REVOLUTION ANALYTICS

Custom Themes with ggplot

Revolution Analytics

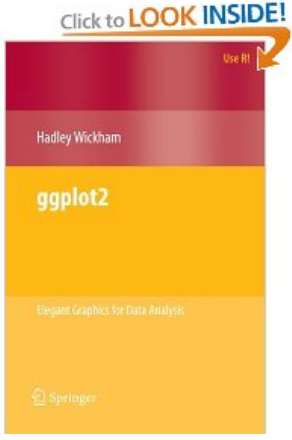
Outline

Intro

Why theme and polish?

- Plotting for the purpose of improving insight and understanding.
- Plotting for the purpose of showing or teaching.

Intro


Click to LOOK INSIDE!
Use R!
Hadley Wickham
ggplot2
Elegant Graphics for Data Analysis
Springer

Intro

- In this section, you change aspects of the plot that are purely aesthetic and not driven by data.
- We'll go over how to define default settings for all successive plots.
- We'll discuss how to prepare your plots for the next step in your analysis presentation/distribution.

Intro

Why use ggplot?

- ggplot separates data centric and non-data plot modifications.
- ggplot makes defining default settings easy.

```
# Recall:
qplot(x = departure.time/100, data = airline, geom = "density", color = carrier,
  xlab = "Departure Time")

# Equivalently:
ggplot(airline, aes(departure.time/100, color = carrier)) + geom_density(na.rm = TRUE) +
  xlab("Departure Time") + ylab("density")
```

Getting started

Read the data and load the required packages

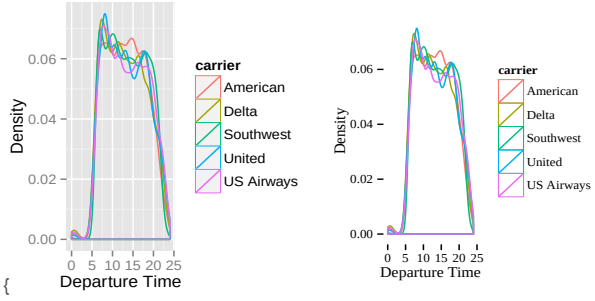
```
dataPath <- "../data"
rdataFile <- file.path(dataPath, "airline.RData")
load(rdataFile)

library(ggplot2)
library(gridExtra)
library(ggthemes)

theme <- theme_gray()
theme <- theme_update(plot.margin = unit(c(0.2, 0, 0, 0), "lines"))
theme <- theme_update(plot.background = element_blank())
```

A first look at themes

```
p1 <- ggplot(airline, aes(departure.time/100, color = carrier)) +
  xlab("Departure Time") + ylab("Density") + geom_density(na.rm = TRUE)
grid.arrange(p1, p1 + theme_tufte(), ncol = 2)  # gridExtra
```



{

}

The ggthemes package

Theme	Description
`theme_tufte`	Based on Tufte's *The Visual Display of Quantitative Information*.
`theme_solarized`	Uses the solarized color palette.
`theme_stata`	Based on Stata graph schemes.
`theme_economist`	Based on the plots in the *The Economist* magazine.
`theme_excel`	Replicates the classic ugly gray charts in Excel
`theme_wsj`	Based on the plots in the the *Wall Street Journal* magazine.
`theme_few`	Stephen Few's "Practical Rules for Using Color in Charts".
`theme_calc`	Based on LibreOffice Calc.
`theme_gdocs`	Based on Google Docs.

More info available at: github.com/jrnold/ggthemes

Outline

1. Theming your ggplot
2. Theme() arguments
3. Custom themes

Theme() Args

Beyond predefined: customize using `theme()` arguments.

Theme Documentation available here

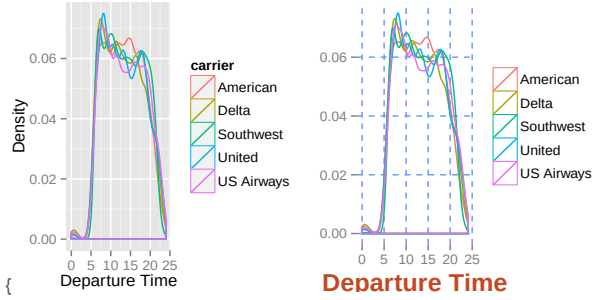
Create a New Theme

```
mytheme <- theme(legend.title = element_blank(),
                 legend.key = element_rect(fill = "white"),
                 axis.title.x = element_text(vjust=0, size=rel(1.5),
                                             colour="#C34922",
                                             face="bold"),
                 axis.title.y = element_blank(),
                 plot.title =   element_text(hjust=0, size=rel(2),
                                             colour="#C34922",
                                             face="bold"),
                 panel.grid.major = element_line(colour="#6699FF",
                                                 linetype = 'dashed'),
                 panel.grid.minor = element_blank(),
                 panel.background = element_rect(fill = 'white'),
                 axis.line = element_blank())
```

Use it!

```
p2 <- p1 + mytheme
grid.arrange(p1, p2, ncol = 2)
```

Theme() Args


Density
0.06
0.04
0.02
0.00
0 5 10 15 20 25
Departure Time
carrier
American
Delta
Southwest
United
US Airways
{
0.06
0.04
0.02
0.00
0 5 10 15 20 25
Departure Time
American
Delta
Southwest
United
US Airways
}

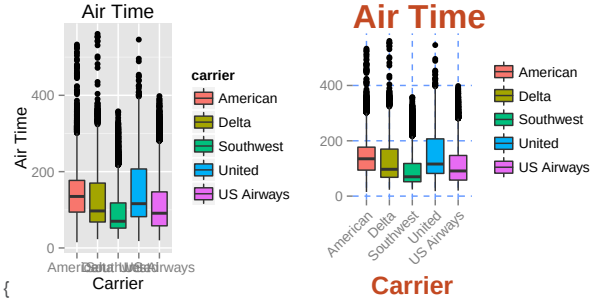
Theme() Args

(Almost) the same `theme()` arguments, different **geom**!

```
p1 <- ggplot(airline, aes(x = carrier, y = air.time, fill = carrier)) +
  xlab("Carrier") + ylab("Air Time") + ggtitle("Air Time") + geom_boxplot()
p2 <- p1 + mytheme + theme(axis.text.x = element_text(angle = 45,
  hjust = 1))
grid.arrange(p1, p2, ncol = 2)
```

What's the one exception to the theme?

See It!


Air Time
Air Time
400
200
0
American
Delta
Southwest
United
US Airways
Carrier
carrier
American
Delta
Southwest
United
US Airways
Air Time
400
200
0
American
Delta
Southwest
United
US Airways
Carrier

{

}

Outline

1. Theming your ggplot
2. Theme() arguments
3. Custom themes

What is a theme?

There is a better way and we saw it in action already!

The Tufte theme was built by modifying an existing theme, which was built, in turn, by modifying the standard black and white theme.

```
theme_tufte
```

```
## function (ticks = TRUE, base_family = "serif", base_size = 11)
## {
##     ret <- theme_bw(base_family = base_family, base_size = base_size) +
##         theme(legend.background = element_blank(), legend.key = element_blank(),
##             panel.background = element_blank(), panel.border = element_blank(),
##             strip.background = element_blank(), plot.background = element_blank(),
##             axis.line = element_blank(), panel.grid = element_blank())
...
```

Taking a look at `theme_bw`

Similarly, `theme_bw` is based on `theme_grey`

```
theme_bw
```

```
## function (base_size = 12, base_family = "")
## {
##     theme_grey(base_size = base_size, base_family = base_family) %+replace%
##      theme(axis.text = element_text(size = rel(0.8)), axis.ticks = element_line(colour = "black")
##         legend.key = element_rect(colour = "grey80"), panel.background = element_rect(fill = "whi
##                 colour = NA), panel.border = element_rect(fill = NA,
##               colour = "grey50"), panel.grid.major = element_line(colour = "grey90",
...
```

So what is theme_grey?

```
theme_grey
```

```
## function (base_size = 12, base_family = "")
## {
##     theme(line = element_line(colour = "black", size = 0.5, linetype = 1,
##         lineend = "butt"), rect = element_rect(fill = "white",
##      colour = "black", size = 0.5, linetype = 1), text = element_text(family = base_family,
##         face = "plain", colour = "black", size = base_size, hjust = 0.5,
##      vjust = 0.5, angle = 0, lineheight = 0.9), axis.text = element_text(size = rel(0.8),
...
```

Writing a custom theme

We will create our own theme by modifying an existing theme in `ggthemes`.

First, we grab some fonts.

blog.revolutionanalytics.com:
how-to-use-your-favorite-fonts-in-r-charts

```
# {extrafont} font_import() # you'll need to run this the first
# time you load extrafont (library(extrafont)) it will take some
# time, but does not need to be repeated
library(extrafont)
fonts()
fonttable()
```

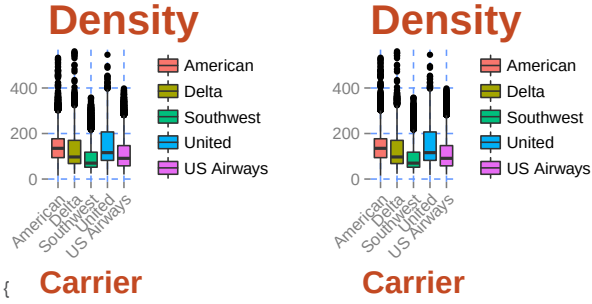
See also here

Writing a custom theme

```
theme_revo <- function(base_size = 14, base_family = "Helvetica") {
  theme_grey(base_size = base_size, base_family = base_family) +
    theme(legend.title = element_blank(), legend.key = element_blank(),
      axis.title.x = element_text(vjust = 0, size = rel(1.5),
        colour = "#C34922", face = "bold"), axis.text.x = element_text(angle = 45,
        hjust = 1), axis.title.y = element_blank(), plot.title = element_text(rel(2),
        hjust = 0, vjust = 1, family = base_family, colour = "#C34922",
        face = "bold"), panel.grid.major = element_line(colour = "#6699FF",
        linetype = "dashed"), panel.grid.minor = element_blank(),
      panel.background = element_rect(fill = "white"), axis.line = element_blank())
}
```

Creating a revo theme

```
p3 <- p1 + labs(title = "Density") + theme_revo()
p4 <- p1 + labs(title = "Density") + theme_revo() + scale_colour_tableau("colorblind10")
grid.arrange(p3, p4, ncol = 2)
```



{ }

Reproducing some classic themes

Scale	Function
Solarized colors	`scale_color_solarized`
The economist	`scale_color_economist`
Stata graph schemes	`scale_color_stata`, `scale_shapes_stata`, `scale_linetype_stata`
Excel	`scale_color_excel`
Tableau	`scale_color_tableau`, `scale_shape_tableau`
Cleveland (1995)	`scale_shape_cleveland`
Tremmel (1995)	`scale_shape_tremmel`
Circle fill	`scale_shape_circlefill` - Lewandowsky and Spence (1989).

See also here.

Working with scales

Some more scales

Scale	Meaning
`scale_color_few`	color palettes from Stephen Few's "Practical Rules for Using Color in Charts".
`scale_color_colorblind`	Colorblind safe palette from [`http://jfly.iam.u-tokyo.ac.jp/color/`]
`scale_color_gdocs`	color palette from Google Docs.
`scale_color_calc`	Color palettes from LibreOffice Calc.
`scale_shape_calc`	Shape palettes from LibreOffice Calc.

For more flexibility and customization see 'Scales' within ggplot2 documentation

Adding images to plots

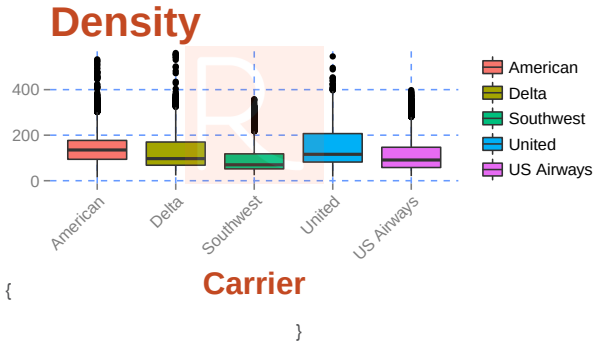
```
library(png)

m <- readPNG(file.path(dataPath,"RA_ICON.png"), FALSE)
w <- matrix(rgb(m[, , 1], m[, , 2], m[, , 3], m[, , 4] * 0.1), # adjust alpha (transparency)
            nrow=dim(m)[1])
(p2 <- p1 + labs(title="Density") +
  theme_revo() +
  scale_colour_tableau("colorblind10") +
  annotation_custom(xmin=-Inf, ymin=-Inf, xmax=Inf, ymax=Inf,
                    rpatternGrob(motif=w, motif.width = unit(3, "cm"))))
```

Colorblind theme

```
m <- readPNG(file.path(dataPath, "RA_ICON.png"), FALSE)
w <- matrix(rgb(m[, , 1], m[, , 2], m[, , 3], m[, , 4] * 0.1), nrow = dim(m)[1])
# adjust alpha (transparency)
(p2 <- p1 + labs(title = "Density") + theme_revo() + scale_colour_tableau("colorblind10") +
  annotation_custom(xmin = -Inf, ymin = -Inf, xmax = Inf, ymax = Inf,
    rasterGrob(w)))
```

Colorblind theme



For more information about working with GIF and JPEG images see this link.

Summary

You can use themes to customize the look and brand of graphics created with `ggplot2`.

Questions?

REVOLUTION ANALYTICS

Thank you

Revolution Analytics is the leading commercial provider of software and support for the popular open source R statistics language.

www.revolutionanalytics.com
1.855.GET.REVO
Twitter: @RevolutionR