Sponsors:

JOHN McNEIL & Company, Inc.

REVOLUTION ANALYTICS

Data visualization with R

San Diego R Users Group | June 2013

Kevin Davenport
kldavenport.com
kevin.davenport@compliancemetrix.com

CMX ComplianceMetrix

What is ggplot2?

- ggplot2 is a data visualization package for R. Created by Hadley Wickham in 2005

- ggplot2 is an implementation of Leland Wilkinson's Grammar of Graphics—a general scheme for data visualization which breaks up graphs into semantic components such as scales and layers.

Advantages of ggplot2

- Consistent underlying grammar of graphics (Wilkinson, 2005)
- Plot specification at a high level of abstraction
- Theme system to refine plot appearance
- Very active development
- Active google group (3786 members as of today)

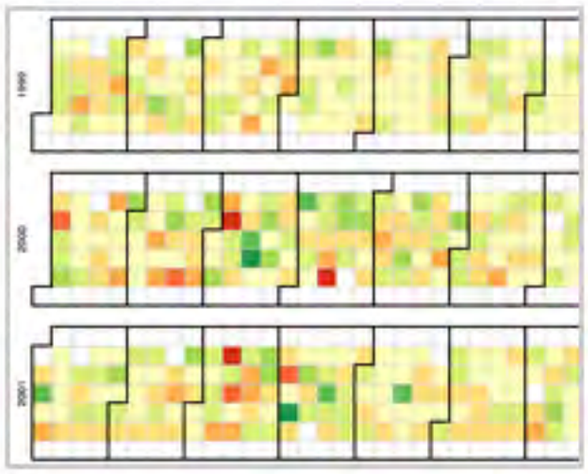
Grammar of graphics

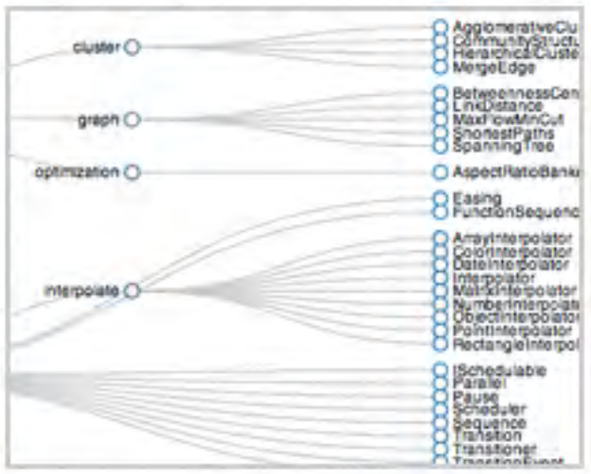
The basic idea: independently specify plot building blocks

Anatomy of a plot:

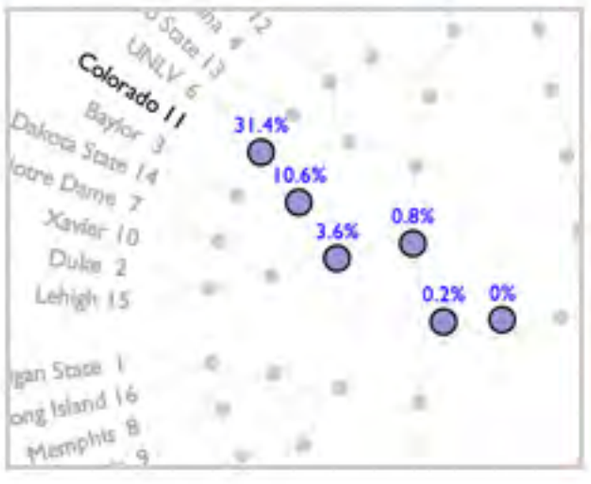
- data
- aesthetic mapping
- geometric object
- statistical transformations
- scales
- coordinate system
- position adjustments
- faceting

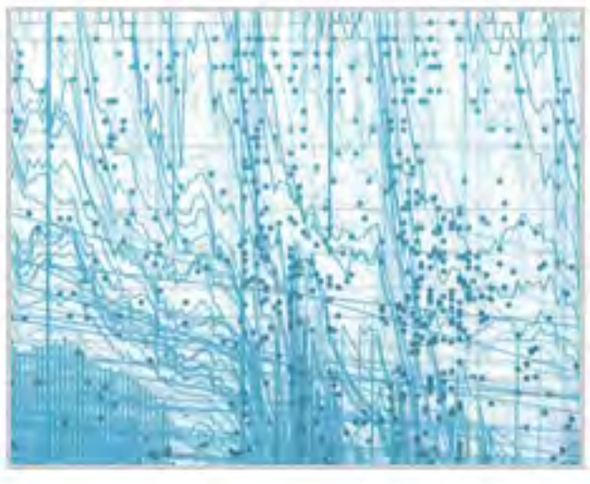
Data-Driven Documents











D3.js is a JavaScript library for manipulating documents based on data. **D3** helps you bring data to life using HTML, SVG and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.

See more examples.

Download the latest version here:

- d3.v3.zip

ggplot2 elements

A plot is comprised of multiple layers

A layer consists of data and a:

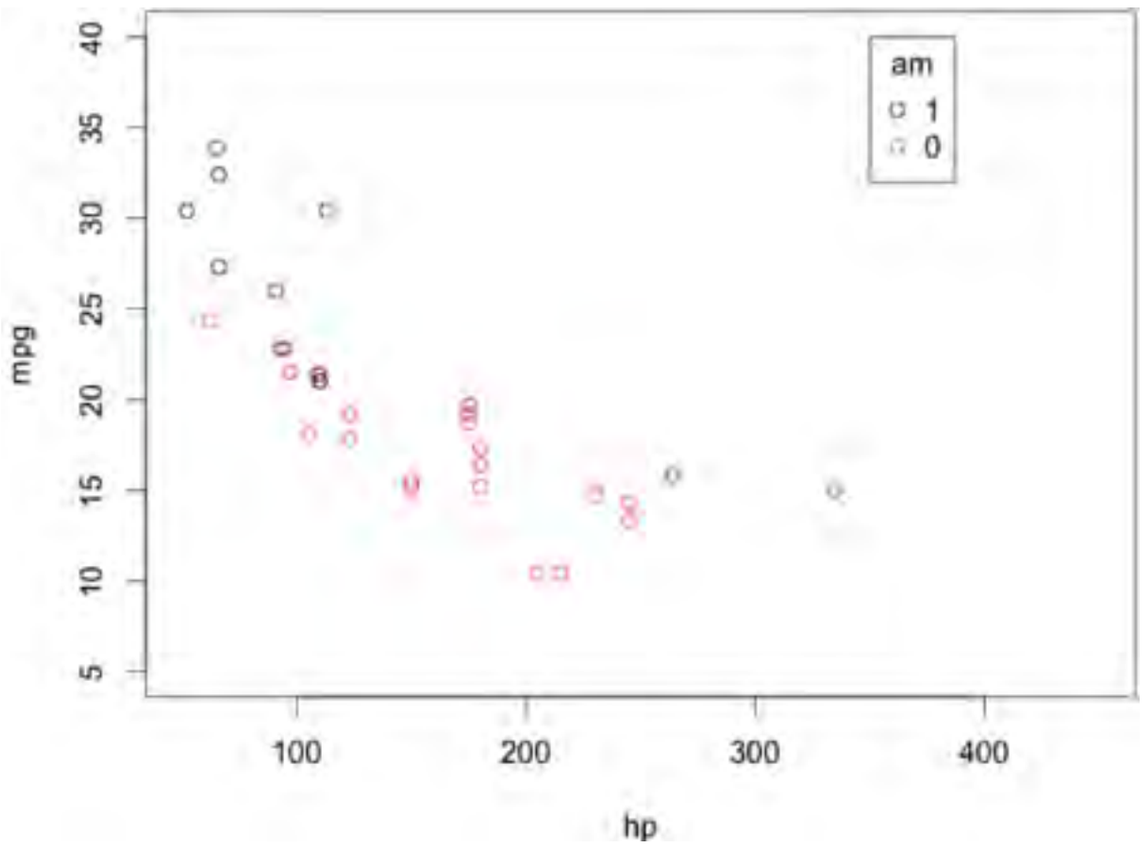
- set of mappings between variables and aesthetics
- geometric object
- statistical transformation

Scales control the details of the mapping.

All components are independent and reusable.

Base graphics vs ggplot: Scatterplots

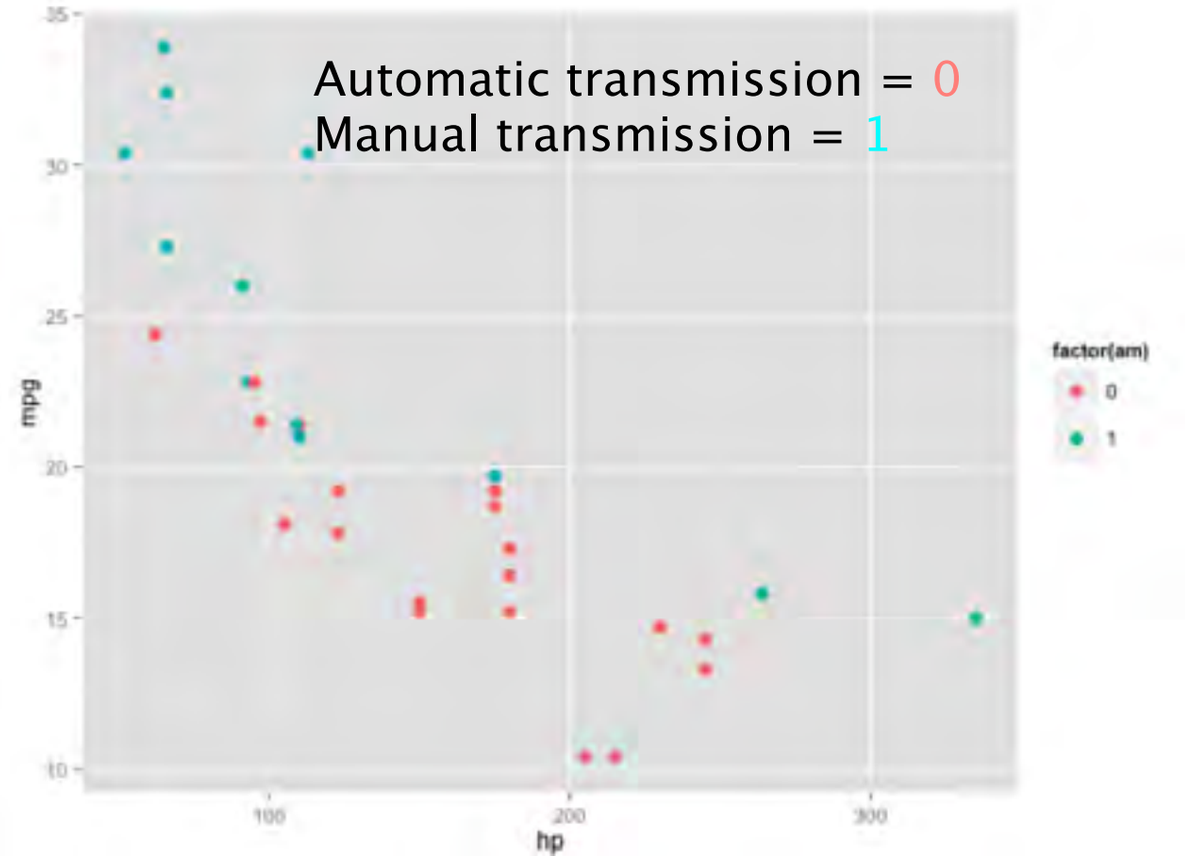
```
par(mar = c(4,4,.1,.1))
    plot(mpg ~ hp,
    data=subset(mtcars, am==1),
    xlim=c(50, 450),ylim=c(5, 40))
    points(mpg ~ hp, col="red",
    data=subset(mtcars, am==0))
    legend(350, 40,
    c("1", "0"), title="am",
    col=c("black", "red"),
    pch=c(1, 1))
```




```
ggplot(mtcars, aes(x=hp,
y=mpg,
color=factor(am)))+
geom_point()
```

OR

```
qplot(hp,mpg,data=mtcars,color=factor
(am))
```

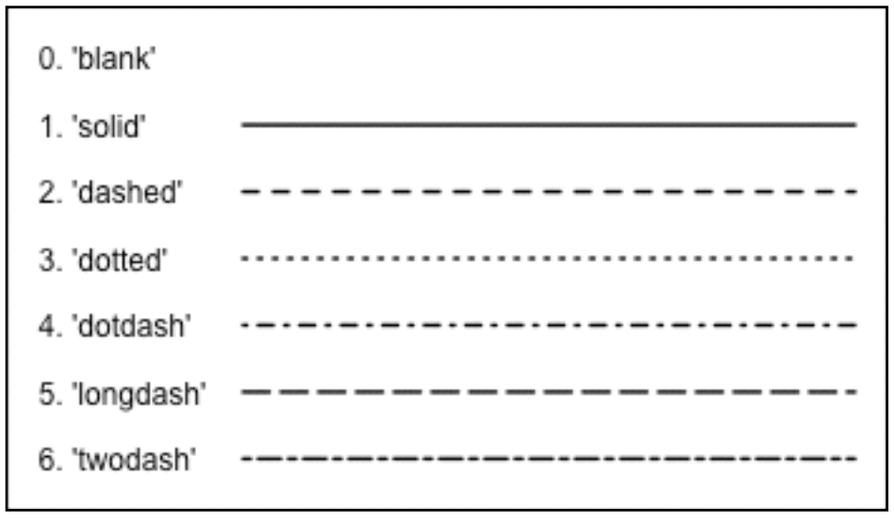


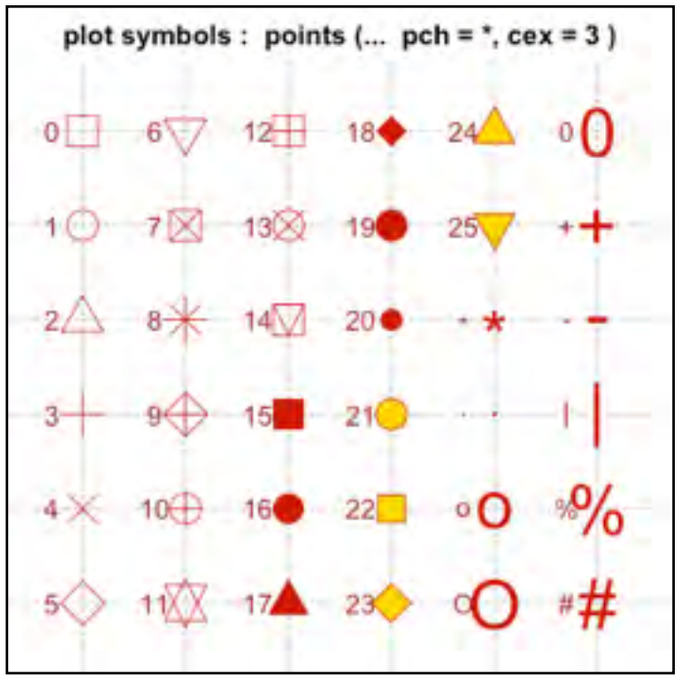
1. Aesthetics and geometric Objects
2. Statistical Transformations
3. Scales
4. Faceting
5. Themes
6. Putting It All Together

1. Aesthetics and geometric Objects
2. Statistical Transformations
3. Scales
4. Faceting
5. Themes
6. Project Example

Aesthetics and geometric Objects:
Mapping aesthetics

- Aesthetic mappings are set with the aes( ) function
- Position (x and y)
- Color (outline color)
- Fill
- Shape (point shapes)
- Line type
- Size
- Each geom accepts only a subset of all aesthetics





Name	Default stat	Aesthetics
abline	abline	colour, linetype, size
area	identity	colour, fill, linetype, size, **x**, **y**
bar	bin	colour, fill, linetype, size, weight, **x**
bin2d	bin2d	colour, fill, linetype, size, weight, **xmax**, **xmin**, **ymax**, **ymin**
blank	identity	
boxplot	boxplot	colour, fill, **lower**, **middle**, size, **upper**, weight, **x**, **ymax**, **ymin**
contour	contour	colour, linetype, size, weight, **x**, **y**
crossbar	identity	colour, fill, linetype, size, **x**, **y**, **ymax**, **ymin**
density	density	colour, fill, linetype, size, weight, **x**, **y**
density2d	density2d	colour, linetype, size, weight, **x**, **y**
errorbar	identity	colour, linetype, size, width, **x**, **ymax**, **ymin**
freqpoly	bin	colour, linetype, size
hex	binhex	colour, fill, size, **x**, **y**
histogram	bin	colour, fill, linetype, size, weight, **x**
hline	hline	colour, linetype, size
jitter	identity	colour, fill, shape, size, **x**, **y**
line	identity	colour, linetype, size, **x**, **y**
linerange	identity	colour, linetype, size, **x**, **ymax**, **ymin**
path	identity	colour, linetype, size, **x**, **y**
point	identity	colour, fill, shape, size, **x**, **y**
pointrange	identity	colour, fill, linetype, shape, size, **x**, **y**, **ymax**, **ymin**
polygon	identity	colour, fill, linetype, size, **x**, **y**
quantile	quantile	colour, linetype, size, weight, **x**, **y**
rect	identity	colour, fill, linetype, size, **xmax**, **xmin**, **ymax**, **ymin**
ribbon	identity	colour, fill, linetype, size, **x**, **ymax**, **ymin**
rug	identity	colour, linetype, size
segment	identity	colour, linetype, size, **x**, **xend**, **y**, **yend**
smooth	smooth	alpha, colour, fill, linetype, size, weight, **x**, **y**
step	identity	colour, linetype, size, **x**, **y**
text	identity	angle, colour, hjust, **label**, size, vjust, **x**, **y**
tile	identity	colour, fill, linetype, size, **x**, **y**
vline	vline	colour, linetype, size

Table 4.3: Default statistics and aesthetics. Emboldened aesthetics are required.

*From Hadley Wickham's excellent book Elegant Graphics for Data Analysis

Aesthetics and geometric Objects:
Geometric objects

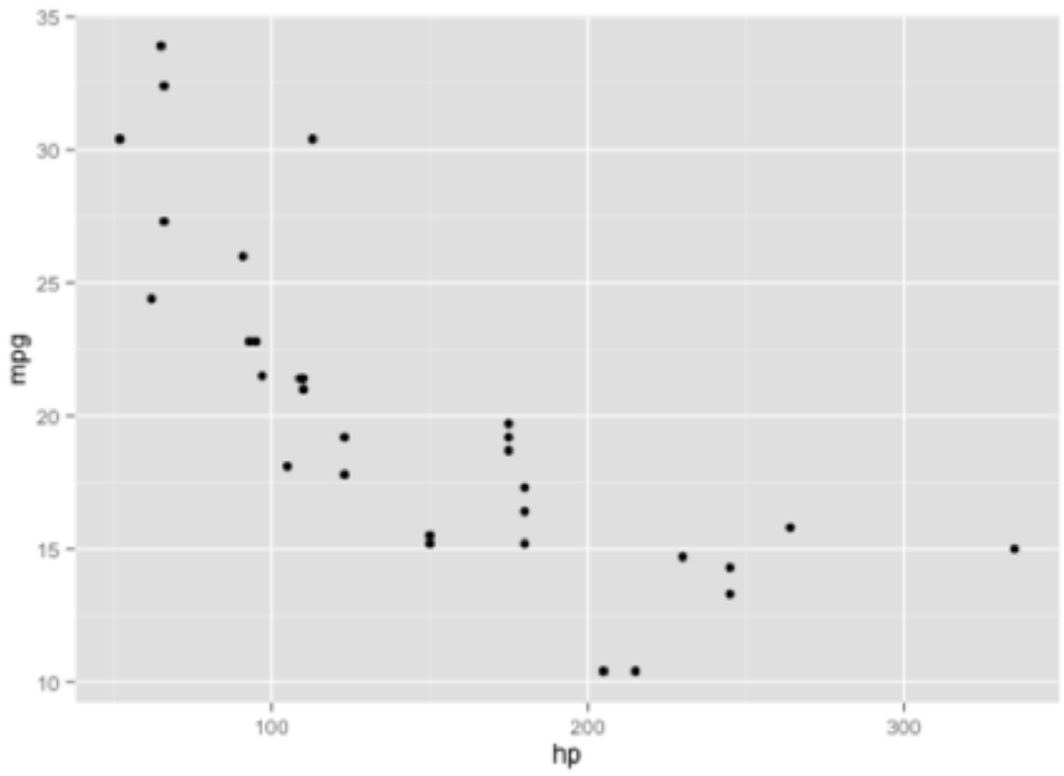
- Geoms are the ways of representing marks on a plot
- Points (geom_point: dot plots, scatter plots)
- Lines (geom_line: trend lines, timeseries)
- Boxplot (geom_boxplot)
- At least one geom is required (no upper limit)
- Geoms are added to the plot using the + operator

```
geoms <- help.search("geom_", package = "ggplot2")

geoms$matches[1:4, 1:2]
```

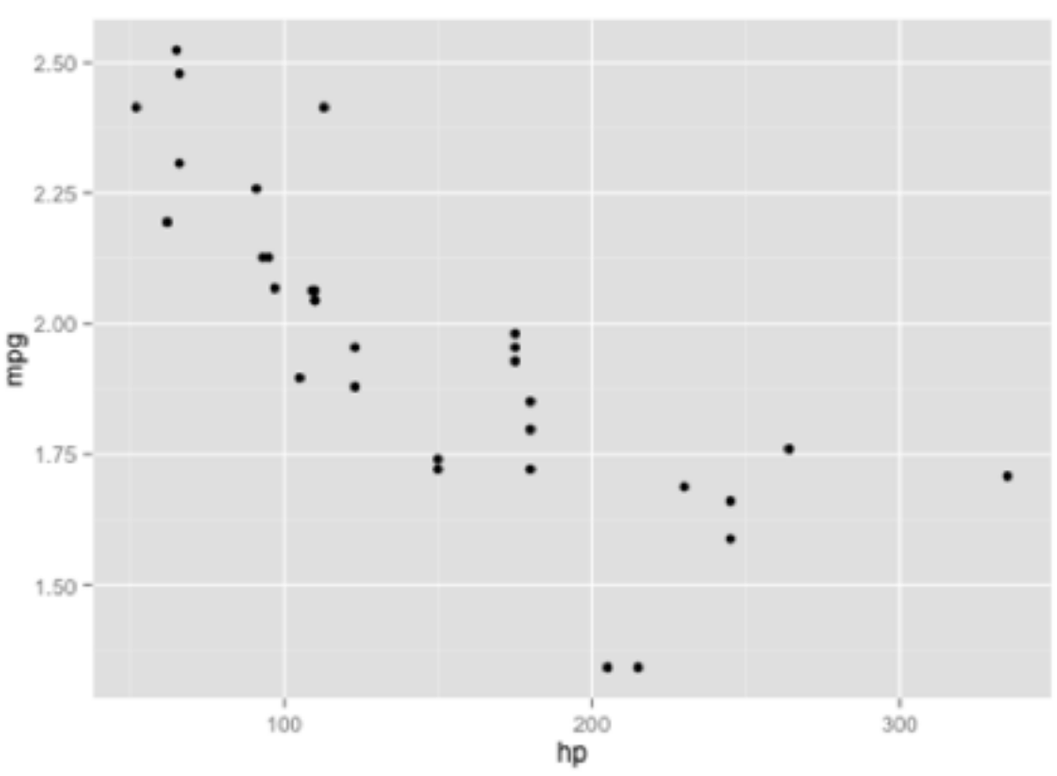
```
topic                      title
[1,] "geom_abline"   "Line specified by slope and intercept."
[2,] "geom_area"     "Area plot."
[3,] "geom_bar"      "Bars, rectangles with bases on x-axis"
[4,] "geom_bin2d"    "Add heatmap of 2d bin counts."
```

Aesthetics and geometric Objects:
geom_point

ggplot(mtcars, aes(x = hp, y = mpg)) +
geom_point()



ggplot(mtcars, aes(x = hp, y = mpg)) +
geom_point(aes(y=log(mpg)-1))



- Remember from slide 10: geom_point requires mappings for x and y, all others are optional: point identity colour, fill, shape, size, x, y
- ggplot() defaults can be overwritten by geom

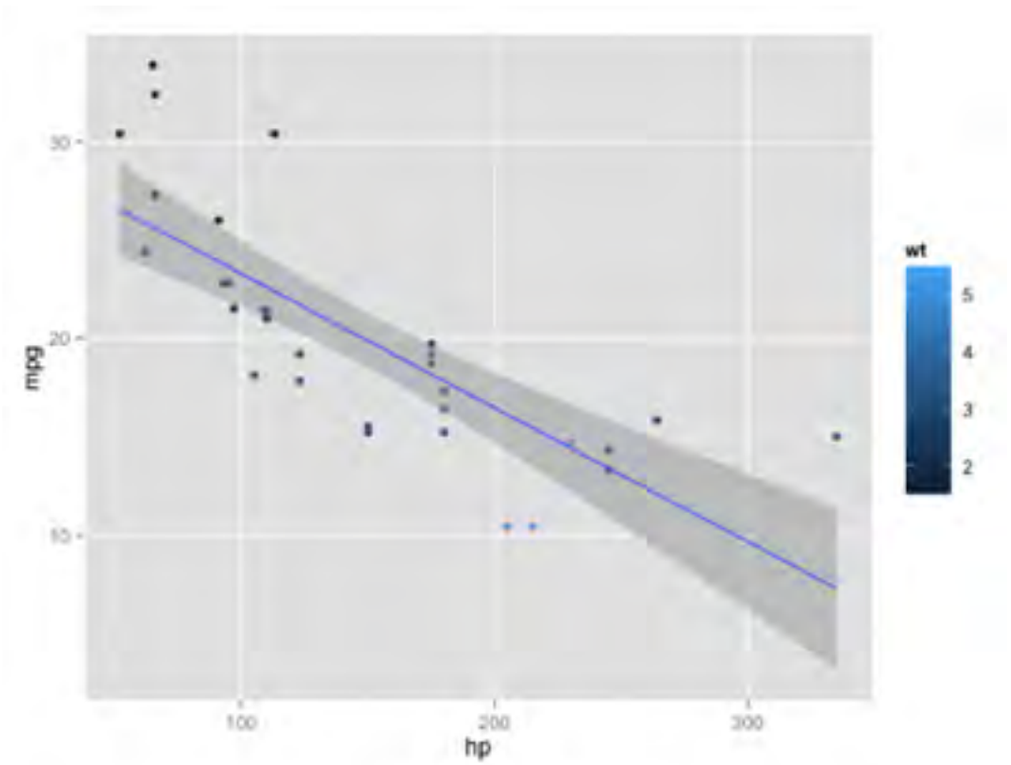
mtcars dataset

```
> head(mtcars)
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

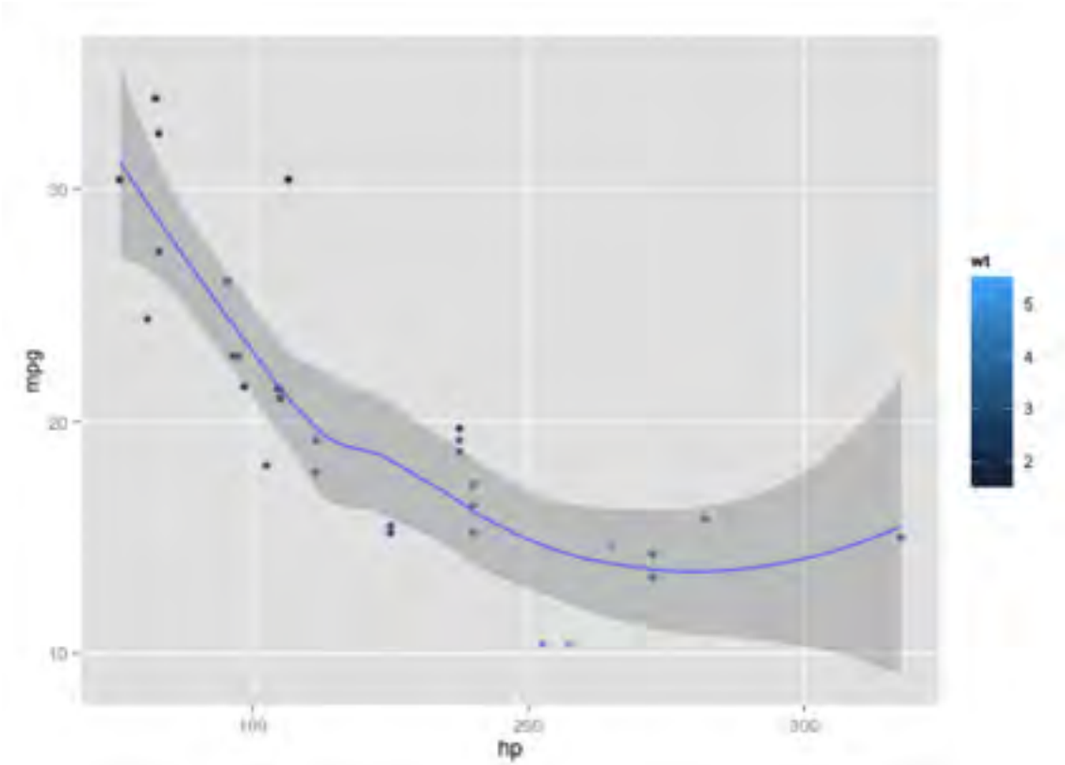
```
> str(mtcars)
'data.frame': 32 obs. of  11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
 $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
 $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
 $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

Aesthetics and geometric Objects:
geom_smooth/geom_line

```
plot <- ggplot(mtcars, aes(x = hp, y = mpg))
plot + geom_point(aes(color = wt)) +
geom_smooth(method = lm)
```




```
plot <- ggplot(mtcars, aes(x = hp, y = mpg))
plot + geom_point(aes(color = wt)) +
geom_smooth(method = loess)
```



- A plot constructed with ggplot can have more than one geom
- We can add a regression line to our previous hp vs mpg plot

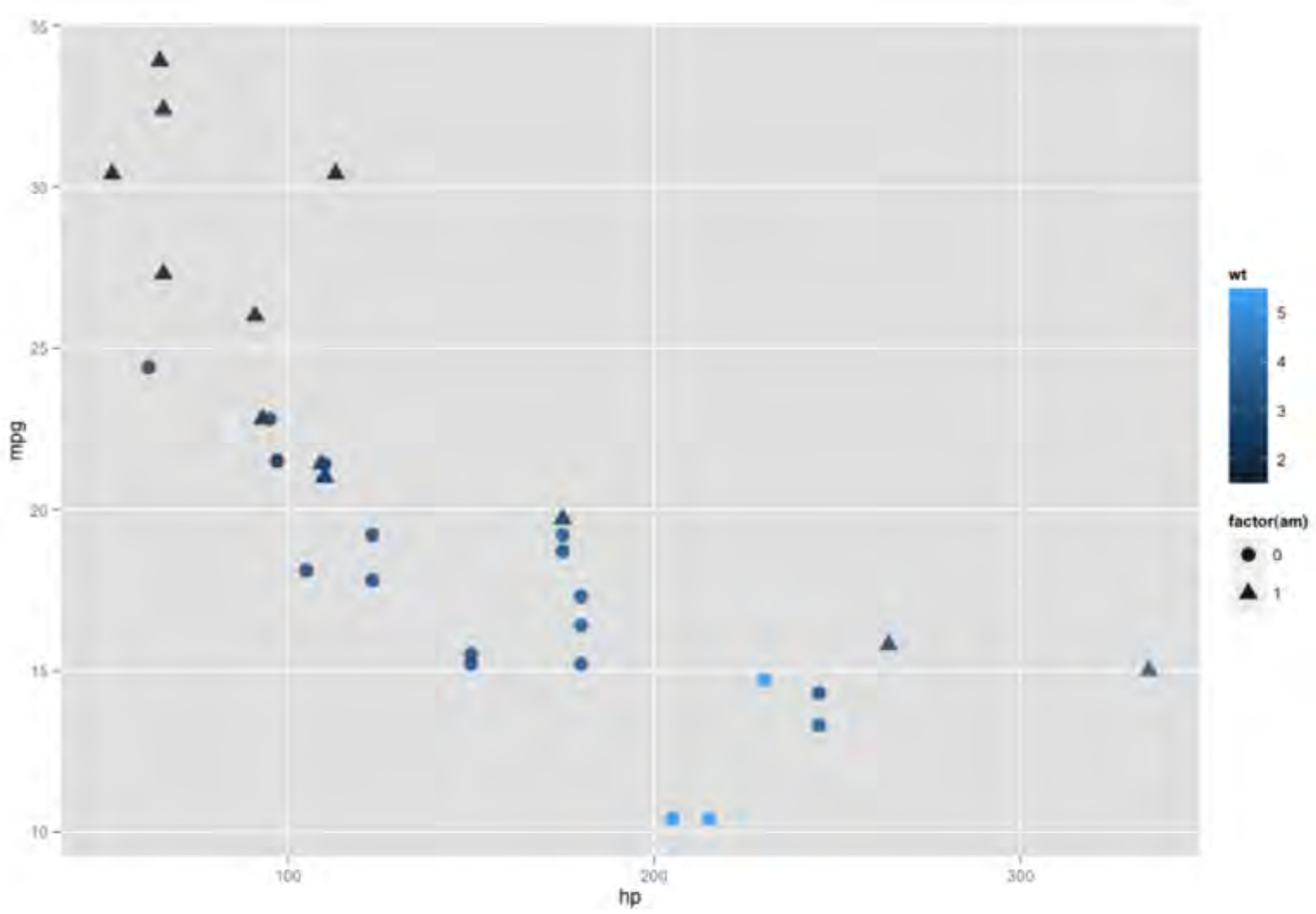
MICHAEL CLARK
CENTER FOR SOCIAL RESEARCH
UNIVERSITY OF NOTRE DAME

GENERALIZED ADDITIVE MODELS

GETTING STARTED WITH ADDITIVE MODELS IN R

Aesthetics and geometric Objects:
Mapping variables to aesthetics

```
ggplot(mtcars, aes(x = hp, y = mpg)) +
geom_point(aes(color = wt, shape = factor(am)), size = 3, alpha = .9)
```



1.Aesthetics and geometric Objects
2.Statistical Transformations
3.Scales
4.Faceting
5.Themes
6.Project Example

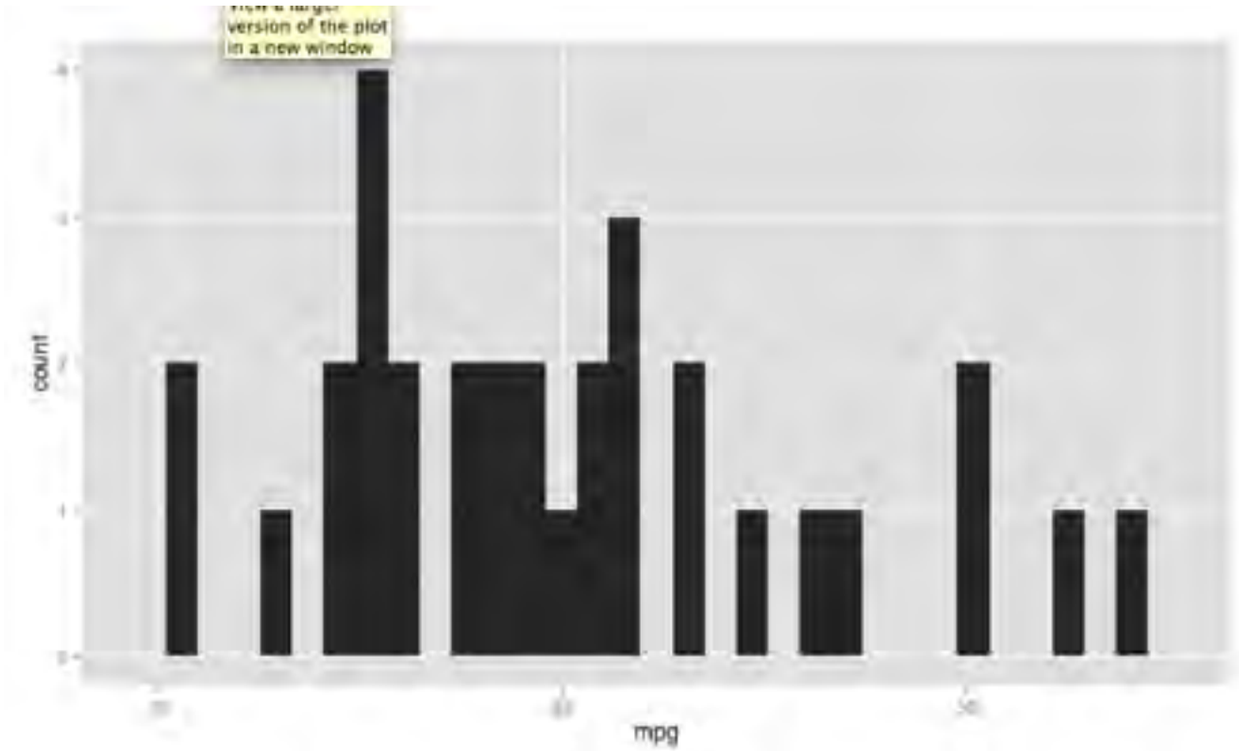
Statistical Transformations:
Intro

- Each geom has default statistics, but these can be changed

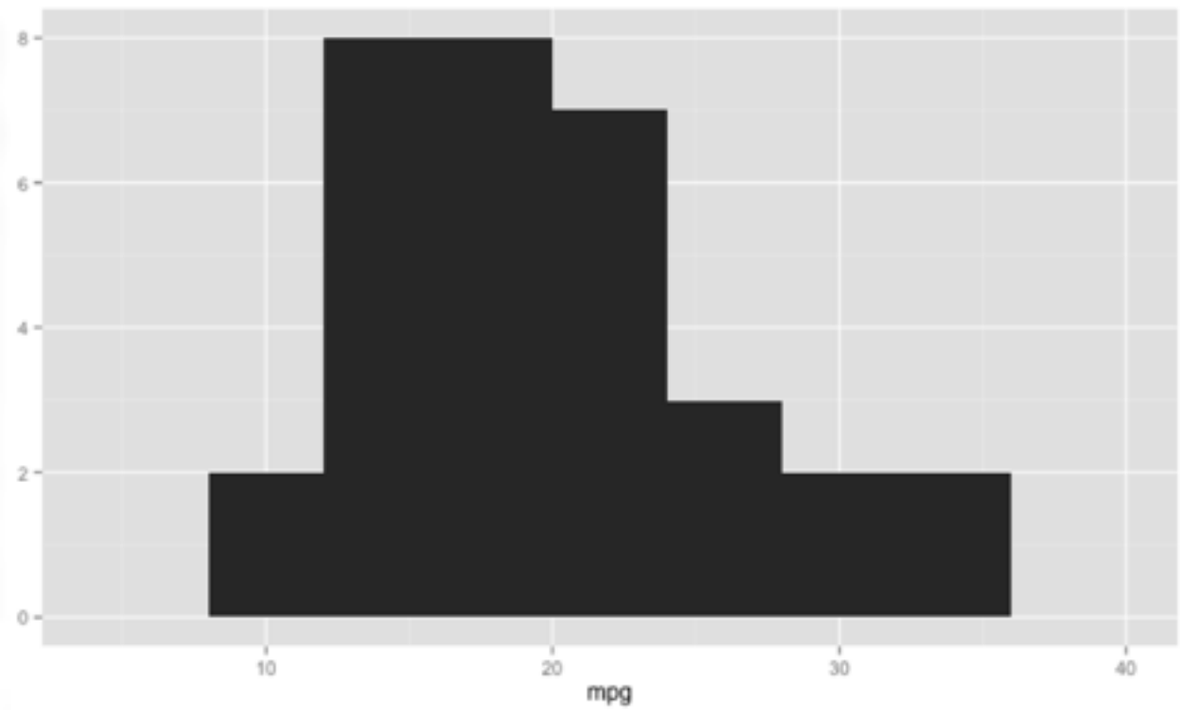
```
args(geom_bar)
function (mapping = NULL, data = NULL, stat = "bin", position = "stack",
    ...)

args(geom_boxplot)
function (mapping = NULL, data = NULL, stat = "boxplot", position = "dodge",
    outlier.color = "black", outlier.shape = 16, outlier.size = 2,
    notch = FALSE, notchwidth = 0.5, ...)

args(geom_histogram)
function (mapping = NULL, data = NULL, stat = "bin", position = "stack",
    ...)
```

Statistical Transformations:
Arguments

```
ggplot(mtcars, aes(x = mpg)) +
geom_bar()
```




```
ggplot(mtcars, aes(x = mpg)) +
geom_bar(stat = "bin", binwidth=4)
```



- Arguments to **stat**_ functions are passed through **geom**_ functions

Statistical Transformations:
Transformation

```
> head(mtcars)
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

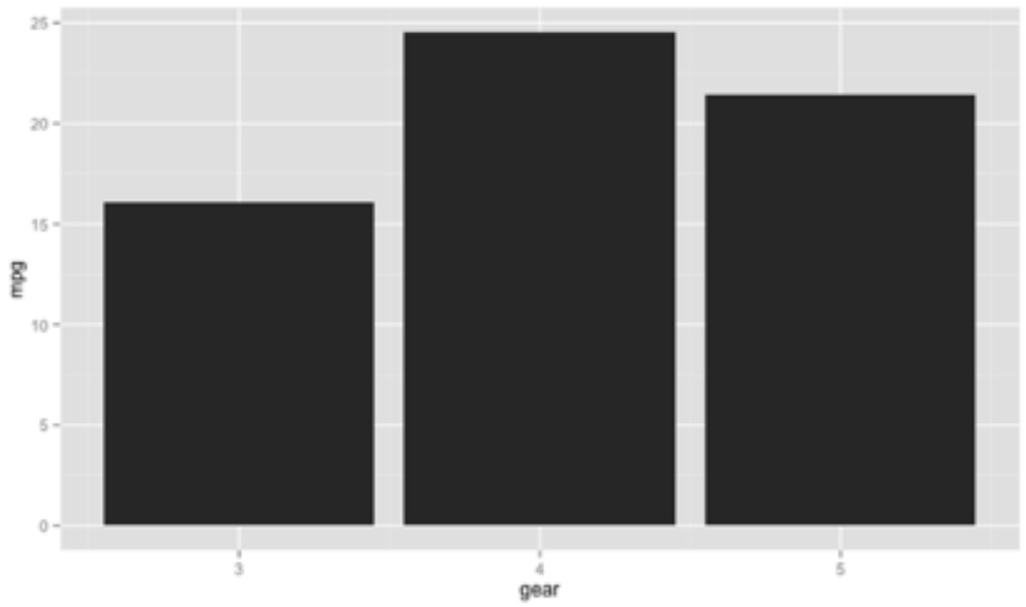
```
> (mtcarsSum <- aggregate(mtcars["mpg"], mtcars["gear"], FUN=mean))
  gear      mpg
1    3 16.10667
2    4 24.53333
3    5 21.38000
```

- With geom_bar(), the default behavior is to use stat="bin", which counts up the number of cases for each group (each x position, in this example).

```
ggplot(mtcarsSum, aes(x=gear, y=mpg)) +
  geom_bar()
```

```
Mapping a variable to y and also using stat="bin".
  With stat="bin", it will attempt to set the y value to the count of cases in each group.
  This can result in unexpected behavior and will not be allowed in a future version of ggplot2.
  If you want y to represent counts of cases, use stat="bin" and don't map a variable to y.
  If you want y to represent values in the data, use stat="identity".
  See ?geom_bar for examples. (Deprecated; last used in version 0.9.2)
stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
Error in pmin(y, 0) : object 'y' not found
```

```
ggplot(mtcarsSum, aes(x=gear, y=mpg)) +
  geom_bar(stat="identity")
```



1.Aesthetics and geometric Objects
2.Statistical Transformations
3.Scales
4.Faceting
5.Themes
6.Project Example

Scales:
Intro

- The domain of each scale corresponds to the range of the variable supplied to the scale, and can be continuous or discrete, ordered or unordered.

- The range consists of the concrete aesthetics that you can perceive and that R can understand: position, color, shape, size and line type

- **name**: the first argument gives the axis or legend title

- **limits**: the minimum and maximum of the scale

- **breaks**: the points along the scale where labels should appear

- **labels**: the labels that appear at each break

Colorbrewer: Color Advice

colorbrewer2.org

number of data classes on your map

3 learn more >

the nature of your data

sequential learn more >

sequential
diverging
qualitative

pick a color scheme

multihue single hue

(optional) only show schemes that are:

colorblind safe print friendly

photocopy-able learn more >

pick a color system

229, 245, 249
153, 216, 201
44, 162, 95

RGB CMYK HEX

adjust map context

roads
cities
borders

select a background

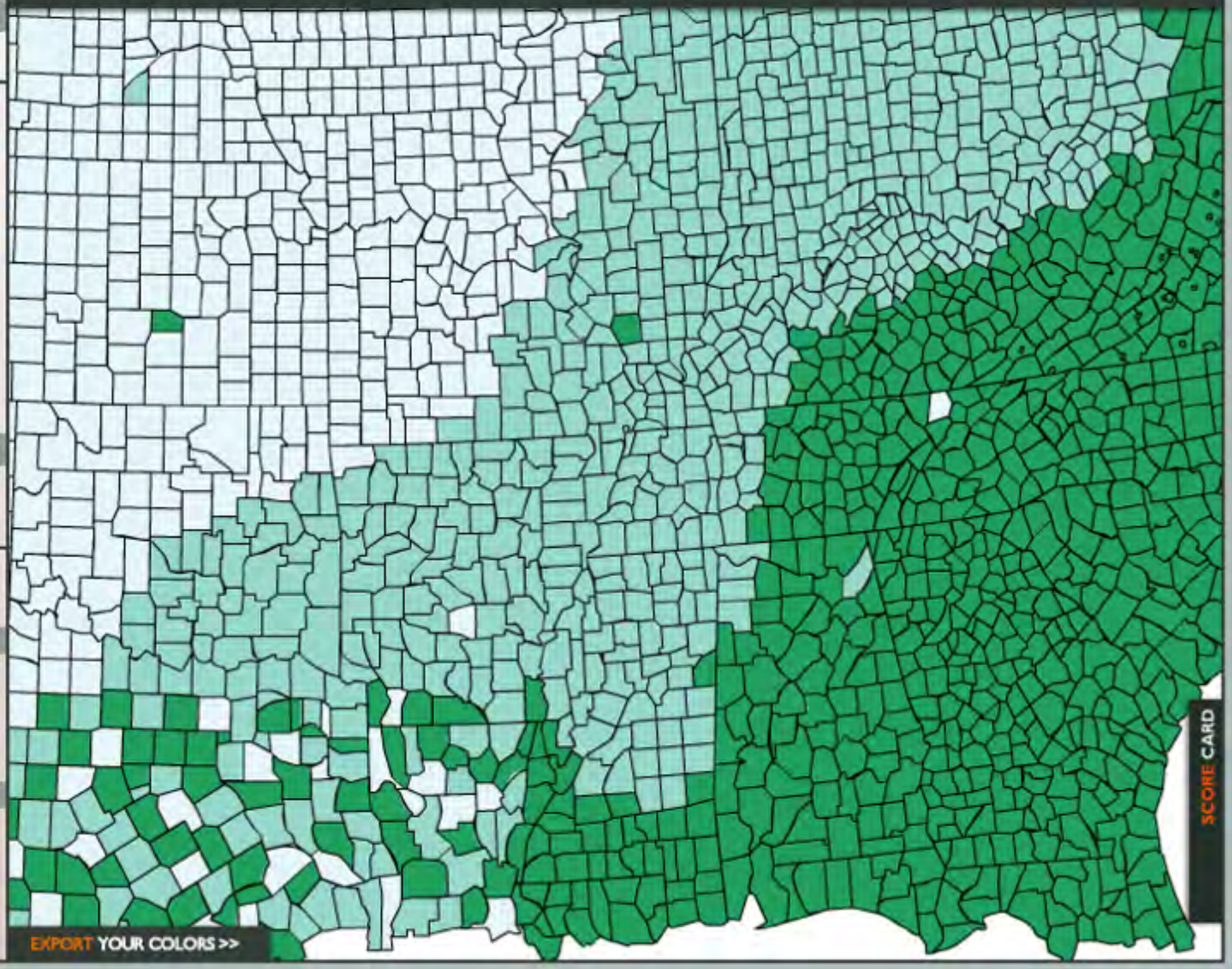
solid color
terrain

color transparency

learn more >

how to use | updates | credits

COLORBREWER 2.0
color advice for cartography



Support
Back to ColorBrewer 1.0

http://colorbrewer2.org/

Colorbrewer: Color Advic

colorbrewer2.org

number of data classes on your map

3 learn more >

how to use | updates | credits

COLORBREWER 2.0

color advice for cartography

the nature of your data

diverging learn more >

pick a color scheme: BrBG

(optional) only show schemes that are:

colorblind safe print friendly

photocopy-able learn more >

pick a color system

216, 179, 101

245, 245, 245

90, 180, 172

RGB CMYK HEX

adjust map context

roads

cities

borders

select a background

solid color

terrain

color transparency

learn more >

SCORE CARD

EXPORT YOUR COLORS >>

Support

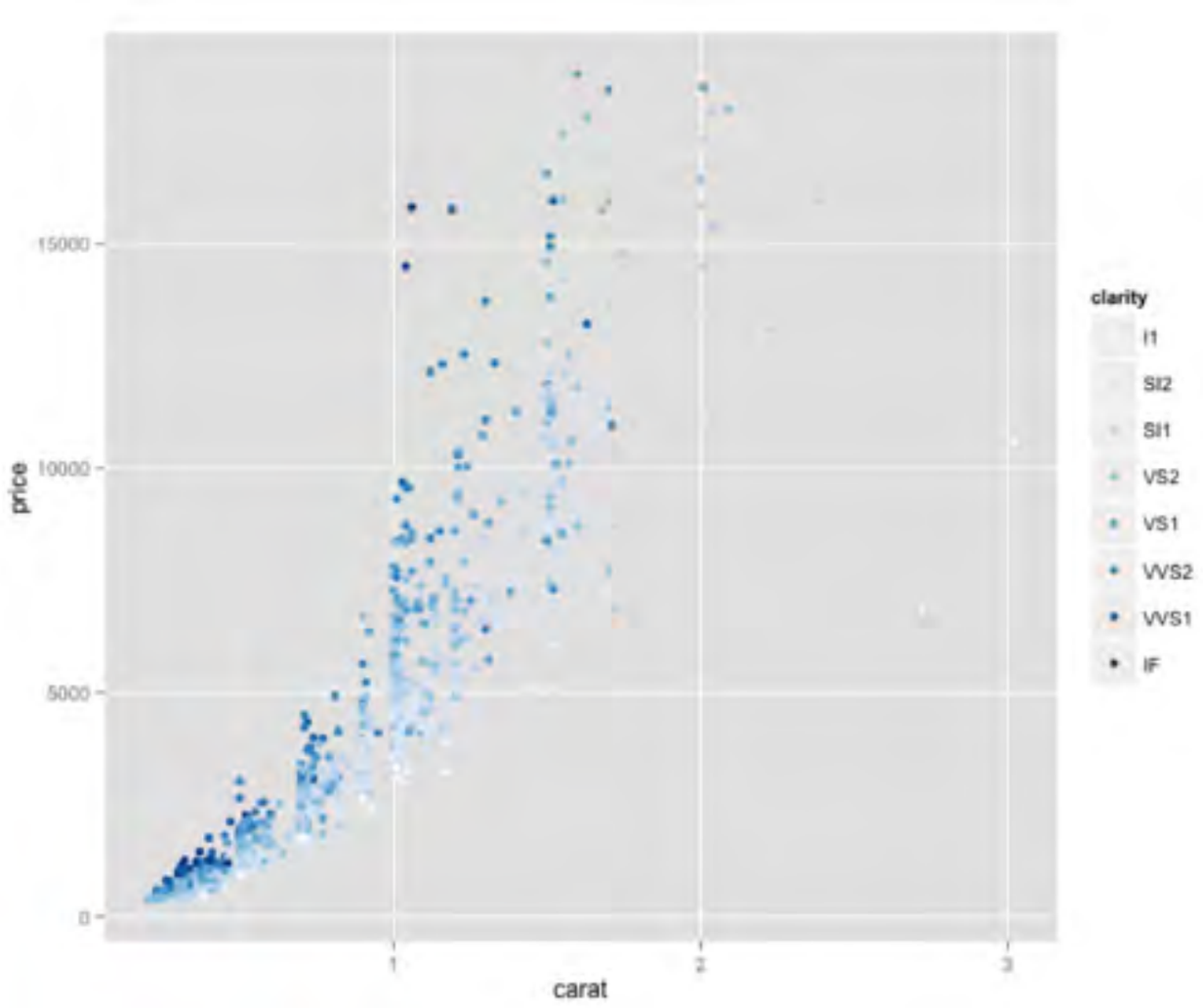
Back to ColorBrewer 1.0

diamonds dataset

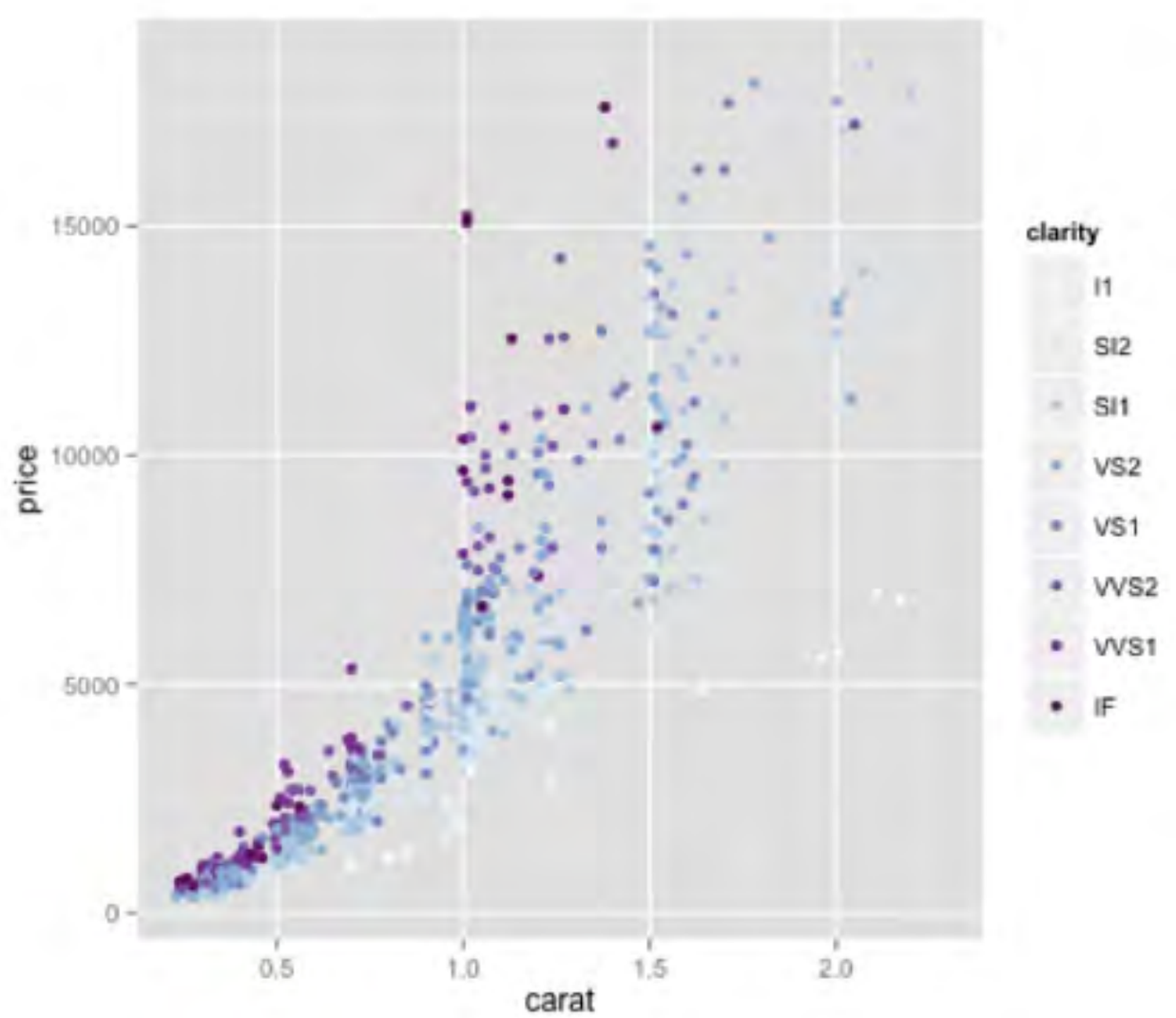
```
> head(diamonds)
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
> str(diamonds)
'data.frame': 53940 obs. of  10 variables:
 $ carat  : num  0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...
 $ cut    : Ord.factor w/ 5 levels "Fair"<"Good"<..: 5 4 2 4 2 3 3 3 1 3 ...
 $ color  : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<..: 2 2 2 6 7 7 6 5 2 5 ...
 $ clarity: Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<..: 2 3 5 4 2 6 7 3 4 5 ...
 $ depth  : num  61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...
 $ table  : num  55 61 65 58 58 57 57 55 61 61 ...
 $ price  : int  326 326 327 334 335 336 336 337 337 338 ...
 $ x      : num  3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
 $ y      : num  3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
 $ z      : num  2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...
```

Scales:
Color

```
dSample <- diamonds[sample(nrow(diamonds), 1000), ]
diamond <- ggplot(dSample,aes(x = carat, y = price, color
= clarity)) + geom_point()
diamond
```

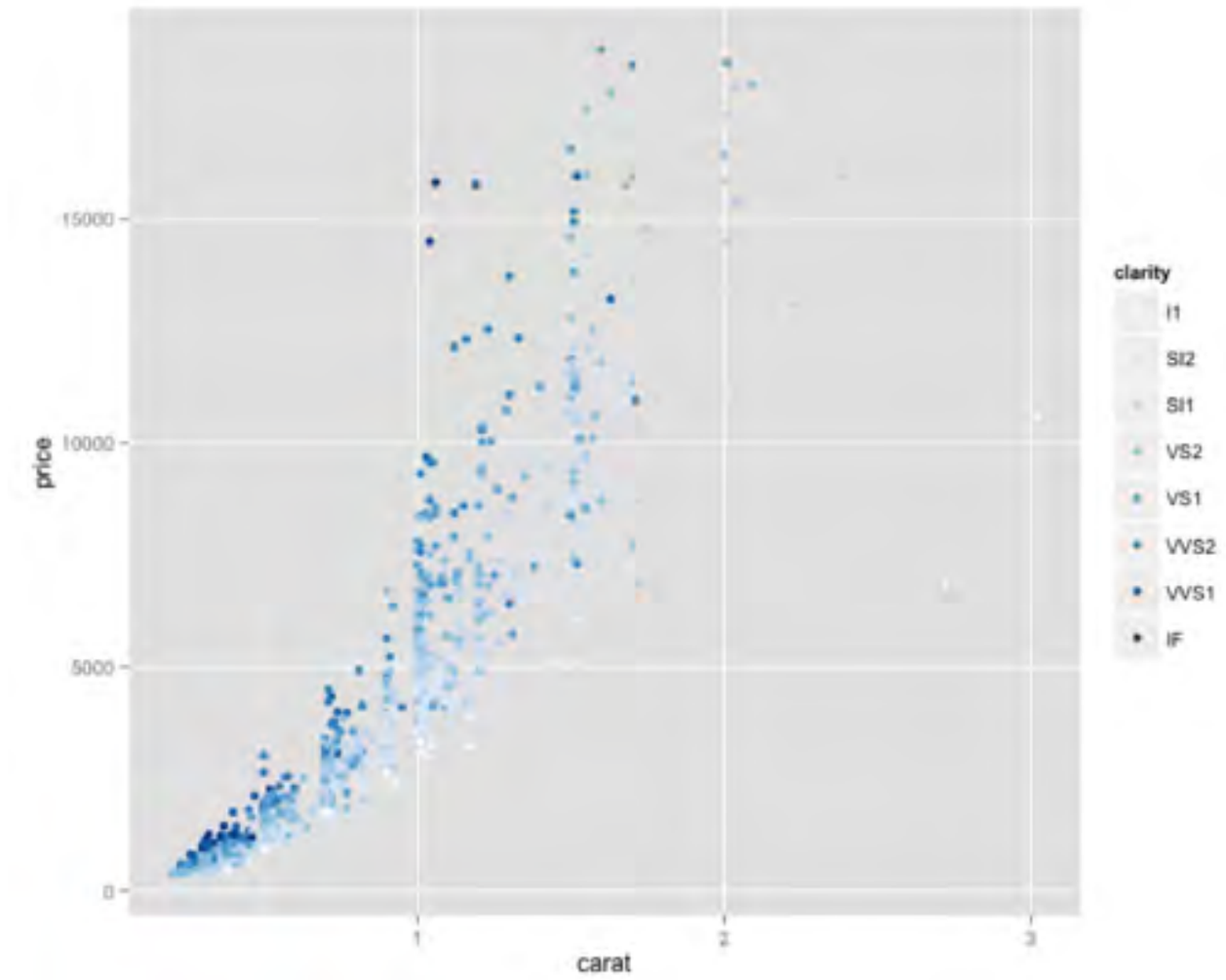



```
diamond + scale_color_brewer(type="seq", palette=3)
```

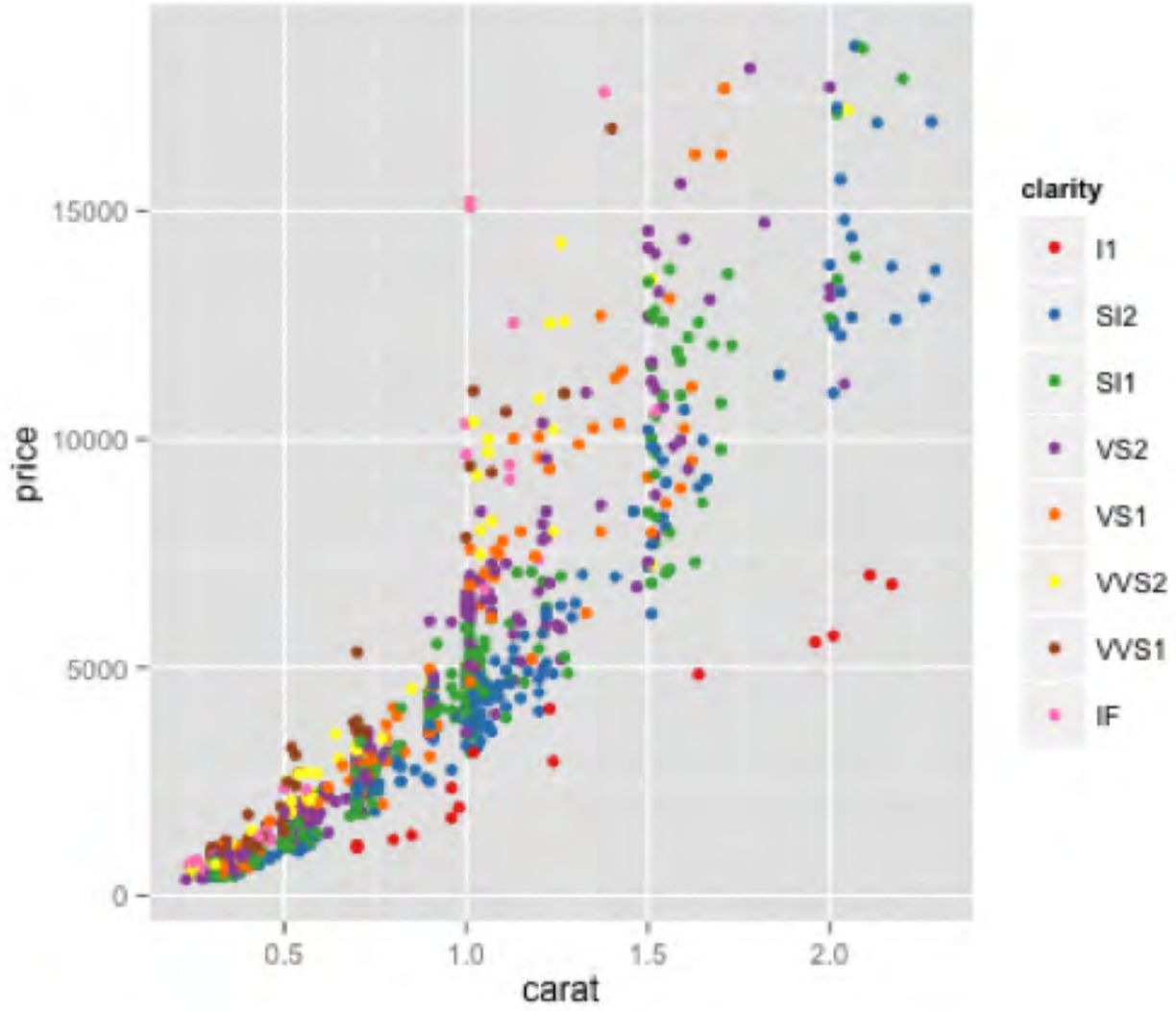


Scales:
Color

```
dSample <- diamonds[sample(nrow(diamonds), 1000), ]
diamond <- ggplot(dSample,aes(x = carat, y = price, color
= clarity)) + geom_point()
diamond
```

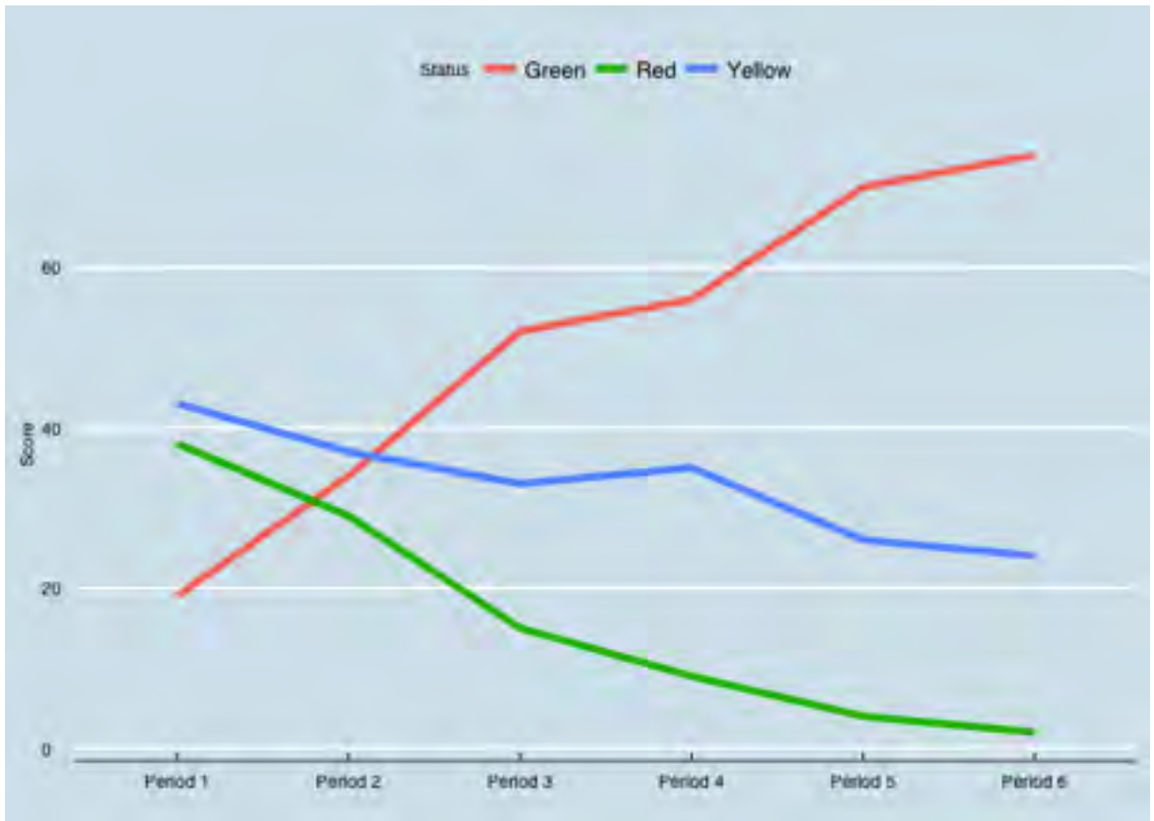


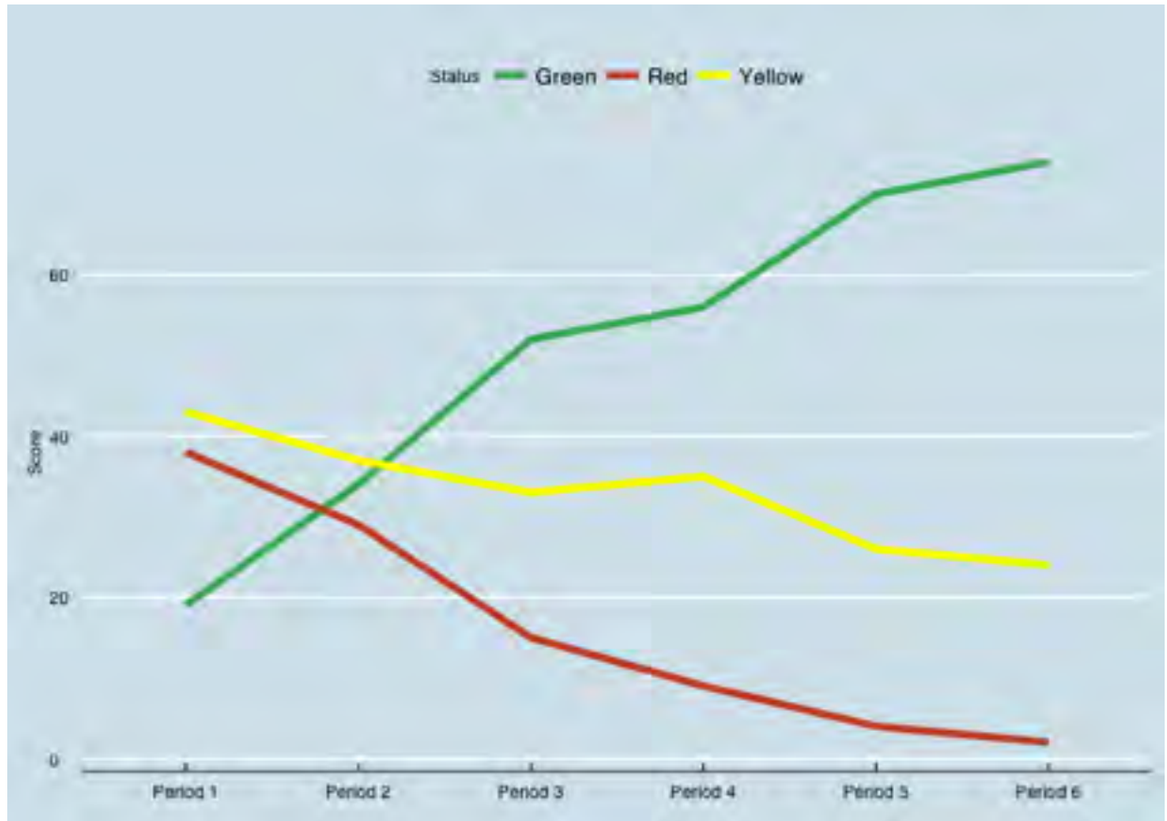

```
diamond + scale_color_brewer(palette="Set1")
```



Scales:
Color (Stroop test)

```
library(ggthemes)
customPalette <- c("mediumseagreen", "tomato3", "yellow1", "#009E73", "#F0E442", "#0072B2")

ggplot(data = df, aes(x = Period, y = Score, group = Status)) + geom_line(aes(color = Status),size = 2) +
theme_economist() + theme(axis.title.x=element_blank()) +  scale_colour_manual(values= customPalette)
```





i want hue

Colors for data scientists. Generate and refine palettes of optimally distinct colors.

Color space

Presets...

H 0 — 360

C 0 — 3

L 0 — 1.5

☐ Dark background

Palette

7 colors | soft (k-Means)

Reroll palette

Colors

#7EC1AA 126,193,170

#C14FAB 193,79,171

JSON

HEX json

```
["#7EC1AA",
"#C14FAB",
"#C25D3C",
"#99BF4E",
"#4F4038",
"#C89B9C",
"#737486"]
```

RGB json

```
[[126,193,170],
[193,79,171],
[194,93,60],
[153,191,78],
[79,64,56],
[200,155,156]
```

CSS

HEX list for CSS

```
#7EC1AA
#C14FAB
#C25D3C
#99BF4E
#4F4038
#C89B9C
#737486
```

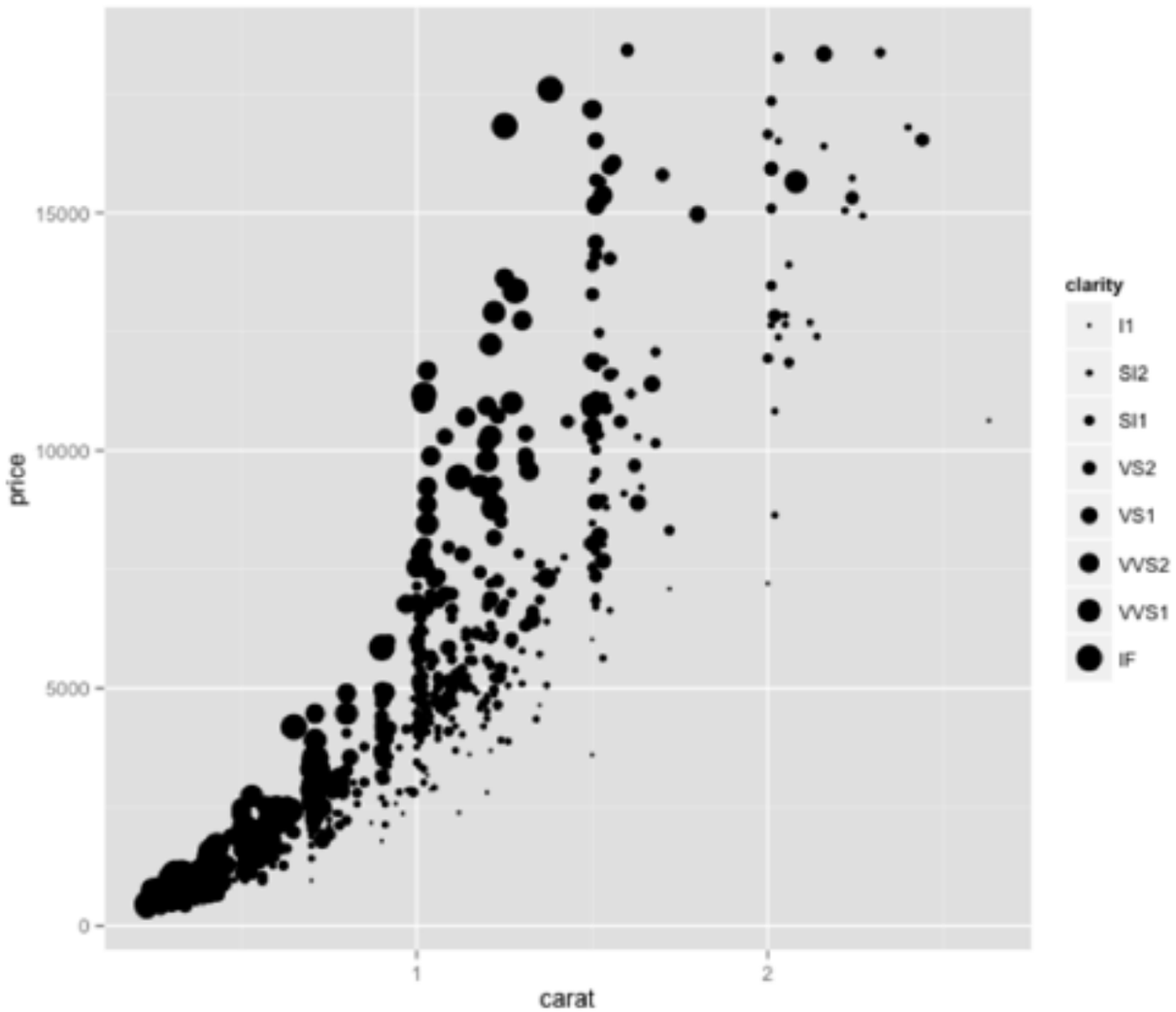
RGB list for CSS

```
rgb(126,193,170)
rgb(193,79,171)
rgb(194,93,60)
rgb(153,191,78)
rgb(79,64,56)
rgb(200,155,156)
```

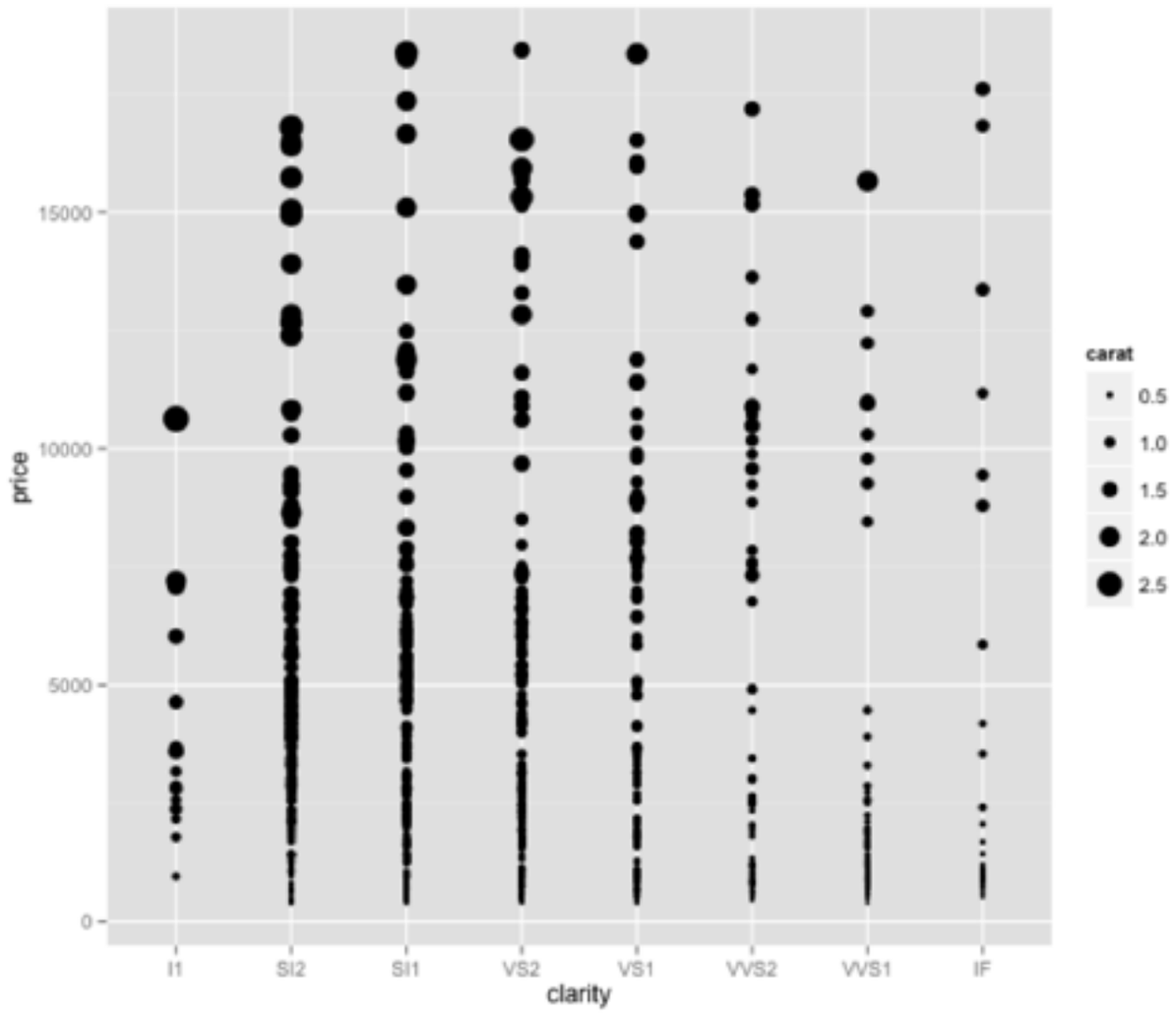
http://tools.medialab.sciences-po.fr/iwanthue/

Scales:
Shapes

```
cars <- ggplot(dSample, aes(x = carat, y = price))
cars + geom_point(aes(size = clarity))
```




```
cars <- ggplot(dSample, aes(x = clarity, y = price))
cars + geom_point(aes(size = carat))
```



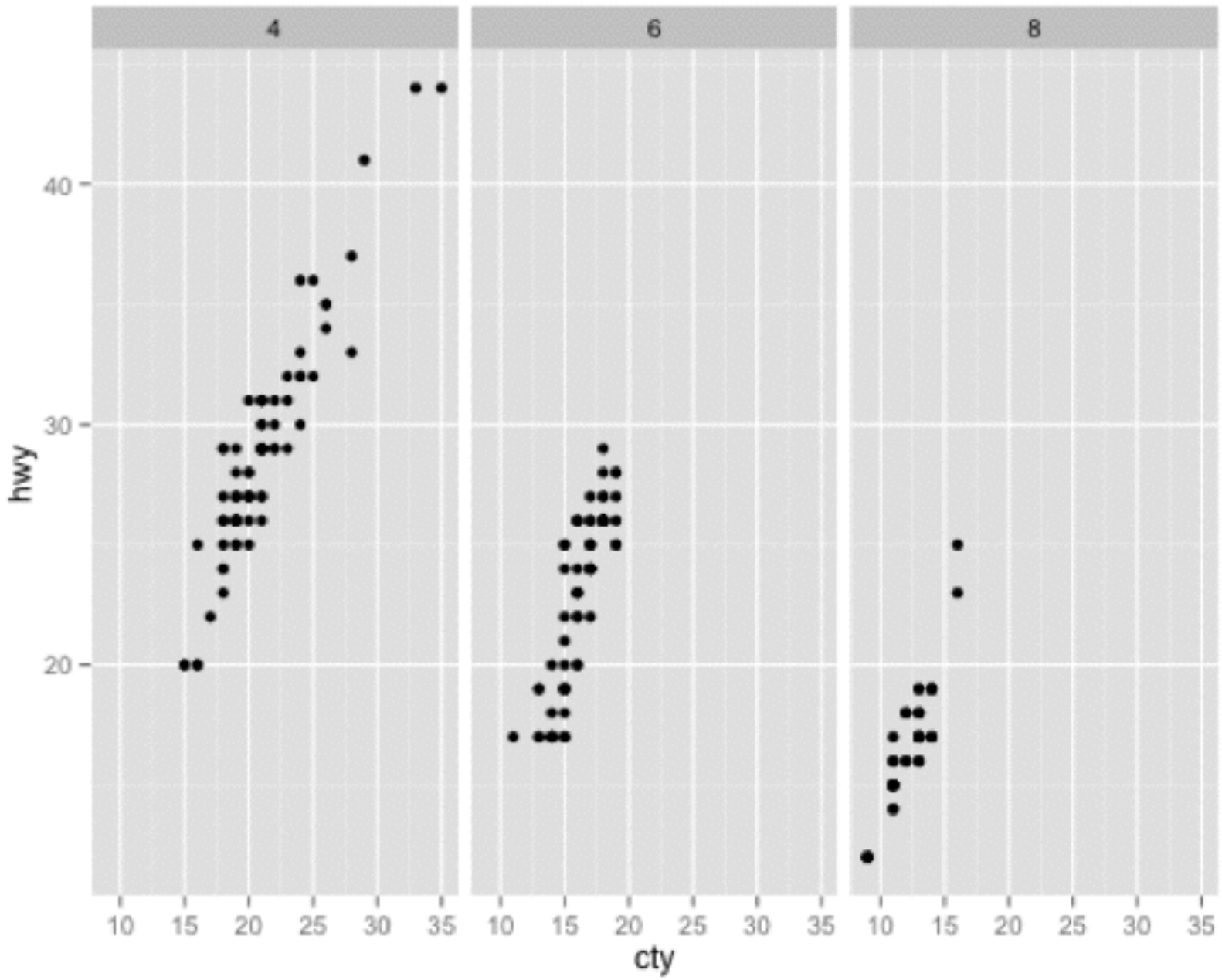
1. Aesthetics and geometric Objects
2. Statistical Transformations
3. Scales
4. Faceting
5. Themes
6. Project Example

mpg dataset

```
> head(mpg)
  manufacturer model displ year cyl      trans drv cty hwy fl   class
1         audi    a4   1.8 1999   4   auto(l5)   f  18  29  p compact
2         audi    a4   1.8 1999   4 manual(m5)   f  21  29  p compact
3         audi    a4   2.0 2008   4 manual(m6)   f  20  31  p compact
4         audi    a4   2.0 2008   4   auto(av)   f  21  30  p compact
5         audi    a4   2.8 1999   6   auto(l5)   f  16  26  p compact
6         audi    a4   2.8 1999   6 manual(m5)   f  18  26  p compact
> str(mpg)
'data.frame': 234 obs. of  11 variables:
 $ manufacturer: Factor w/ 15 levels "audi","chevrolet",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ model       : Factor w/ 38 levels "4runner 4wd",..: 2 2 2 2 2 2 2 3 3 3 ...
 $ displ       : num  1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
 $ year        : int  1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
 $ cyl         : int  4 4 4 4 6 6 6 4 4 4 ...
 $ trans       : Factor w/ 10 levels "auto(av)","auto(l3)",..: 4 9 10 1 4 9 1 9 4 10 ...
 $ drv         : Factor w/ 3 levels "4","f","r": 2 2 2 2 2 2 2 1 1 1 ...
 $ cty         : int  18 21 20 21 16 18 18 18 16 20 ...
 $ hwy         : int  29 29 31 30 26 26 27 26 25 28 ...
 $ fl          : Factor w/ 5 levels "c","d","e","p",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ class       : Factor w/ 7 levels "2seater","compact",..: 2 2 2 2 2 2 2 2 2 2 ...
```

Faceting:
facet_grid

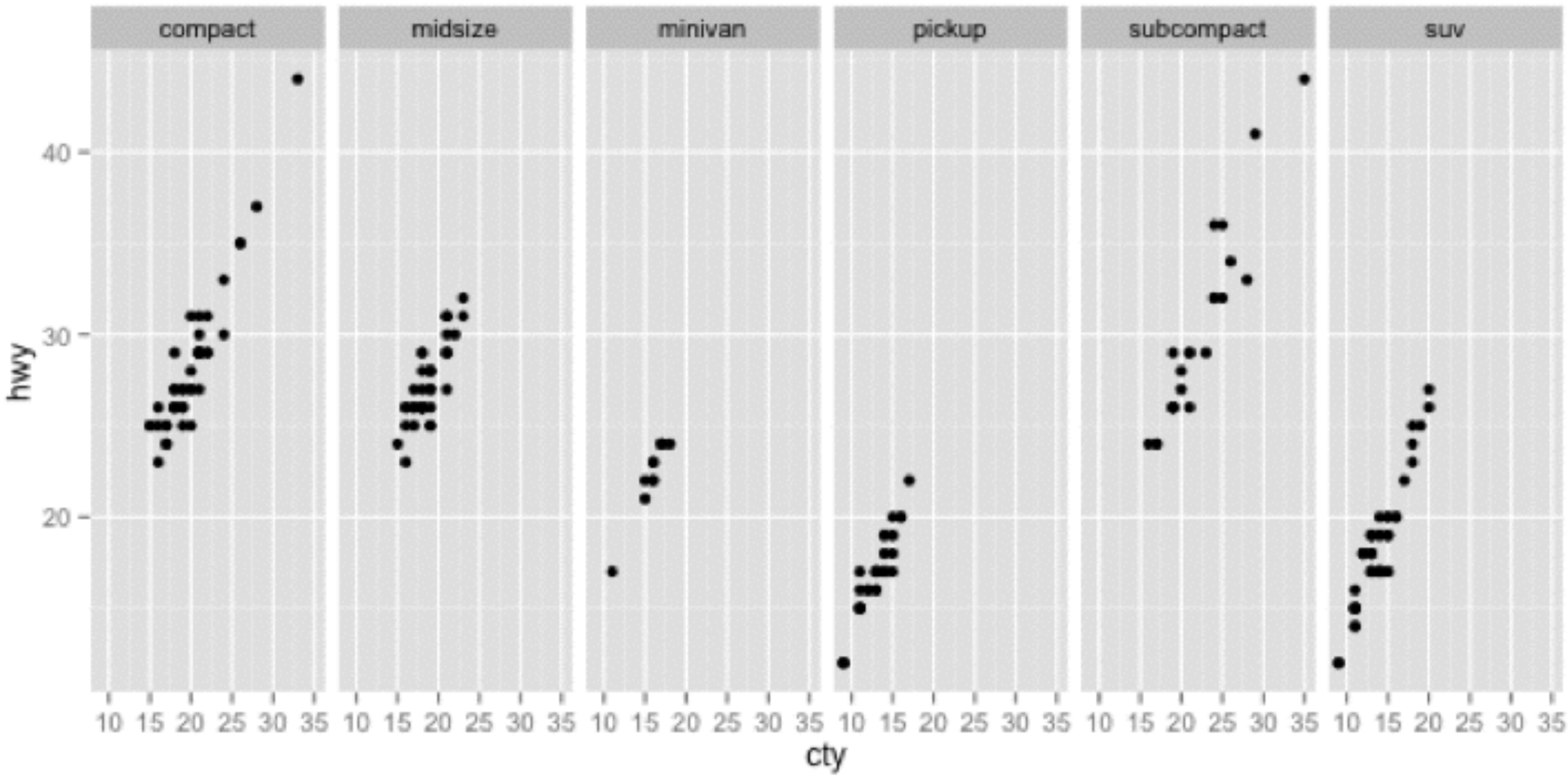
```
mpg2 <- subset(mpg, cyl != 5 & drv %in% c("4", "f"))

qplot(data = mpg2, cty, hwy) + facet_grid(. ~ cyl)
```



Faceting:
facet_grid

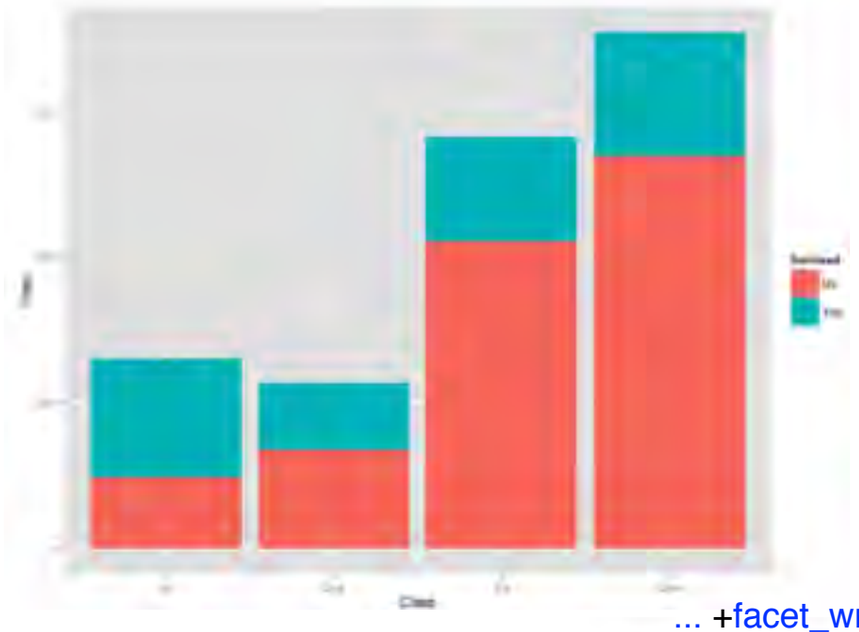
---

qplot(cty, hwy, data = mpg2) + facet_grid(. ~ class)

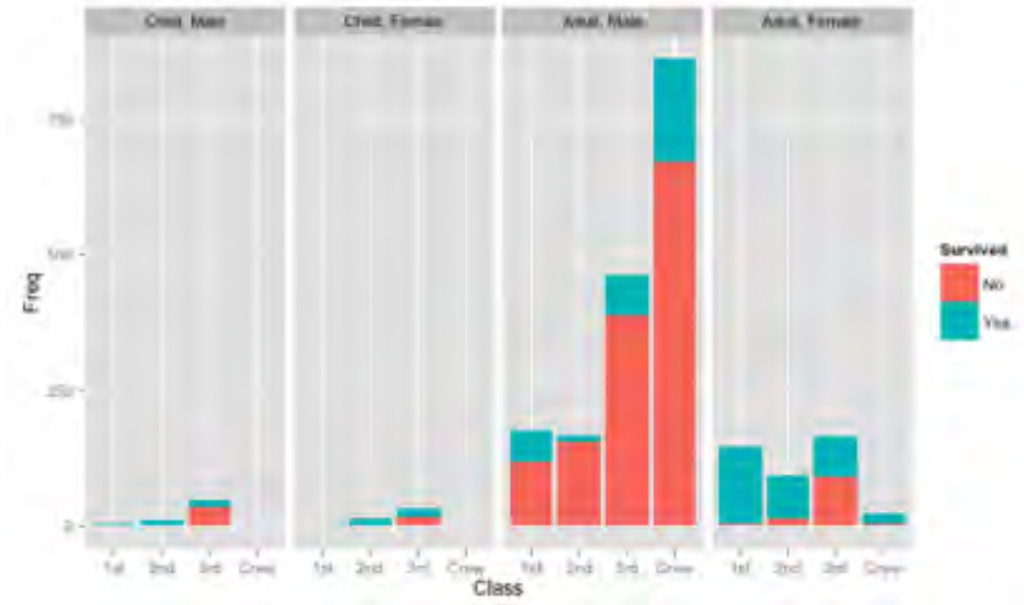


Faceting:
facet_wrap

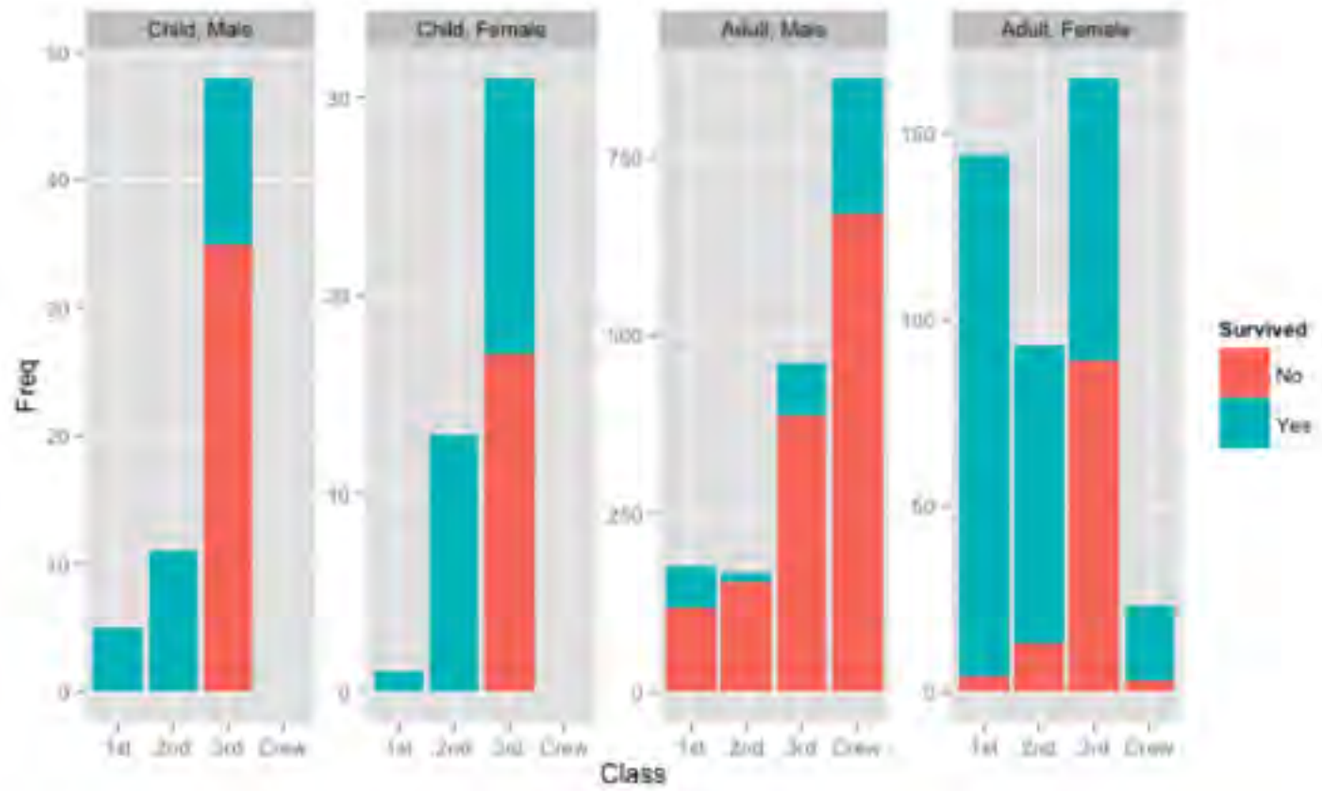
ggplot(as.data.frame(Titanic), aes(x = Class, y = Freq, fill = Survived)) +
geom_bar(stat = "identity")



ggplot(as.data.frame(Titanic), aes(x = Class, y = Freq, fill = Survived)) +
geom_bar(stat = "identity") + **facet_wrap(~Age + Sex, nrow = 1)**



... +facet_wrap(~Age + Sex, nrow = 1, **scales = "free"**)



1. Aesthetics and geometric Objects
2. Statistical Transformations
3. Scales
4. Faceting
5. Themes
6. Project Example

Themes:
Theme system

- The theme system handles non-data plot elements:
- Axis labels
- Facet label background
- Legend appearance
- Plot background

https://github.com/hadley/ggplot2/wiki/Themes

https://github.com/jrnold/ggthemes

Themes:
A lot of granularity

line all line elements (element_line)
rect all rectangluar elements (element_rect)
text all text elements (element_text)
title all title elements: plot, axes, legends (element_text; inherits from text)
axis.title

label of axes (element_text; inherits from text)
axis.title.x x axis label (element_text; inherits from axis.title)
axis.title.y y axis label (element_text; inherits from axis.title)
axis.text tick labels along axes (element_text; inherits from text)
axis.text.x x axis tick labels (element_text; inherits from axis.text)
axis.text.y y axis tick labels (element_text; inherits from axis.text)
axis.ticks tick marks along axes (element_line; inherits from line)
axis.ticks.x x axis tick marks (element_line; inherits from axis.ticks)
axis.ticks.y y axis tick marks (element_line; inherits from axis.ticks)
axis.ticks.length length of tick marks (unit)
axis.ticks.margin space between tick mark and tick label (unit)
axis.line lines along axes (element_line; inherits from line)
axis.line.x line along x axis (element_line; inherits from axis.line)
axis.line.y line along y axis (element_line; inherits from axis.line)

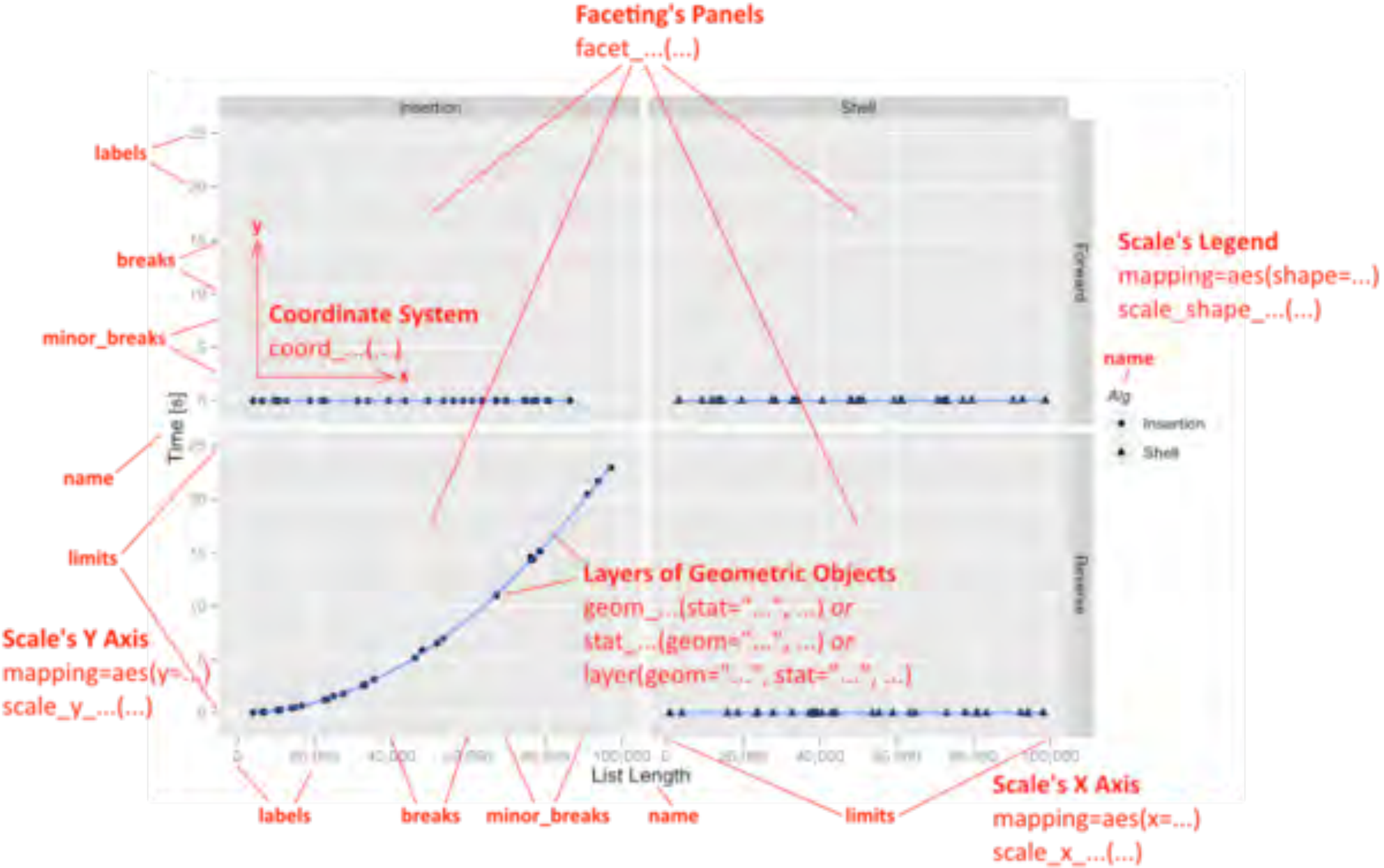
legend.background

background of legend (element_rect; inherits from rect)
legend.margin extra space added around legend (unit)
legend.key background underneath legend keys (element_rect; inherits from rect)
legend.key.size size of legend keys (unit; inherits from legend.key.size)
legend.key.height key background height (unit; inherits from legend.key.size)
legend.key.width key background width (unit; inherits from legend.key.size)
legend.text legend item labels (element_text; inherits from text)
legend.text.align alignment of legend labels (number from 0 (left) to 1 (right))
legend.title title of legend (element_text; inherits from title)
legend.title.align alignment of legend title (number from 0 (left) to 1 (right))

legend.position the position of legends. ("left", "right", "bottom", "top", or two-element numeric vector)
legend.direction layout of items in legends ("horizontal" or "vertical")
legend.justification anchor point for positioning legend inside plot ("center" or two-element numeric vector)
legend.box arrangement of multiple legends ("horizontal" or "vertical")
panel.background
background of plotting area (element_rect; inherits from rect)
panel.border border around plotting area (element_rect; inherits from rect)
panel.margin margin around facet panels (unit)
panel.grid grid lines (element_line; inherits from line)
panel.grid.major major grid lines (element_line; inherits from panel.grid)
panel.grid.minor minor grid lines (element_line; inherits from panel.grid)
panel.grid.major.x vertical major grid lines (element_line; inherits from panel.grid.major)
panel.grid.major.y horizontal major grid lines (element_line; inherits from panel.grid.major)
panel.grid.minor.x vertical minor grid lines (element_line; inherits from panel.grid.minor)
panel.grid.minor.y horizontal minor grid lines (element_line; inherits from panel.grid.minor)
plot.background

background of the entire plot (element_rect; inherits from rect)
plot.title plot title (text appearance) (element_text; inherits from title)
plot.margin margin around entire plot (unit)
strip.background

background of facet labels (element_rect; inherits from rect)
strip.text facet labels (element_text; inherits from text)
strip.text.x facet labels along horizontal direction (element_text; inherits from strip.text)
strip.text.y facet labels along vertical direction (element_text; inherits from strip.text)



http://sape.inf.usi.ch/quick-reference/

ggplot2 resources

Hadley's ggplot2 book
http://amzn.com/0387981403

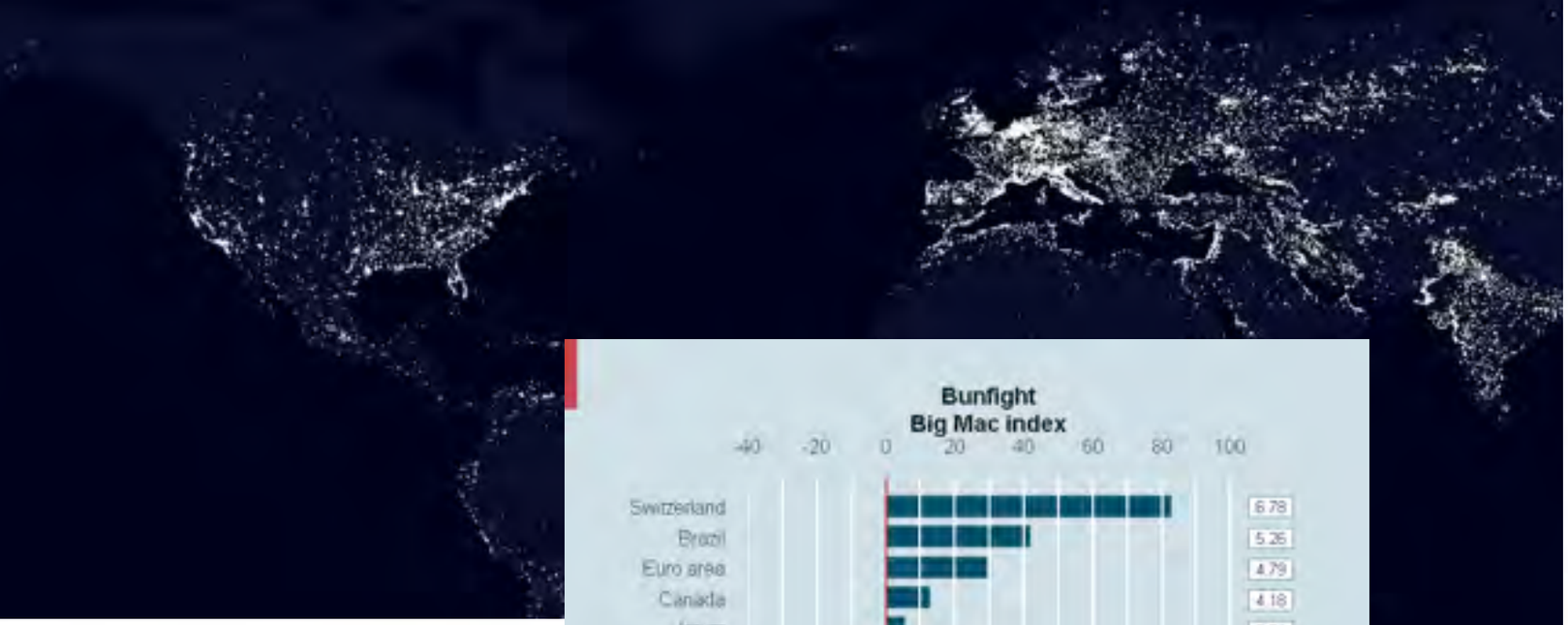
ggplot2 google group
http://groups.google.com/group/ggplot2

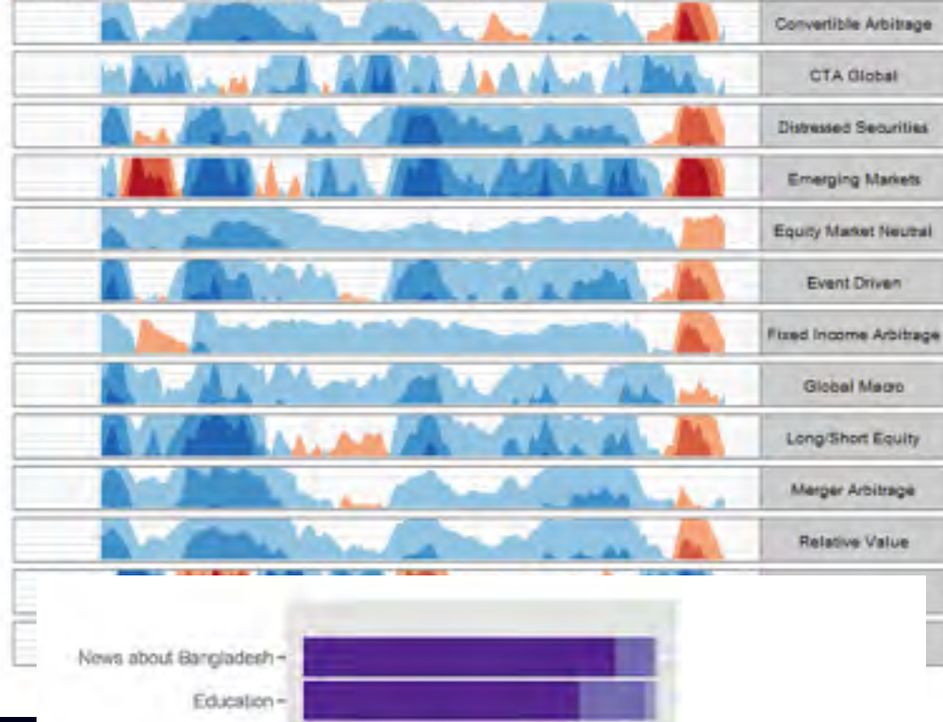
stackoverflow
http://stackoverflow.com/tags/ggplot2

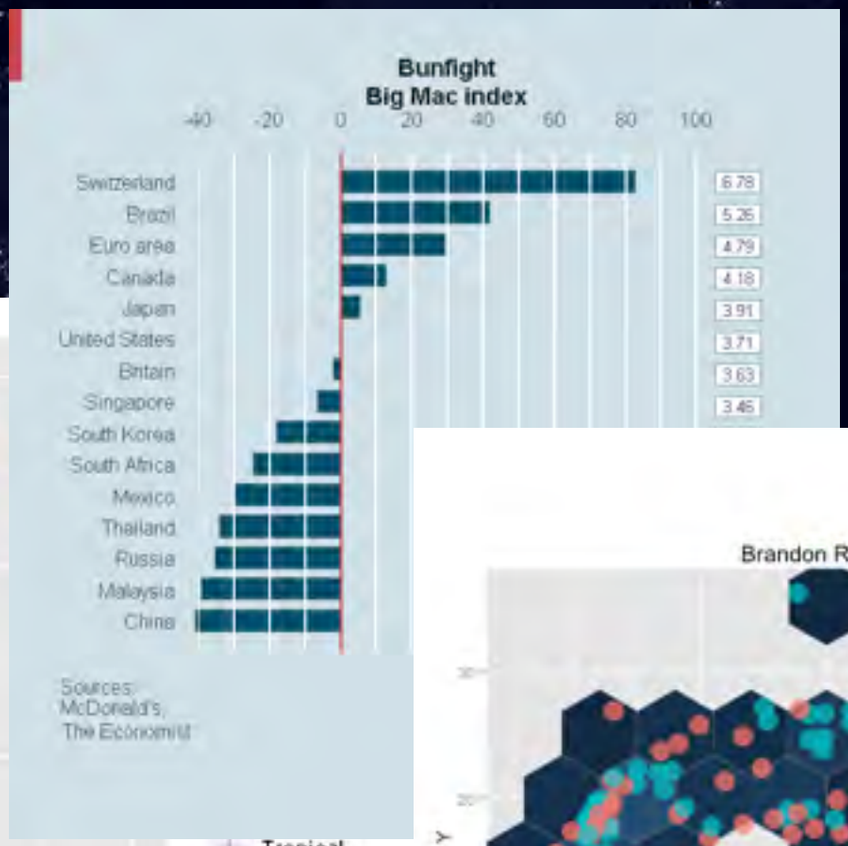
Lattice to ggplot2 conversion
http://learnr.wordpress.com/?s=lattice

Winston Chang's Cookbook for common graphics
http://wiki.stdout.org/rcookbook/Graphs/

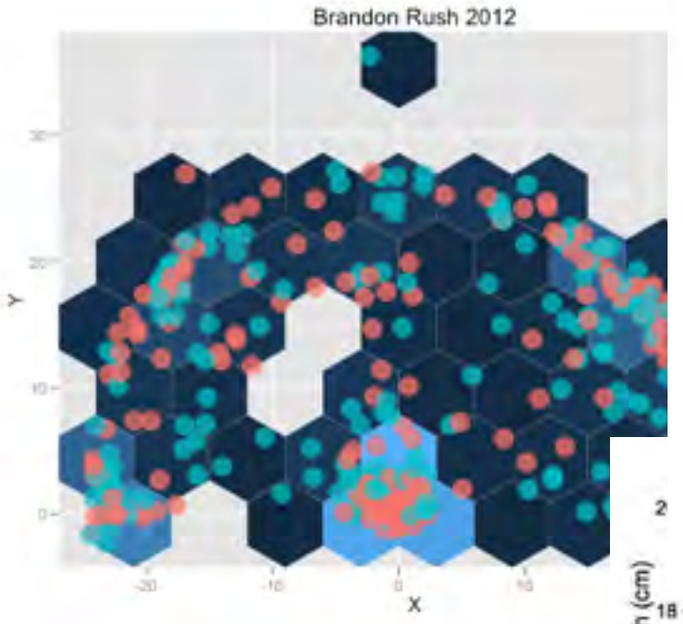
Winston Chang's R Graphics Cookbook
http://www.amazon.com/R-Graphics-Cookbook-Winston-Chang/dp/1449316956

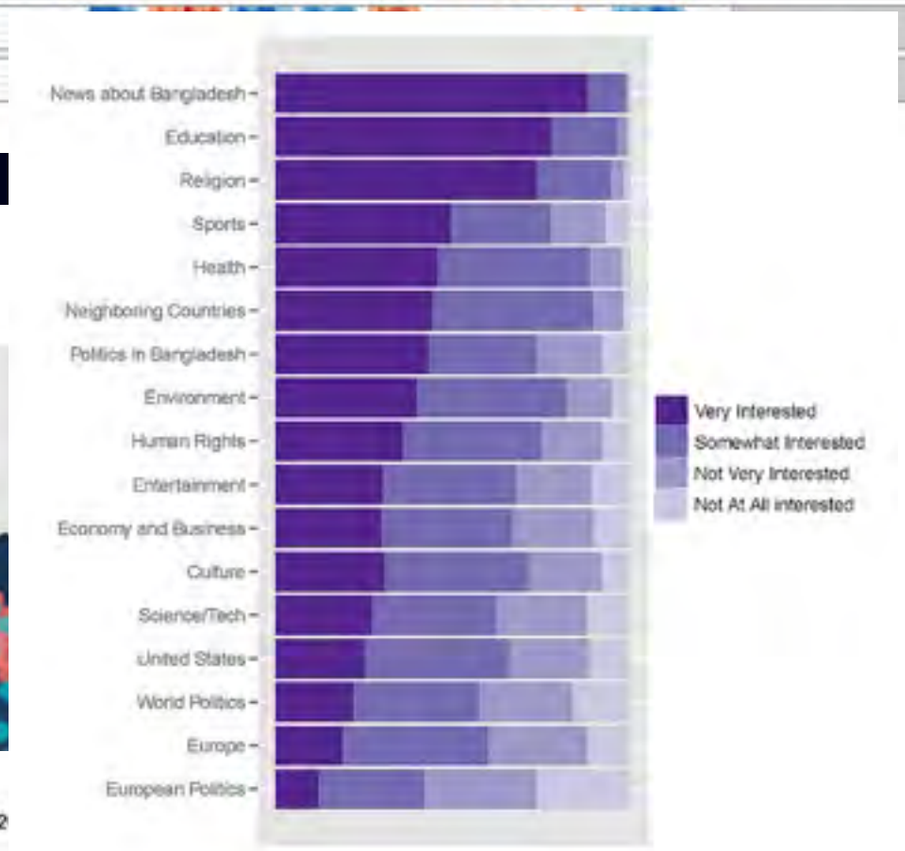



Convertible Arbitrage
CTA Global
Distressed Securities
Emerging Markets
Equity Market Neutral
Event Driven
Fixed Income Arbitrage
Global Macro
Long/Short Equity
Merger Arbitrage
Relative Value

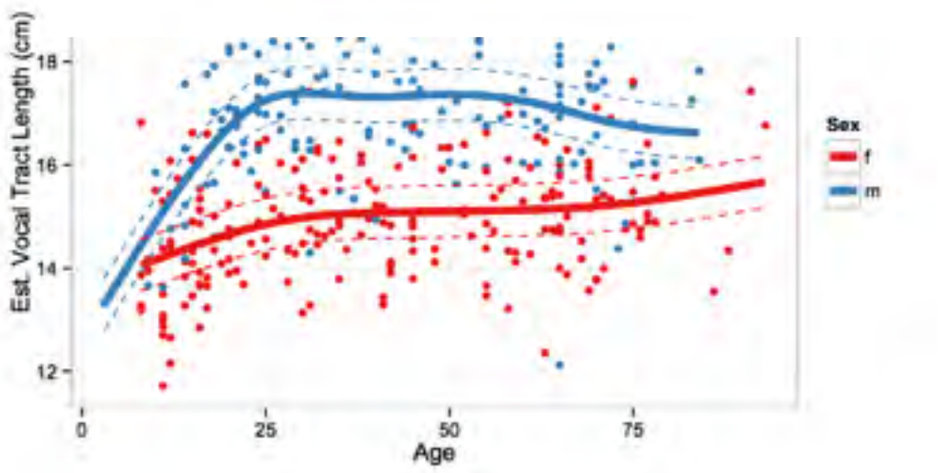

Bunfight
Big Mac index
-40
-20
0
20
40
60
80
100
Switzerland
Brazil
Euro area
Canada
Japan
United States
Britain
Singapore
South Korea
South Africa
Mexico
Thailand
Russia
Malaysia
China
6.78
5.26
4.79
4.18
3.91
3.71
3.63
3.45
Sources:
McDonald's;
The Economist


25
50
75
100
Field
Tropical


Brandon Rush 2012
Y
X


News about Bangladesh
Education
Religion
Sports
Health
Neighboring Countries
Politics in Bangladesh
Environment
Human Rights
Entertainment
Economy and Business
Culture
Science/Tech
United States
World Politics
Europe
European Politics
Very Interested
Somewhat Interested
Not Very Interested
Not At All Interested

0
Latitude
MAP (mm/yr)
2500
2000
1500
1000


Est. Vocal Tract Length (cm)
18
16
14
12
0
25
50
75
Age
Sex
f
m