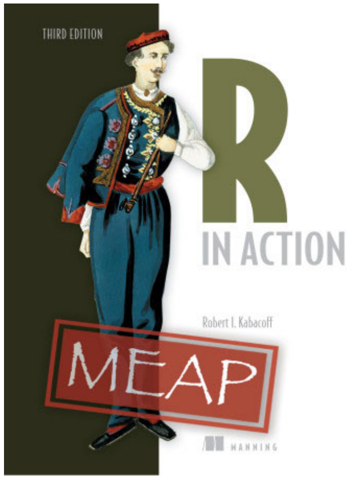


40% discount: twitkaba40

Welcome

We'll be staring in a few minutes

Github: github.com/Rkabacoff/twitch

ggplot2

MODERN DATA VISUALIZATION WITH R
ROB KABACOFF

Who am I?

Not a clue, but here goes…

- Professor, Quantitative Analysis Center, Wesleyan University
- 35 years experience as data scientist (20 in industry)
- Author - R in Action (3rd ed), Quick-R website

What is R

- A language and environment for statistical computing and graphics
- Based on the "S" Language developed at Bell Labs (1976)
- R was first created by **R**oss Ihaka and **R**obert Gentleman at the Univ. of Auckland in 1993

Why R?

- Free
- Open source
- State-of-the-art graphics and data analysis
- Platform for programming new methods
- Runs on Windows, Linux, Mac OS X
- Enormous user base

Packages

- Collections of R functions, data, and compiled code in well-defined format
- Massively extend the functionality of R
- Thousands of user written packages on CRAN http://cran.r-project.or/web/packages

21,276 as of last night

Gaphics

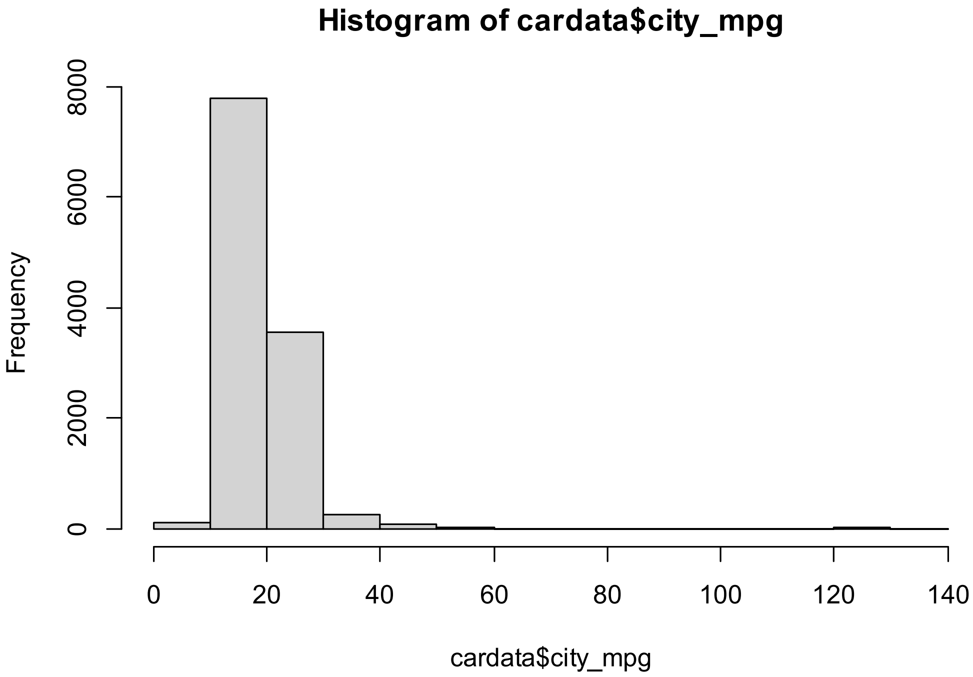
Static

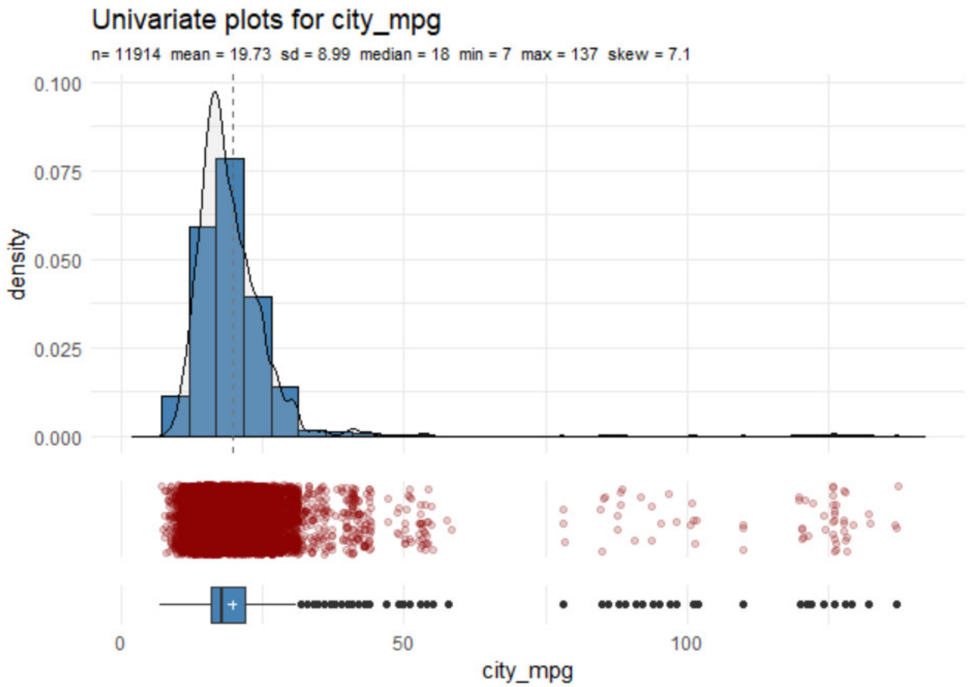
- base
- grid
- lattice
- D3 (via r2d3)
- **ggplot2**

Interactive

- **leaflet**
- **plotly**
- rbokeh
- rCharts
- highcharter
- base (very limited)
- Shiny (as platform)

Examples





hist(cardata$city_mpg)

base graphics

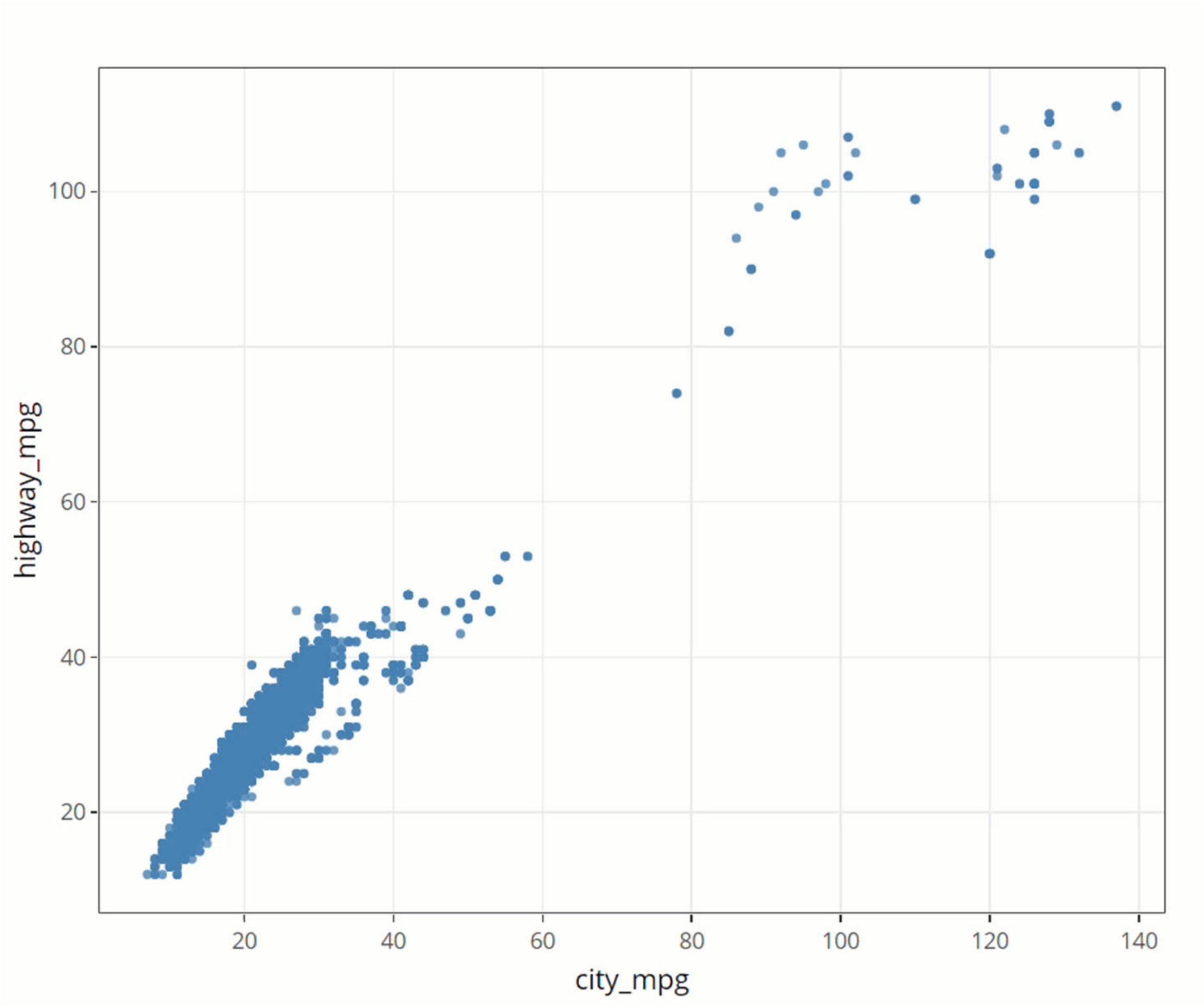
univariate_plot(cardata, city_mpg)

ggplot2 graphics

Examples

Interactive

ggplot + plotly



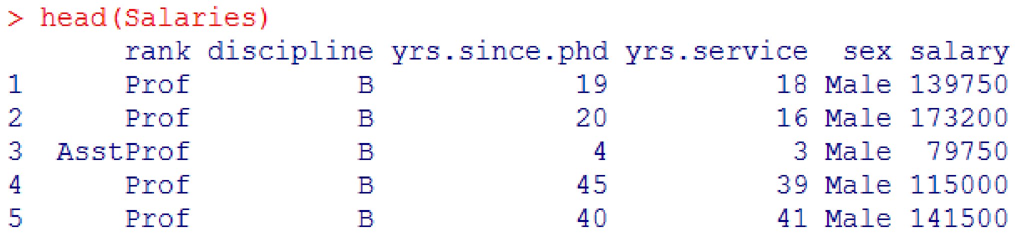
Datasets

data(Salaries, package="carData")

Salaries from the **carData** package
(2008-2009 9 month academic salaries n=397)

```
url <- "https://bit.ly/3bsMwsS"
Salaries <- read.csv(url)
```

1. rank (AssocProf, AsstProf, Prof)
2. salary in dollars
3. discipline (A=theoretical, B=applied)
4. sex (Female, Male)
5. yrs.since.phd.
6. yrs.service



```
> head(Salaries)
       rank discipline yrs.since.phd yrs.service  sex salary
1      Prof          B            19          18 Male 139750
2      Prof          B            20          16 Male 173200
3  AsstProf          B             4           3 Male  79750
4      Prof          B            45          39 Male 115000
5      Prof          B            40          41 Male 141500
```

Grammar of Graphics

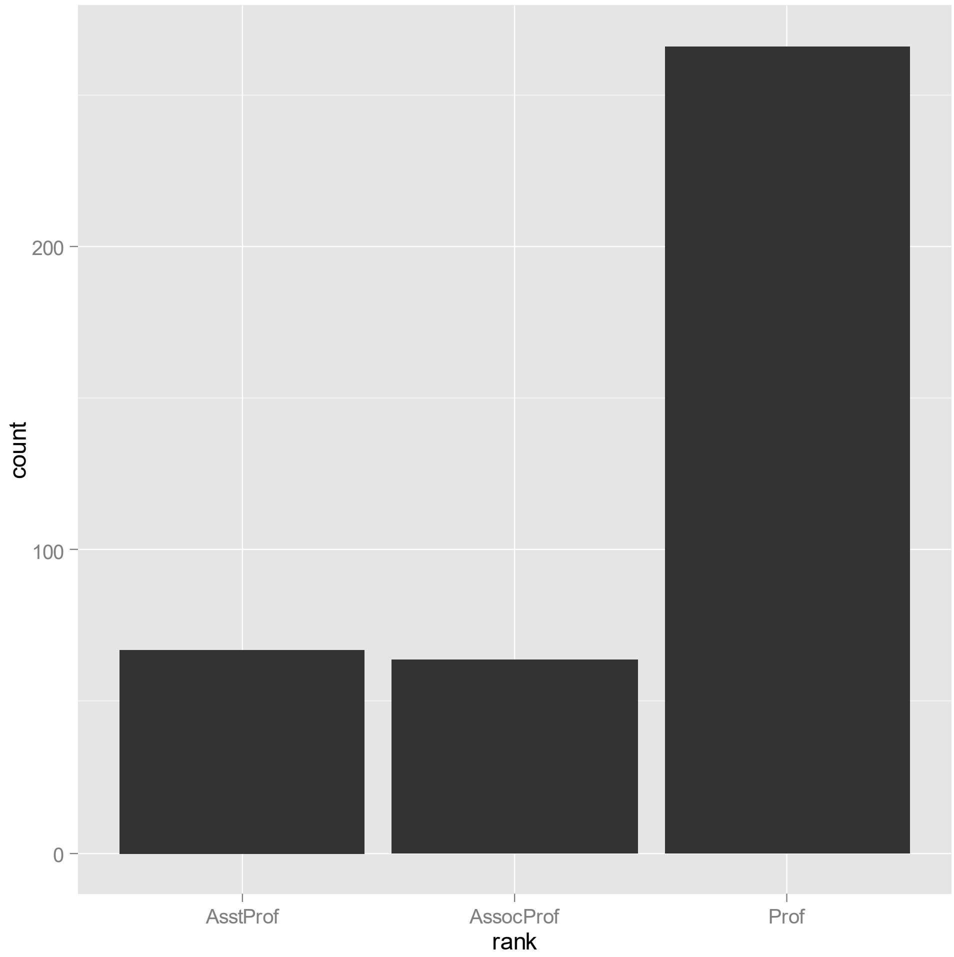
- data: an R data frame
- coordinate system: 2-D space data projected onto (e.g. Cartesian coordinates, polar coordinates, map projections)
- geoms: type of geometric objects that represent data (e.g. points, lines, bars)
- aesthetics: visual characteristics that represent data (e.g. position, size, color, shape, transparency, fill)
- scales: for each aesthetic, how visual characteristic is converted to display values
- stats: statistical transformations that summarize data (e.g., counts, means, trend lines)
- facets: how data is split into subsets and displayed as small multiples

Thanks to Dawn Koffman ggplot2 presentation

Simple bar plot

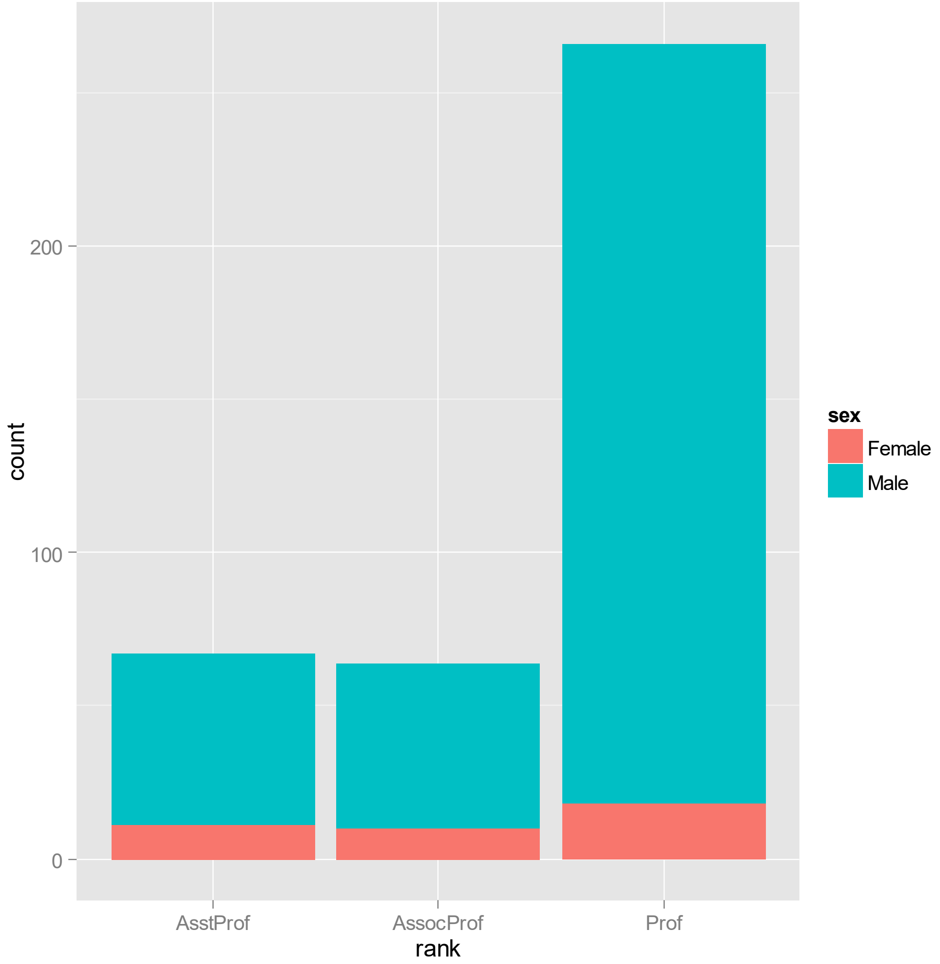
```
ggplot(data=Salaries,
    aes(x=rank)) +
geom_bar()
```

common geom_bar options:
- width
- fill
- color (border)
- position



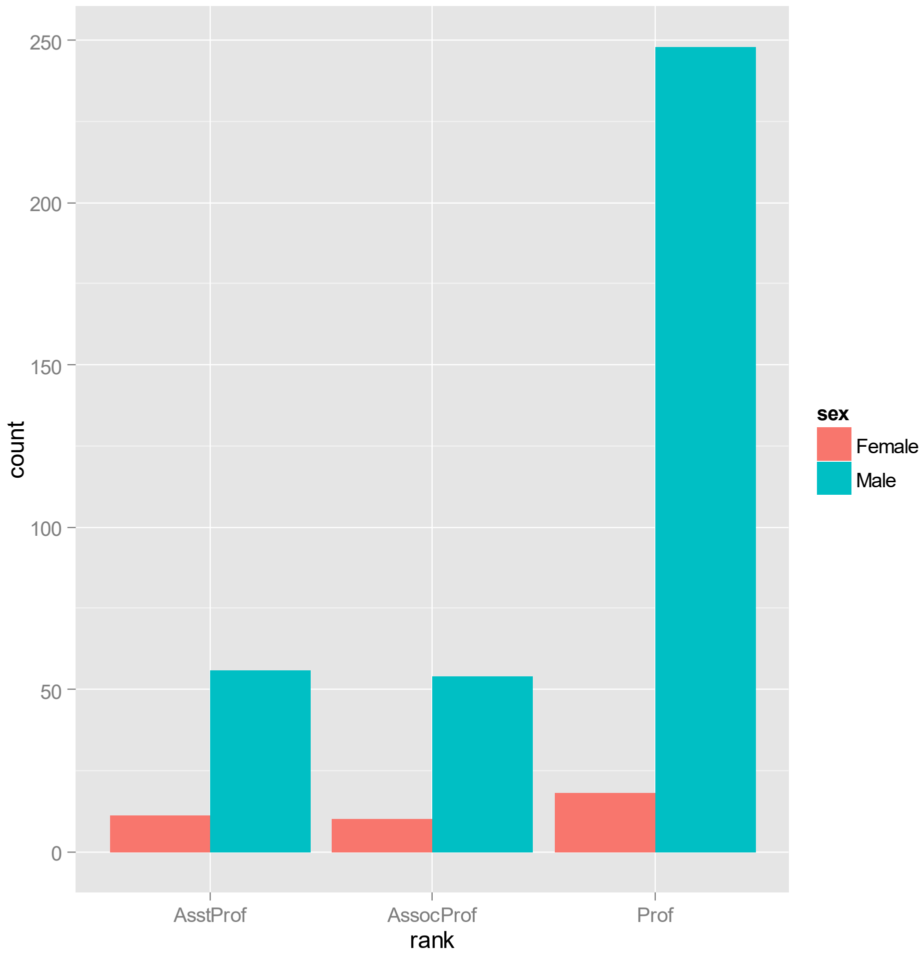
Stacked bar plot

```
ggplot(data=Salaries,
    aes(x=rank, fill=sex)) +
geom_bar()
```



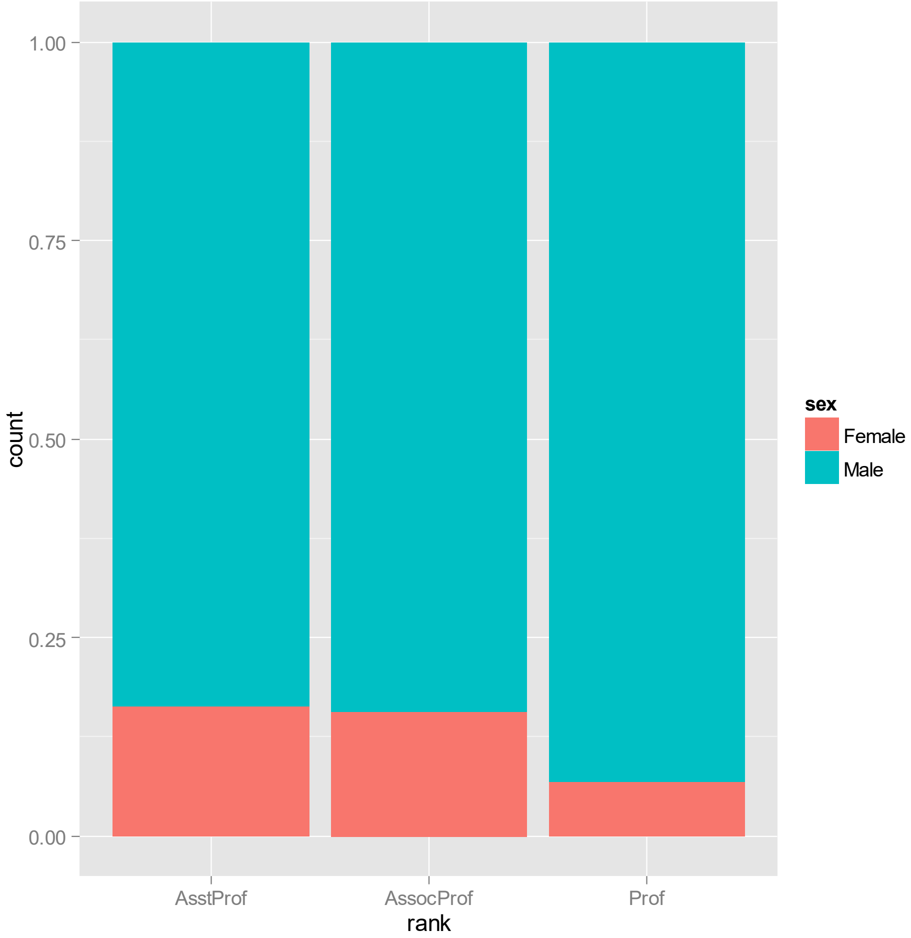
Grouped bar plot

```
ggplot(data=Salaries,
    aes(x=rank, fill=sex)) +
geom_bar(position="dodge")
```



Spinogram

```
ggplot(data=Salaries,
    aes(x=rank, fill=sex)) +
geom_bar(position="fill")
```

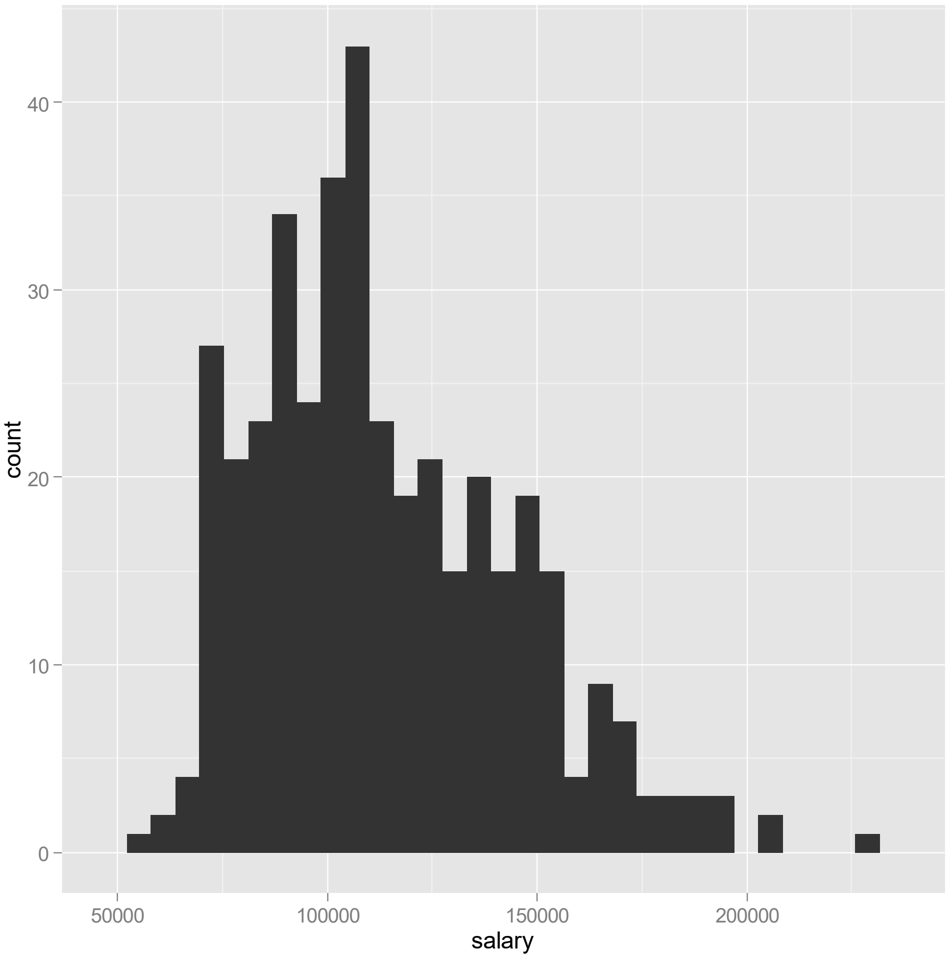


Histogram

```
ggplot(data=Salaries,
    aes(x=salary)) +
geom_histogram()
```

common geom_histogram options:

- binwidth
- bins
- color (border)
- fill

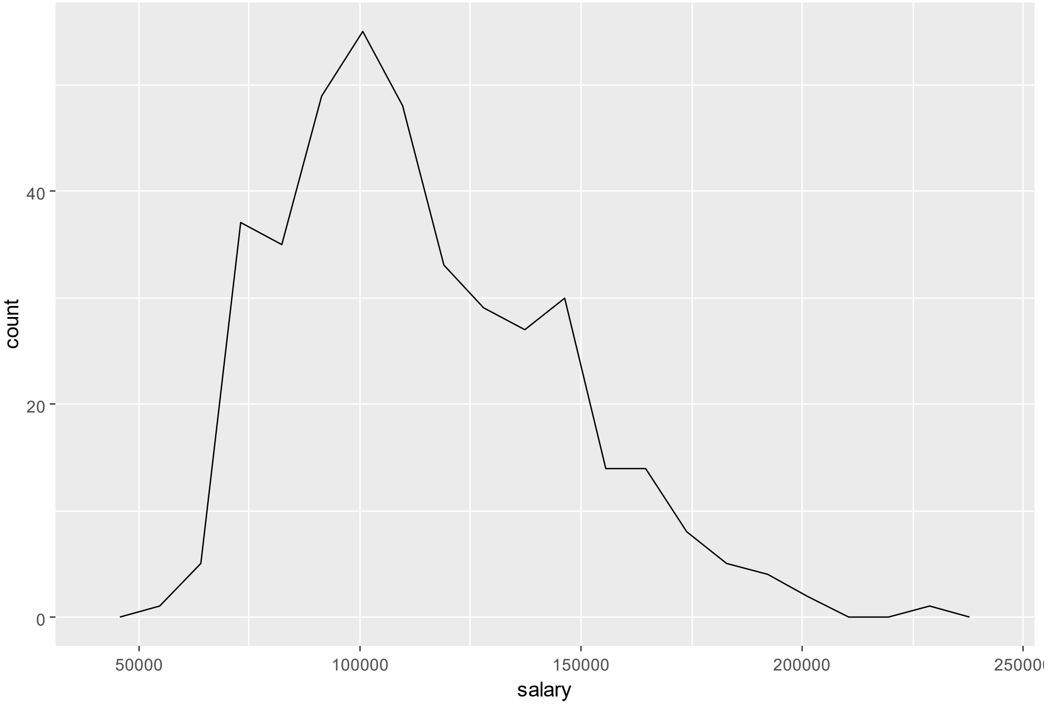


Frequency polygons

```
ggplot(data=Salaries,
    (x=salary)) +
geom_freqpoly()
```

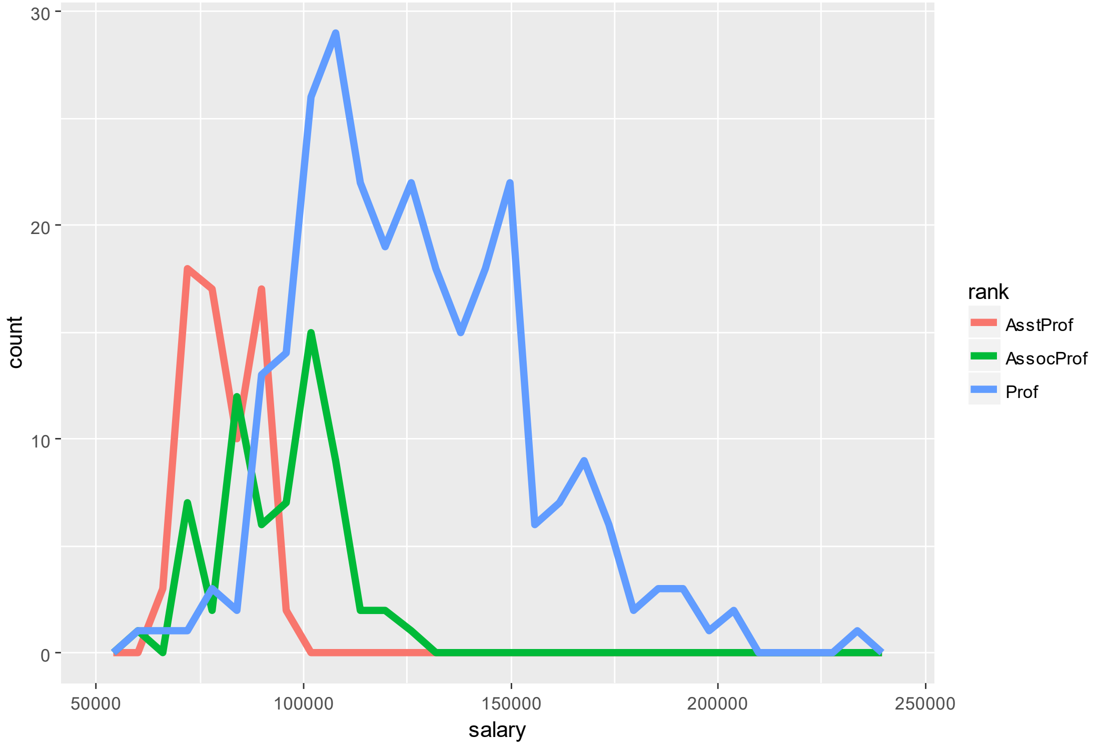
common geom_freqpoly options:

- binwidth
- bins
- color
- size (thickness of line)



Frequency polygons

```
ggplot(data=Salaries,
    aes(x=salary, color=rank)) +
geom_freqpoly(size=2)
```



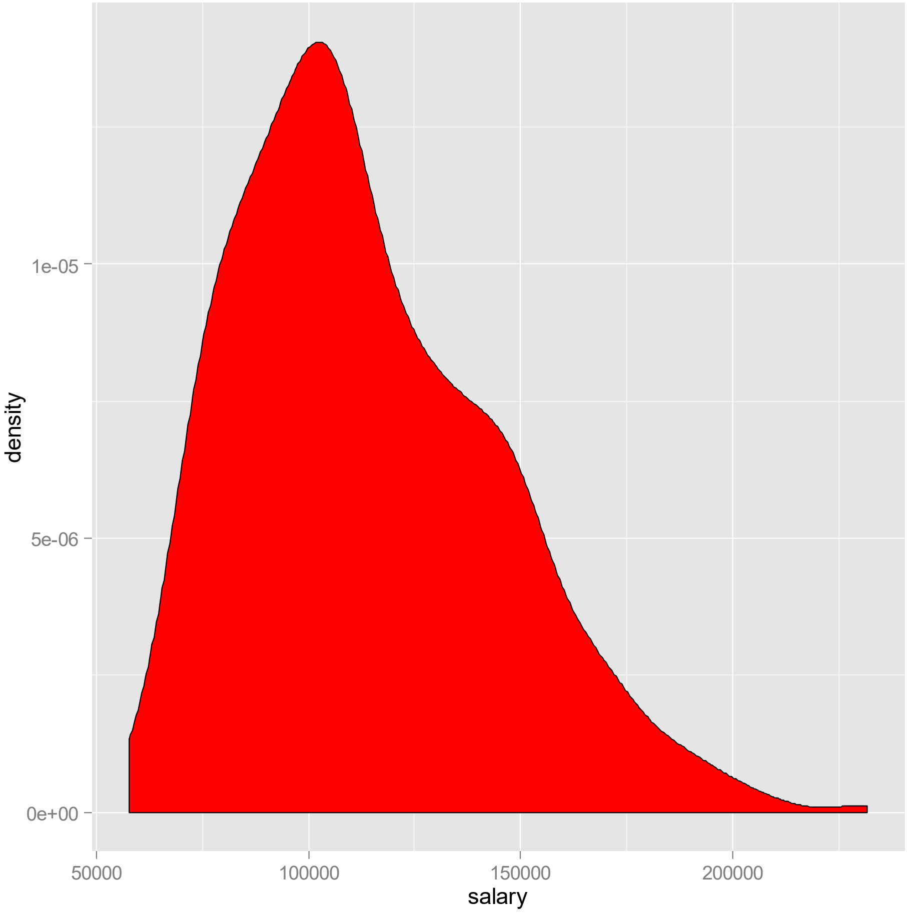
common geom_freqpoly options:

- binwidth
- bins
- color
- size (thickness of line)

Kernel density plot

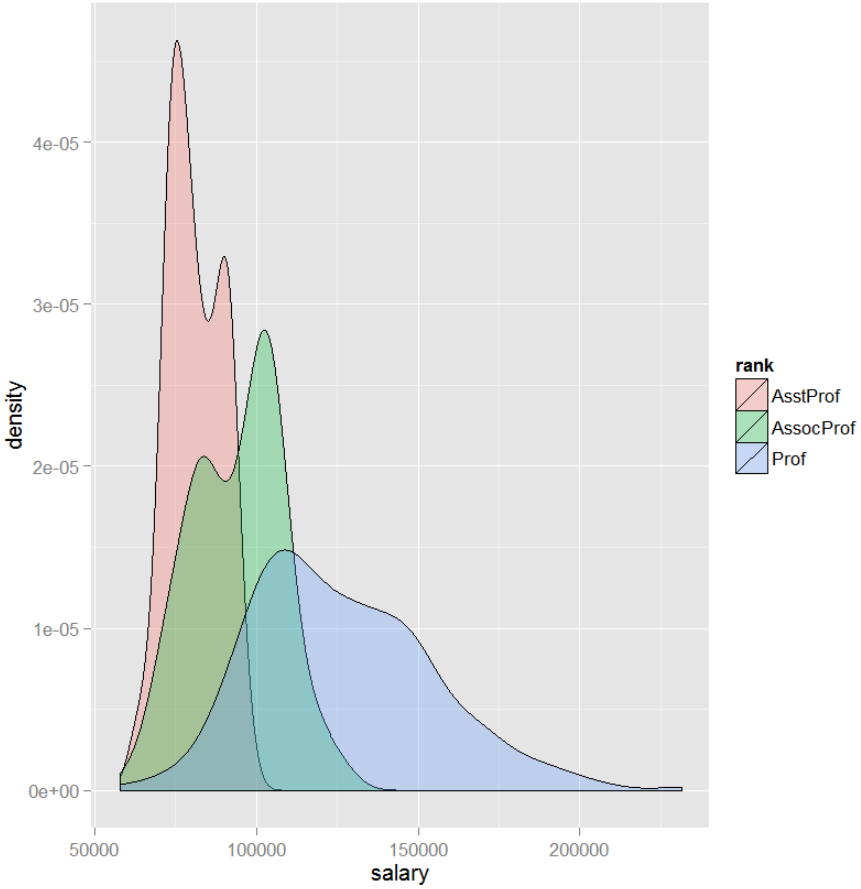
```
ggplot(data=Salaries,
    aes(x=salary)) +
geom_density(fill="red")
```

common geom_density options:
- fill
- color
- alpha



Kernel density plot - multiple groups

```
ggplot(data=Salaries,
    aes(x=salary, fill=rank)) +
geom_density(alpha=.3)
```

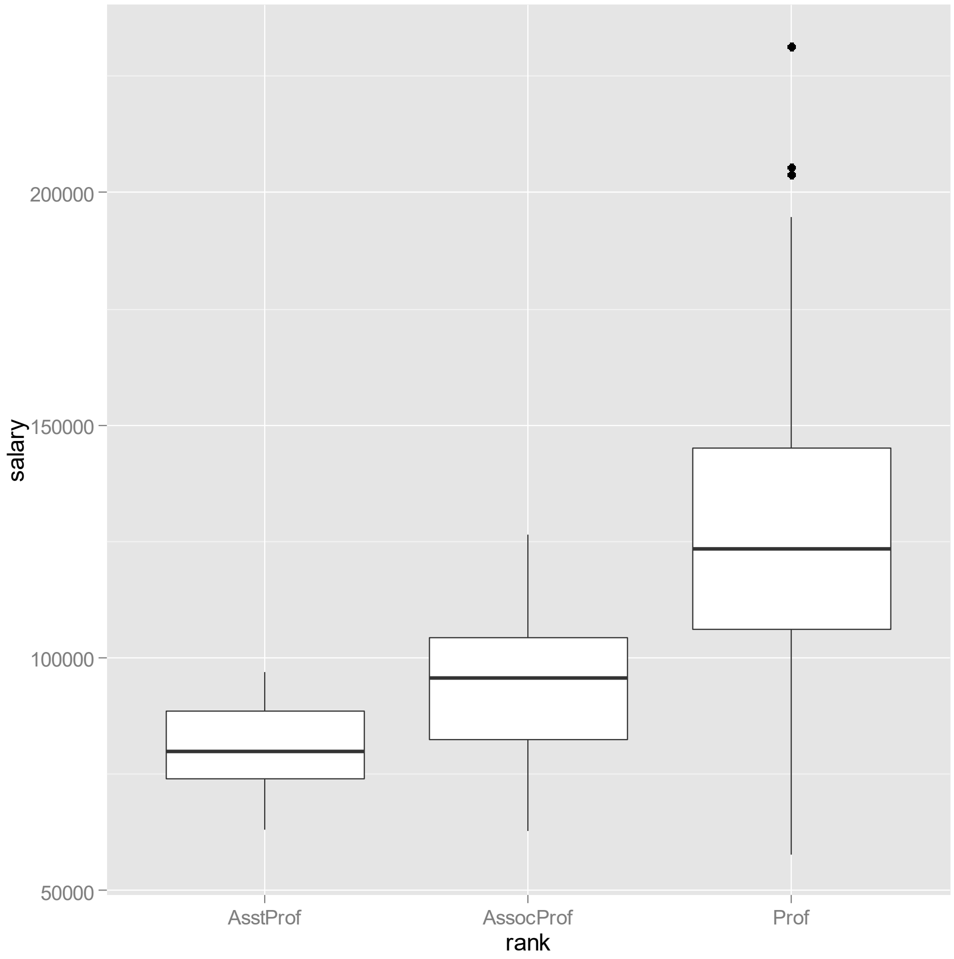


Box plot

```
ggplot(data=Salaries,
    aes(x=rank, y=salary)) +
geom_boxplot()
```

common geom_boxplot options:

fill
color
notch (=TRUE or FALSE)
outlier. -color shape size

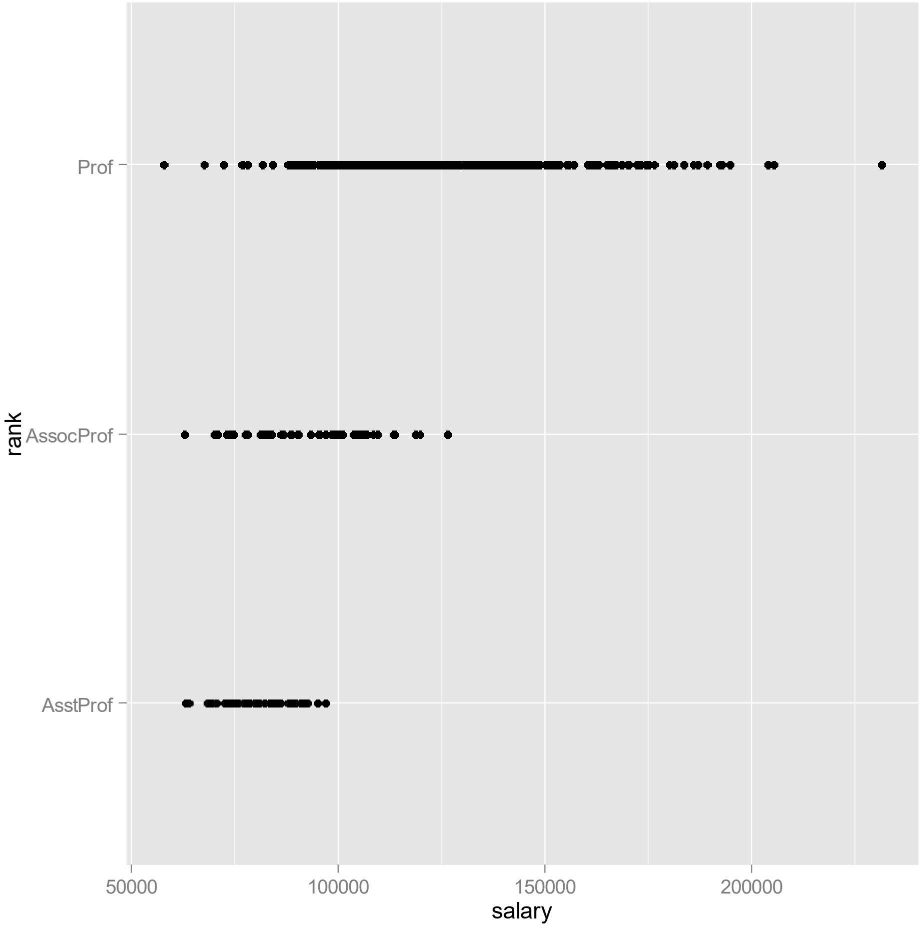


Strip plot

```
ggplot(data=Salaries,
    aes(x=salary, y=rank)) +
geom_point()
```

common geom_point options:

- color
- alpha
- shape
- size

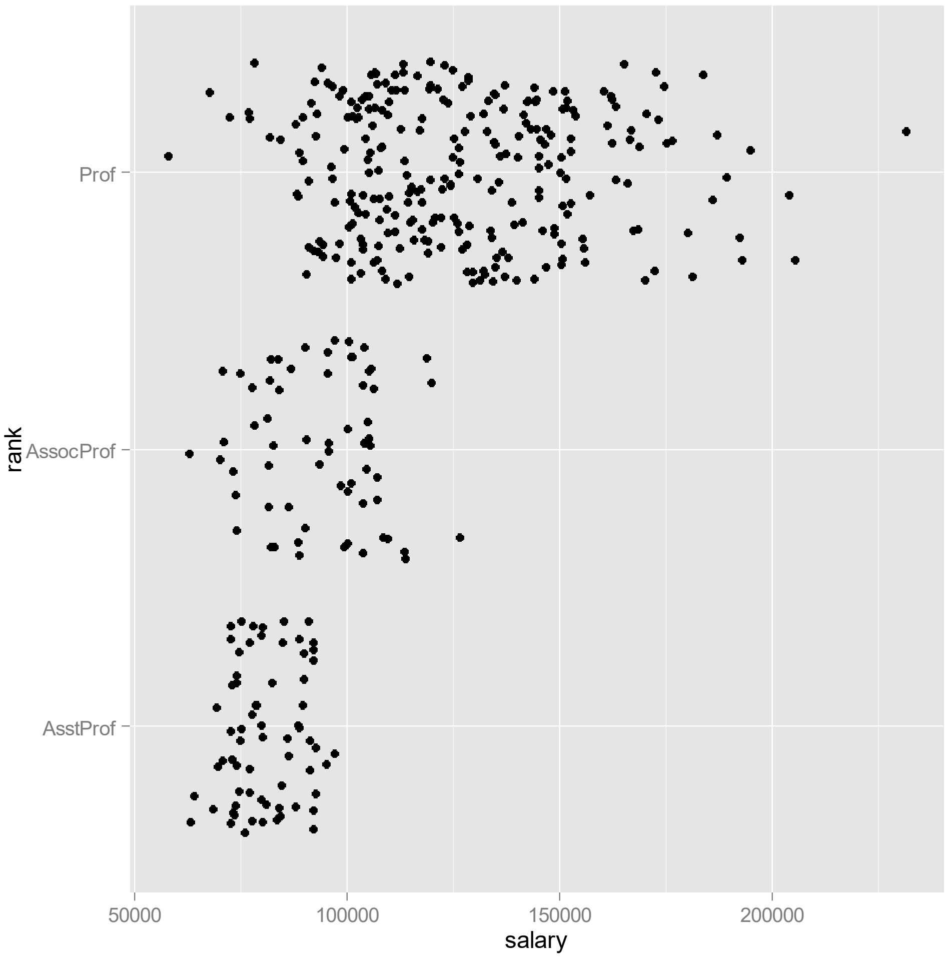


Jittered Strip plot

```
ggplot(data=Salaries,
    aes(x=salary, y=rank)) +
geom_jitter()
```

common geom_jitter options:

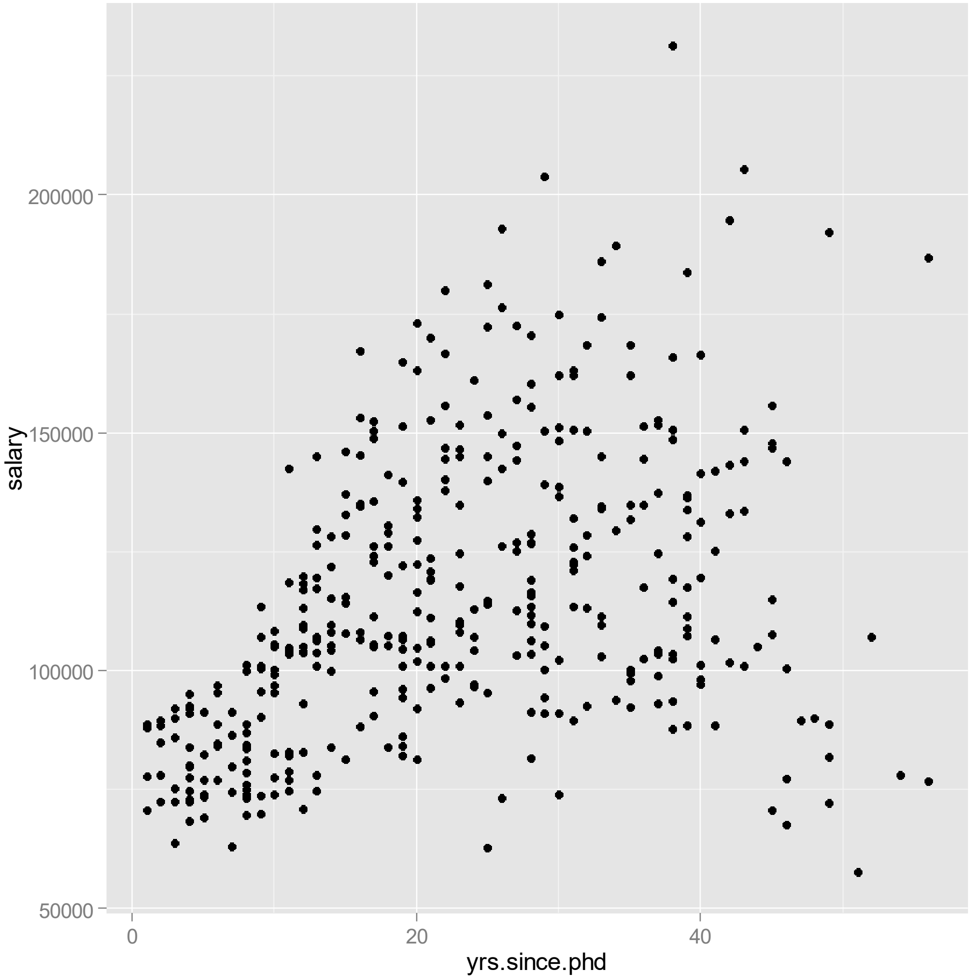
- color
- alpha
- shape
- size



Scatter plot

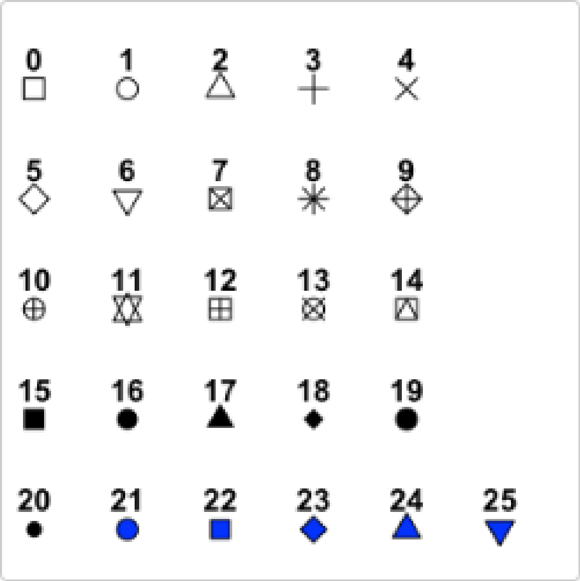
```
ggplot(data=Salaries,
    aes(x=yrs.since.phd,
        y=salary)) +
geom_point()
```

common geom_point options:
- color
- alpha
- shape
- size



Changing point shapes

```
+ geom_point(shape = 15)
```

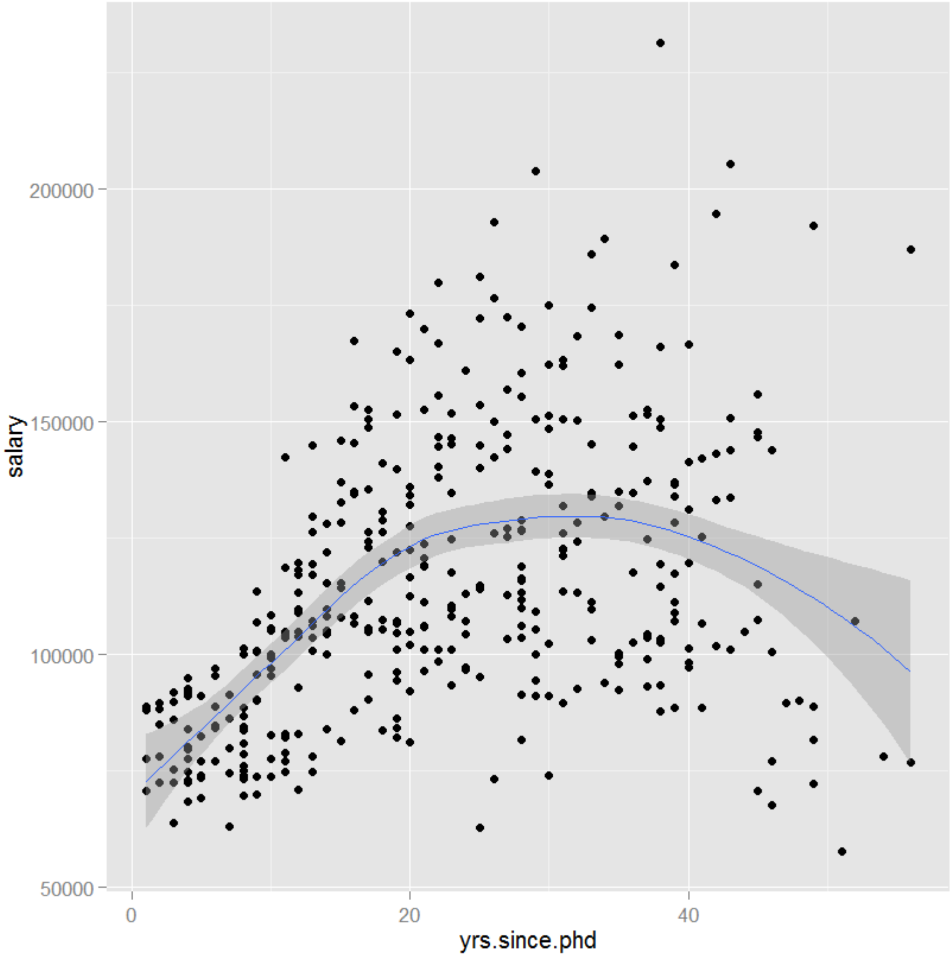


for 21-25 you
can control both
the fill and the border

Scatterplot with fit

```
ggplot(data=Salaries,
   aes(x=yrs.since.phd,
       y=salary)) +
geom_point() +
geom_smooth()
```

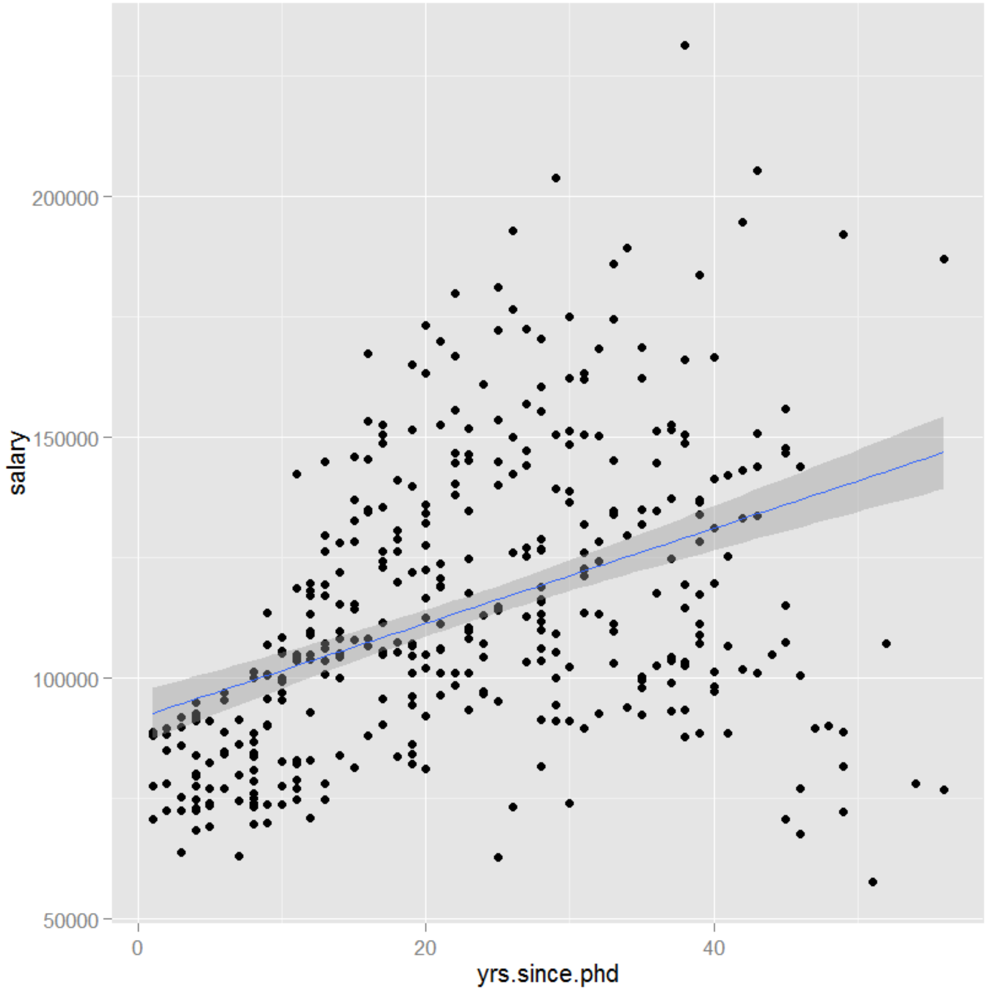
common geom_smooth options
  method ("lm", "loess", "gam")
  se (TRUE or FALSE)
  formula



Scatterplot with fit

```
ggplot(data=Salaries,
  aes(x=yrs.since.phd,
      y=salary)) +
geom_point() +
geom_smooth(method="lm",
            formula=y~x)
```

try formula = y~poly(x, 2)



Grouping

Add

- *color,*
- *shape,*
- *size,*
- *alpha*

to

aes()

or the

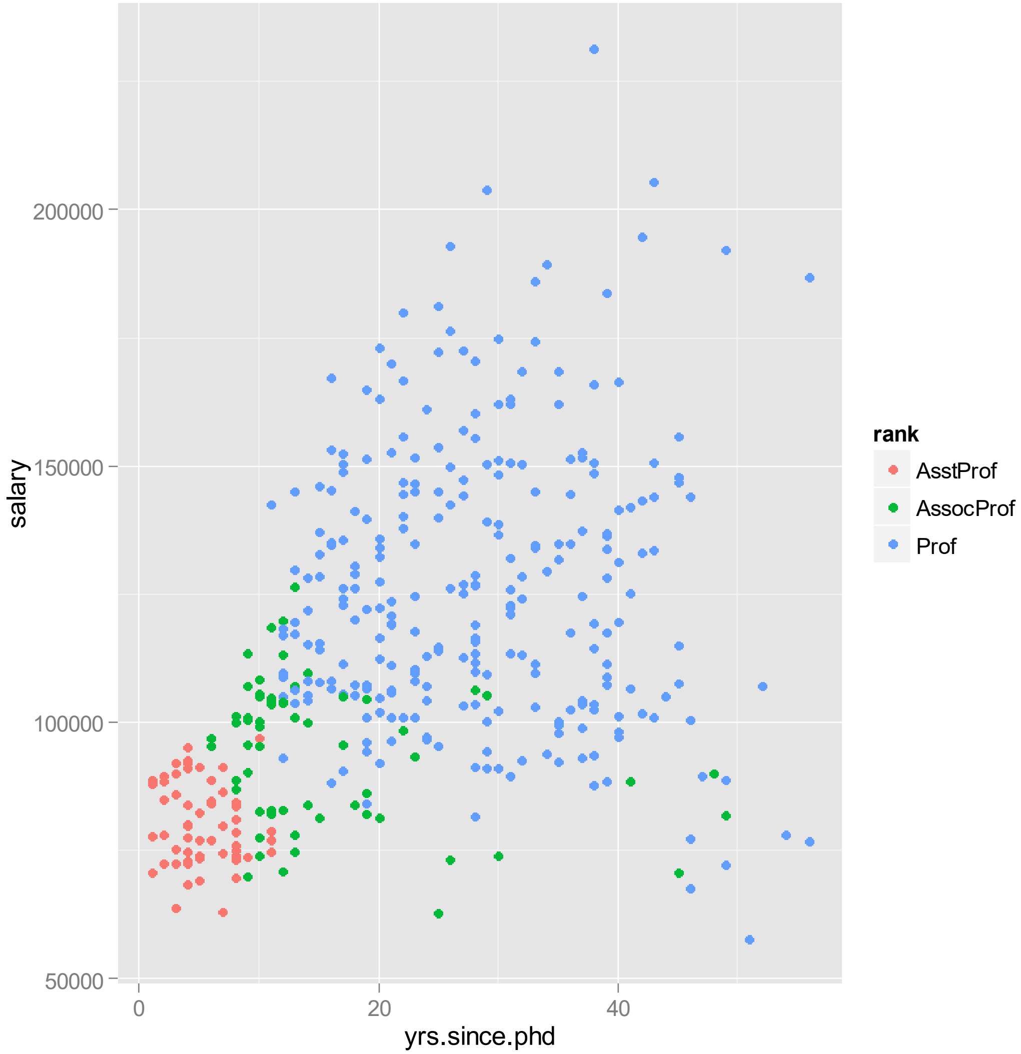
geom_xxx()

be careful of aesthetics
fill=sex

vs attributes!
fill="blue"

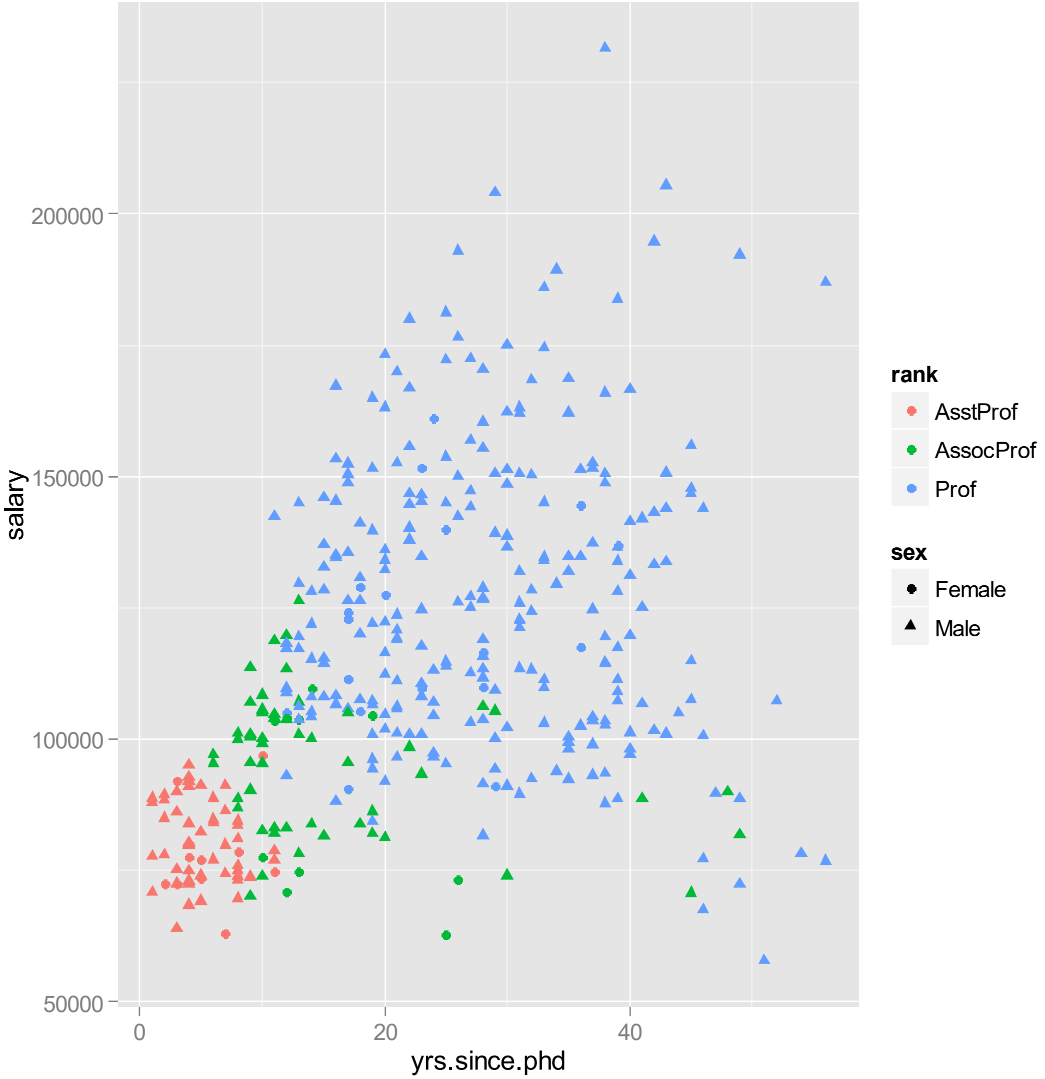
Grouping

```
ggplot(data=Salaries,
  aes(x=yrs.since.phd,
      y=salary,
      color=rank)) +
geom_point()
```



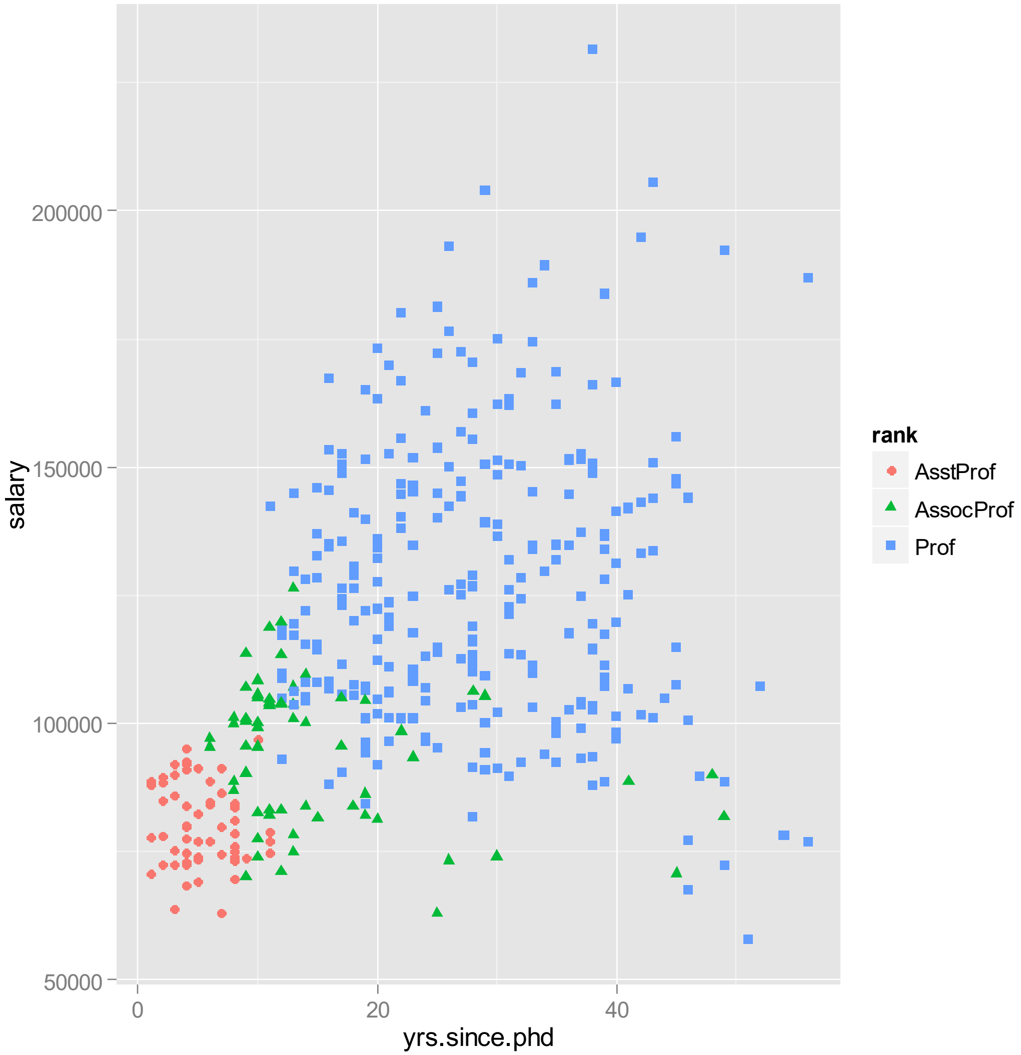
Grouping

```
ggplot(data=Salaries,
  aes(x=yrs.since.phd,
      y=salary,
      color=rank,
      shape=sex)) +
geom_point()
```



Grouping

```
ggplot(data=Salaries,
  aes(x=yrs.since.phd,
      y=salary,
      color=rank,
      shape=rank)) +
geom_point()
```

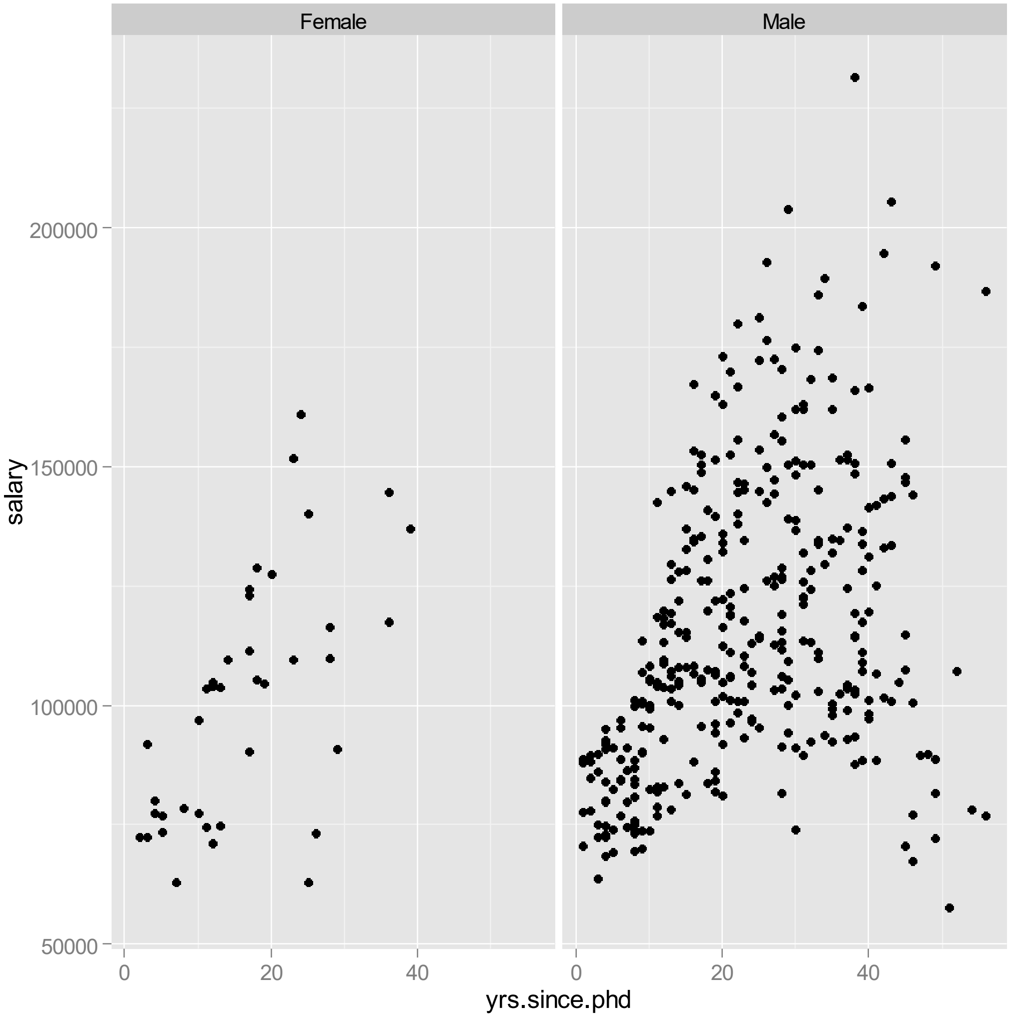


Facets

facets_grid(rowvar ~ colvar)	rows by columns
facets_grid(. ~ colvar)	just columns
facets_grid(rowvar ~ .)	just rows
facets_wrap(~ var, ncol=#)	one classification variable wrapped to fill page

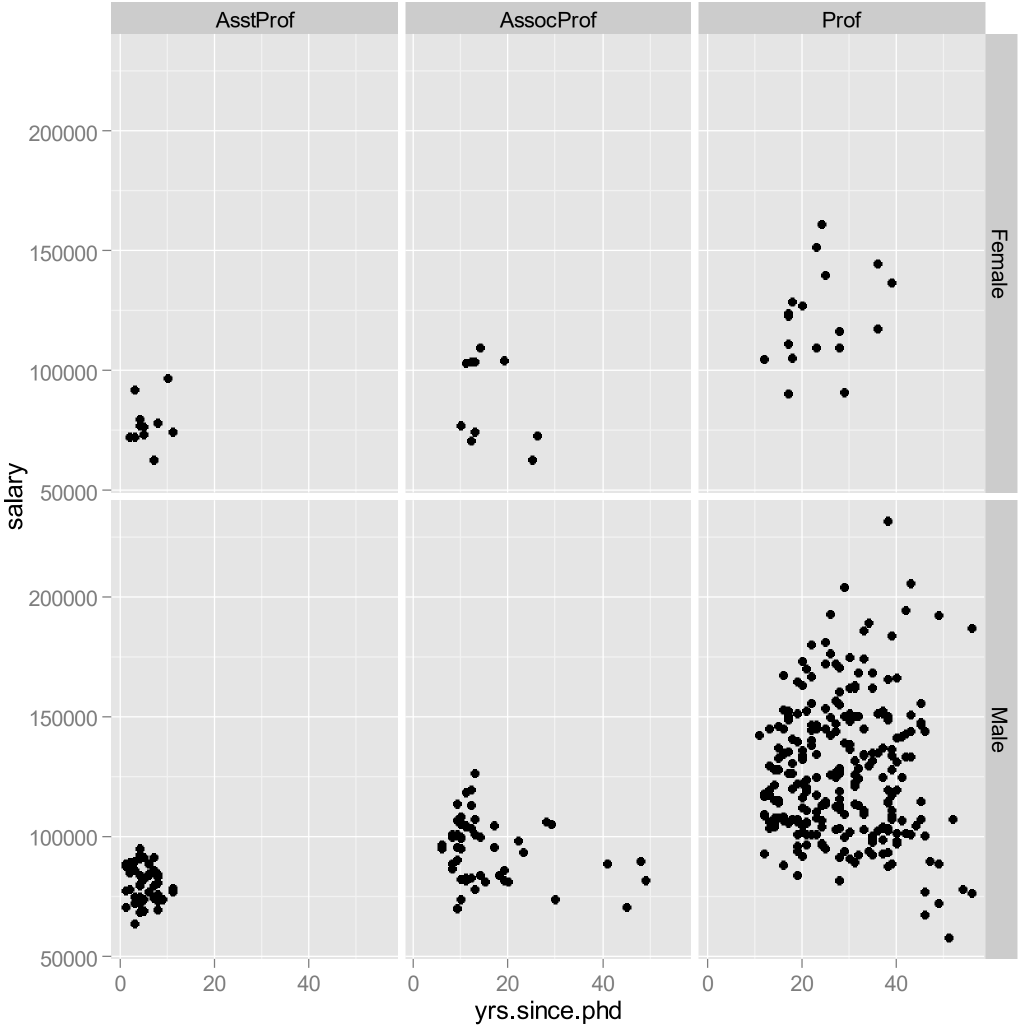
Facets

```
ggplot(data=Salaries,
  aes(x=yrs.since.phd,
      y=salary)) +
geom_point() +
facet_grid(. ~ sex)
```



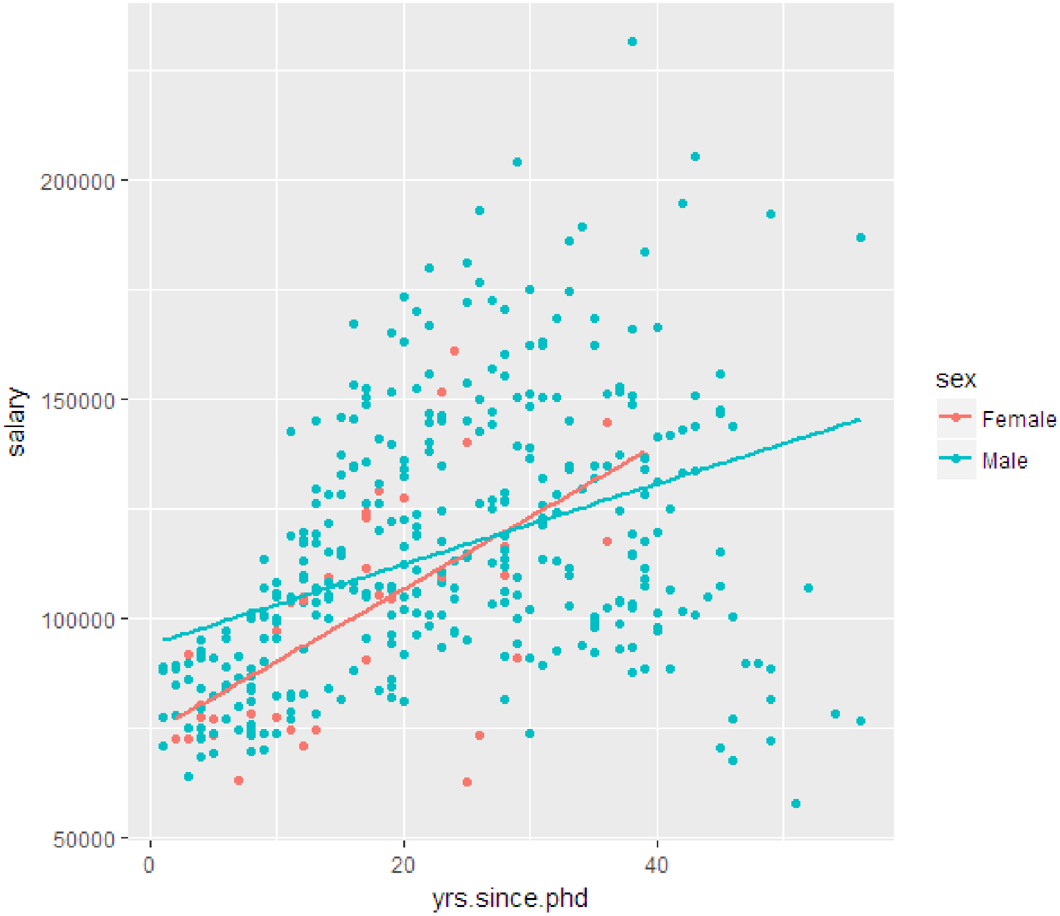
Facets

```
ggplot(data=Salaries,
  aes(x=yrs.since.phd,
      y=salary)) +
geom_point() +
facet_grid(sex ~ rank)
```



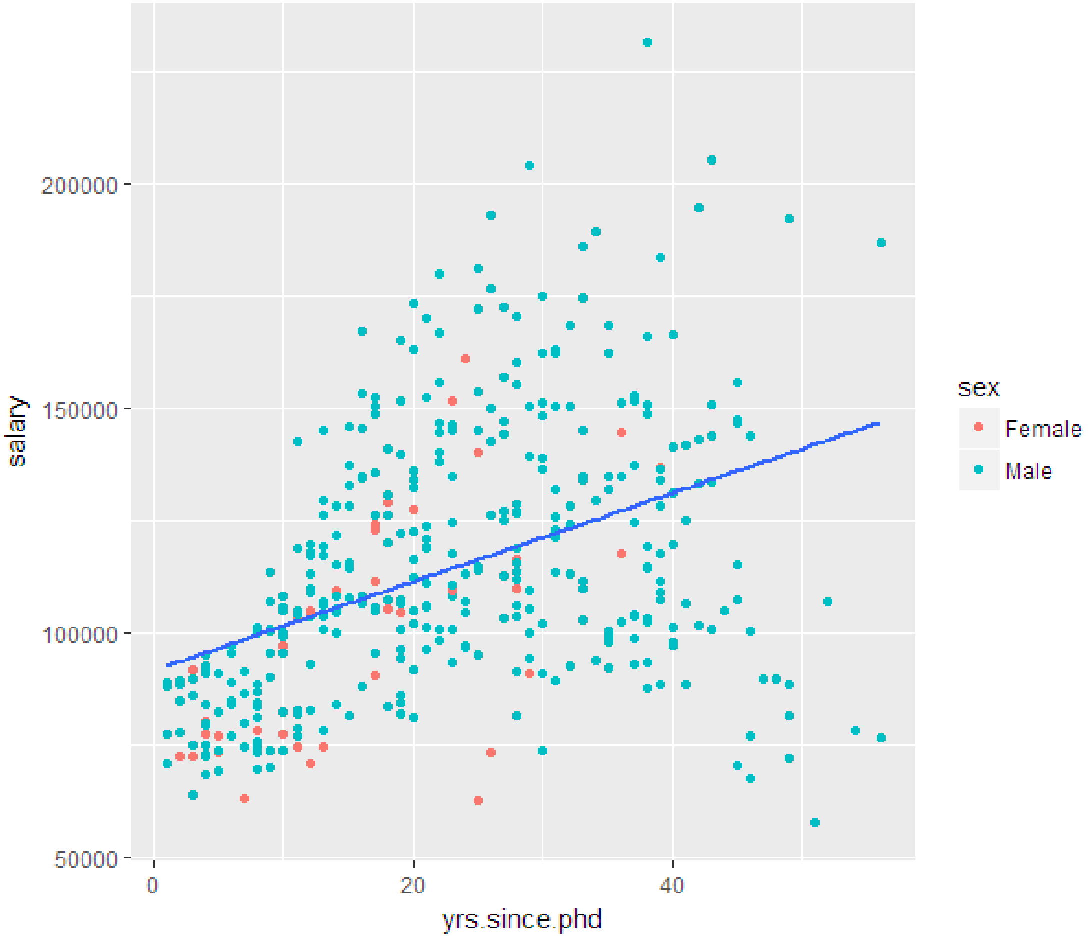
Aesthetics in ggplot( ) vs geom_xxx( )

```
ggplot(data=Salaries,
    aes(x=yrs.since.phd,
        y=salary,
        color=sex )) +
  geom_point() +
  geom_smooth(method="lm", se=FALSE)
```



Aesthetics in ggplot( ) vs geom_xxx( )

```
ggplot(data=Salaries,
       aes(x=yrs.since.phd,
           y=salary )) +
  geom_point(aes(color=sex)) +
  geom_smooth(method="lm", se=FALSE)
```

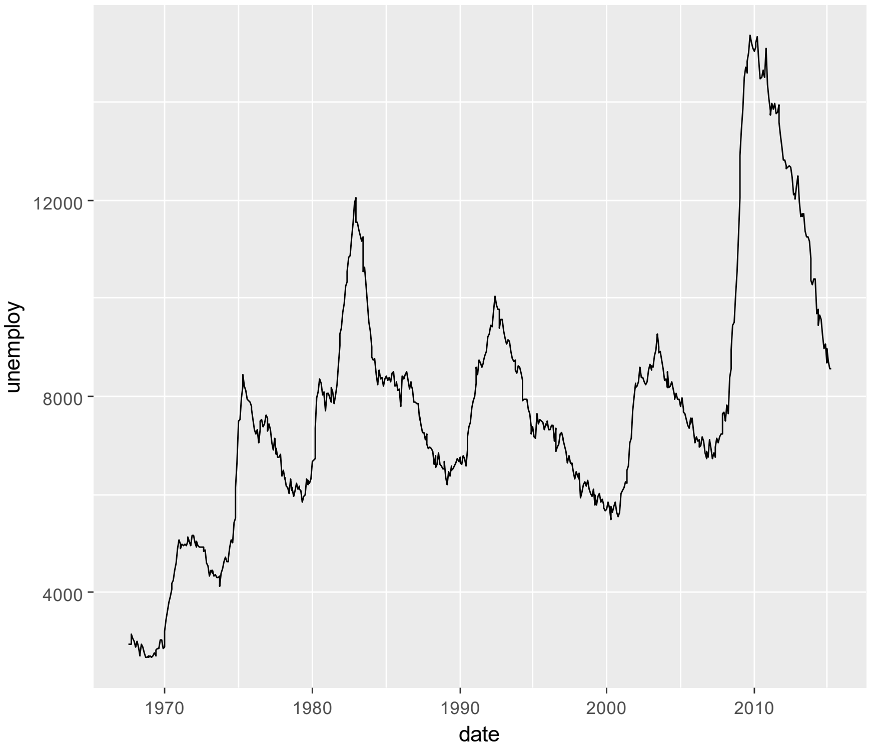


Geoms

geom_abline	Reference lines: horizontal, vertical, and diagonal
geom_bar	Bars charts
geom_bin2d	Heatmap of 2d bin counts
geom_blank	Draw nothing
geom_boxplot	A box and whiskers plot (in the style of Tukey)
geom_contour	2d contours of a 3d surface
geom_count	Count overlapping points
geom_density	Smoothed density estimates
geom_density_2d	Contours of a 2d density estimate
geom_dotplot	Dot plot
geom_errorbarh	Horizontal error bars
geom_hex	Hexagonal heatmap of 2d bin counts
geom_freqpoly	Histograms and frequency polygons
geom_jitter	Jittered points
geom_crossbar	Vertical intervals: lines, crossbars & errorbars
geom_map	Polygons from a reference map
geom_path	Connect observations
geom_point	Points
geom_polygon	Polygons
geom_qq	A quantile-quantile plot
geom_quantile	Quantile regression
geom_ribbon	Ribbons and area plots
geom_rug	Rug plots in the margins
geom_segment	Line segments and curves
geom_smooth	Smoothed conditional means
geom_spoke	Line segments parameterised by location, direction and distance
geom_label	Text
geom_raster	Rectangles
geom_violin	Violin plot

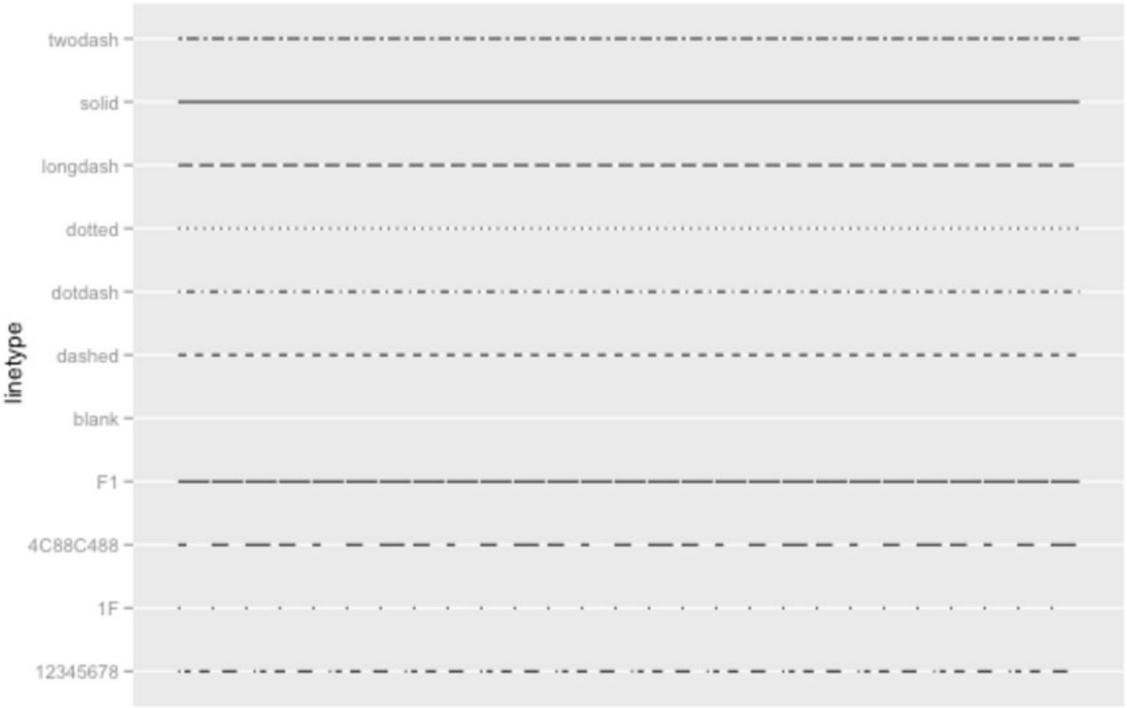
Line charts

```
ggplot(economics, aes(date, unemploy)) + geom_line()
```



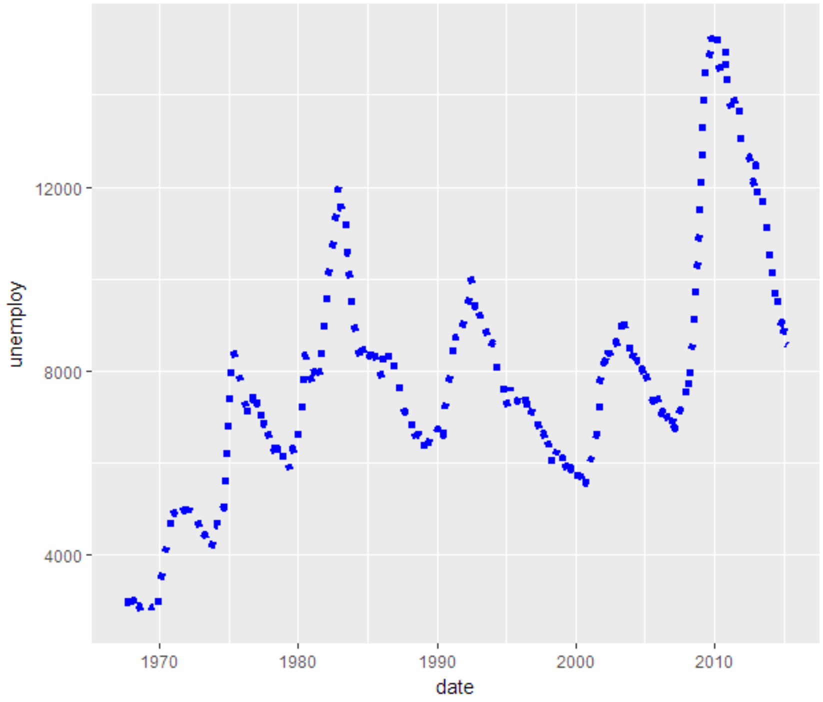
Line charts

Changing the linetype



Line charts

```
ggplot(economics, aes(date, unemploy)) +
geom_line(linetype="dotted", color="blue", size=1)
```



Scales

```
scale_x_continuous()
scale_y_continuous()

scale_x_discrete()
scale_y_discrete()
```

Axes

```
scale_color_continuous()
scale_color_manual()
scale_color_brewer()
```

Colors

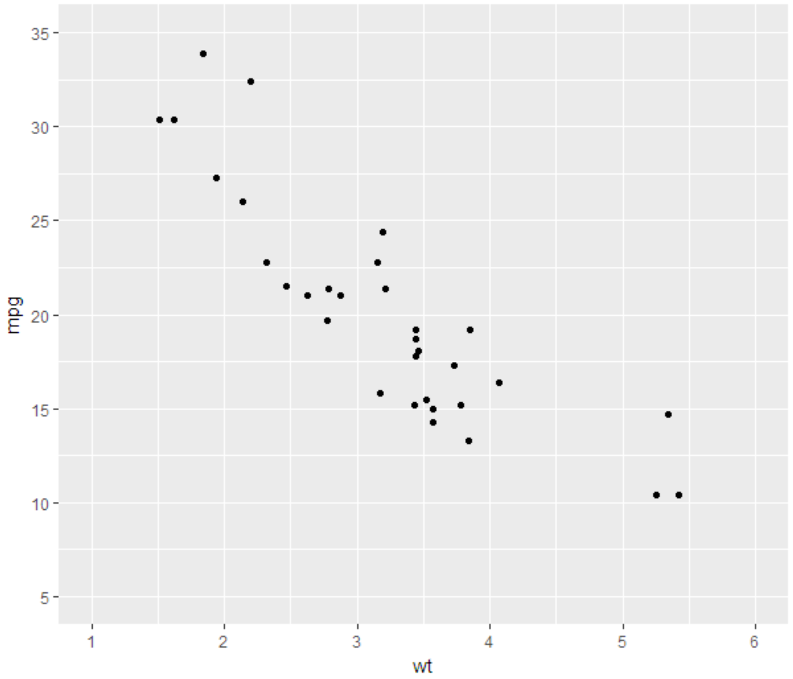
```
scale_fill_continuous()
scale_fill_manual()
```

Fill

Also shape,
and size

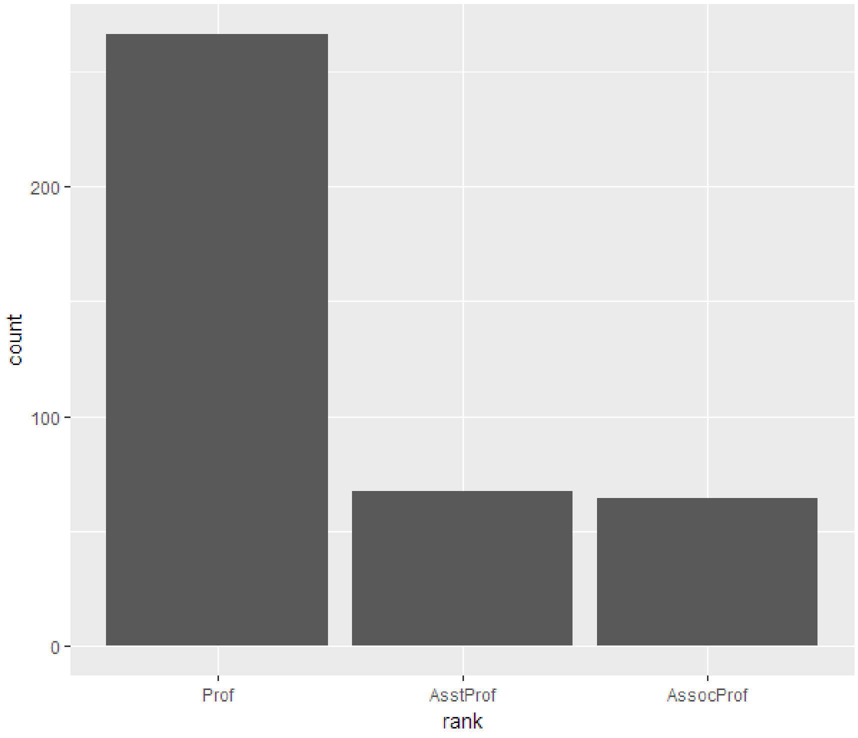
Scales

```
ggplot(mtcars, aes(x=wt, y=mpg)) + geom_point() +
  scale_x_continuous(breaks=seq(1,6,1), limits=c(1, 6)) +
  scale_y_continuous(breaks=seq(5, 35, 5),limits=c(5,35))
```



Scales

```
ggplot(Salaries, aes(x=rank)) + geom_bar() +
  scale_x_discrete(limits = c("Prof", "AsstProf", "AssocProf"))
```



breaks,
limits,
labels

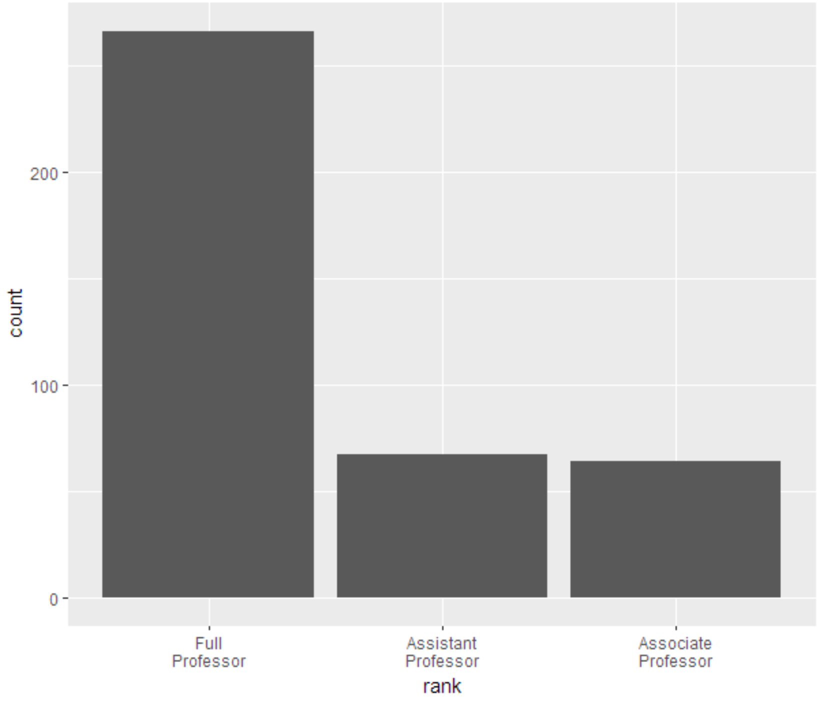
use limits
to reorder
levels

Scales

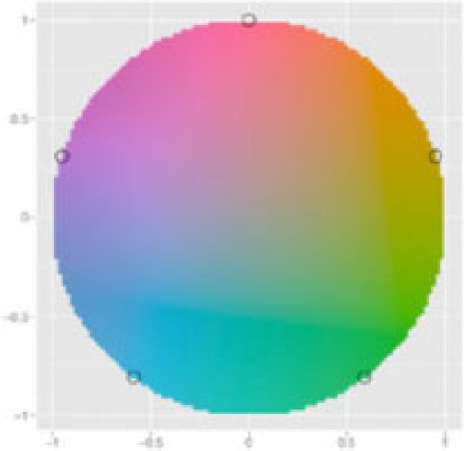
```
ggplot(Salaries, aes(x=rank)) + geom_bar() +
  scale_x_discrete(limits = c("Prof", "AsstProf", "AssocProf"),
     labels = c("Full\nProfessor", "Assistant\nProfessor" ,
                "Associate\nProfessor"))
```

breaks,
limits,
labels

use limits
to reorder
levels



Scales – Color and Fill



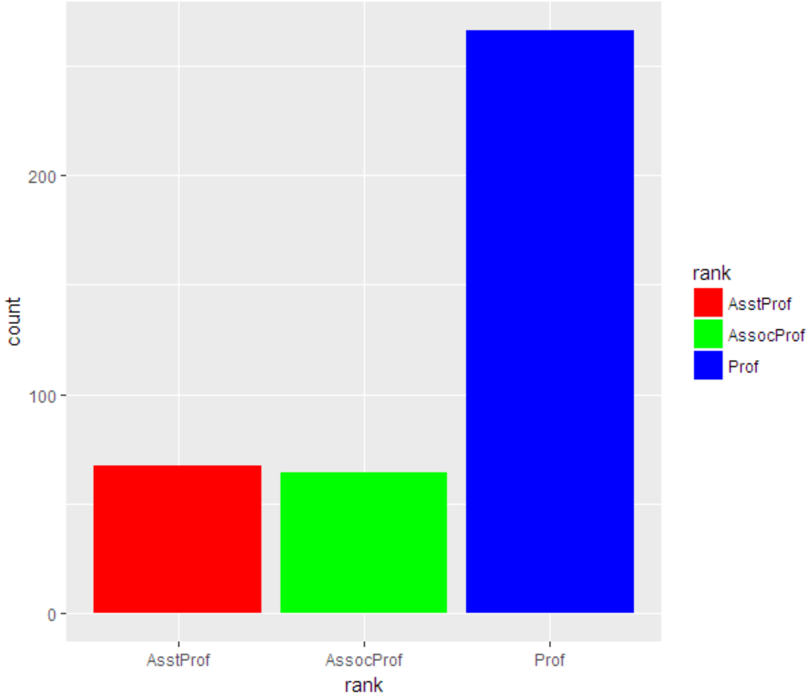
picking colors by name - "red"
or hex #ff0000

try colors() to list all built in colors

ggplot2 picks colors from around the circle
for example the 5 points above if there are five levels

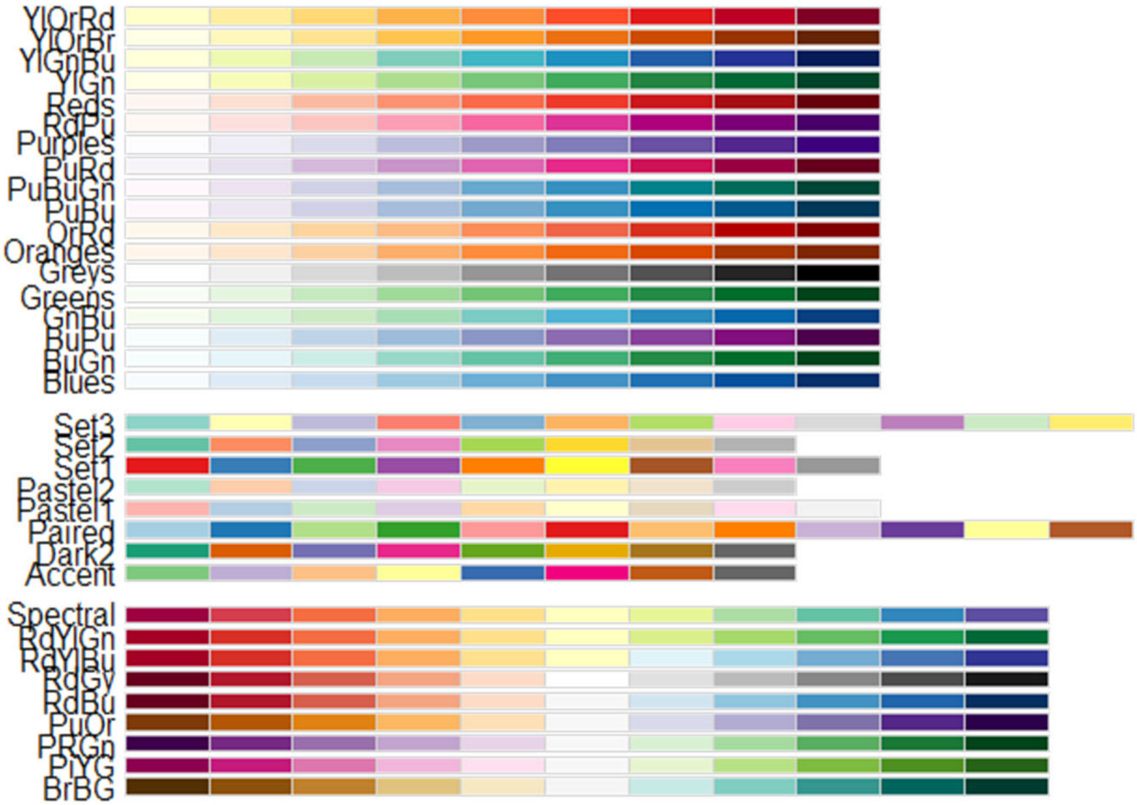
Scales – Color/Fill

```
ggplot(Salaries, aes(x=rank, fill=rank)) + geom_bar() +
  scale_fill_manual(values=c("red", "green", "blue"))
```



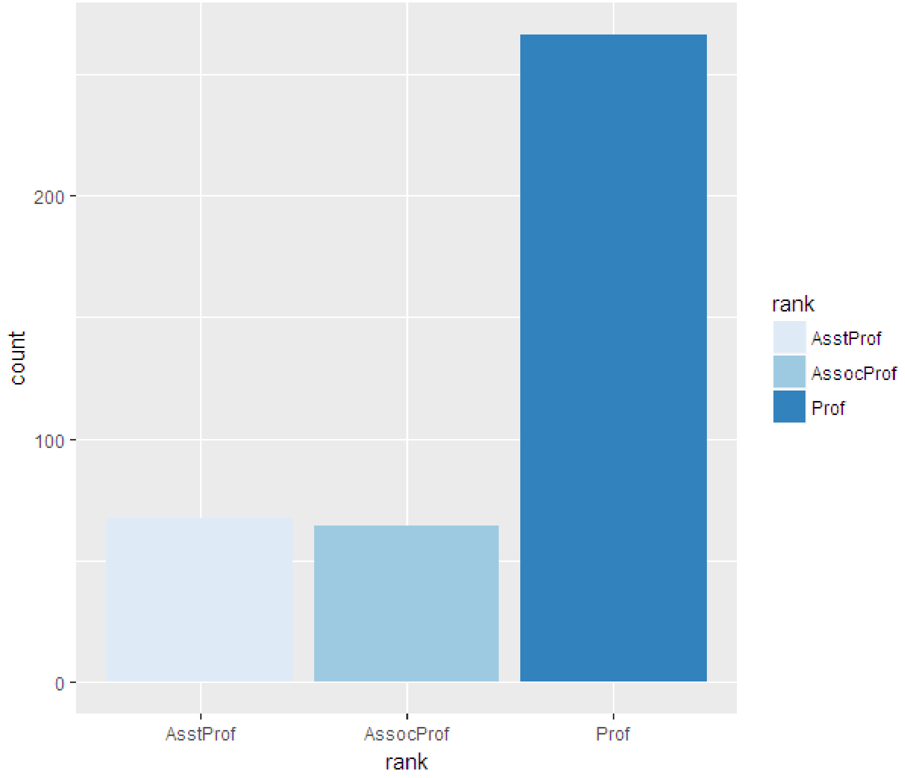
Scales – Color / Fill

Specify a color palate using
`scale_fill_brewer()`
`scale_color_brewer()`
using `palette=` option



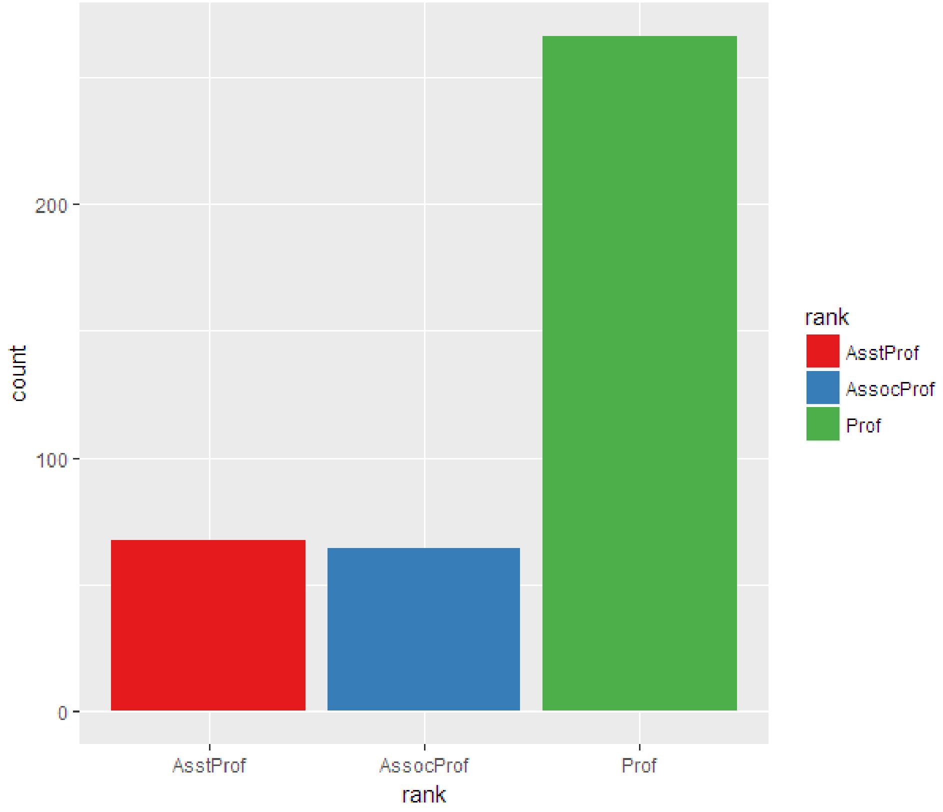
Scales – Color/Fill

```
ggplot(Salaries, aes(x=rank, fill=rank)) +
  geom_bar() +
  scale_fill_brewer()
```



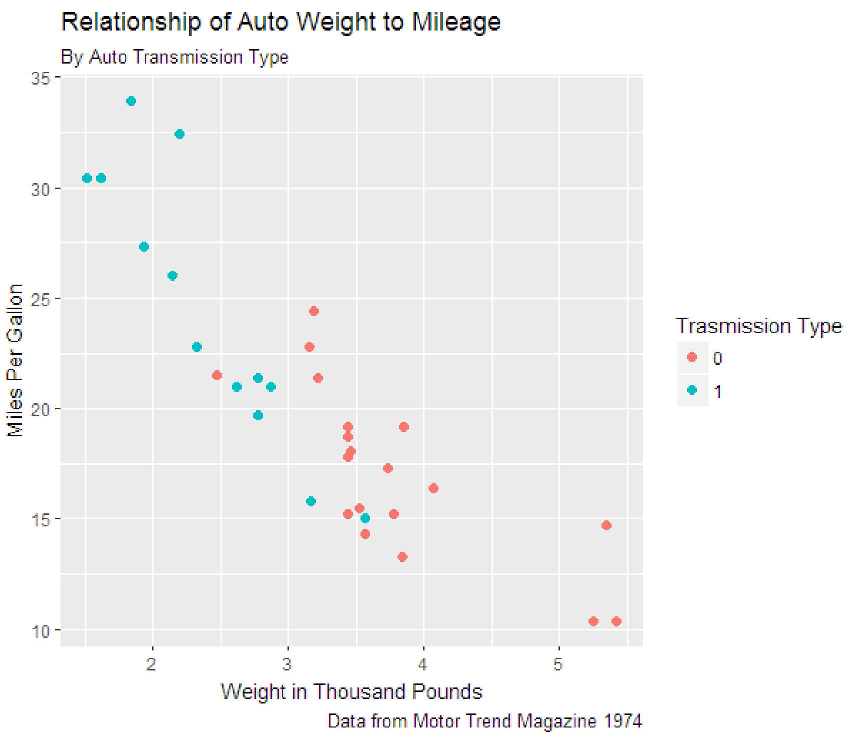
Scales – Color/Fill

```
ggplot(Salaries, aes(x=rank, fill=rank)) +
  geom_bar() +
  scale_fill_brewer(palette = "Set1")
```



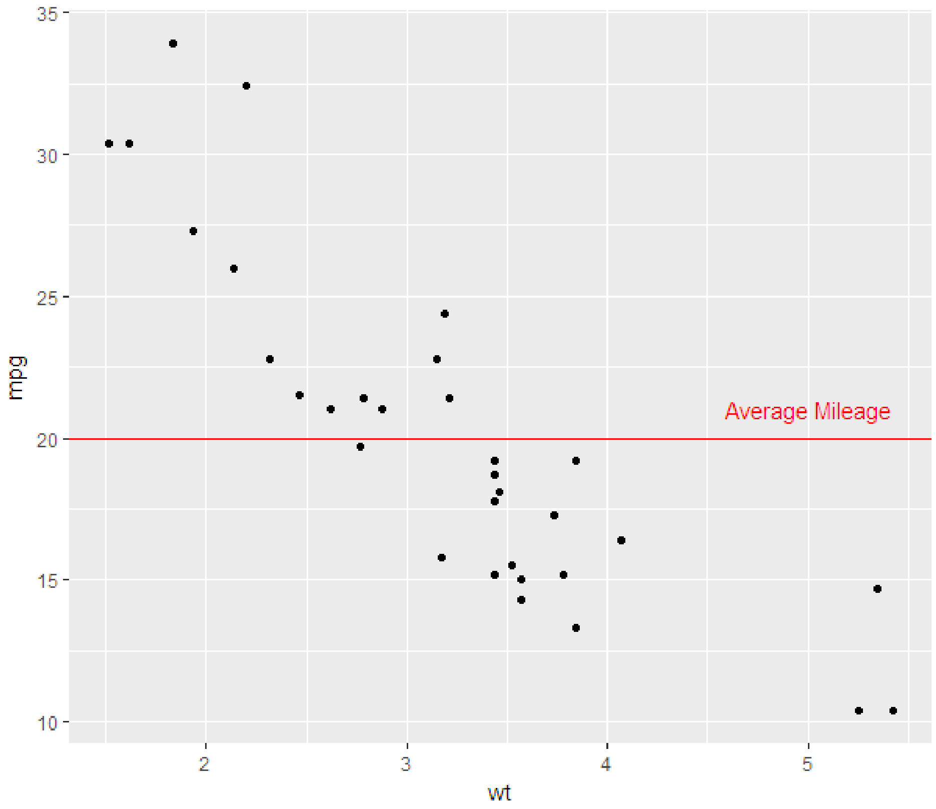
Annotations - Labels

```
p <- ggplot(data=mtcars, aes(x=wt, y=mpg, color=factor(am))) +
     geom_point(size=2) +
     labs(title="Relationship of Auto Weight to Mileage"
       subtitle="By Auto Transmission Type",
       caption = "Data from Motor Trend Magazine 1974",
       x = "Weight in Thousand Pounds",
       y="Miles Per Gallon",
       color = "Transmission Type")

p
```



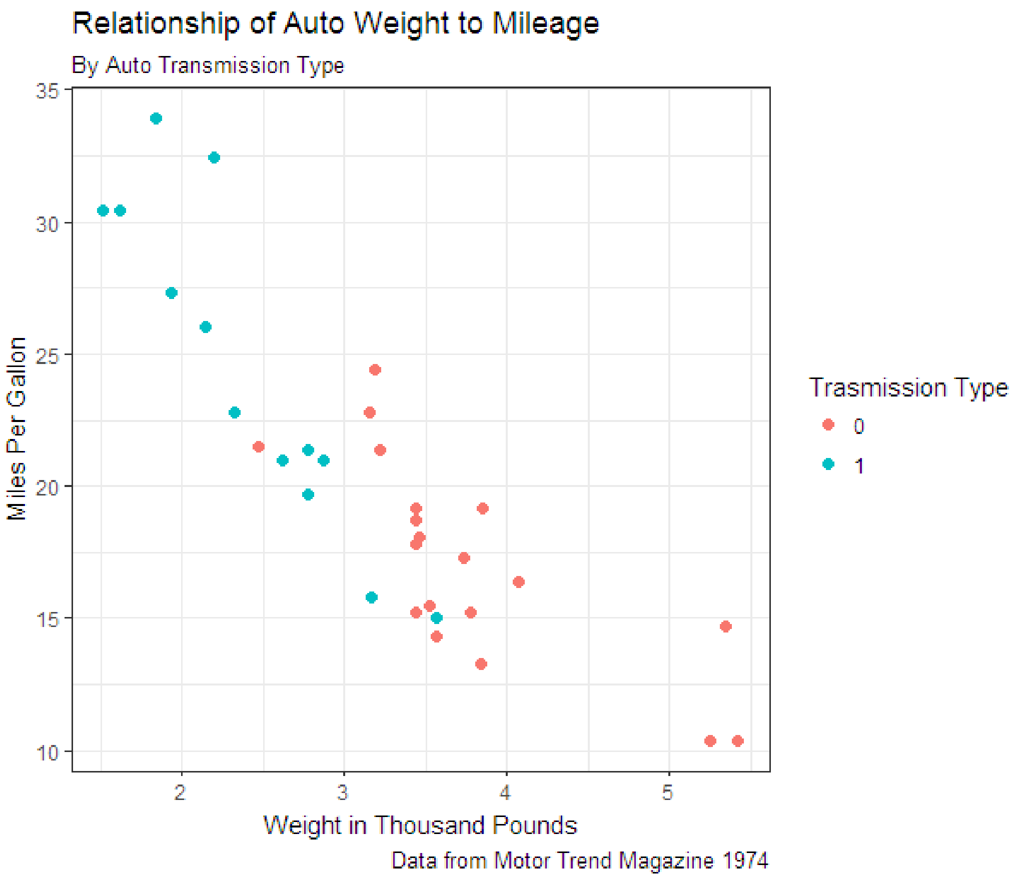
Annotations – reference lines and labels

```
ggplot(data=mtcars, aes(x=wt, y=mpg)) + geom_point() +
  geom_hline(yintercept=20, color="red") +
  annotate("text", x=5, y=21,
             label="Average Mileage", color="red")
```



Themes - prepackaged

```
p + theme_bw()
```



Themes - prepackaged

```
library(ggthemes)
```

theme_base: a theme resembling the default base graphics in R. See also theme_par.

theme_calc: a theme based on LibreOffice Calc.

theme_economist: a theme based on the plots in the The Economist magazine.

theme_excel: a theme replicating the classic ugly gray charts in Excel

theme_few: theme from Stephen Few's "Practical Rules for Using Color in Charts".

theme_fivethirtyeight: a theme based on the plots at fivethirtyeight.com.

theme_gdocs: a theme based on Google Docs.

theme_hc: a theme based on Highcharts JS.

theme_par: a theme that uses the current values of the base graphics parameters in par.

theme_pander: a theme to use with the pander package.

theme_solarized: a theme using the solarized color palette.

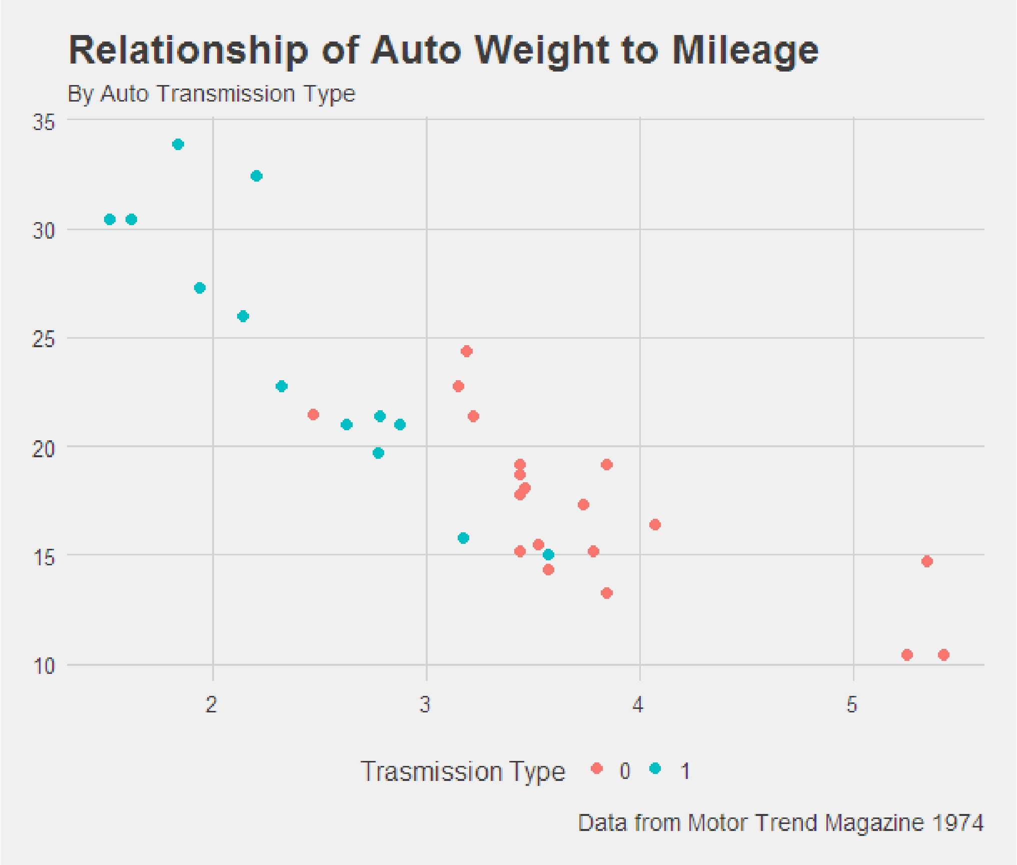
theme_stata: themes based on Stata graph schemes.

theme_tufte: a minimal ink theme based on Tufte's The Visual Display of Quantitative Information.

theme_wsj: a theme based on the plots in the The Wall Street Journal.

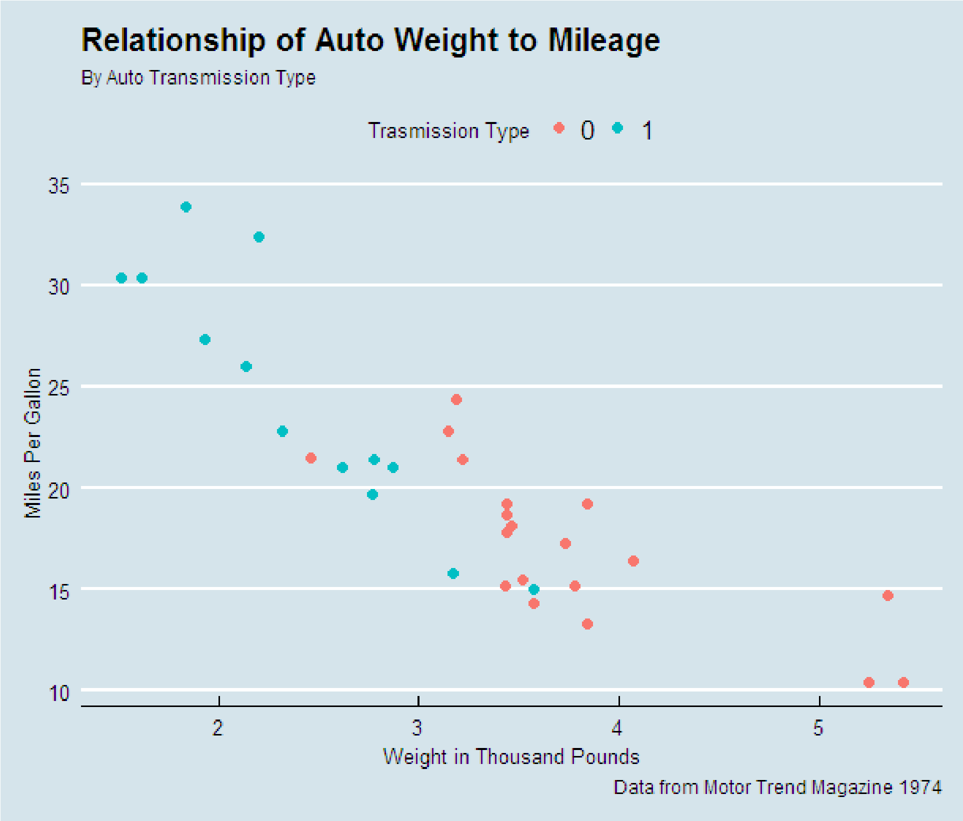
Themes - prepackaged

```
library(ggthemes)
p + theme_fivethirtyeight()
```

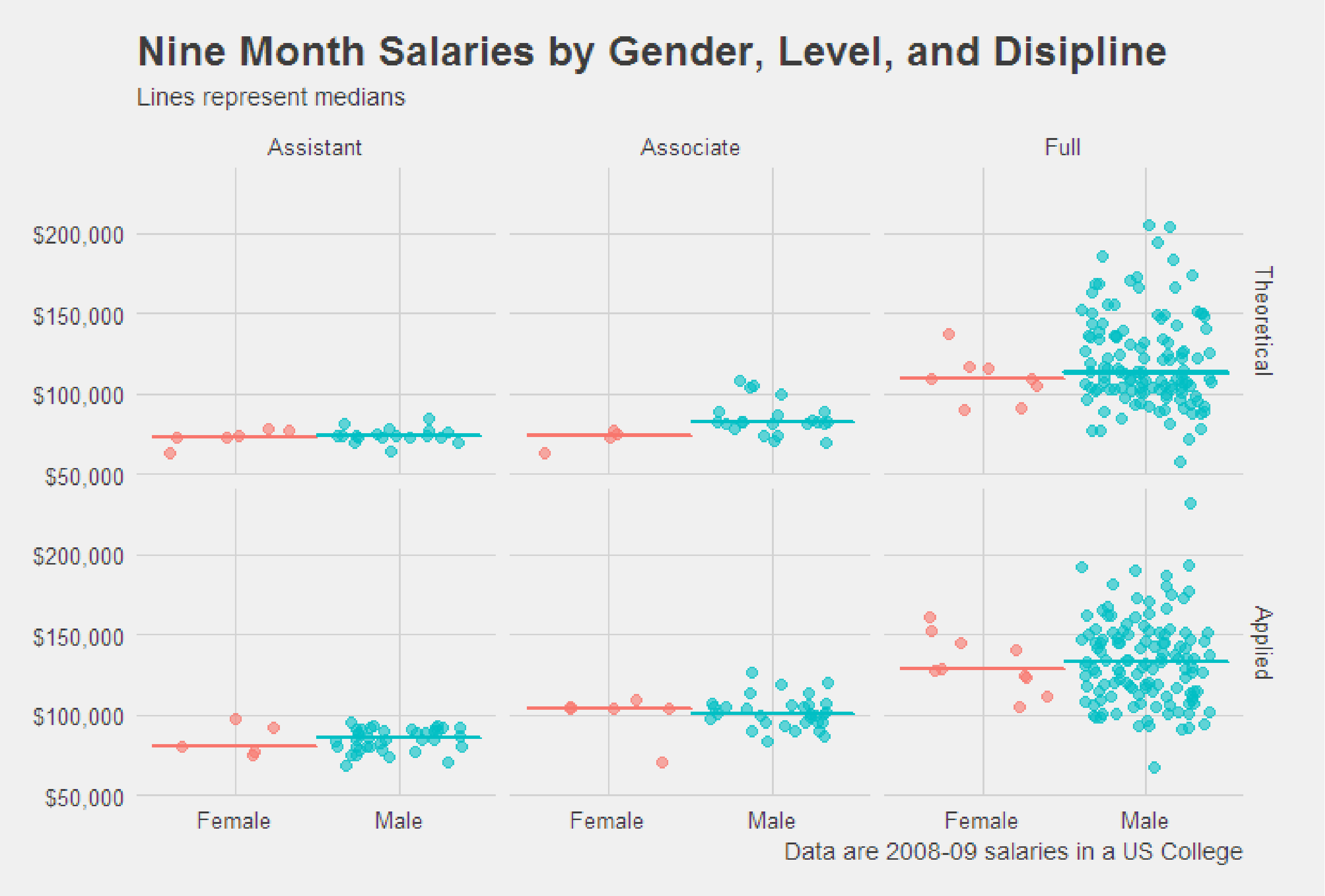


Themes - prepackaged

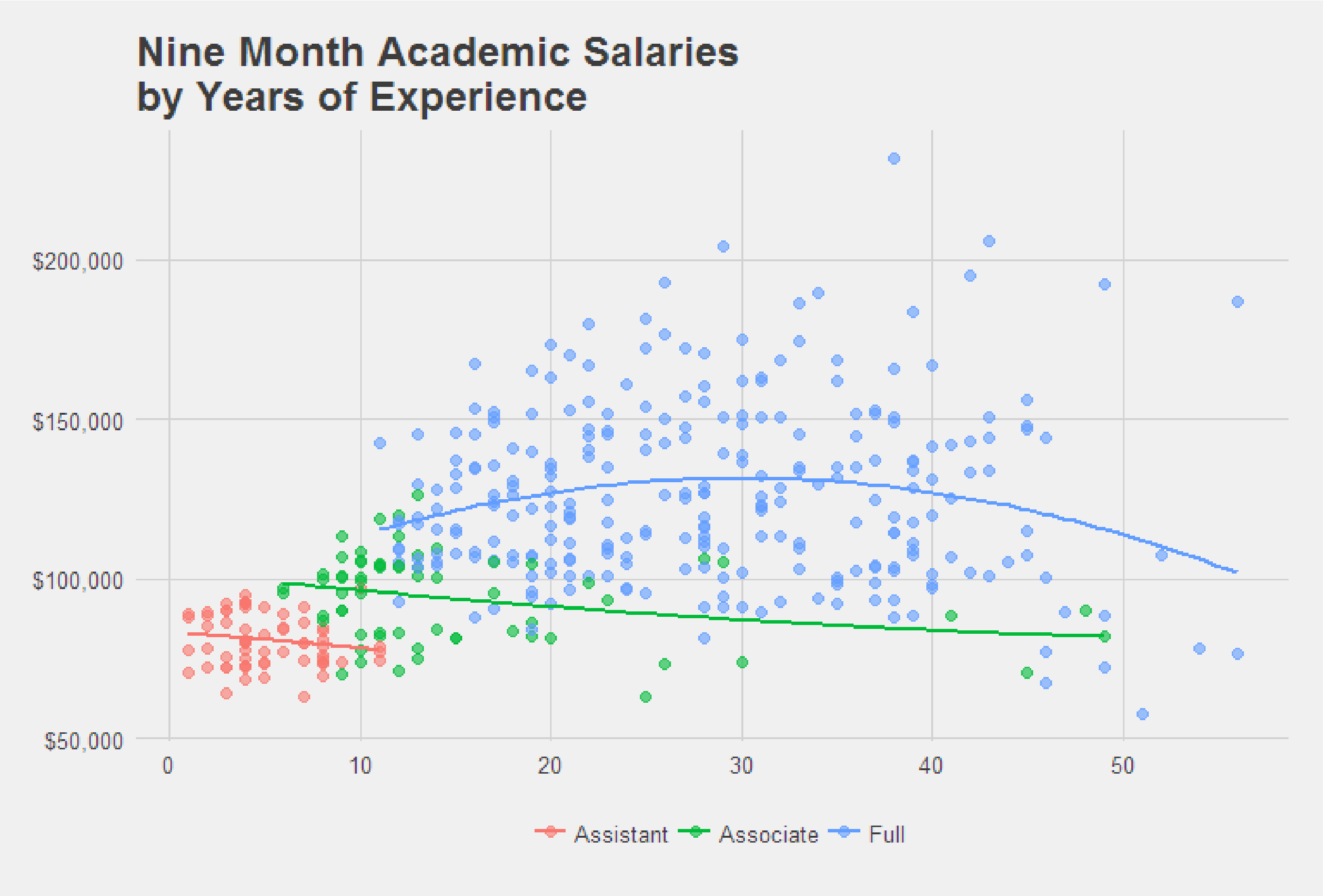
```
library(ggthemes)
p + theme_economist()
```



Getting Fancy

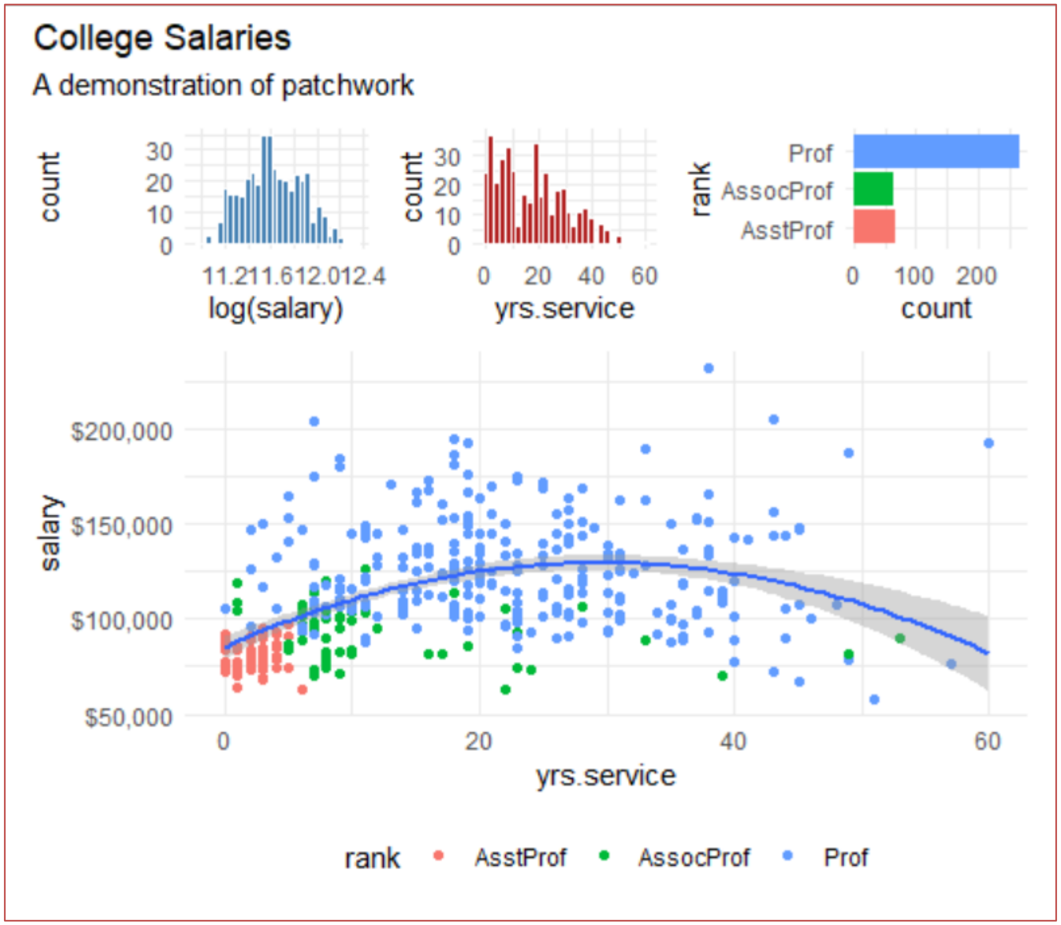


Getting Fancy



Combining graphs

```
theme_set(theme_minimal())

p1 <- ggplot(Salaries, aes(x=log(salary))) +
  geom_histogram(fill="steelblue", color="white")

p2 <- ggplot(Salaries, aes(x=yrs.service)) +
  geom_histogram(fill="firebrick", color="white")

p3 <- ggplot(Salaries, aes(x=rank, fill=rank)) +
  geom_bar() + theme(legend.position="none") +
  labs(x="") + coord_flip()

p4 <- ggplot(Salaries, aes(x=yrs.service, y=salary)) +
  geom_point(aes(color=rank)) +
  geom_smooth(method="lm", formula=y~poly(x, 2)) +
  scale_y_continuous(labels=scales::dollar_format()) +
  theme(legend.position="bottom")

library(patchwork)
(p1 + p2 + p3)/ p4 +
  plot_layout(heights=c(1,3)) +
  plot_annotation(title="College Salaries",
              subtitle="A demonstration of patchwork")
```



Saving your work

- `ggsave(filename="filename.ext", plot=p)`
  - ext can be
    `eps, ps, tex, pdf, jpeg, tiff, png, bmp, svg, wmf`
  - plot defaults to last one created
  - wmf on windows platforms only
  - svg can be edited using Inkscape or illustrator
  - be careful of transparencies with Microsoft Office (use bitmap not wmf)

Creating interactive graphs

```
library(ggplot2)
data(mtcars)
mtcars$auto <- row.names(mtcars)
p <- ggplot(data=mtcars, aes(x=wt, y=mpg, text=auto)) +
  geom_point()

library(plotly)
ggplotly(p, tooltip=c("x", "y", "text"))
```

Learning more

- R in Action (3rd ed) https://www.manning.com/books/r-in-action-third-edition
- Data Visualization with R - http://rkabacoff.github.io/datavis
- Hadley Wickham – http://docs.ggplot2.org/
- Winston Chang- http://wiki.stdout.org/rcookbook/Graphs/