

# ggplot-theme

Pawan

2022-12-08

## Themes in ggplot2

**ggthemes package**

**hrbrthemes package**

**ggthemr package**

**ggtech package**

**ggdark package**

## Set an active theme

You can customize the appearance of your plots made with ggplot2 using themes created by other users. Click on the buttons of each section to visualize each theme and its corresponding code.

Note that you can override theme elements making use of the theme function, like the background color, the grid lines or the margins, among others.

**In-built themes** The ggplot2 package comes with eight different themes. By default it uses the theme named `theme_grey` (`theme_gray`), so you don't really need to specify it.

Choose a theme

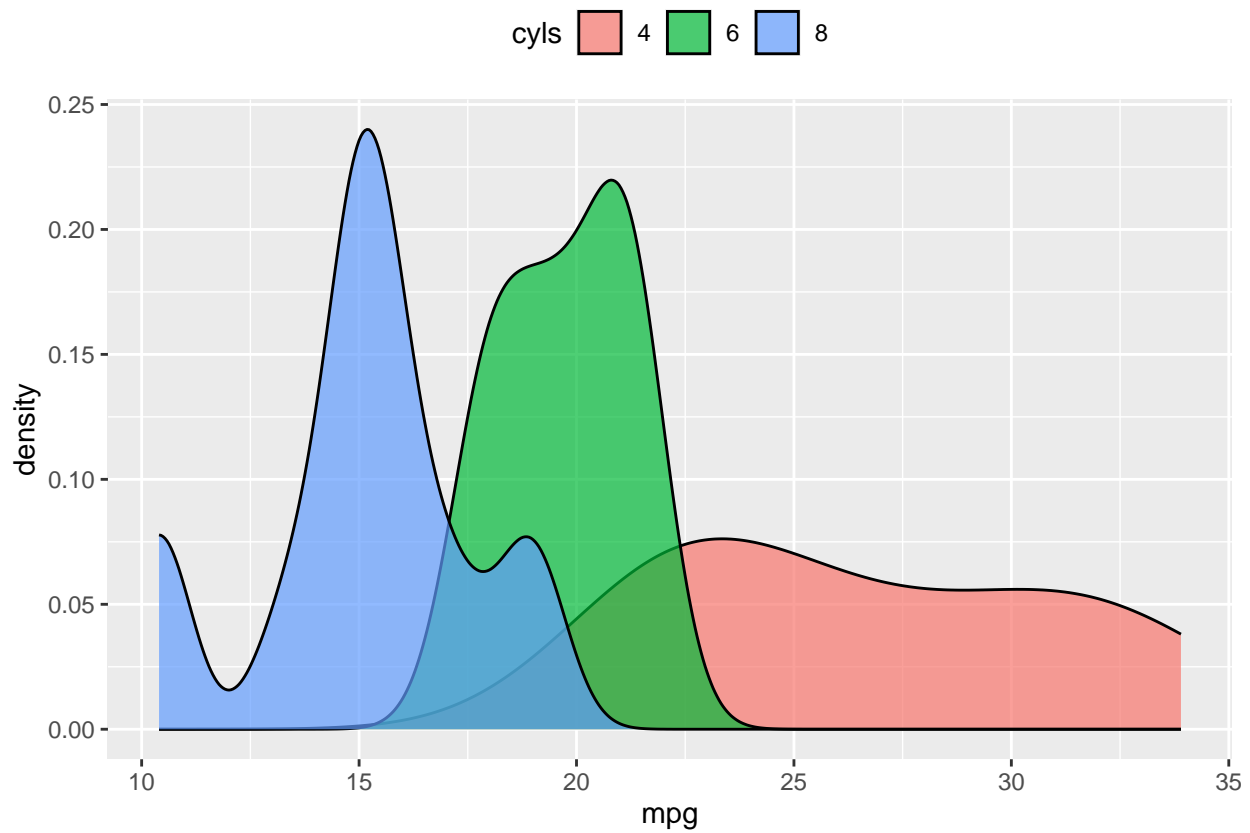
Note that there is an additional theme named `theme_test` to conduct visual unit tests by advanced users.

Default ggplot2 theme

```
library(ggplot2)

cyls <- as.factor(mtcars$cyl)

ggplot(mtcars, aes(x = mpg, fill = cyls)) +
  geom_density(alpha = 0.7) +
  theme_grey() + # Default
  theme(legend.position = "top")
```

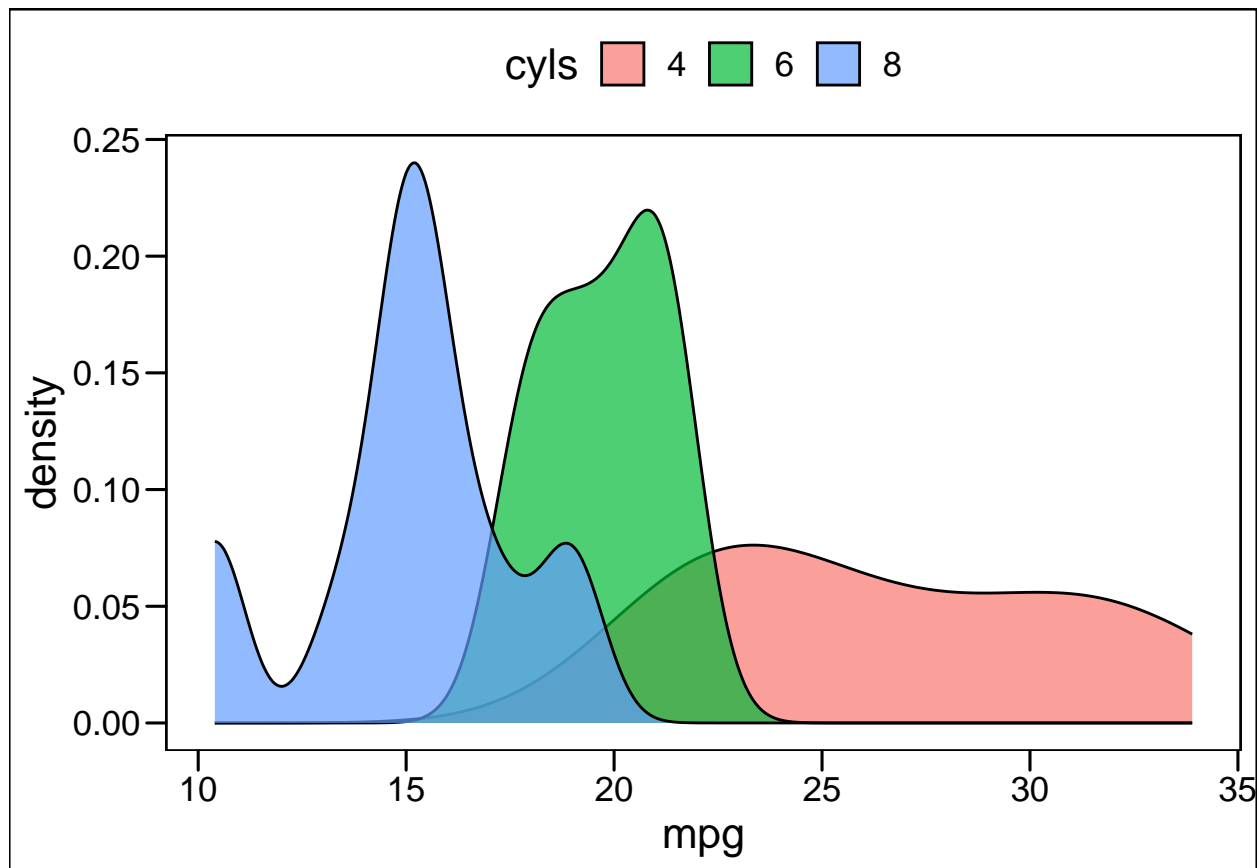


## ggthemes package

```
library(ggplot2)
library(ggthemes)

cyls <- as.factor(mtcars$cyl)

ggplot(mtcars, aes(x = mpg, fill = cyls)) +
  geom_density(alpha = 0.7) +
  theme_base() +
  theme(legend.position = "top")
```



The ggthemes package contains several very popular themes. Some of them also come with their corresponding color scales. Use the scales properly according to your data.

## hrbrthemes package

hrbrthemes provides “typography-centric themes and theme components for ggplot2”. The available themes are listed below, but you can also check the available scales, palettes, fonts and utilities.

```
library(ggplot2)
## install.packages("hrbrthemes")
library(hrbrthemes)

cyls <- as.factor(mtcars$cyl)

ggplot(mtcars, aes(x = mpg, fill = cyls)) +
  geom_density(alpha = 0.7) +
  ggtitle("Plot title") +
  theme_ipsum() + # Arial Narrow
  scale_fill_ipsum() +
  theme(legend.position = "top")
```

## ggthemr package

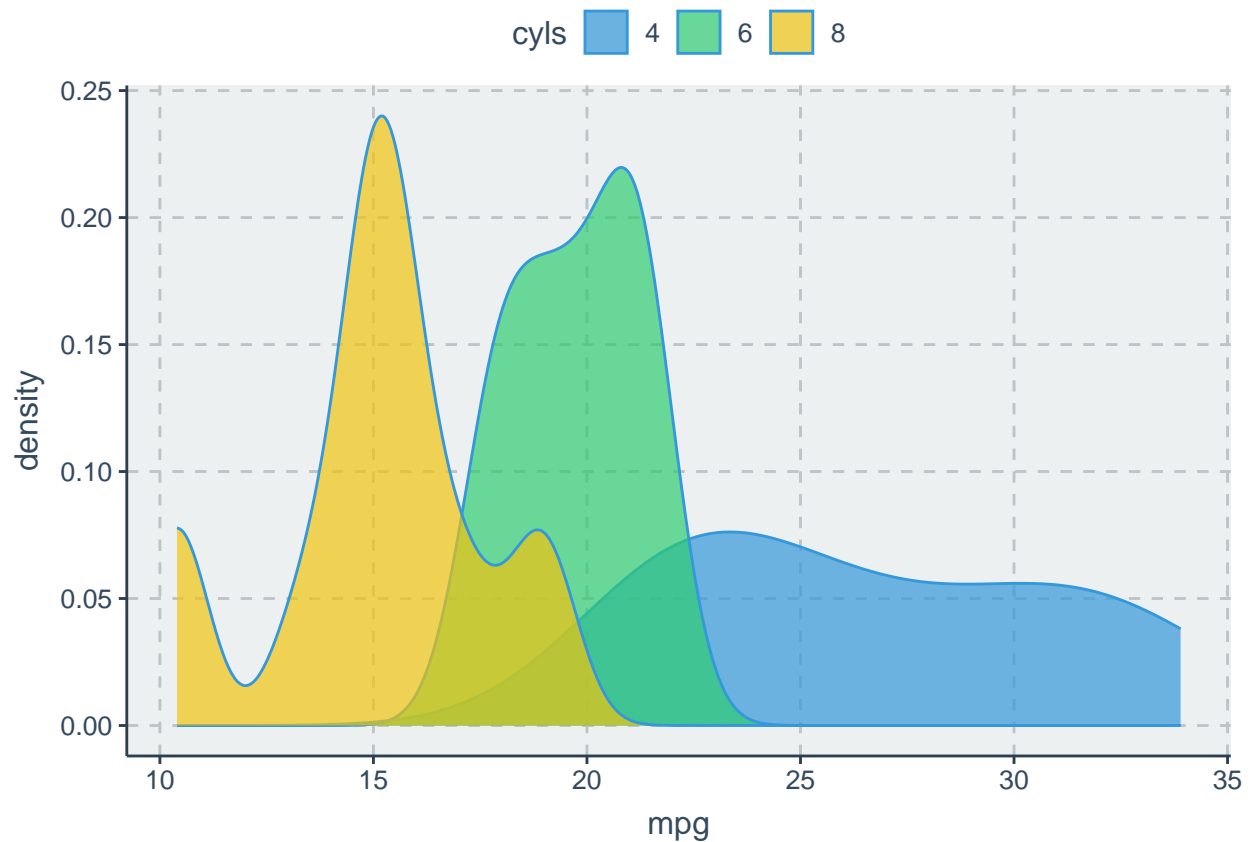
```
library(ggplot2)
##devtools::install_github('Mikata-Project/ggthemr')
```

```
library(ggthemr)

cyls <- as.factor(mtcars$cyl)

ggthemr("flat")

ggplot(mtcars, aes(x = mpg, fill = cyls)) +
  geom_density(alpha = 0.7) +
  theme(legend.position = "top")
```



Note that you will need to call `ggthemr_reset()` to reset to the default `ggplot2` theme.

The `ggthemr` package works different than the other packages. Instead of using the `theme` function and set a theme you can set a theme globally making use of the `ggthemr` function and passing the theme as string as argument.

`ggtech` package The `ggtech` package provides themes inspired by tech companies, such as Airbnb, Google, Twitter or Facebook.

```
library(ggplot2)
##devtools::install_github("ricardo-bion/ggtech",
##                           dependencies=TRUE)
library(ggtech)
```

```
cyls <- as.factor(mtcars$cyl)

ggplot(mtcars, aes(x = mpg, fill = cyls)) +
  geom_density(alpha = 0.7) +
```

```

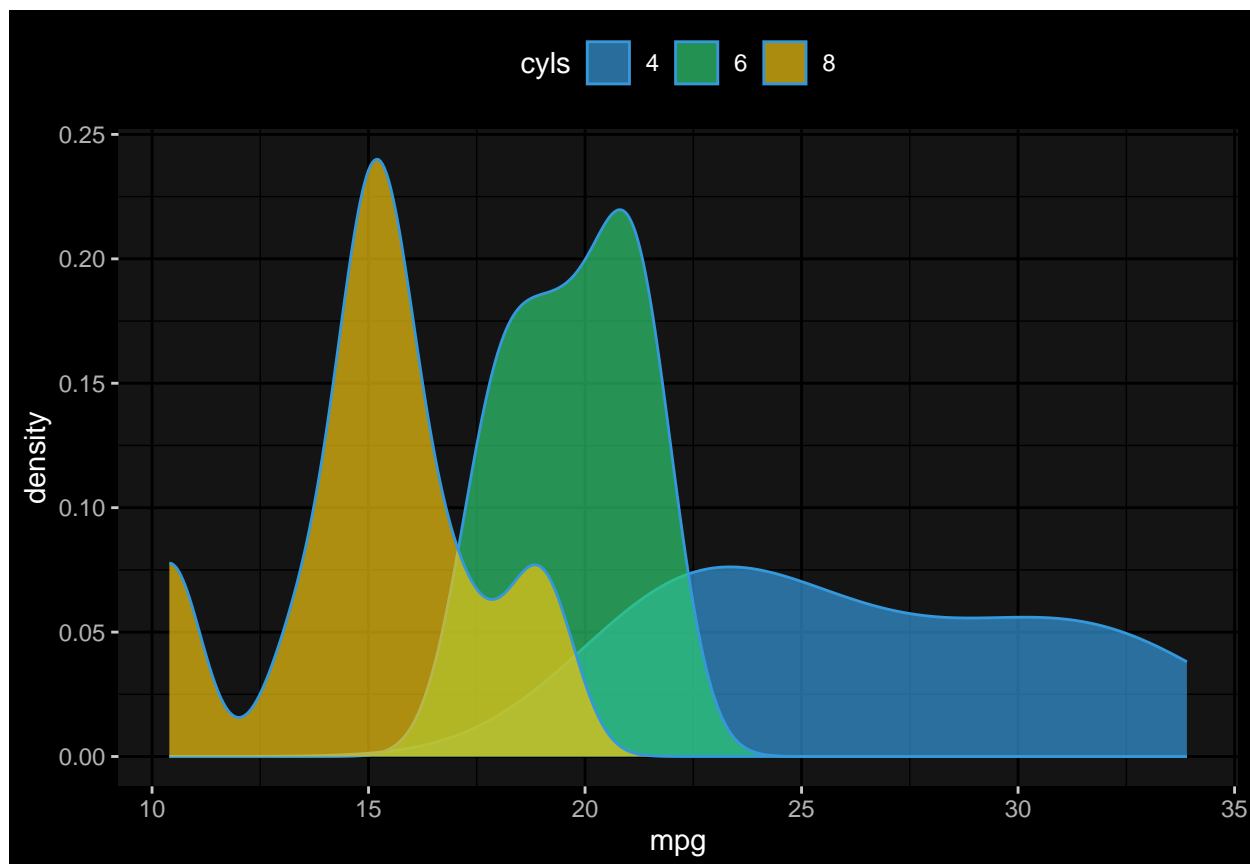
ggtitle("Title of the plot") +
  theme_tech(theme = "airbnb") +
  scale_fill_tech(theme = "airbnb") +
  theme(legend.position = "top")

ggdark package
library(ggplot2)
## install.packages("ggdark")
library(ggdark)

cyls <- as.factor(mtcars$cyl)

ggplot(mtcars, aes(x = mpg, fill = cyls)) +
  geom_density(alpha = 0.7) +
  dark_theme_gray() + # Default
  theme(legend.position = "top")

```



ggdark provides the dark mode themes of the in-built ggplot2 themes. In addition, the package can convert any theme into a dark theme making use of the `dark_mode` function.

Set an active theme The `theme_set` function can be used to set a theme globally. In the following example we are setting the in-built `theme_dark` to all the generated plots. Then, you can you back to the default theme as follows.

```

library(ggplot2)

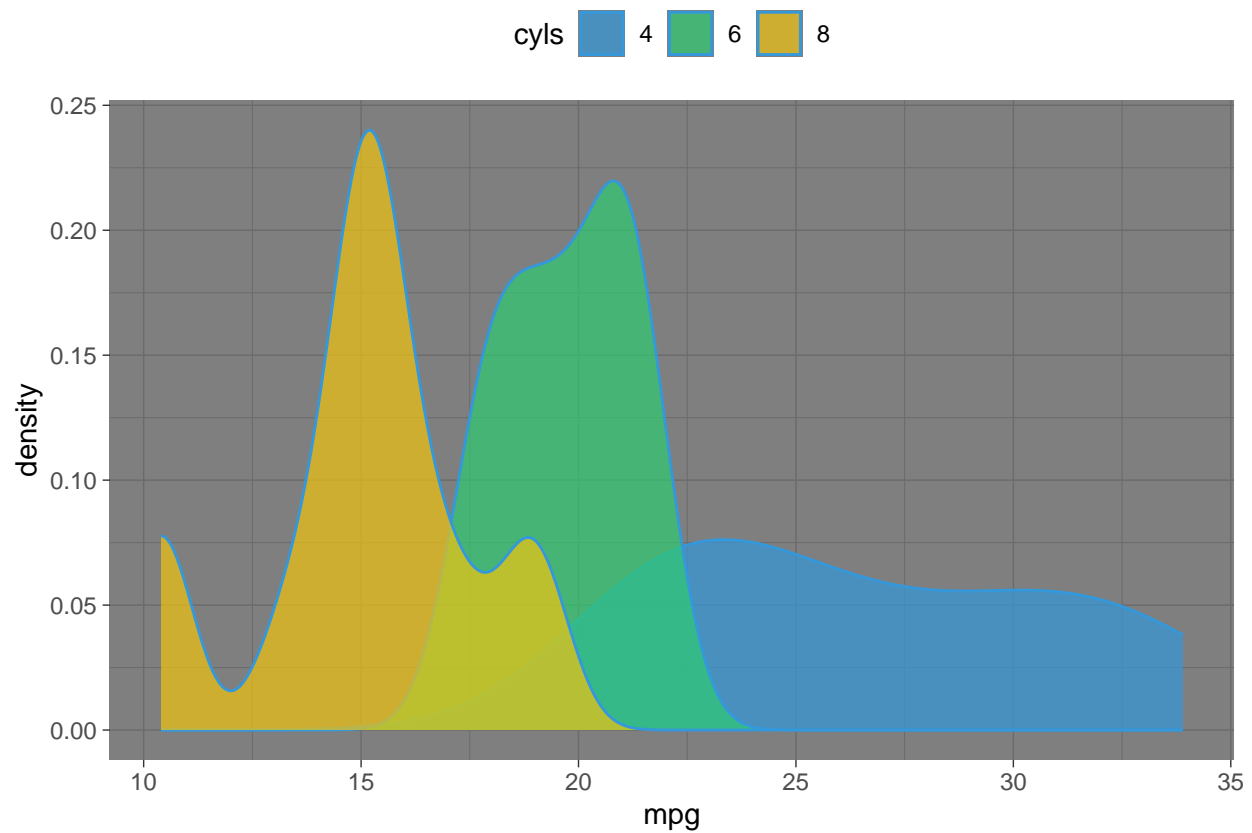
# Global theme

```

```
old <- theme_set(theme_dark())

cyls <- as.factor(mtcars$cyl)

ggplot(mtcars, aes(x = mpg, fill = cyls)) +
  geom_density(alpha = 0.7) +
  theme(legend.position = "top")
```



```
# Reset to default
theme_set(old)
```

Set an active theme in ggplot2 with the `theme_set` function