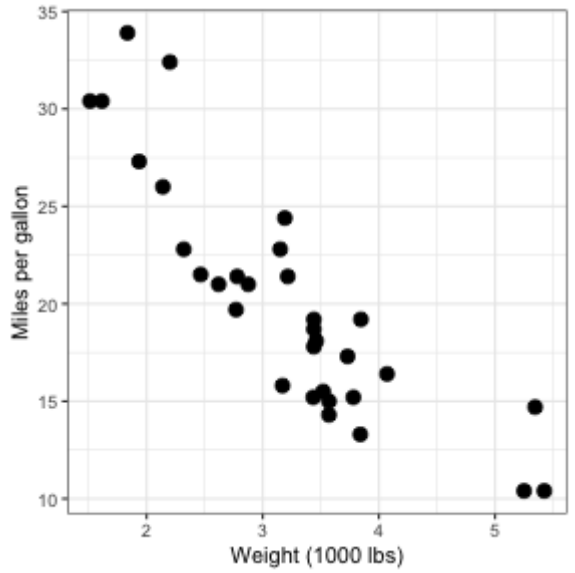
Built-in themes

There are many existing themes

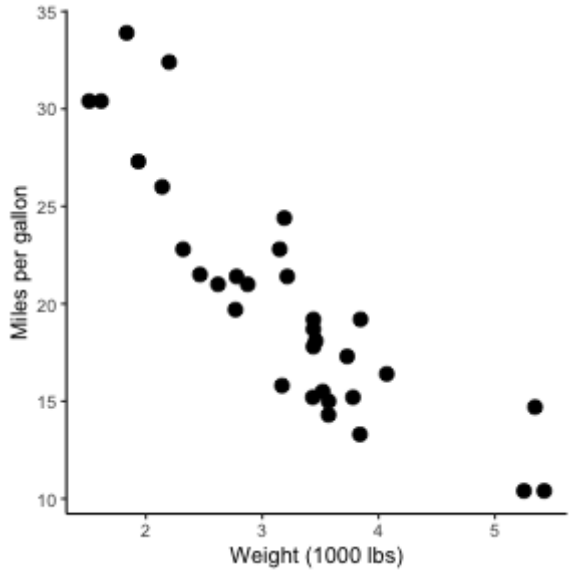
(you can also make your own)

```
g <- ggplot(mtcars_df, aes(wt, mpg)) + geom_point(size = 3) +
  labs(x = "Weight (1000 lbs)", y = "Miles per gallon")
```

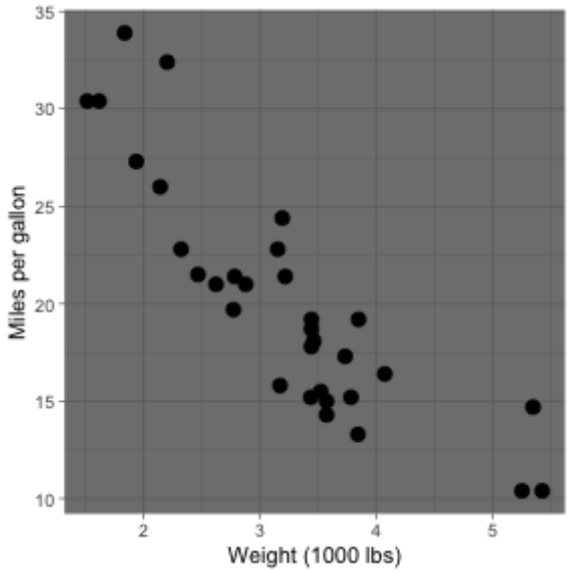
`g + theme_bw()`



`g + theme_classic()`

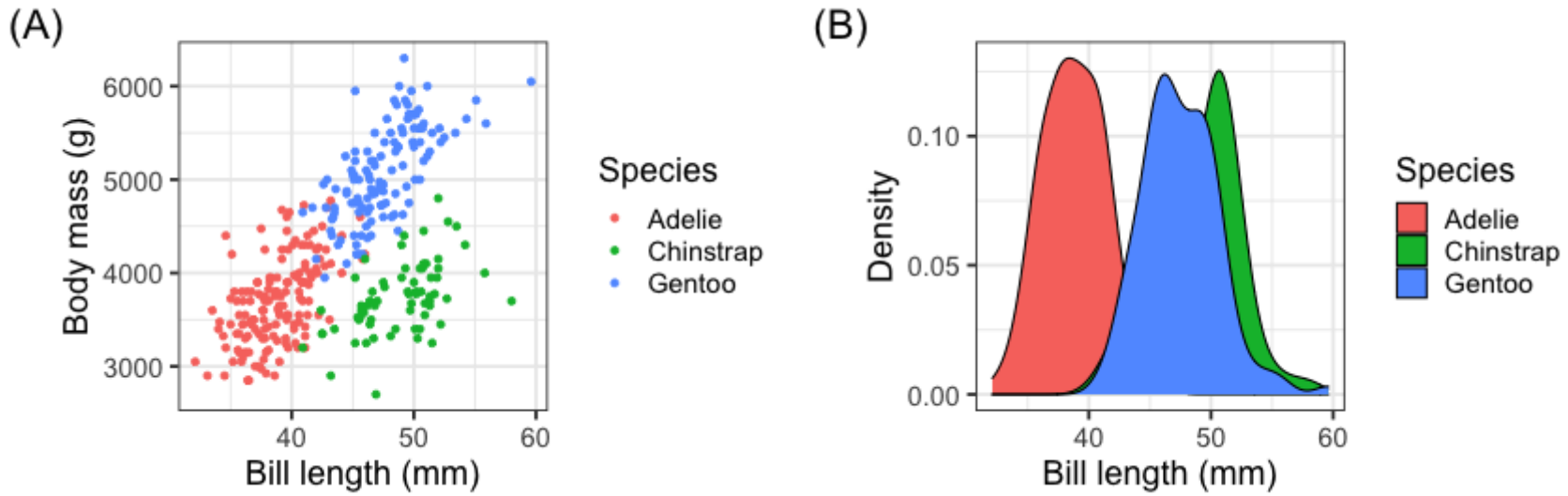


`g + theme_dark()`



Combining plots with patchwork

```
library(patchwork)
g1 <- ggplot(penguins, aes(bill_length_mm, body_mass_g, color = species)) +
  geom_point() + theme_bw(base_size = 18) +
  labs(tag = "(A)", x = "Bill length (mm)", y = "Body mass (g)", color = "Species")
g2 <- ggplot(penguins, aes(bill_length_mm, fill = species)) +
  geom_density() + theme_bw(base_size = 18) +
  labs(tag = "(B)", x = "Bill length (mm)", y = "Density", fill = "Species")
g1 + g2
```



Thomas Lin Pedersen (2020). patchwork: The Composer of Plots. R package version 1.0.1. https://CRAN.R-project.org/package=patchwork

</> Open `part1-exercise-04.Rmd`

15:00

Session Information

```
devtools::session_info()
```

```
## ─ Session info  🗾  📣  😜  ─────────────────────────────────────────────
##  hash: map of Japan, speaker medium volume, winking face with tongue
##
##  setting  value
##  version  R version 4.1.2 (2021-11-01)
##  os       macOS Big Sur 10.16
##  system   x86_64, darwin17.0
##  ui       X11
##  language (EN)
##  collate  en_AU.UTF-8
##  ctype    en_AU.UTF-8
##  tz       Australia/Melbourne
##  date     2021-11-30
##  pandoc   2.11.4 @ /Applications/RStudio.app/Contents/MacOS/pandoc/ (via rmarkdown)
```

These slides are licensed under