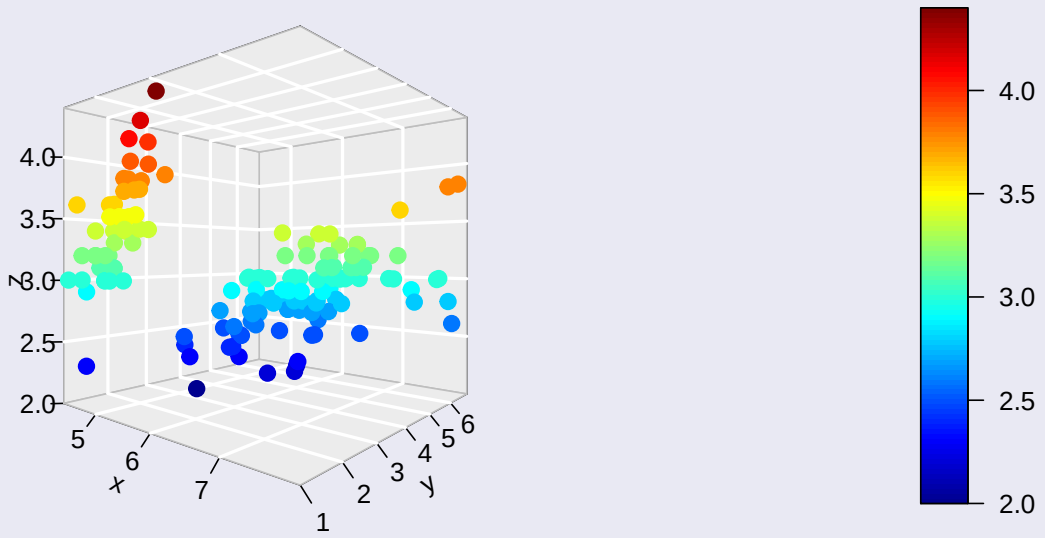
R language and data analysis: ggplot2

Qiang Shen

Jan 16, 2018

Visualization

- A picture is worth a thousand words



data visualization

- base package
- plot for summary statistics

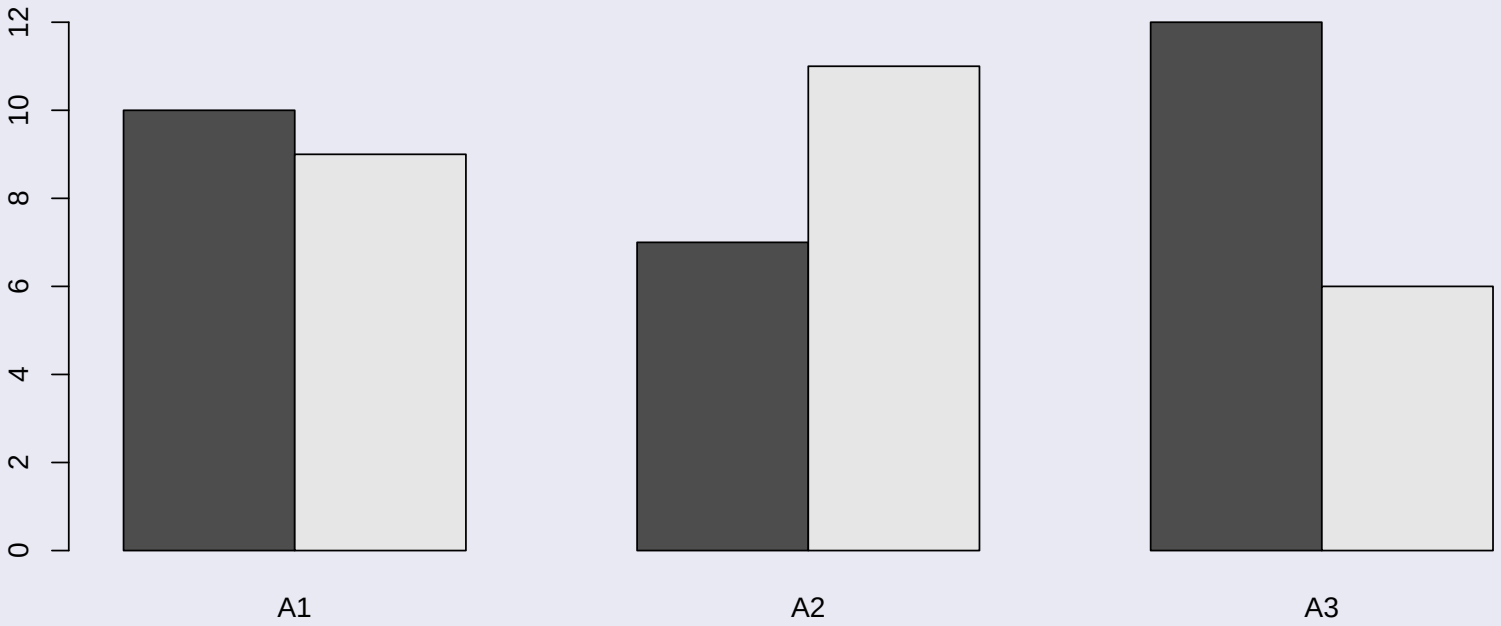
motivation

```
library(gcookbook)
simpledat
```

```
##    A1 A2 A3
## B1 10  7 12
## B2  9 11  6
```

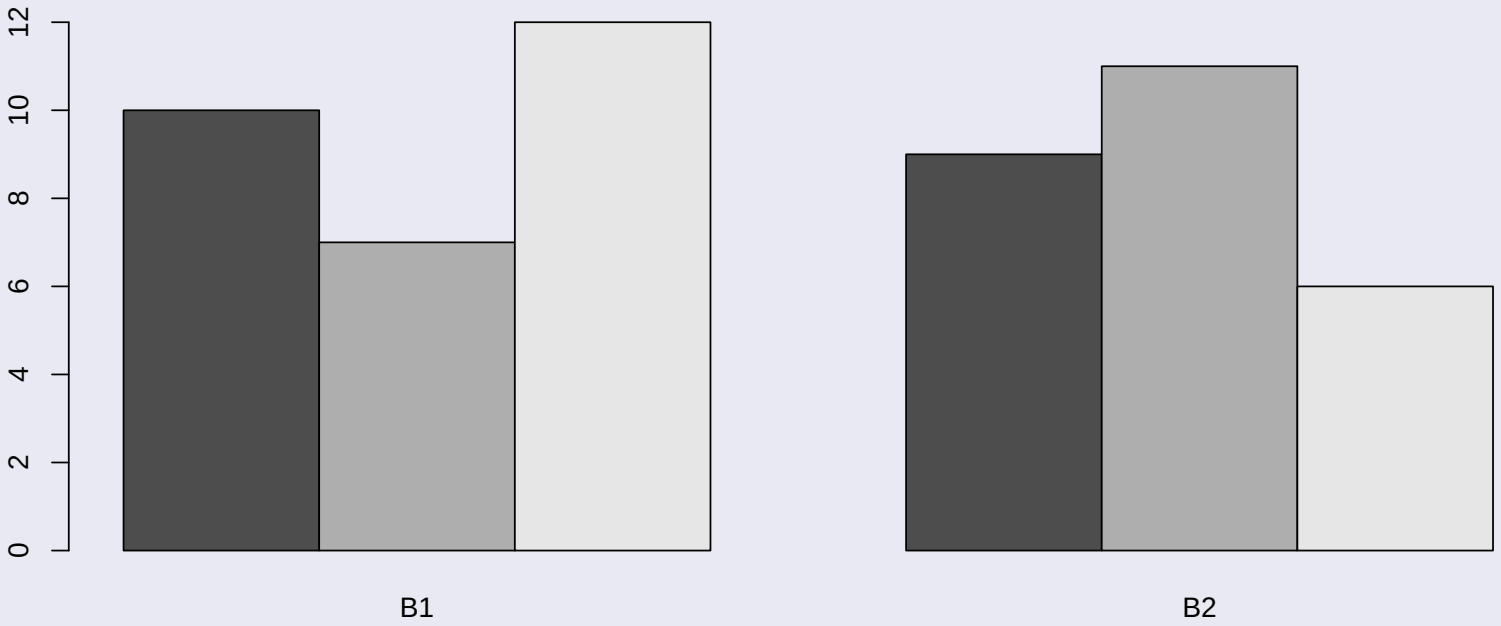
motivation

```
library(gcookbook)
barplot(simpledat,beside=T)
```



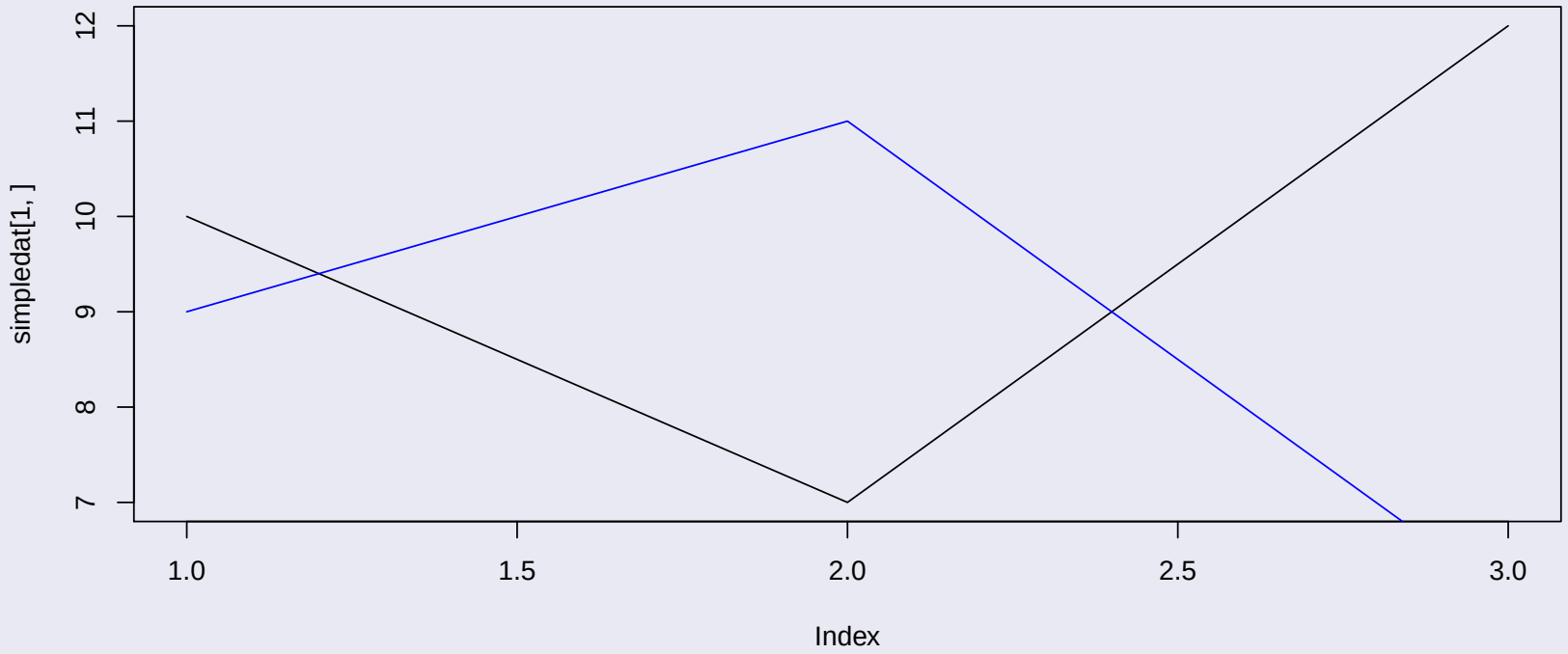
motivation

```
library(gcookbook)
barplot(t(simpledat),beside=T)
```



motivation

```
##     A1 A2 A3
## B1 10  7 12
## B2  9 11  6
```



motivation

```
library(gcookbook)
simpledat
barplot(simpledat,beside=T)
barplot(t(simpledat),beside=T)
plot(simpledat[1,],type='l')
lines(simpledat[2,],type='l',col='blue')
```

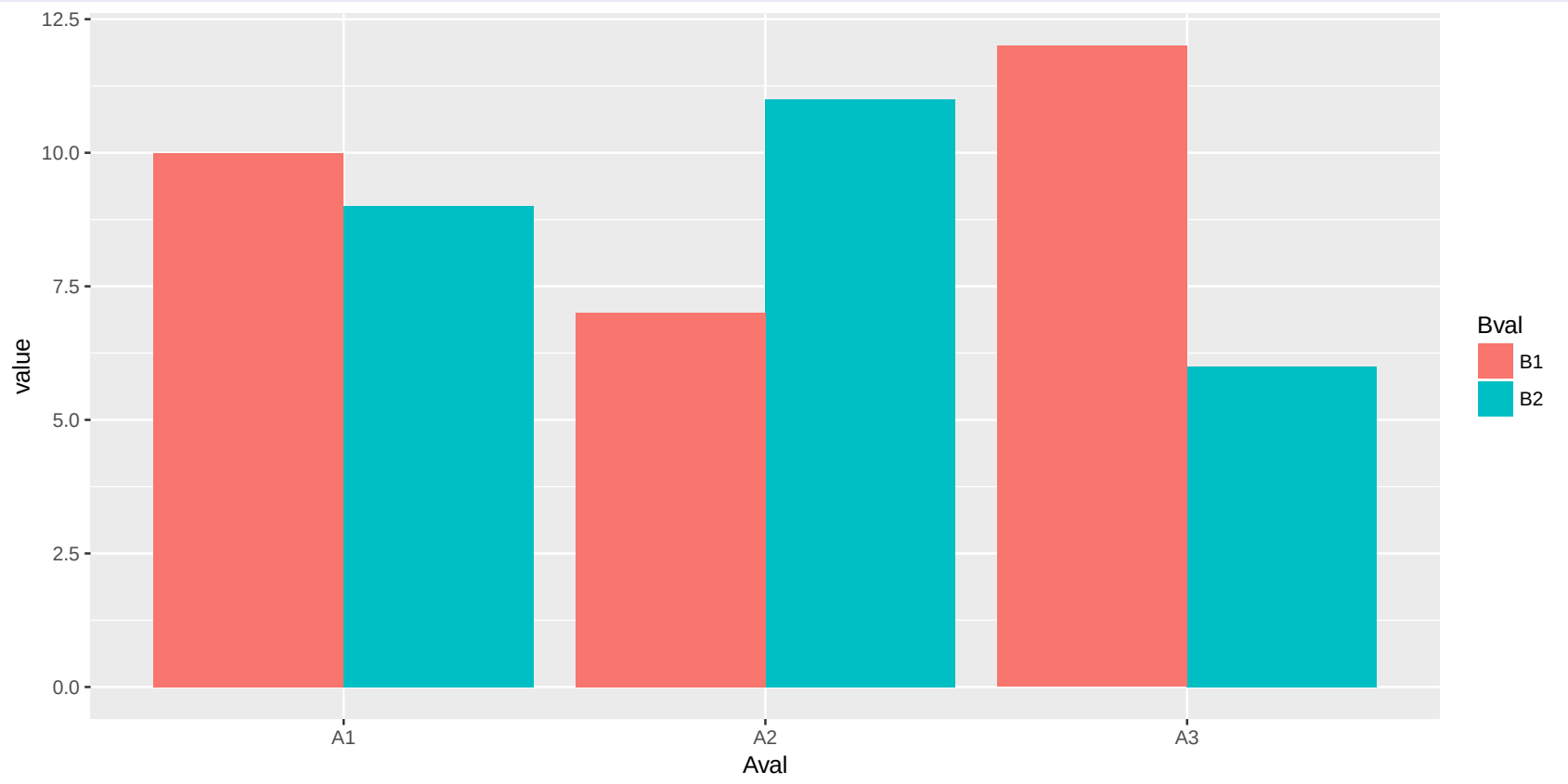
How is the case in ggplot2

```
library(ggplot2)
simpledat_long
```

```
##   Aval Bval value
## 1   A1   B1    10
## 2   A1   B2     9
## 3   A2   B1     7
## 4   A2   B2    11
## 5   A3   B1    12
## 6   A3   B2     6
```

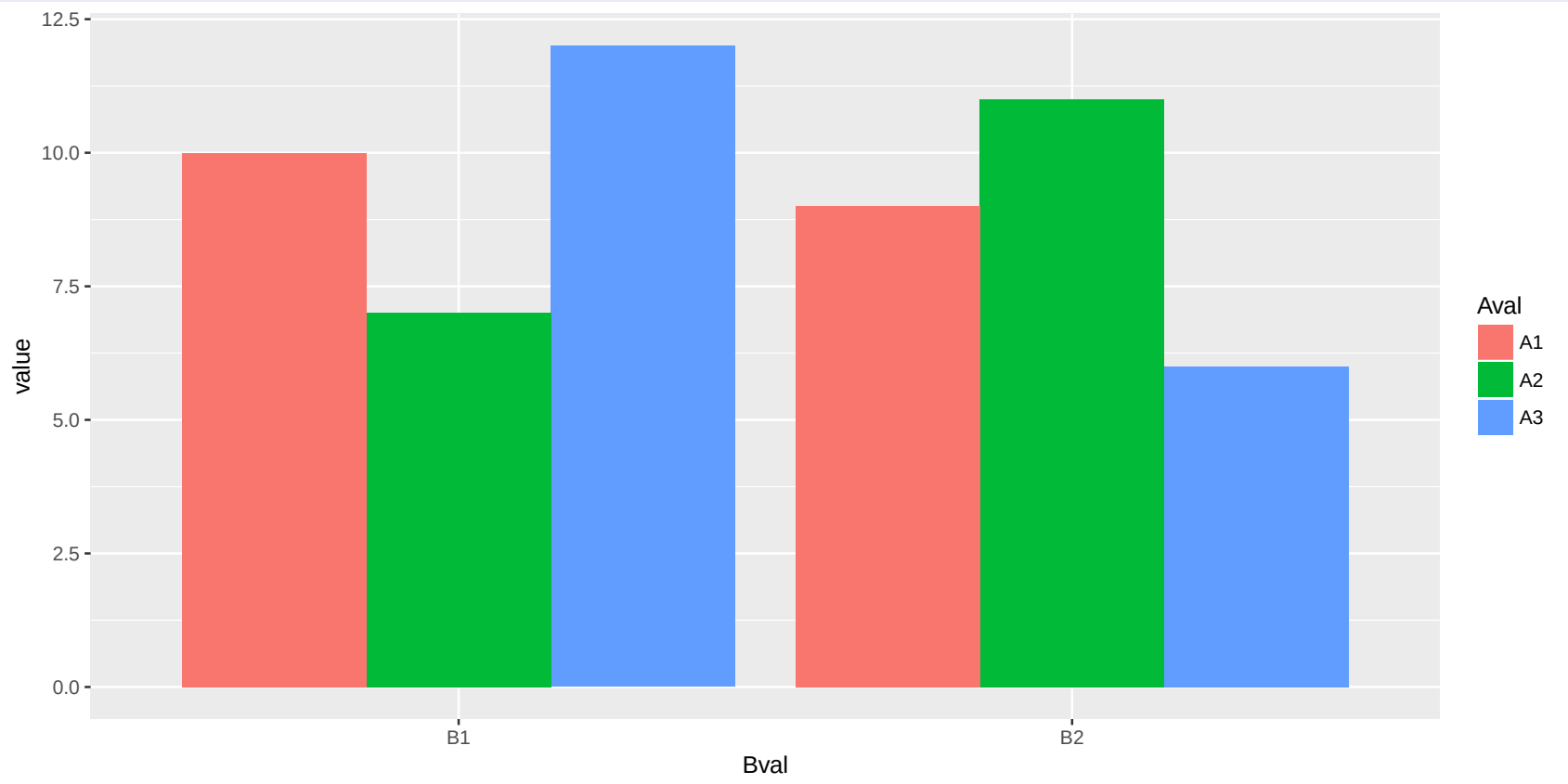
ggplot2

```
ggplot(simpledat_long,aes(x=Aval,y=value,fill=Bval))+
  geom_bar(stat='identity',position='dodge')
```



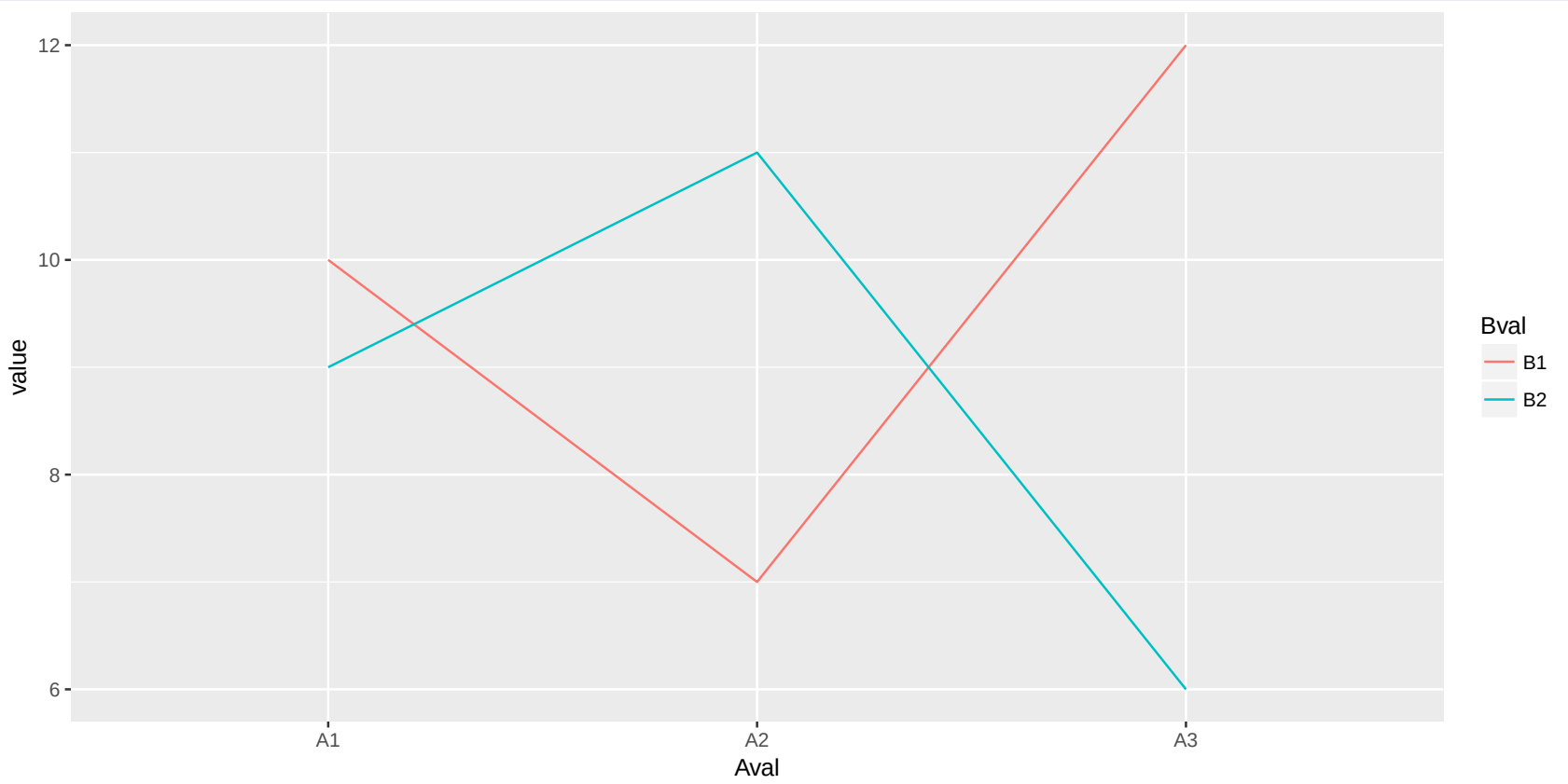
ggplot2

```
ggplot(simpledat_long,aes(x=Bval,y=value,fill=Aval))+
  geom_bar(stat='identity',position='dodge')
```



ggplot2

```
ggplot(simpledat_long,aes(x=Aval,y=value,colour=Bval,
                          group=Bval))+geom_line()
```



How is the case in ggplot2

```
library(ggplot2)
simpledat_long
ggplot(simpledat_long,aes(x=Aval,y=value,fill=Bval))+
  geom_bar(stat='identity',position='dodge')
ggplot(simpledat_long,aes(x=Bval,y=value,fill=Aval))+
  geom_bar(stat='identity',position='dodge')
ggplot(simpledat_long,aes(x=Aval,y=value,colour=Bval,
                          group=Bval))+geom_line()
```

four graphics system in R.

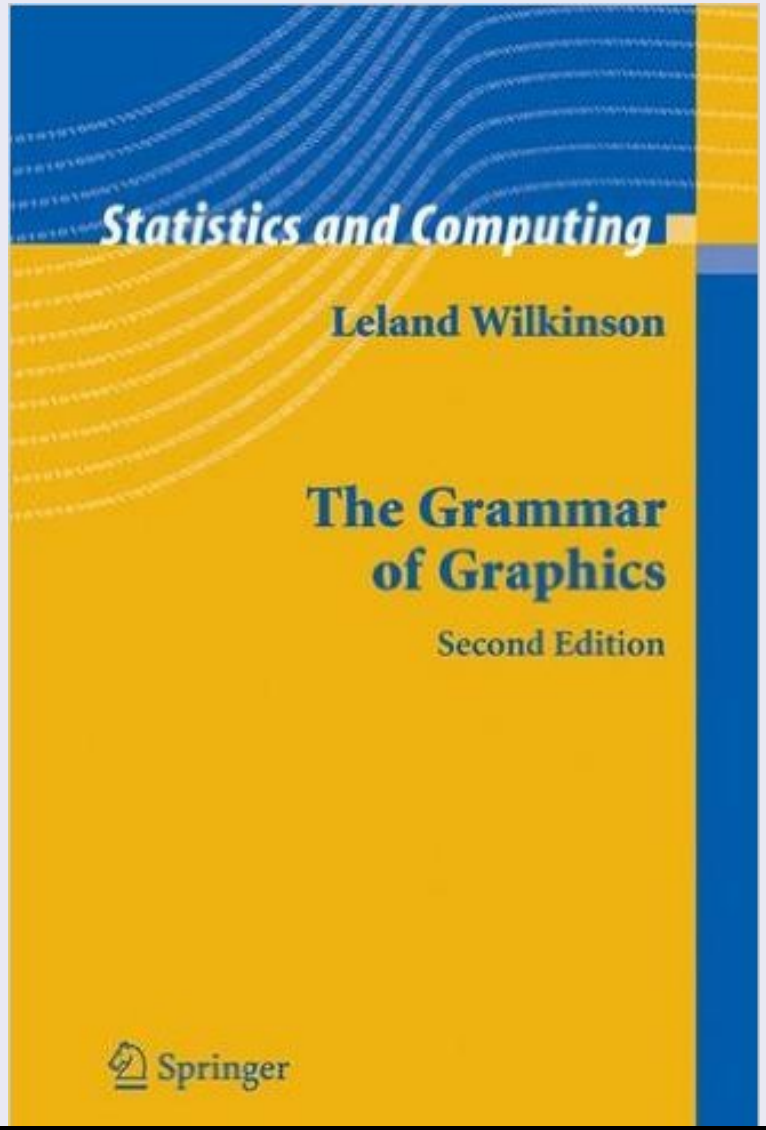
System	Included in base installation?	Must be explicitly loaded?
`base`	Yes	No
`grid`	Yes	Yes
`lattice`	Yes	Yes
`ggplot2`	No	Yes

package requirement for the chapter

- ggplot2
- car
- gridExtra
- ggthemes

ggplot2

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.



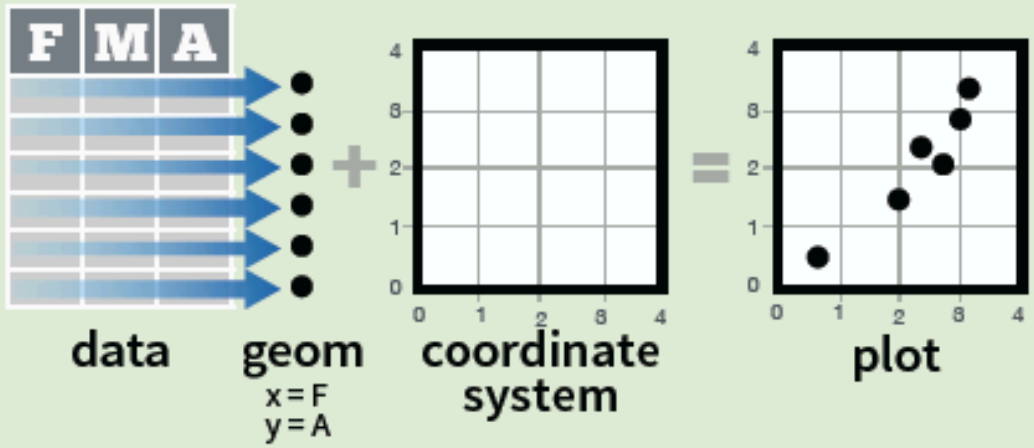
ggplot2

Plot = data + Aesthetics + Geometry.

- data is a data frame
- Aesthetics (aes) is used to indicate x and y variables. It can also be used to control the color, the size or the shape of a point...
- Geometry (geom) corresponds to the type of graphics (histogram, box plot, line plot,density plot, dot plot)

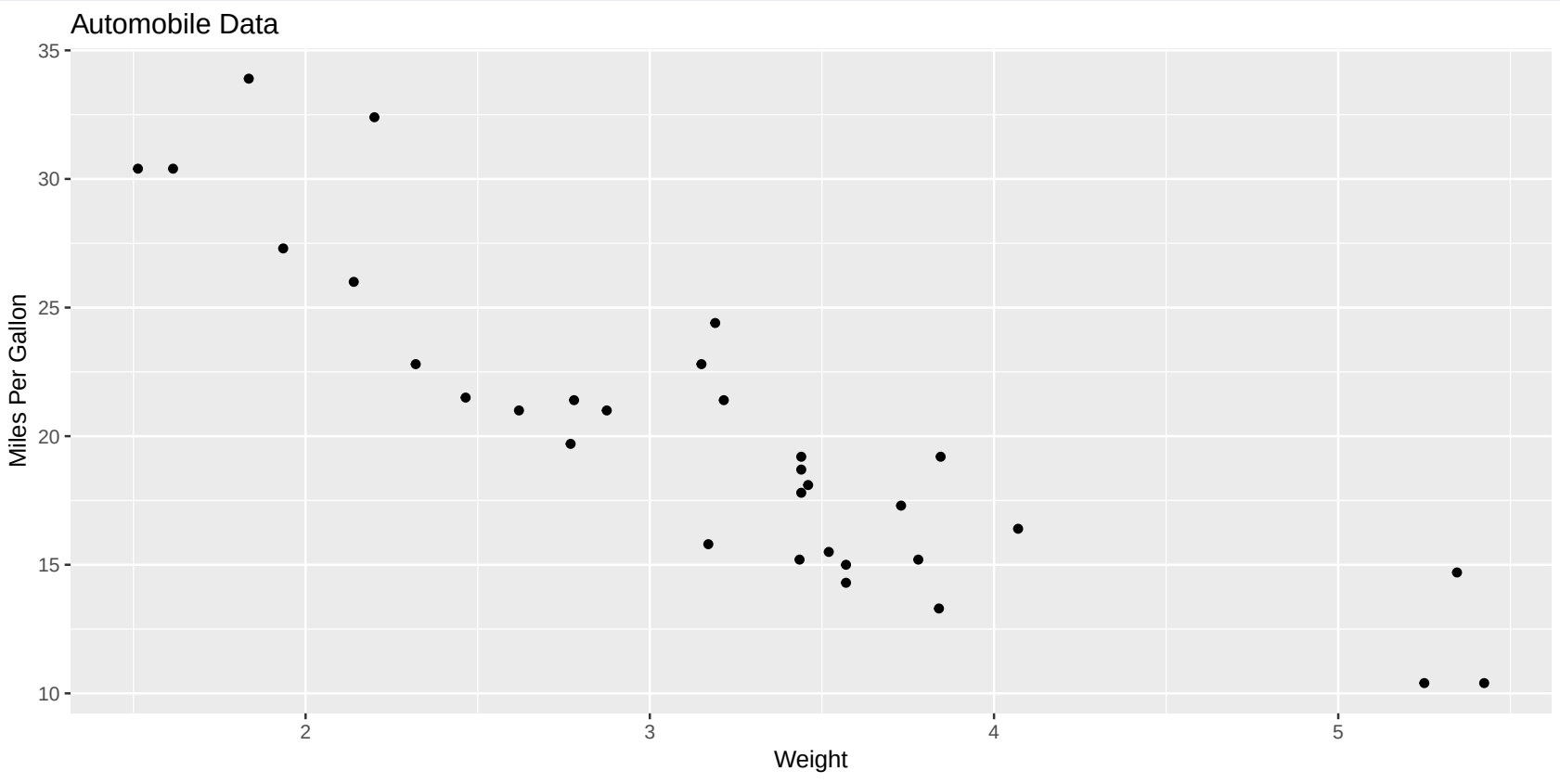
ggplot2

ggplot2 is based on the **grammar of graphics**, the idea that you can build every graph from the same few components: a **data** set, a set of **geoms**—visual marks that represent data points, and a **coordinate system.**



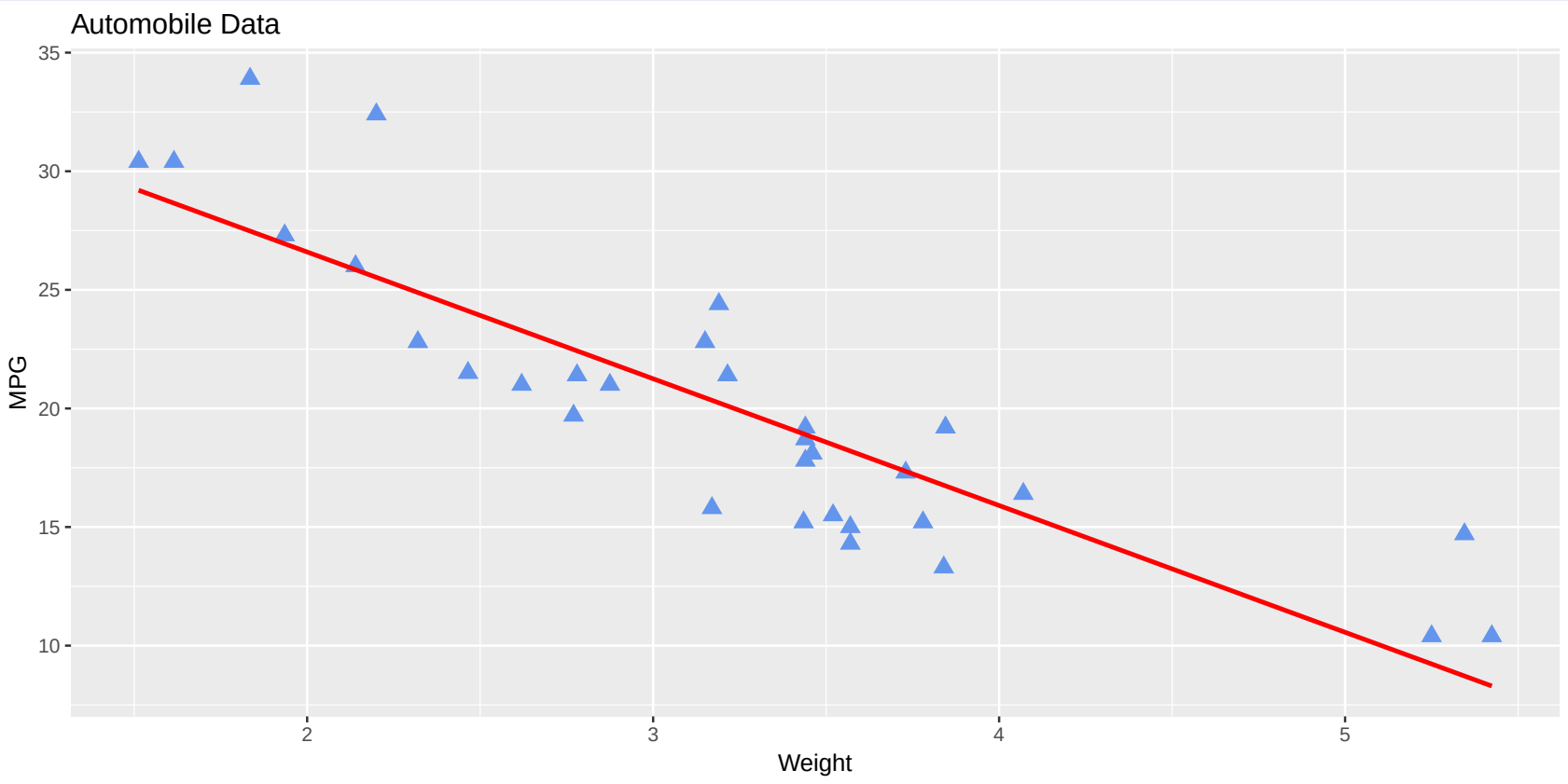
Basic scatterplot

```
library(ggplot2)
# with(mtcars,plot(wt,mpg))
ggplot(data=mtcars, aes(x=wt, y=mpg))+
  geom_point()+labs(title="Automobile Data",
                    x="Weight", y="Miles Per Gallon")
```



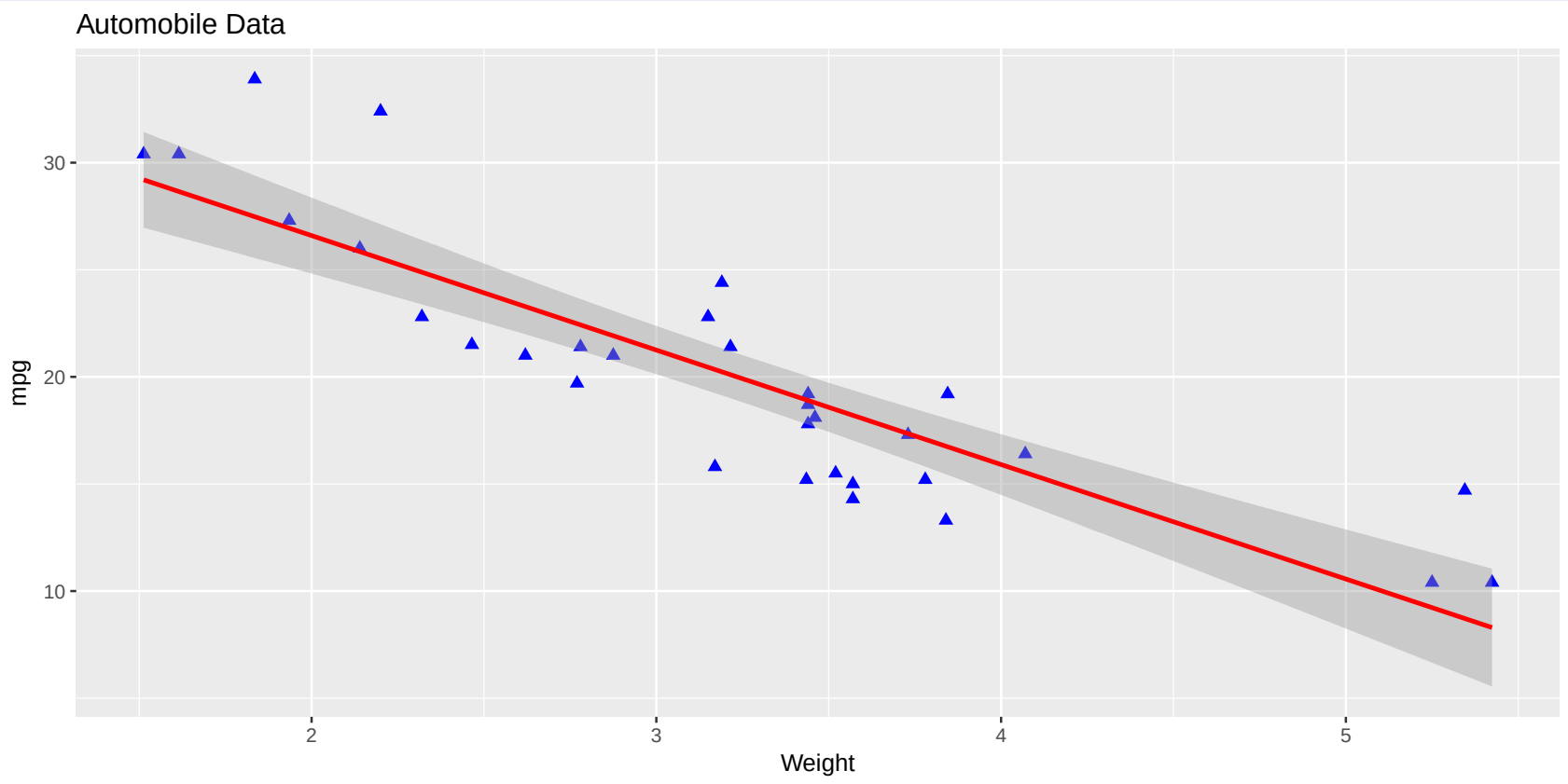
Scatter plot with additional options

```
ggplot(data=mtcars, aes(x=wt, y=mpg)) +
  geom_point(pch=17,color="cornflowerblue",size=3)+
  geom_smooth(method="lm",color="red",se=F,linetype=1)+
labs(title="Automobile Data", x="Weight", y="MPG")
```



ggplot2

```
p<-ggplot(data=mtcars, aes(x=wt, y=mpg))
p+ geom_point(pch=17, color="blue", size=2)+
geom_smooth(method="lm",color="red",linetype=1)+
labs(title="Automobile Data",x="Weight",y="mpg")
```

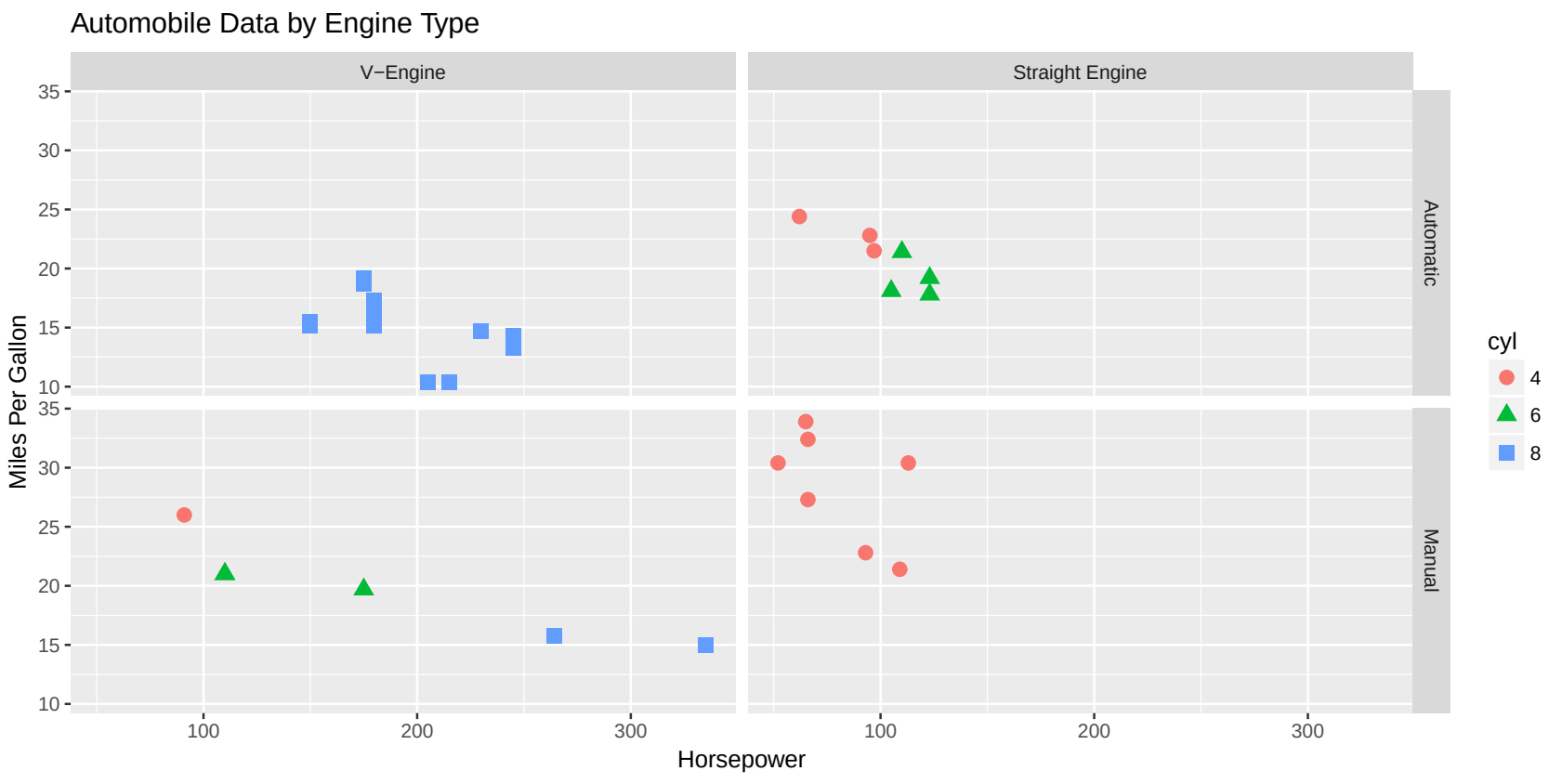


Scatter plot with grouping and faceting

- grouping (display groups in a single plot)
- faceting (display observations in separate,side-by-side plots)
- need to be factors

```
data(mtcars)
mtcars$am <- factor(mtcars$am, levels=c(0,1),
                    labels=c("Automatic", "Manual"))
mtcars$vs <- factor(mtcars$vs, levels=c(0,1),
                    labels=c("V-Engine", "Straight Engine")
mtcars$cyl <- factor(mtcars$cyl)
```

Scatter plot with grouping and faceting



Scatter plot with grouping and faceting

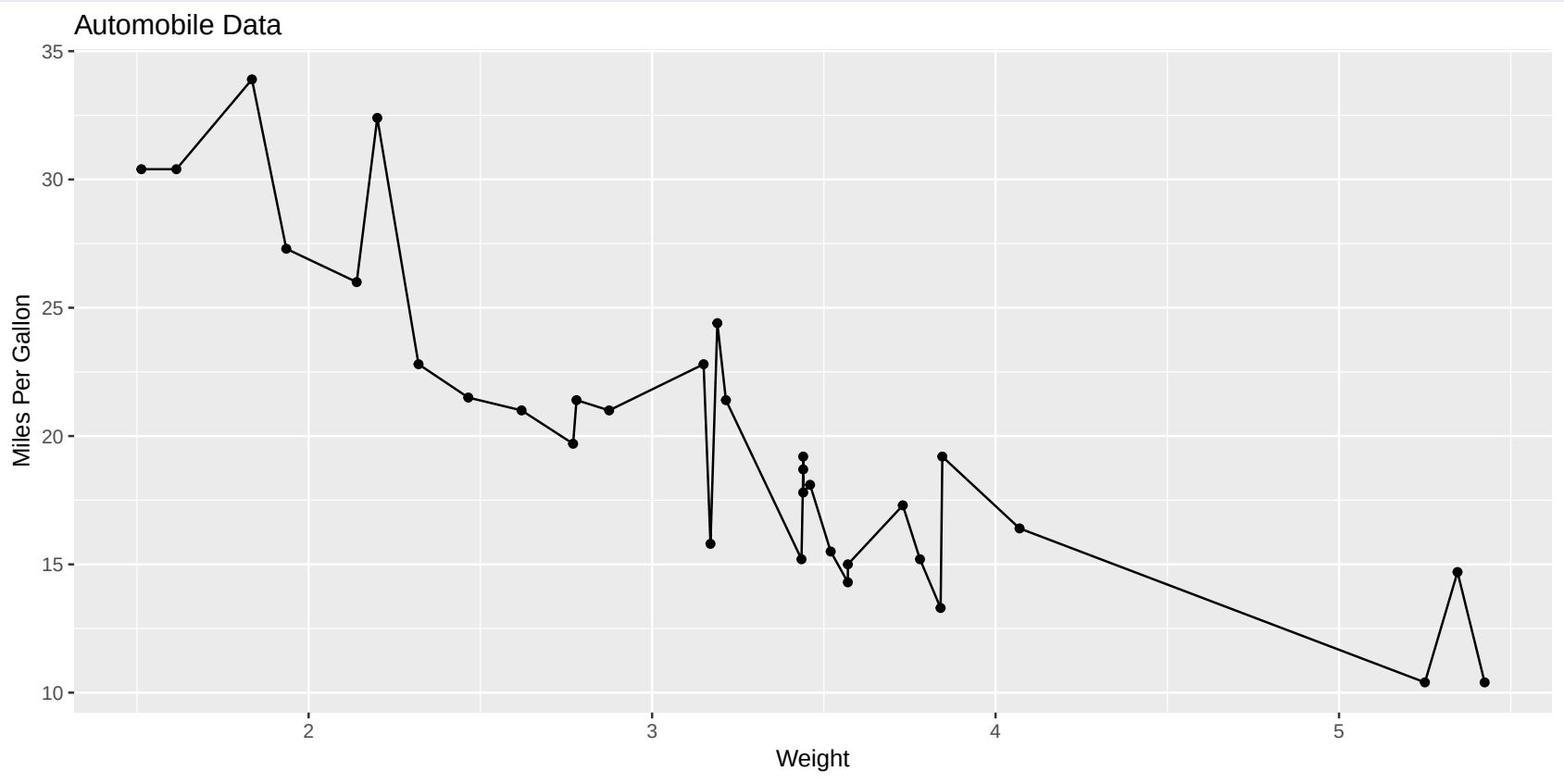
```
library(ggplot2)
ggplot(data=mtcars, aes(x=hp, y=mpg,
                        shape=cyl, color=cyl)) +
  geom_point(size=3) +
  facet_grid(am~vs) +
  labs(title="Automobile Data by Engine Type",
       x="Horsepower", y="Miles Per Gallon")
```

geometric objects (geom)

Function	Adds	Options
geom_bar()	Bar chart	color, fill, alpha
geom_boxplot()	Box plot	color, fill, alpha, notch, width
geom_density()	Density plot	color, fill, alpha, linetype
geom_histogram()	Histogram	color, fill, alpha, linetype, binwidth
geom_hline()	Horizontal lines	color, alpha, linetype, size
geom_jitter()	Jittered points	color, size, alpha, shape
geom_line()	Line graph	colorvalpha, linetype, size
geom_point()	Scatterplot	color, alpha, shape, size
geom_rug()	Rug plot	color, side
geom_smooth()	Fitted line	method, formula, color, fill, linetype, size
geom_text()	Text annotations	Many; see the help for this function
geom_violin()	Violin plot	color, fill, alpha, linetype
geom_vline()	Vertical lines	color, alpha, linetype, size

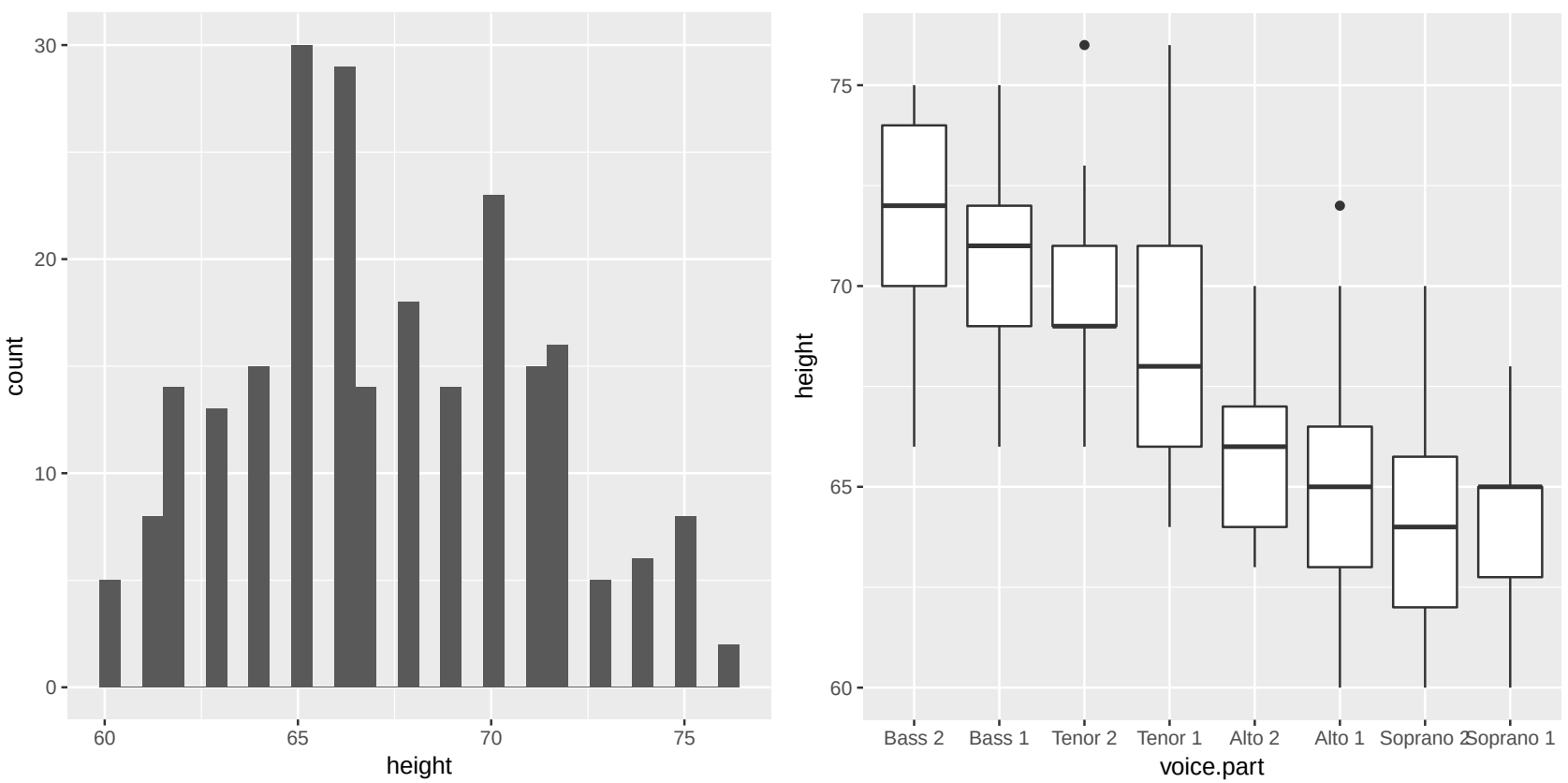
Using geoms: line

```
library(ggplot2)
ggplot(data=mtcars, aes(x=wt, y=mpg))+geom_line()+
  geom_point() + labs(title="Automobile Data",
                      x="Weight", y="Miles Per Gallon")
```



Using geoms:histogram,geom_boxplot

```
data(singer, package="lattice")
p1<-ggplot(singer, aes(x=height)) + geom_histogram()
p2<-ggplot(singer,aes(x=voice.part,y=height))+geom_boxplot(
library(gridExtra);grid.arrange(p1, p2, ncol=2)
```

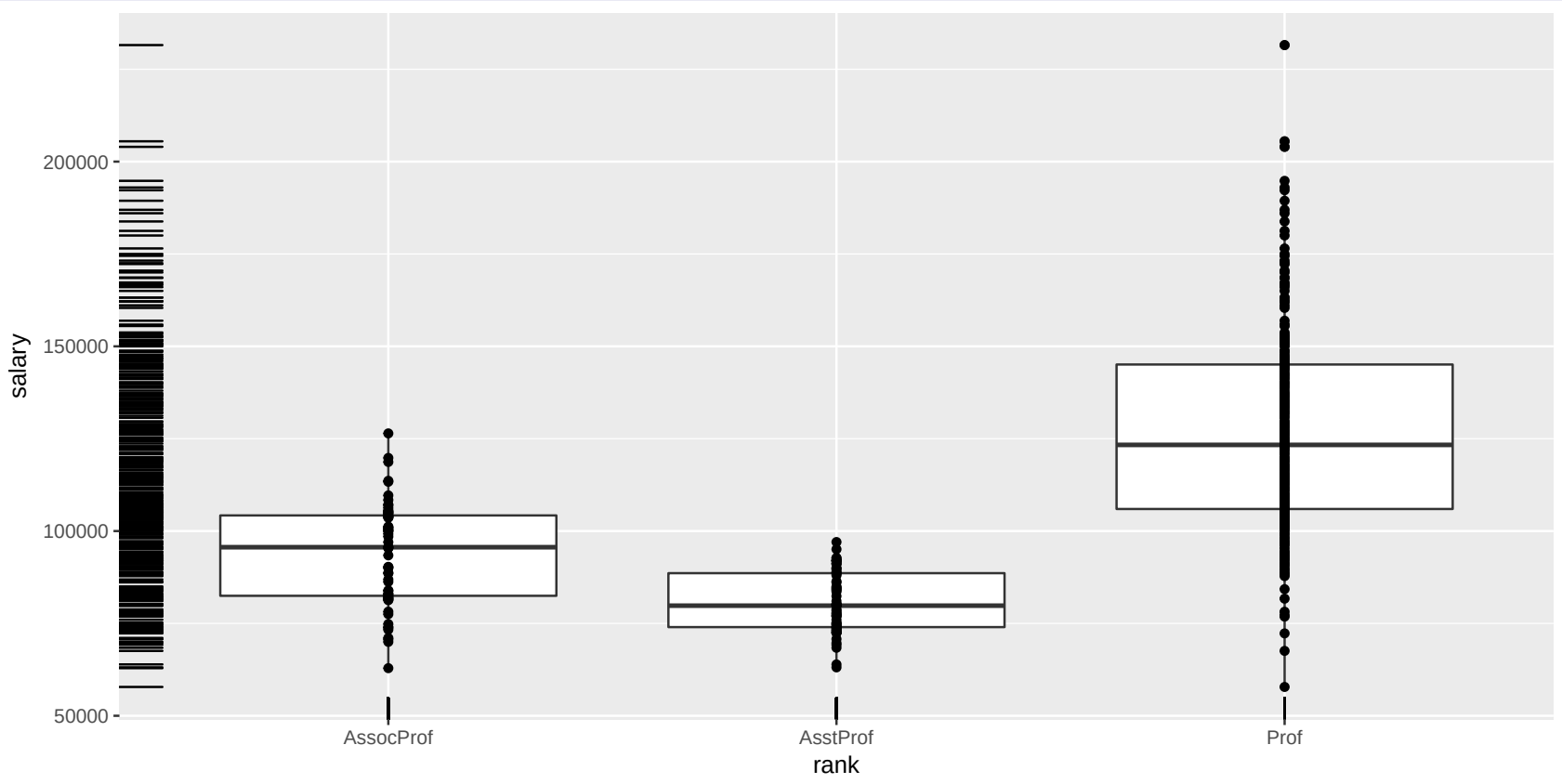


common options for geom functions

Option	Specifies
color	Color of points, lines, and borders around filled regions.
fill	Color of filled areas such as bars and density regions.
alpha	Transparency of colors, ranging from 0 (fully transparent) to 1 (opaque).
linetype	Pattern for lines (1 = solid, 2 = dashed, 3 = dotted, 4 = dotdash, 5 = longdash, 6 = twodash).
size	Point size and line width.
shape	Point shapes (same as pch, with 0 = open square, 1 = open circle, 2 = open triangle, and so on). See figure 3.4 for examples.
position	Position of plotted objects such as bars and points. For bars, "dodge" places grouped bar charts side by side, "stacked" vertically stacks grouped bar charts, and "fill" vertically stacks grouped bar charts and standardizes their heights to be equal. For points, "jitter" reduces point overlap.
binwidth	Bin width for histograms.
notch	Indicates whether box plots should be notched (TRUE/FALSE).
sides	Placement of rug plots on the graph ("b" = bottom, "l" = left, "t" = top, "r" = right, "bl" = both bottom and left, and so on).
width	Width of box plots.

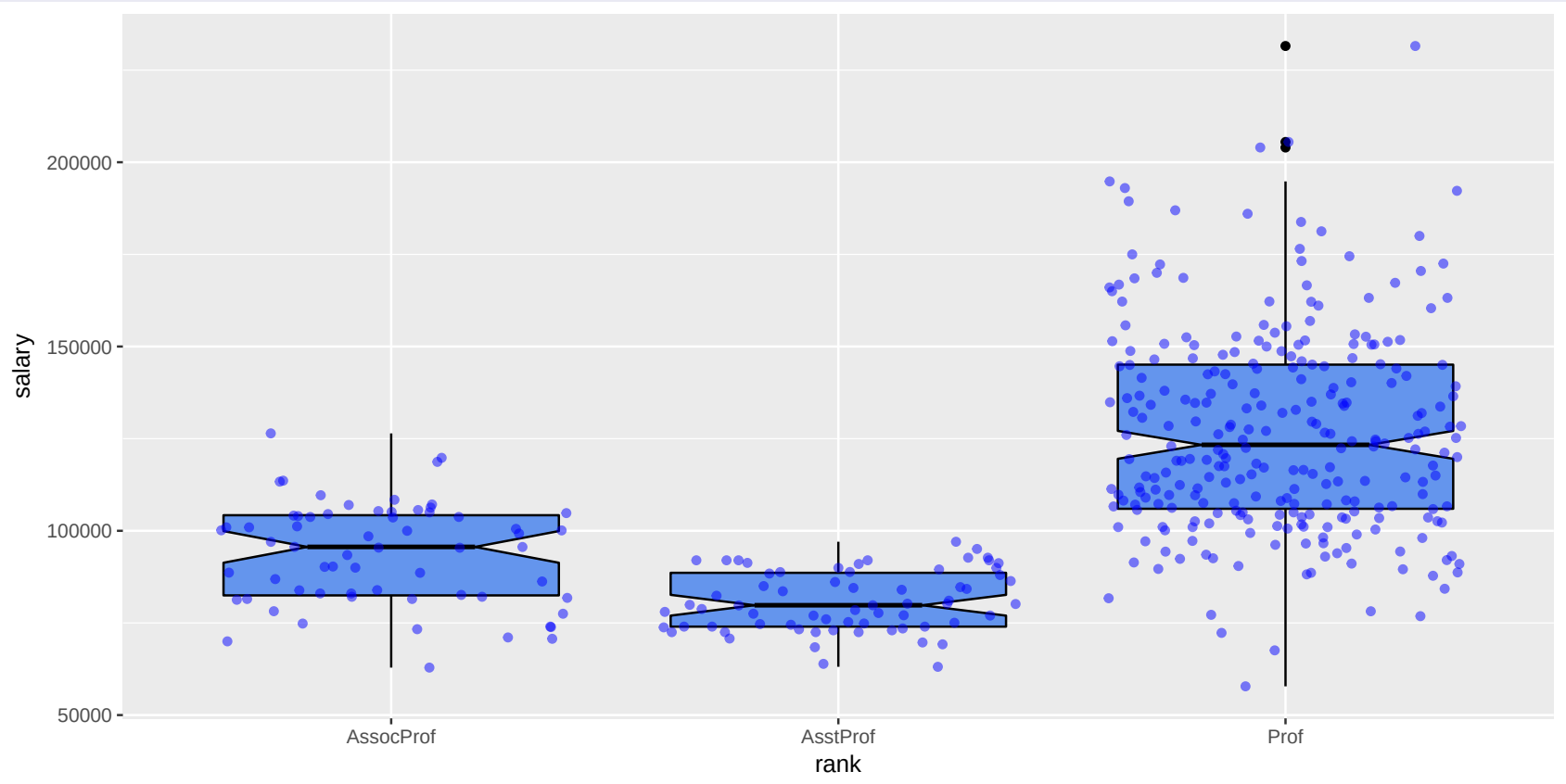
common options for geom functions

```
Salaries <- read.csv("salaries.csv")
library(ggplot2)
ggplot(Salaries, aes(x=rank, y=salary)) +
  geom_boxplot()+ geom_point()+ geom_rug()
```



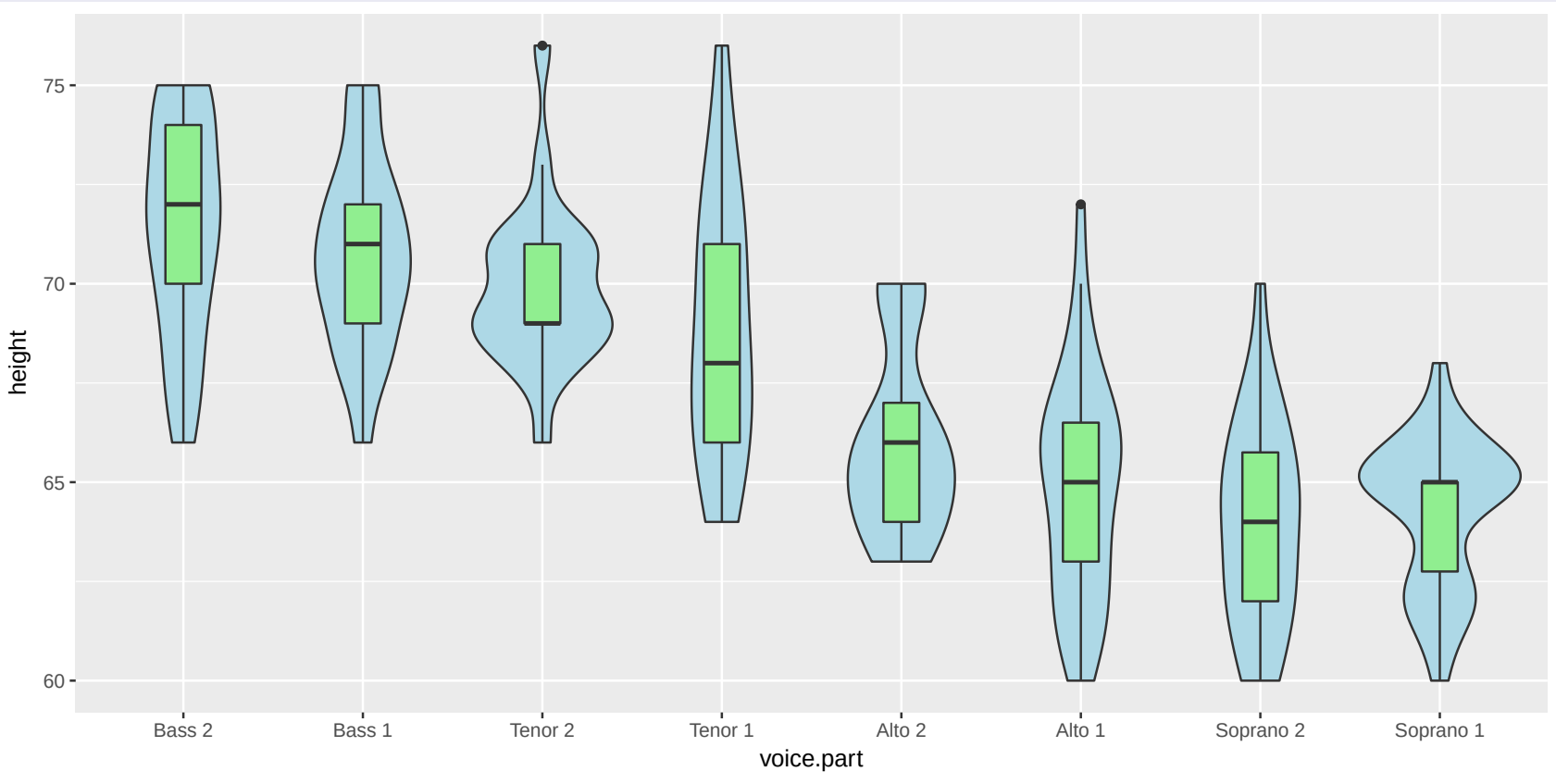
common options for geom functions

```
Salaries <- read.csv("salaries.csv")
ggplot(Salaries, aes(x=rank, y=salary)) +
geom_boxplot(fill="cornflowerblue",color="black",notch=T)+
  geom_point(position='jitter',color="blue", alpha=.5)
```



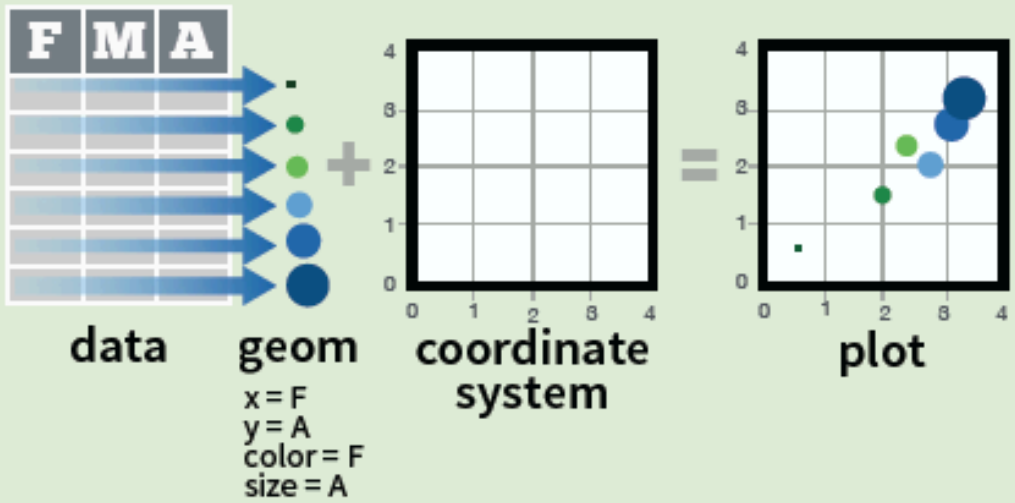
common options for geom functions

```
library(ggplot2)
data(singer, package="lattice")
ggplot(singer, aes(x=voice.part,y=height))+
geom_violin(fill="lightblue")+
geom_boxplot(fill="lightgreen", width=0.2)
```



Grouping

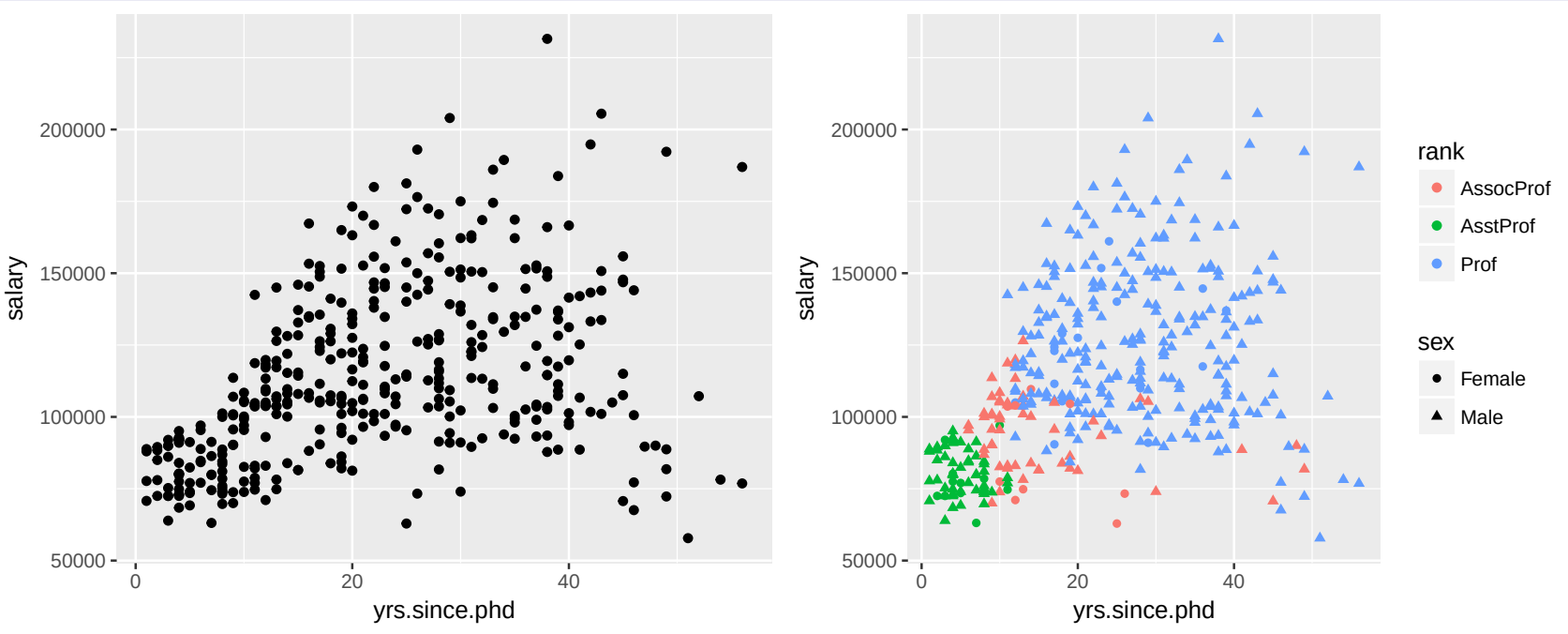
To display data values, map variables in the data set to aesthetic properties of the geom like **size**, **color**, and **x** and **y** locations.



Grouping

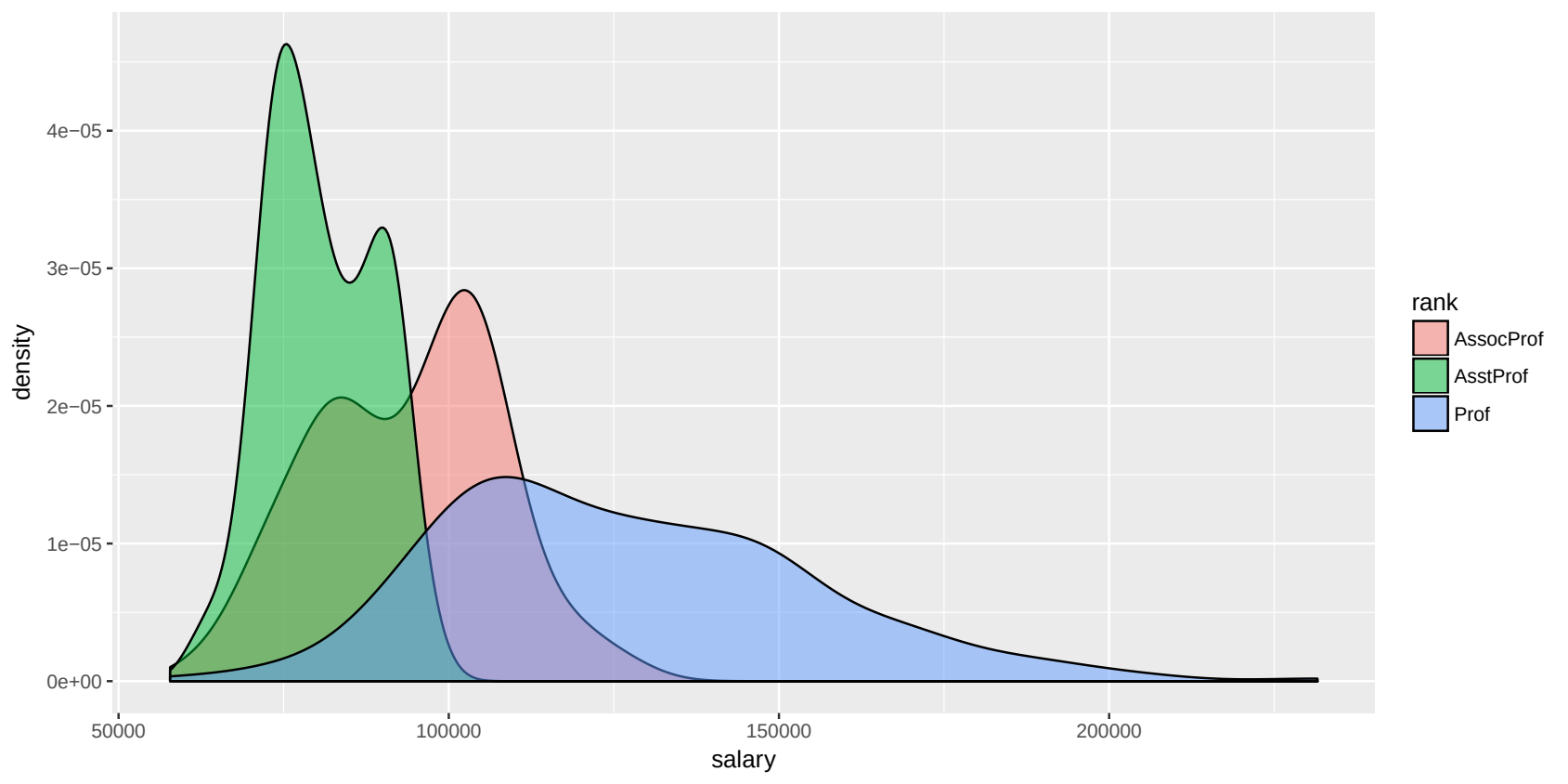
- color,fill, size,shape or combination.

```
Salaries <- read.csv("salaries.csv")
p<-ggplot(Salaries,aes(x=yrs.since.phd,y=salary))+geom_poi
q<-ggplot(Salaries,aes(x=yrs.since.phd,y=salary,
shape=sex,color=rank))+geom_point()
library(gridExtra);grid.arrange(p,q,ncol=2)
```



Grouping: kernal density plot

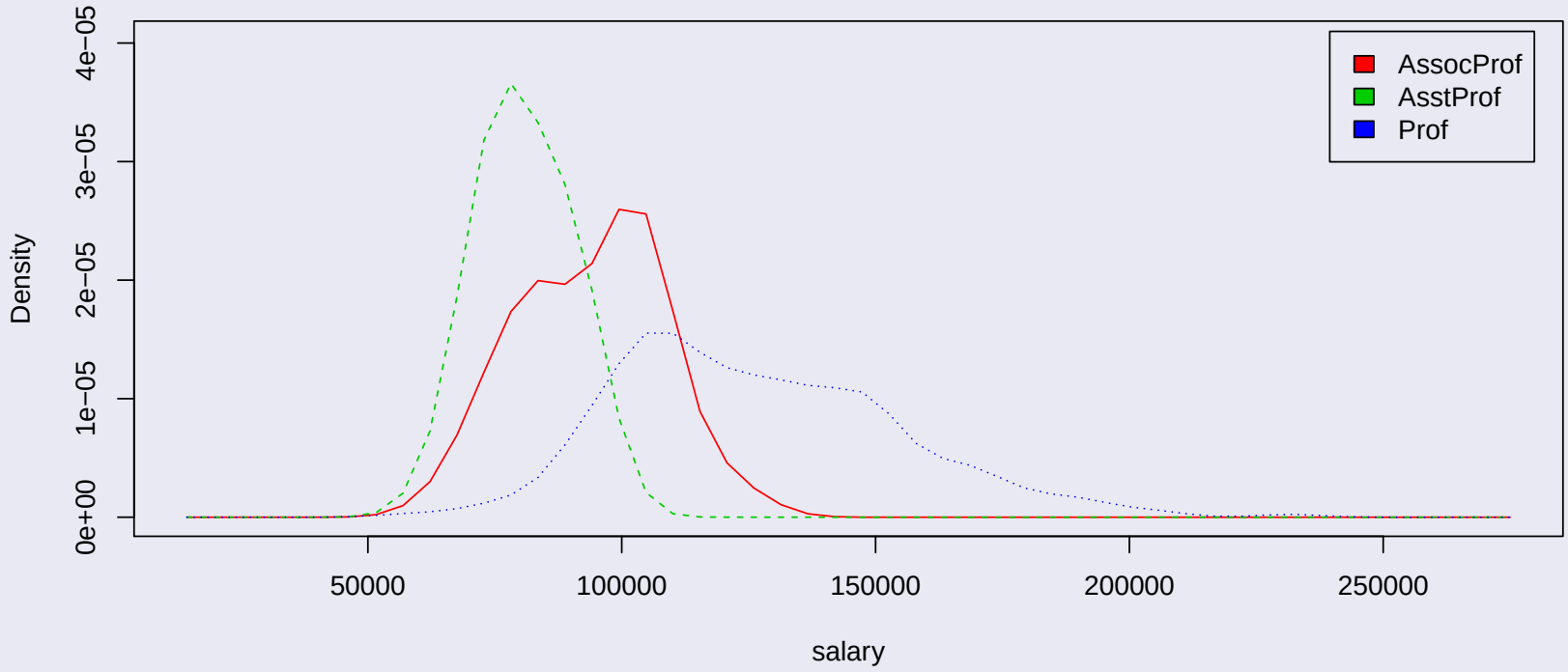
```
Salaries <- read.csv("salaries.csv")
ggplot(data=Salaries,aes(x=salary,fill=rank))+
  geom_density(alpha=.5,col='black')
```




```
# +facet_grid(rank~.)
```

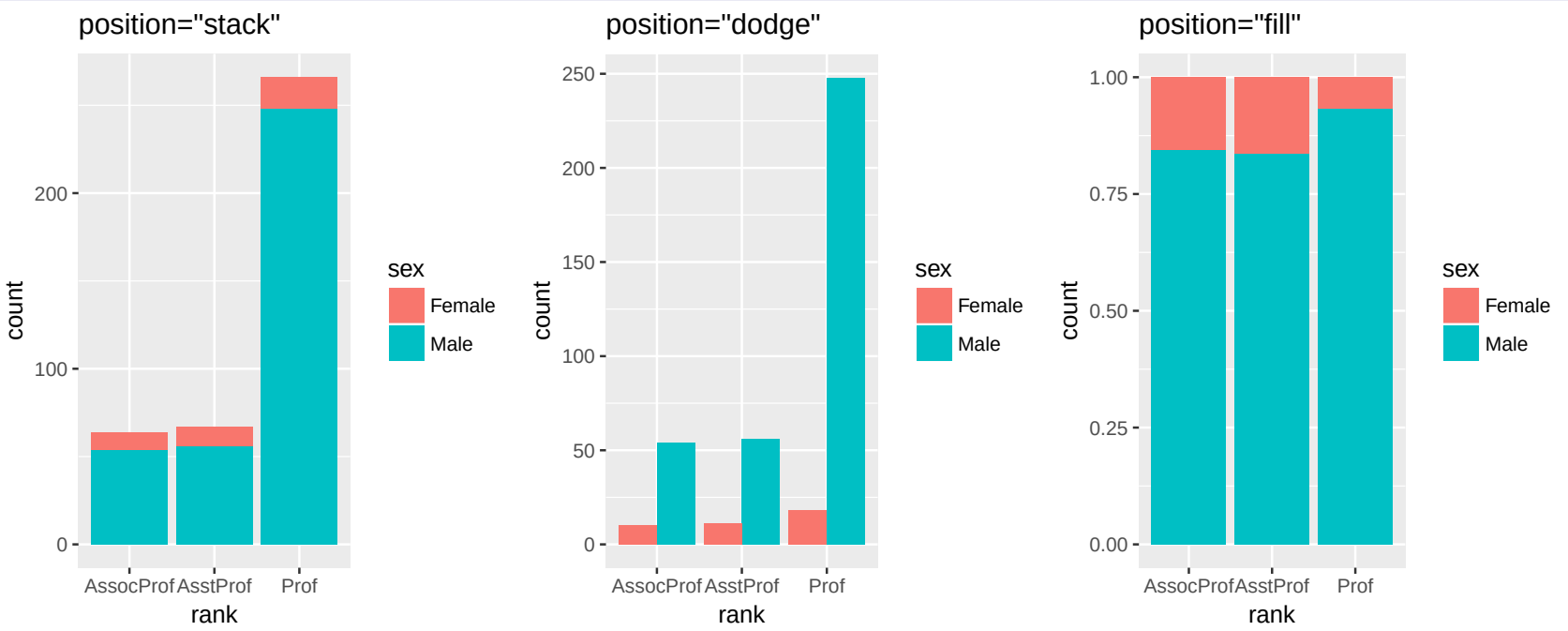
Grouping in base graphics

```
library(sm);Salaries <- read.csv("salaries.csv")
attach(Salaries);sm.density.compare(salary,rank)
colfill<-c(2:(2+length(levels(rank))))
legend('topright',levels(rank),fill=colfill,inset=0.02)
```



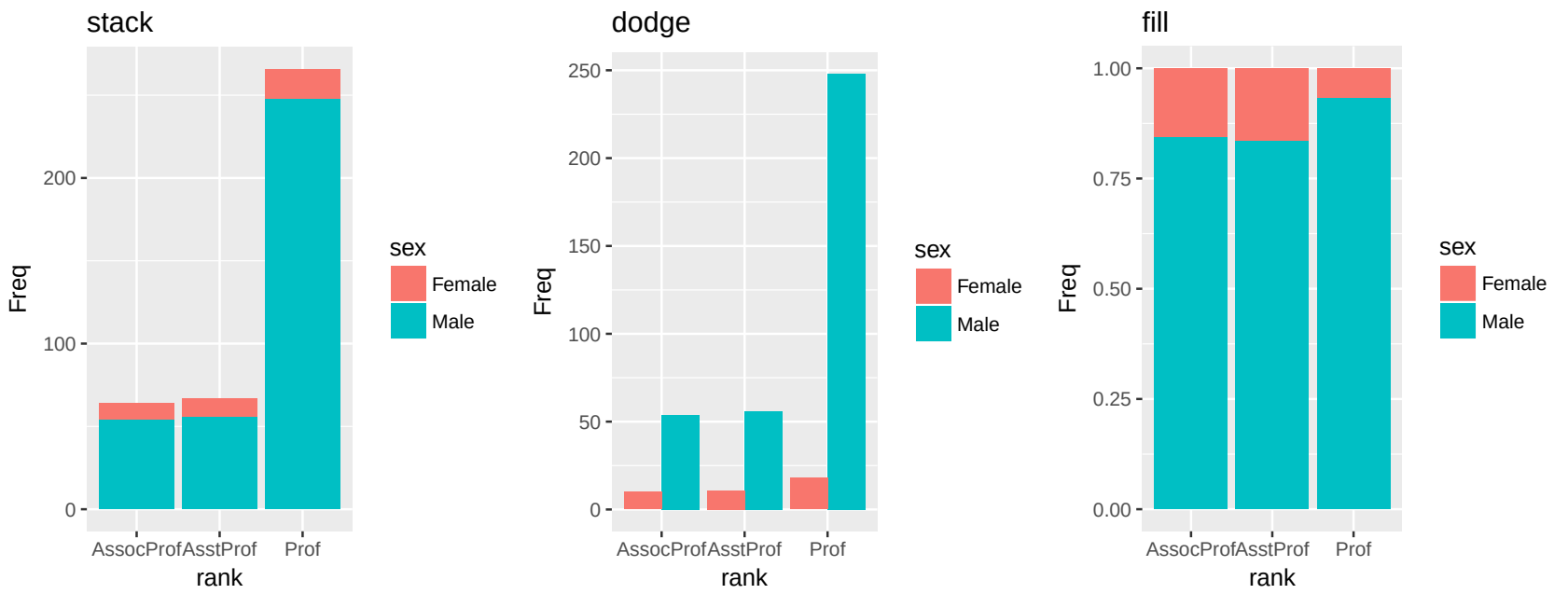
Grouping: barplot

```
a<-ggplot(Salaries, aes(x=rank, fill=sex))+
geom_bar(position="stack")+labs(title='position="stack"')
b<-ggplot(Salaries, aes(x=rank, fill=sex))+
geom_bar(position="dodge")+labs(title='position="dodge"')
c<-ggplot(Salaries, aes(x=rank, fill=sex))+
geom_bar(position="fill")+labs(title='position="fill"')
library(gridExtra);grid.arrange(a,b,c,ncol=3)
```



how to cope with summarized data:barplot

```
data<-data.frame(with(Salaries,table(sex,rank)))
a<-ggplot(data, aes(x=rank,y=Freq,fill=sex))+geom_bar(
stat='identity',position="stack")+labs(title='stack')
b<-ggplot(data, aes(x=rank,y=Freq,fill=sex))+geom_bar(
stat='identity',position="dodge")+labs(title='dodge')
c<-ggplot(data, aes(x=rank,y=Freq,fill=sex))+geom_bar(
stat='identity',position="fill")+labs(title='fill')
library(gridExtra);grid.arrange(a,b,c,ncol=3)
```



Facet:trellis graphs

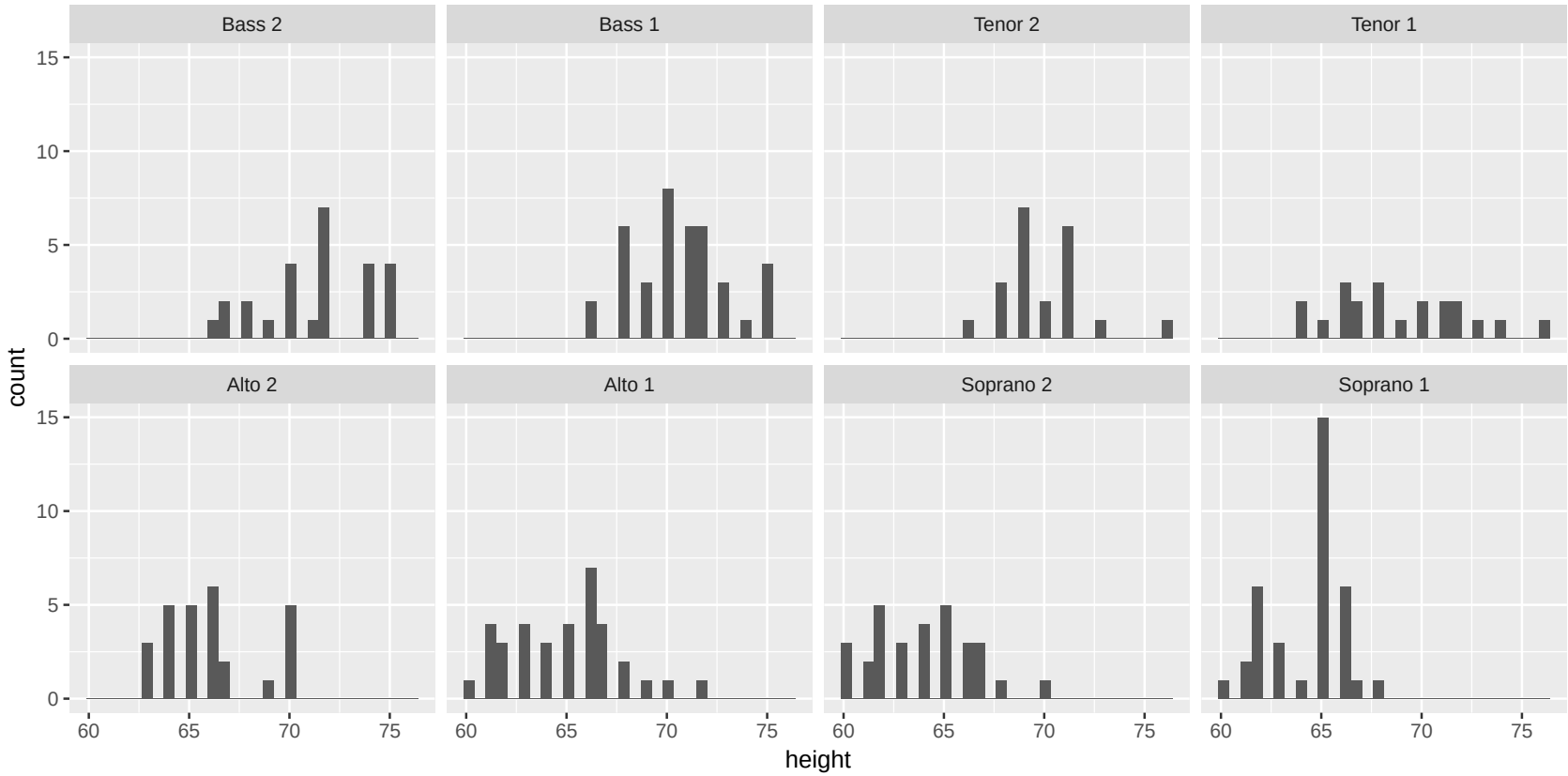
side-by-side graphs vs. single graph (grouping) - facet_wrap - facet_grid

Syntax	Results
`facet_wrap(~var, ncol=n)`	Separate plots for each level of *var* arranged into *n* columns
`facet_wrap(~var, nrow=n)`	Separate plots for each level of *var* arranged into *n* rows

Syntax	Results
`facet_grid(rowvar~colvar)`	Separate plots for each combination of *rowvar* and *colvar*, where *rowvar* represents rows and *colvar* represents columns
`facet_grid(rowvar~.)`	Separate plots for each level of *rowvar*, arranged as a single column
`facet_grid(.~colvar)`	Separate plots for each level of *colvar*, arranged as a single row

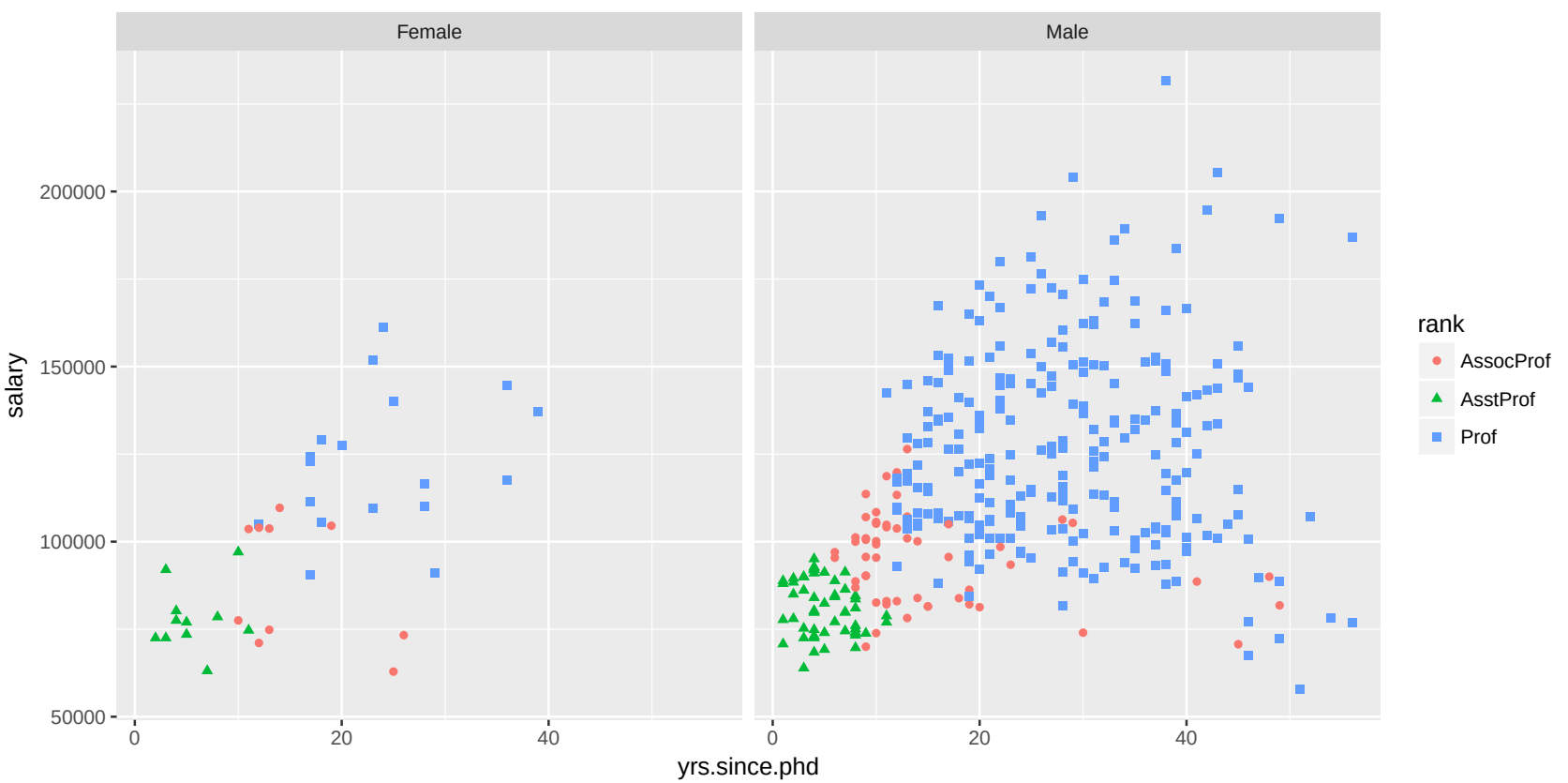
Faceting: facet_wrap

```
data(singer, package="lattice")
ggplot(data=singer, aes(x=height)) +
geom_histogram() + facet_wrap(~voice.part,nrow=2)
```



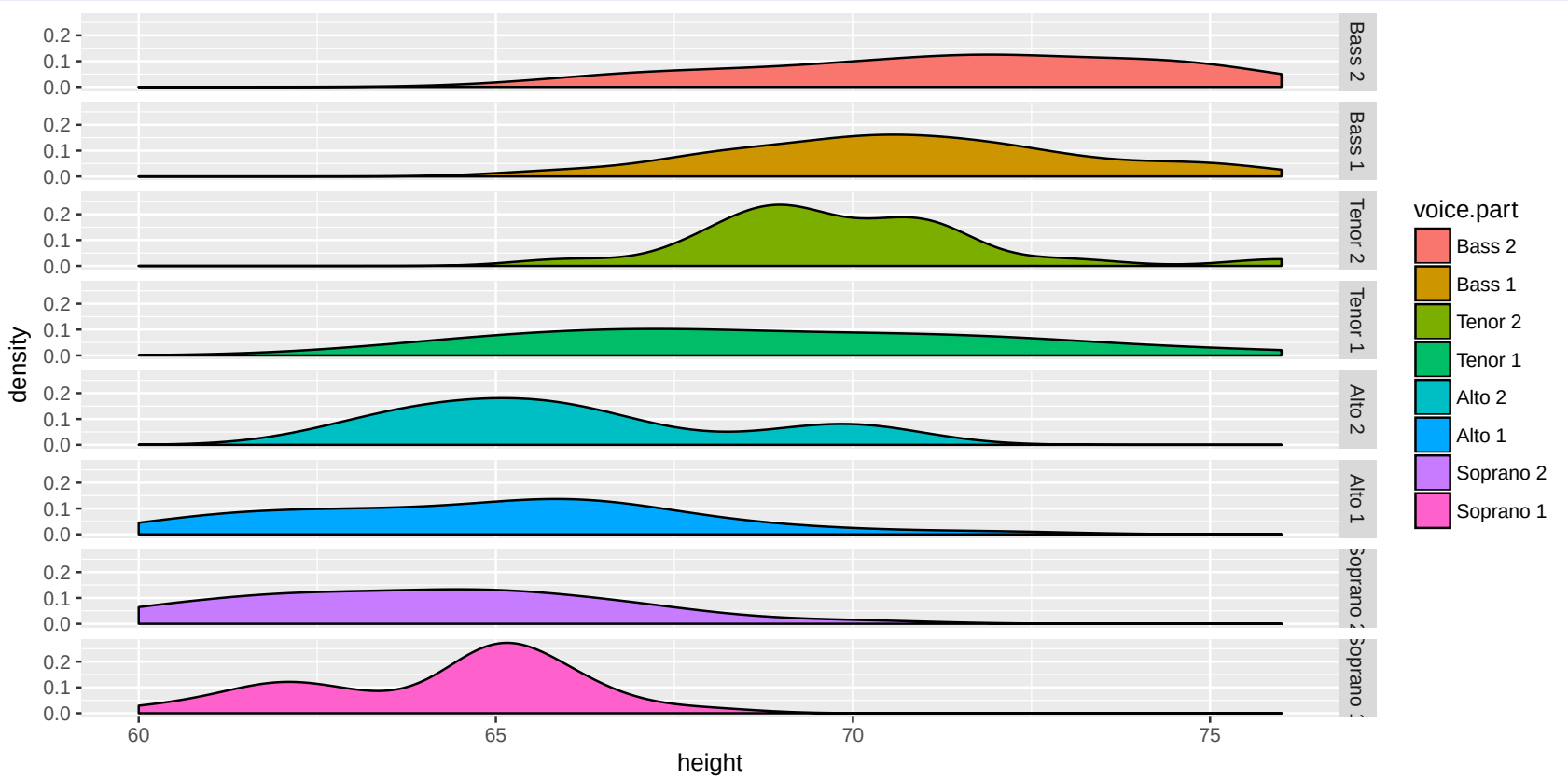
Faceting: facet_grid

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



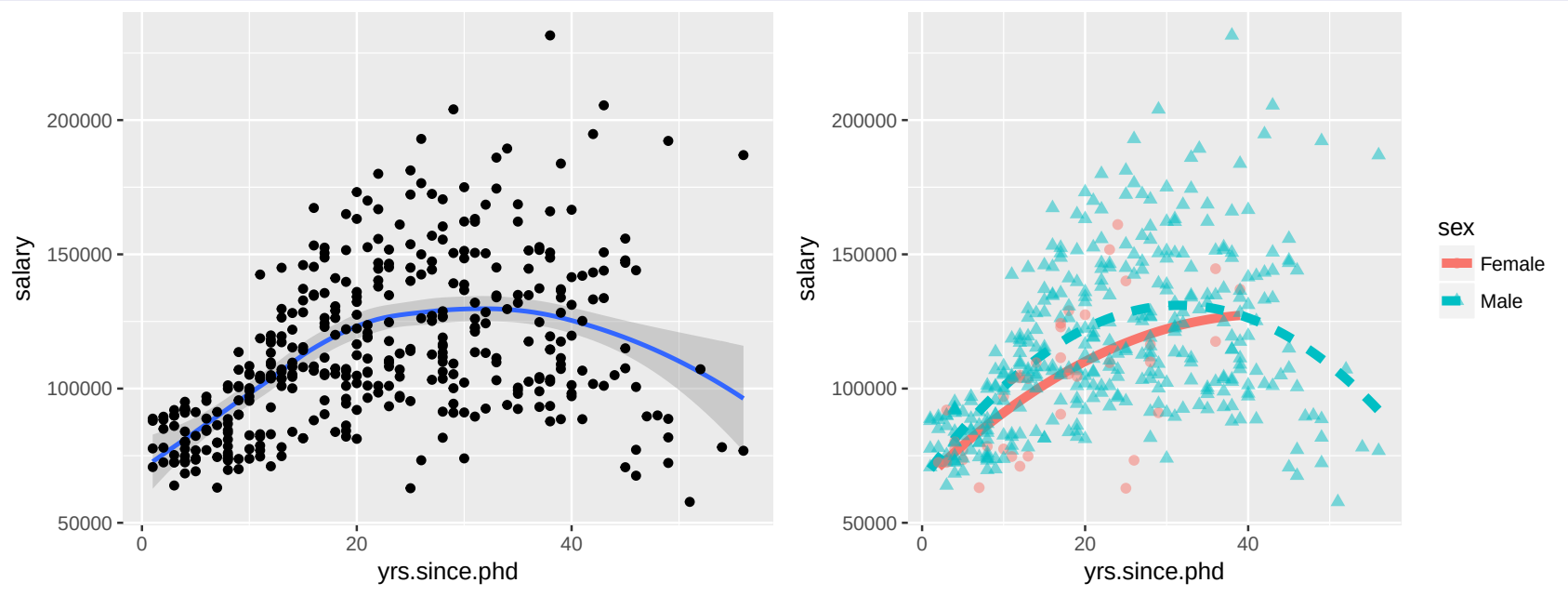
facet_grid

```
data(singer, package="lattice")
library(ggplot2)
ggplot(data=singer, aes(x=height, fill=voice.part))+
  geom_density()+ facet_grid(voice.part~.)
```



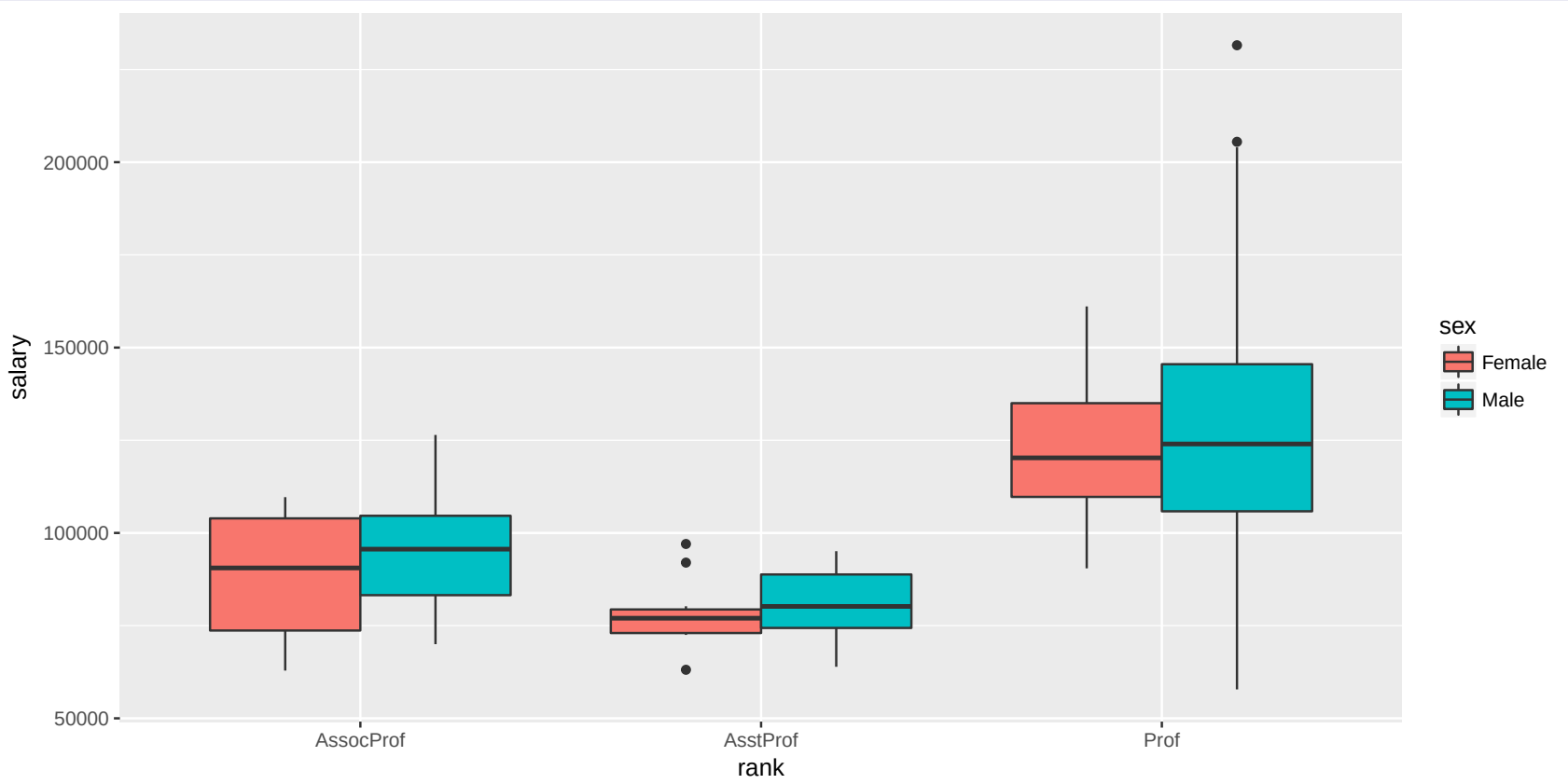
Adding smoothed lines

```
Salaries <- read.csv("salaries.csv")
a<-ggplot(data=Salaries, aes(x=yrs.since.phd, y=salary)) +
geom_smooth() + geom_point()
b<-ggplot(data=Salaries, aes(x=yrs.since.phd, y=salary,
linetype=sex, shape=sex, color=sex)) +
geom_smooth(method=lm, formula=y~poly(x,2),se=F,size=2)+
geom_point(size=2,alpha=0.5)# quadratic
library(gridExtra);grid.arrange(a,b,ncol=2)
```

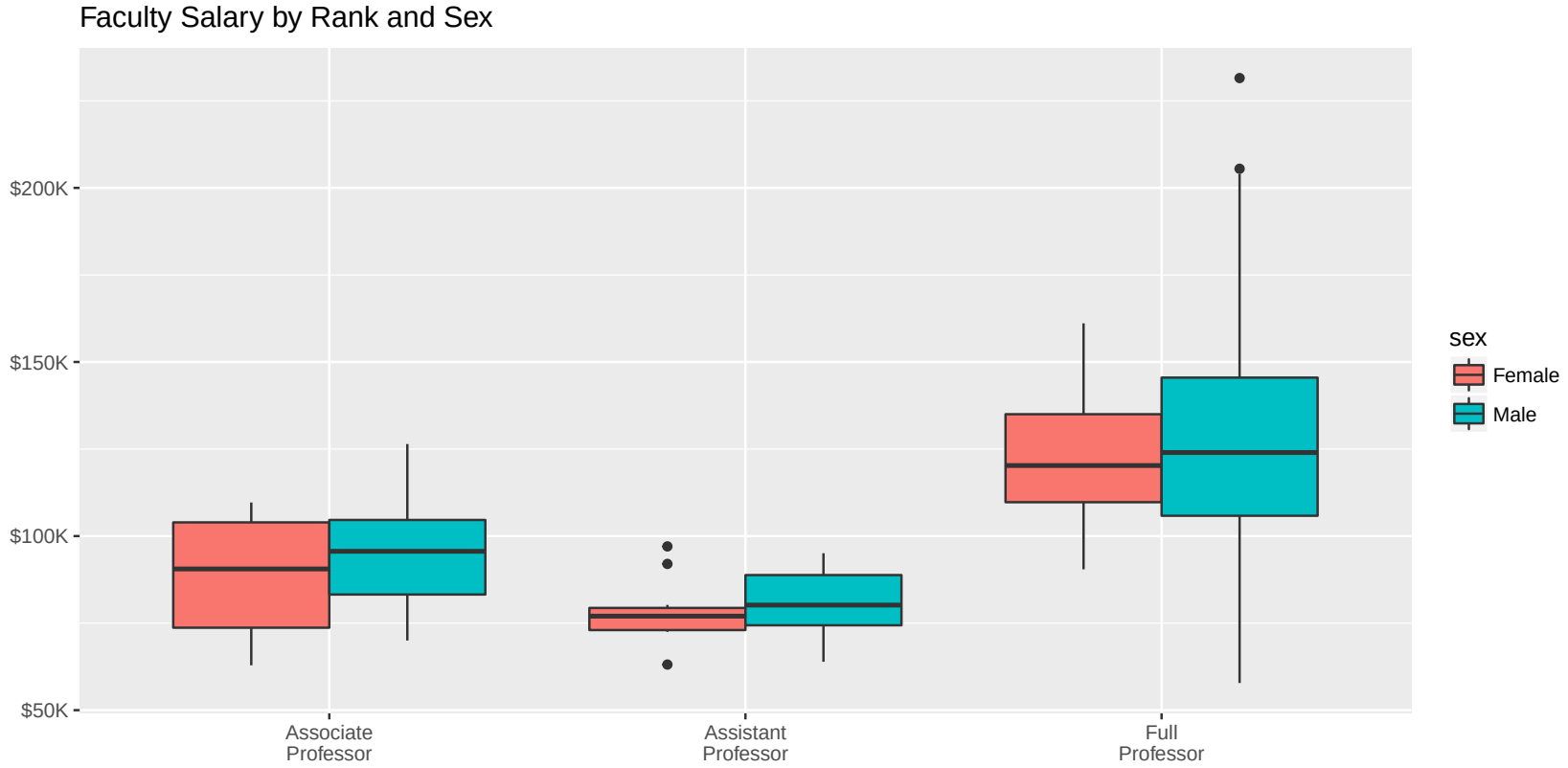


Modifying axes

```
Salaries <- read.csv("salaries.csv")
library(ggplot2)
ggplot(Salaries,aes(x=rank, y=salary,fill=sex))+
geom_boxplot()
```



Modifying axes



Modifying axes

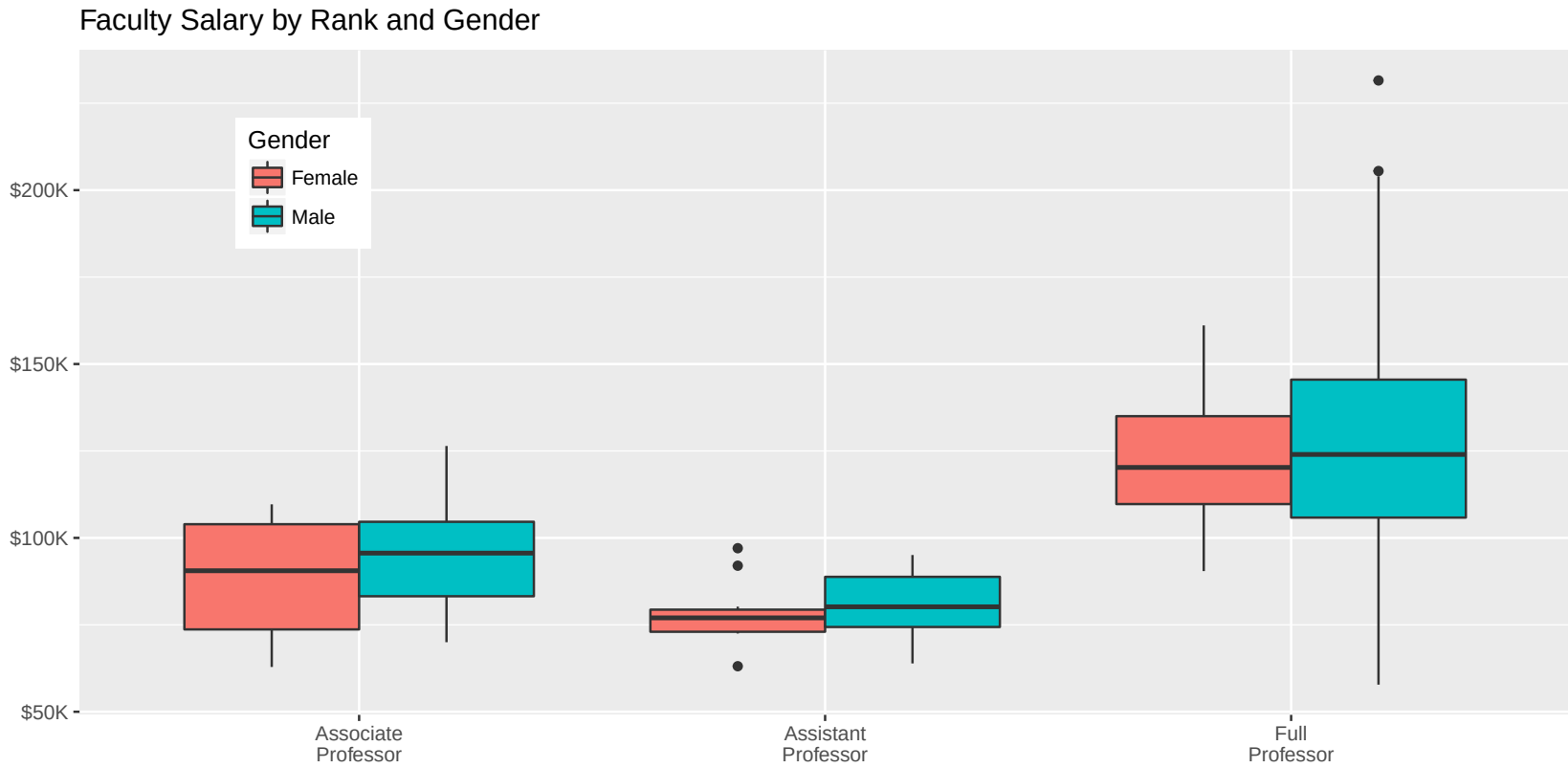
Function	Options
`scale_x_continuous()`, `scale_y_continuous()`	`breaks=` specifies tick marks, `labels=` specifies labels for tick marks, and `limits=` controls the range of the values displayed.

Function	Options
`scale_x_discrete()`, `scale_y_discrete()`	`breaks=` places and orders the levels of a factor, `labels=` specifies the labels for these levels, and `limits=` indicates which levels should be displayed.
`coord_flip()`	Reverses the x and y axes.

Modifying axes

```
Salaries <- read.csv("salaries.csv")
library(ggplot2)
ggplot(data=Salaries, aes(x=rank, y=salary,fill=sex)) +
  geom_boxplot() +
  scale_x_discrete(breaks=c("AsstProf", "AssocProf", "Prof"
                   labels=c("Assistant\nProfessor",
                            "Associate\nProfessor",
                            "Full\nProfessor")) +
  scale_y_continuous(breaks=c(50000, 100000, 150000, 20000
                     labels=c("$50K", "$100K", "$150K", "$2
  labs(title="Faculty Salary by Rank and Sex", x="", y="")
```

Legends

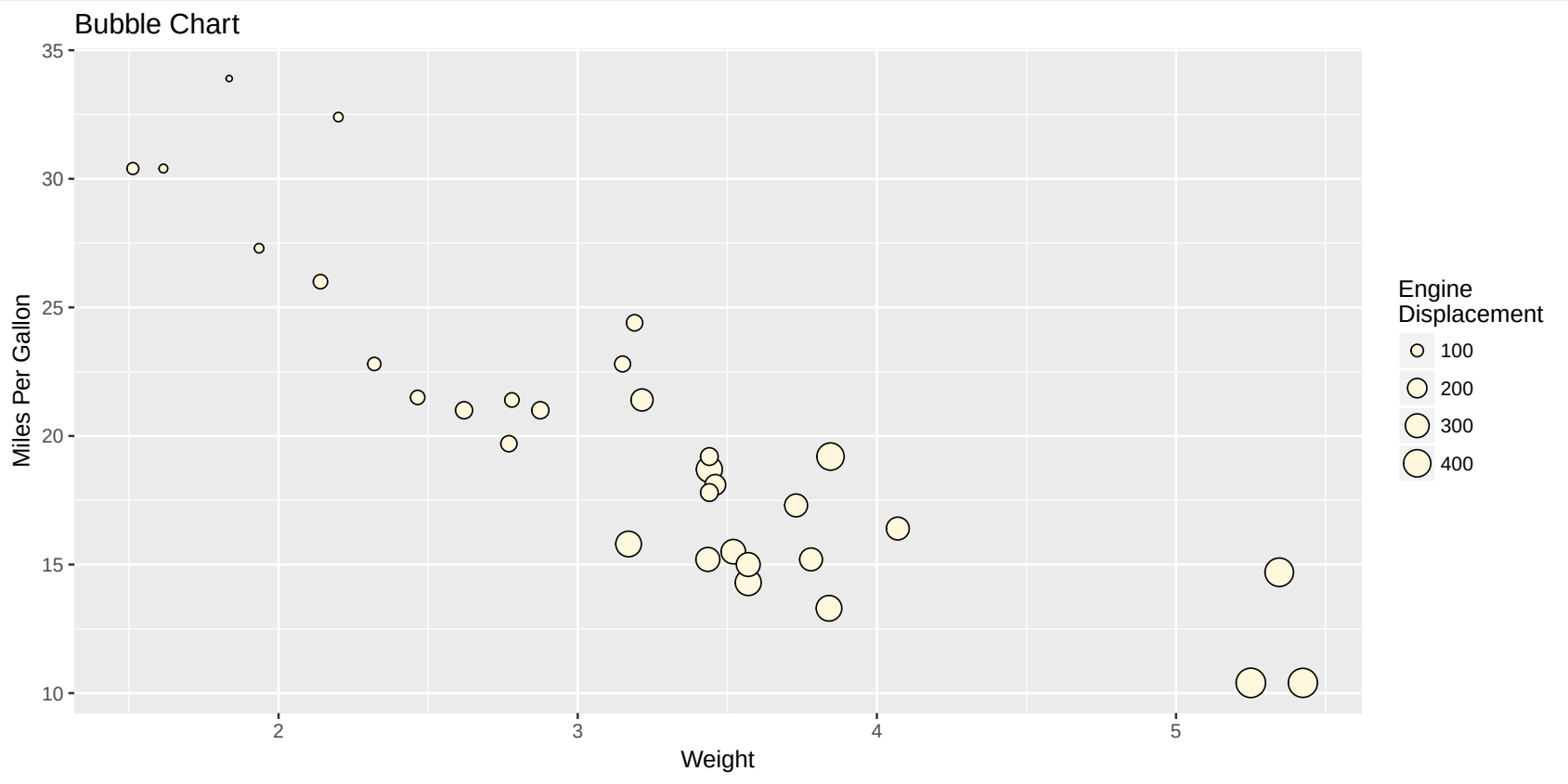

Faculty Salary by Rank and Gender
Gender
Female
Male
$200K
$150K
$100K
$50K
Associate Professor
Assistant Professor
Full Professor

Legends

```
Salaries <- read.csv("salaries.csv")
ggplot(data=Salaries, aes(x=rank, y=salary, fill=sex)) +
  geom_boxplot() +
scale_x_discrete(breaks=c("AsstProf", "AssocProf", "Prof"),
                 labels=c("Assistant\nProfessor",
                          "Associate\nProfessor",
                          "Full\nProfessor")) +
  scale_y_continuous(breaks=c(50000, 100000, 150000, 200000
                     labels=c("$50K", "$100K", "$150K", "$2
  labs(title="Faculty Salary by Rank and Gender",
       x="", y="", fill="Gender") +
  theme(legend.position=c(.15,.8))
```

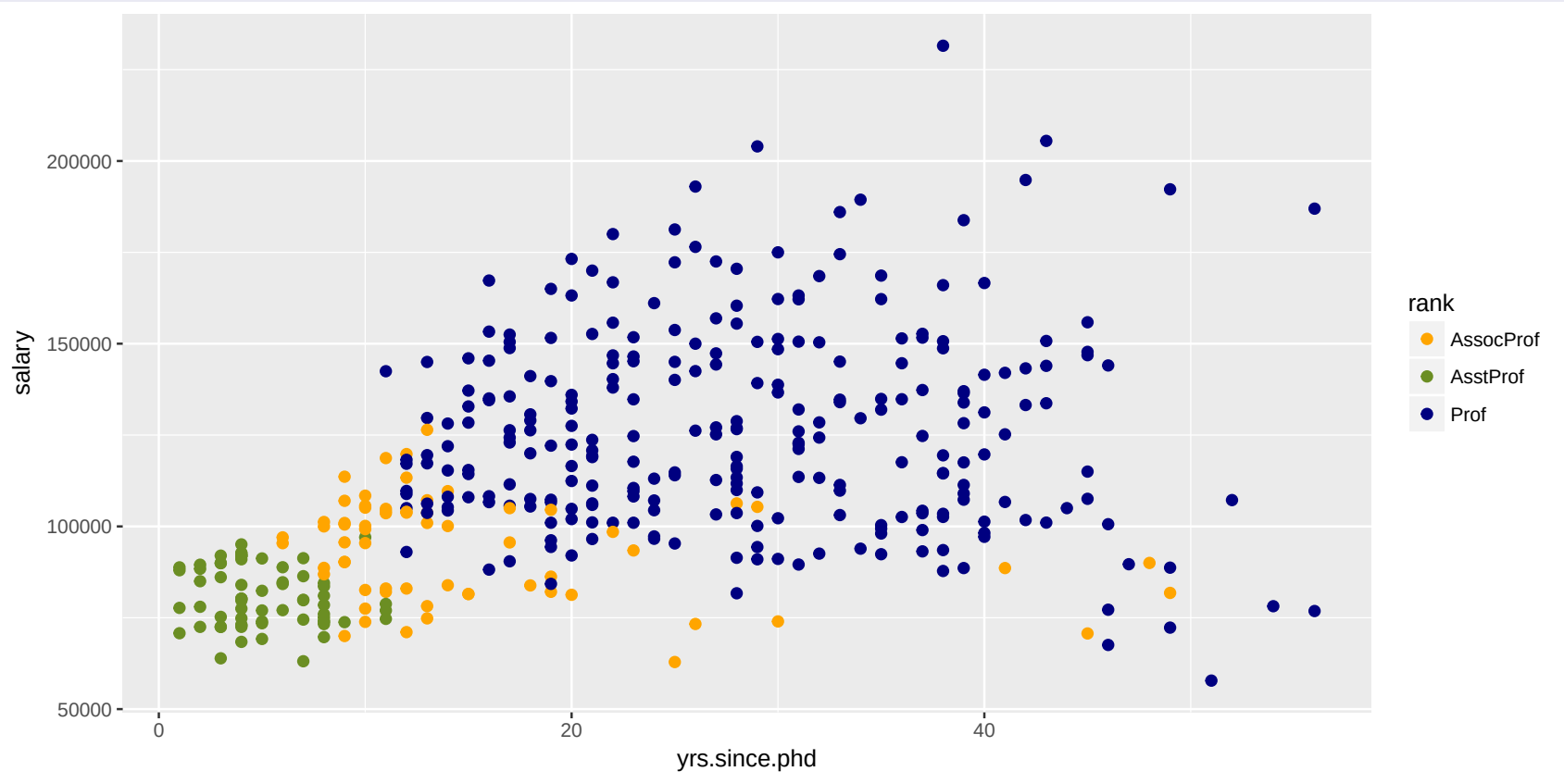
Scale

```
# table(mtcars$disp)
ggplot(mtcars, aes(x=wt, y=mpg, size=disp)) +
  geom_point(shape=21, color="black", fill="cornsilk") +
  labs(x="Weight", y="Miles Per Gallon",
title="Bubble Chart", size="Engine\nDisplacement")
```



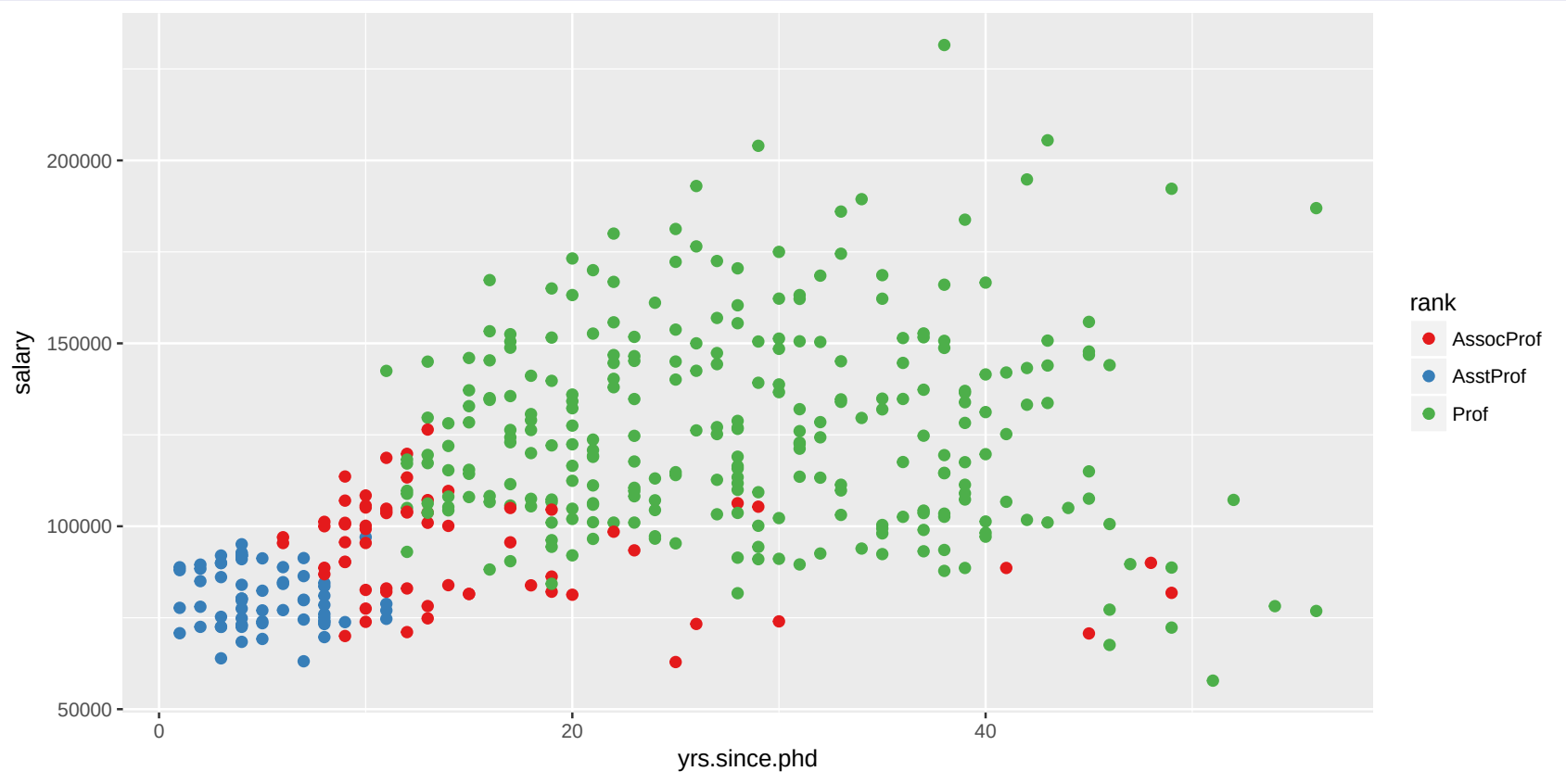
Scales

```
Salaries <- read.csv("salaries.csv")
ggplot(data=Salaries, aes(x=yrs.since.phd, y=salary, color=
  geom_point(size=2)+
  scale_color_manual(values=c("orange", "olivedrab", "navy"
```



Scales

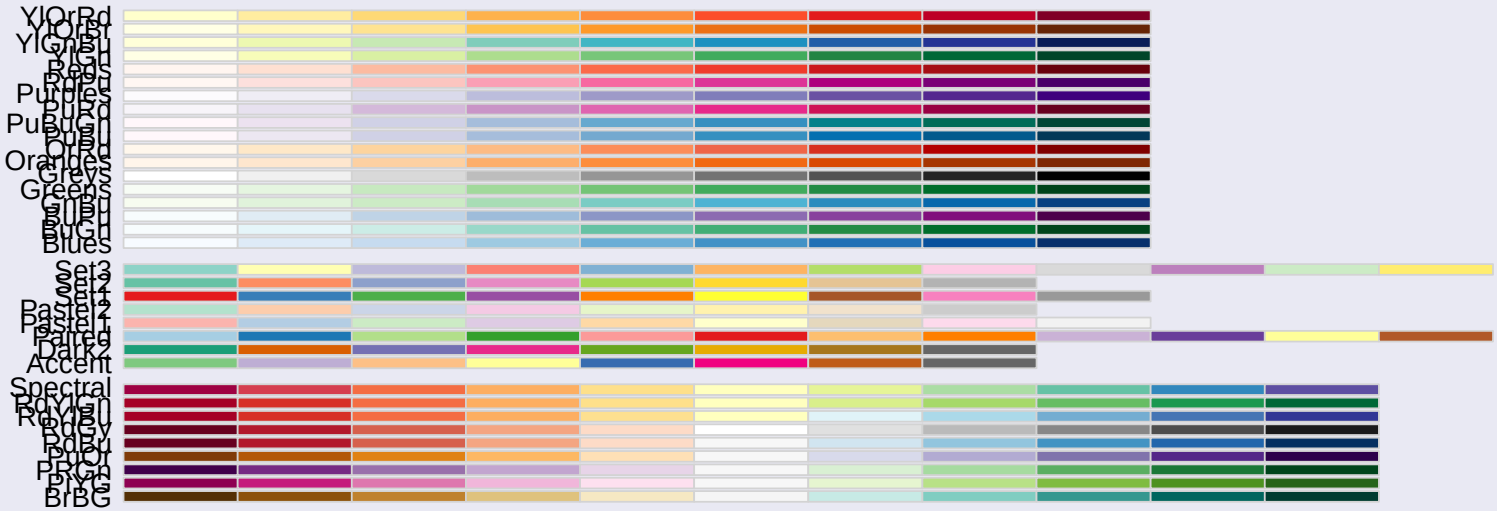
```
Salaries <- read.csv("salaries.csv")
ggplot(Salaries,aes(x=yrs.since.phd,y=salary,color=rank))+
scale_color_brewer(palette="Set1")+
geom_point(size=2)
```




```
# "Set1", "Set2", "Pastel1", "Pastel2", "Paired", "Dark2"
```

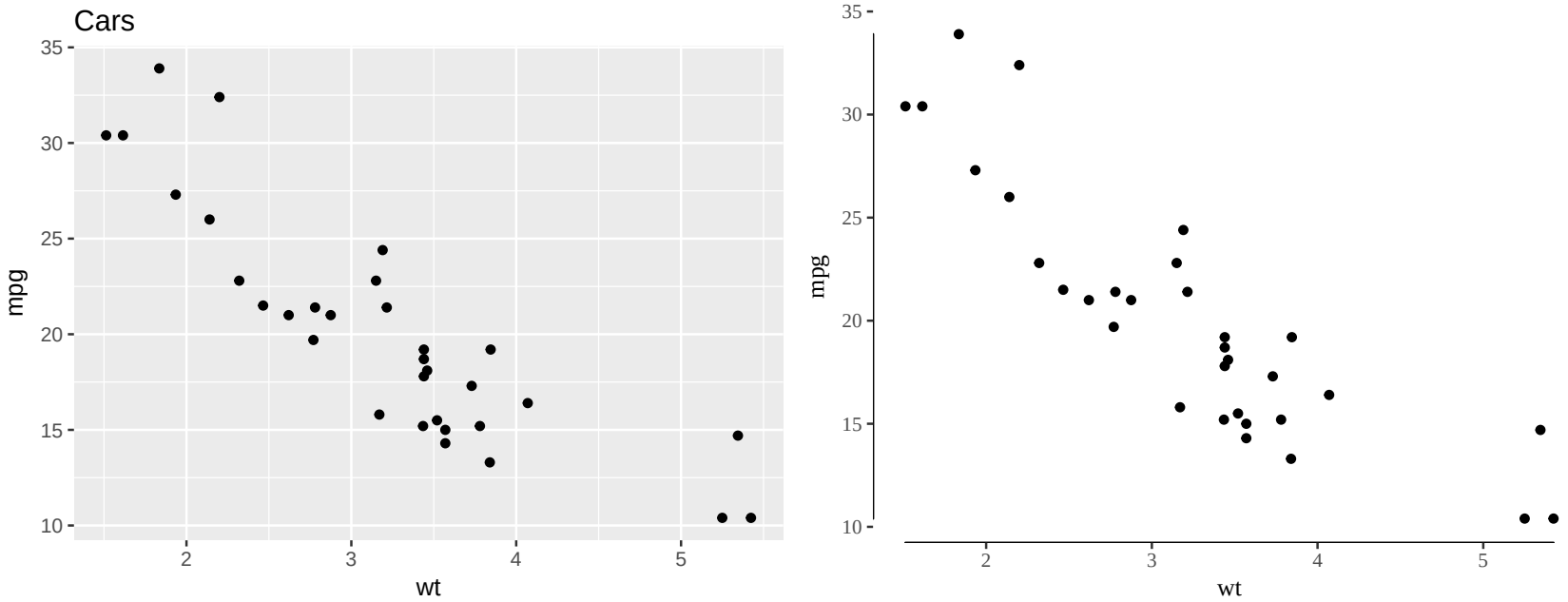
Scales

```
library(RColorBrewer);display.brewer.all()
```

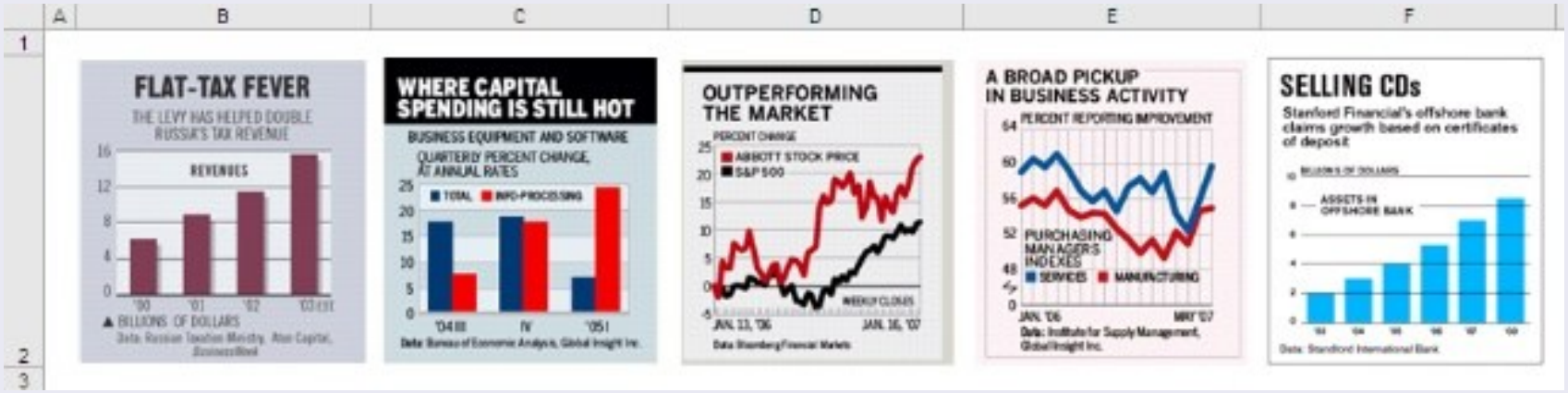


Themes

```
library(ggthemes)
p1<-ggplot(mtcars, aes(x = wt, y = mpg)) +
  geom_point() +labs(title='Cars')
p2<-ggplot(mtcars, aes(wt, mpg)) +
geom_point() + geom_rangeframe() +
theme_tufte()
library(gridExtra);grid.arrange(p1,p2,ncol=2)
```

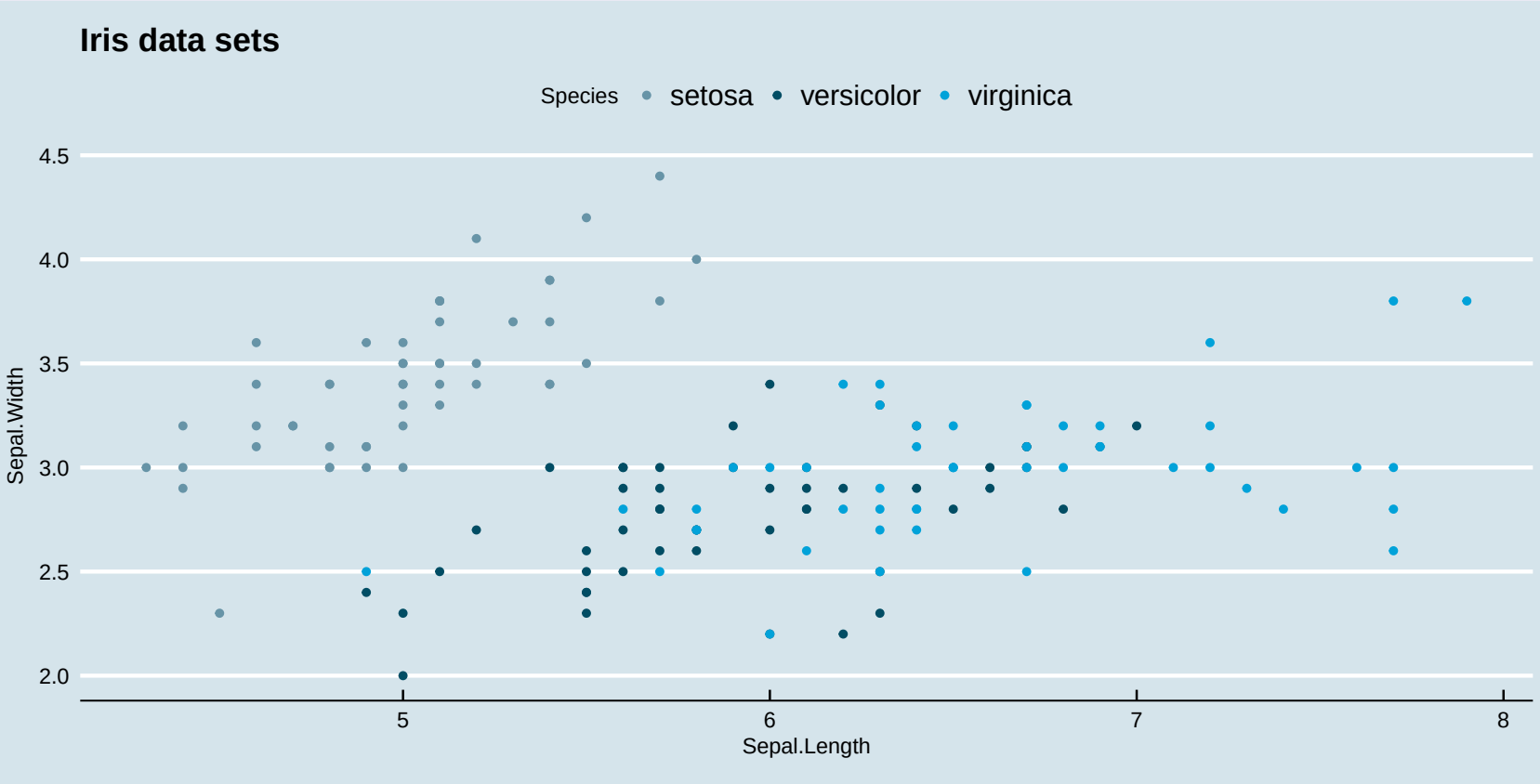


Themes


A
B
C
D
E
F
1
2
3
FLAT-TAX FEVER
THE LEVY HAS HELPED DOUBLE RUSSIA'S TAX REVENUE
REVENUES
16
12
8
4
0
'00
'01
'02
'03 EST
▲ BILLIONS OF DOLLARS
WHERE CAPITAL SPENDING IS STILL HOT
BUSINESS EQUIPMENT AND SOFTWARE
QUARTERLY PERCENT CHANGE, AT ANNUAL RATES
TOTAL
INFO-PROCESSING
25
20
15
10
5
0
'04 III
IV
'05 I
OUTPERFORMING THE MARKET
PERCENT CHANGE
ABBOTT STOCK PRICE
S&P 500
25
20
15
10
5
0
-5
JAN. 13, '06
JAN. 16, '07
A BROAD PICKUP IN BUSINESS ACTIVITY
PERCENT REPORTING IMPROVEMENT
64
60
56
52
48
0
PURCHASING MANAGERS INDEXES
SERVICES
MANUFACTURING
JAN. '06
MAY '07
SELLING CDs
Stanford Financial's offshore bank claims growth based on certificates of deposit
ASSETS IN OFFSHORE BANK

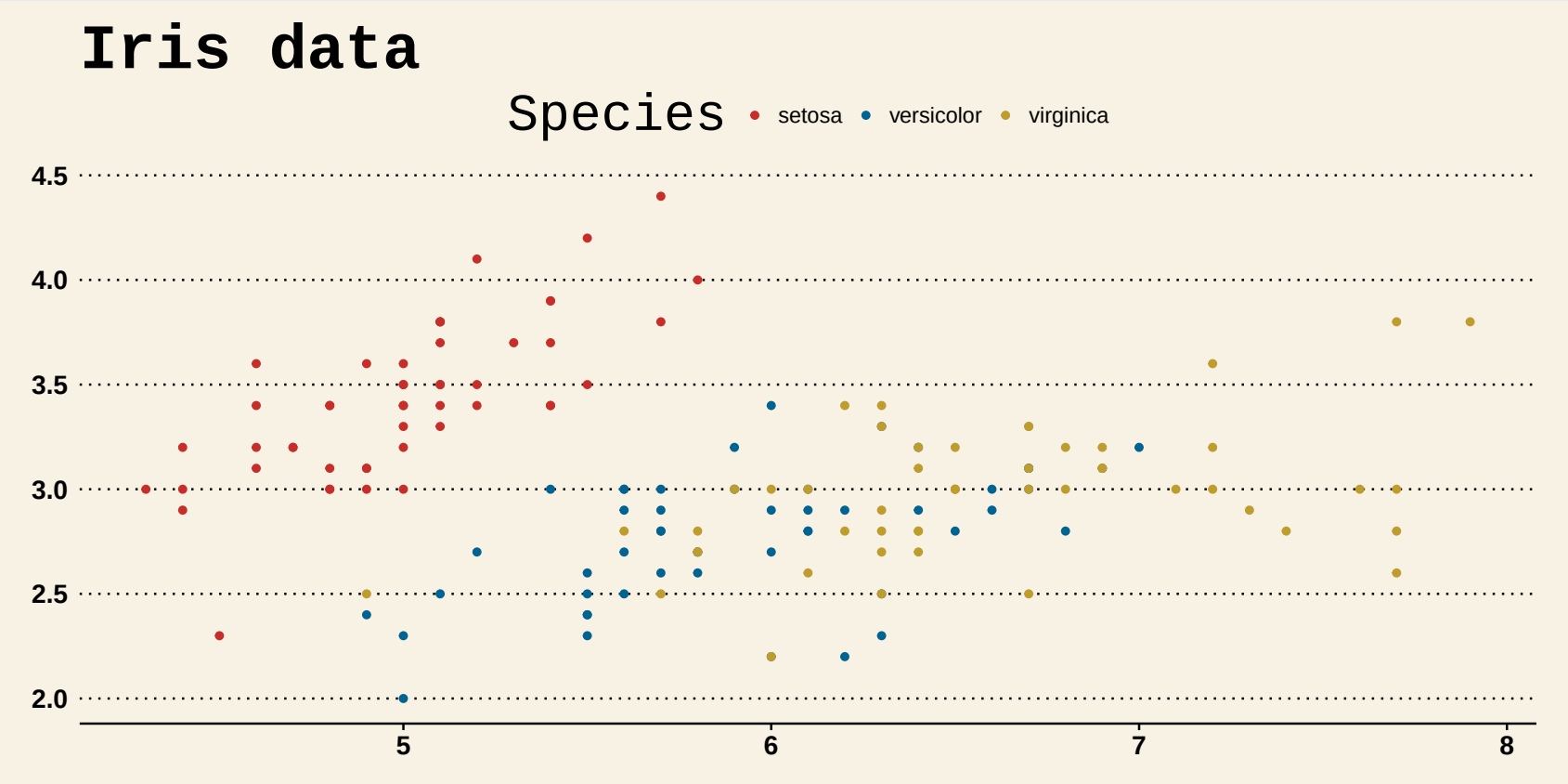
Economist template

```
p<-ggplot(iris,aes(Sepal.Length,Sepal.Width,colour
= Species))+ geom_point(size=1.3)
p + theme_economist() +
scale_color_economist()+
ggtitle("Iris data sets")
```



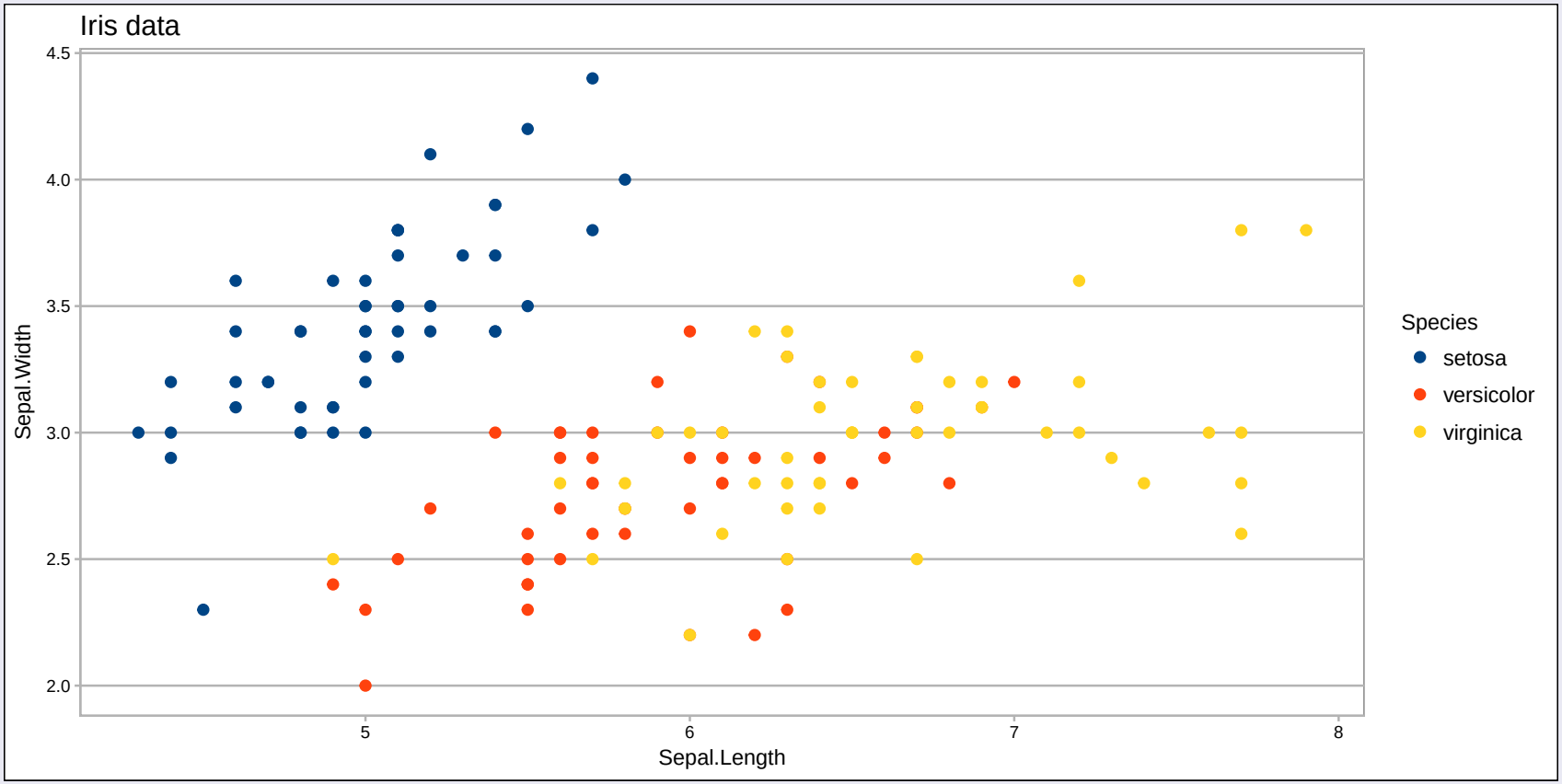
Wall Street Journal template

```
p<-ggplot(iris,aes(Sepal.Length,Sepal.Width,colour
= Species))+ geom_point(size=1.3)
p + theme_wsj()+ scale_colour_wsj("colors6")+
ggtitle("Iris data")# rgby, red_green, black_green, dem_re
```



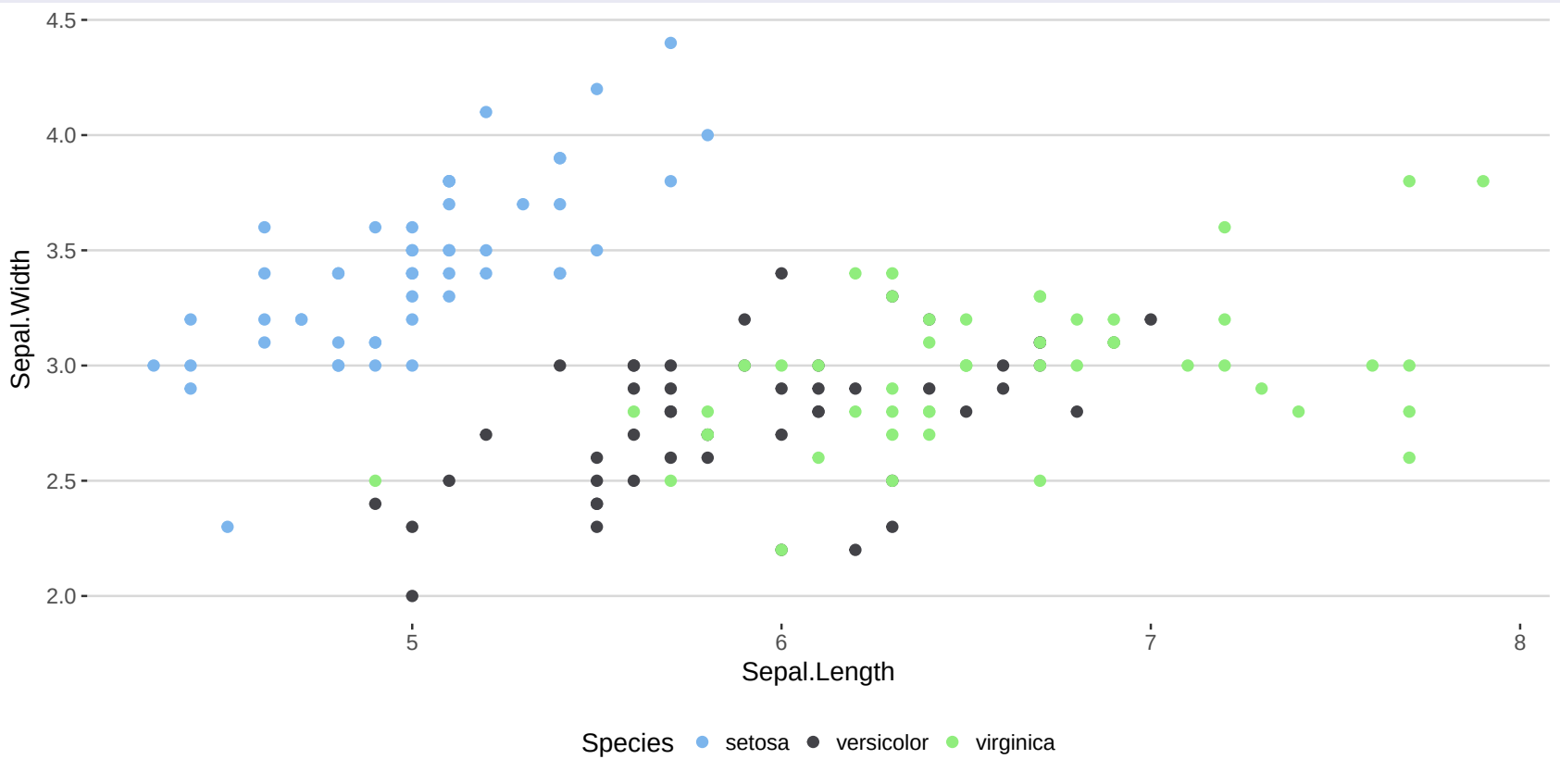
Google Docs

```
p<-ggplot(iris,aes(Sepal.Length,Sepal.Width,colour
= Species))+ geom_point(size=2)
p + theme_calc()+ scale_colour_calc()+
ggtitle("Iris data")
```



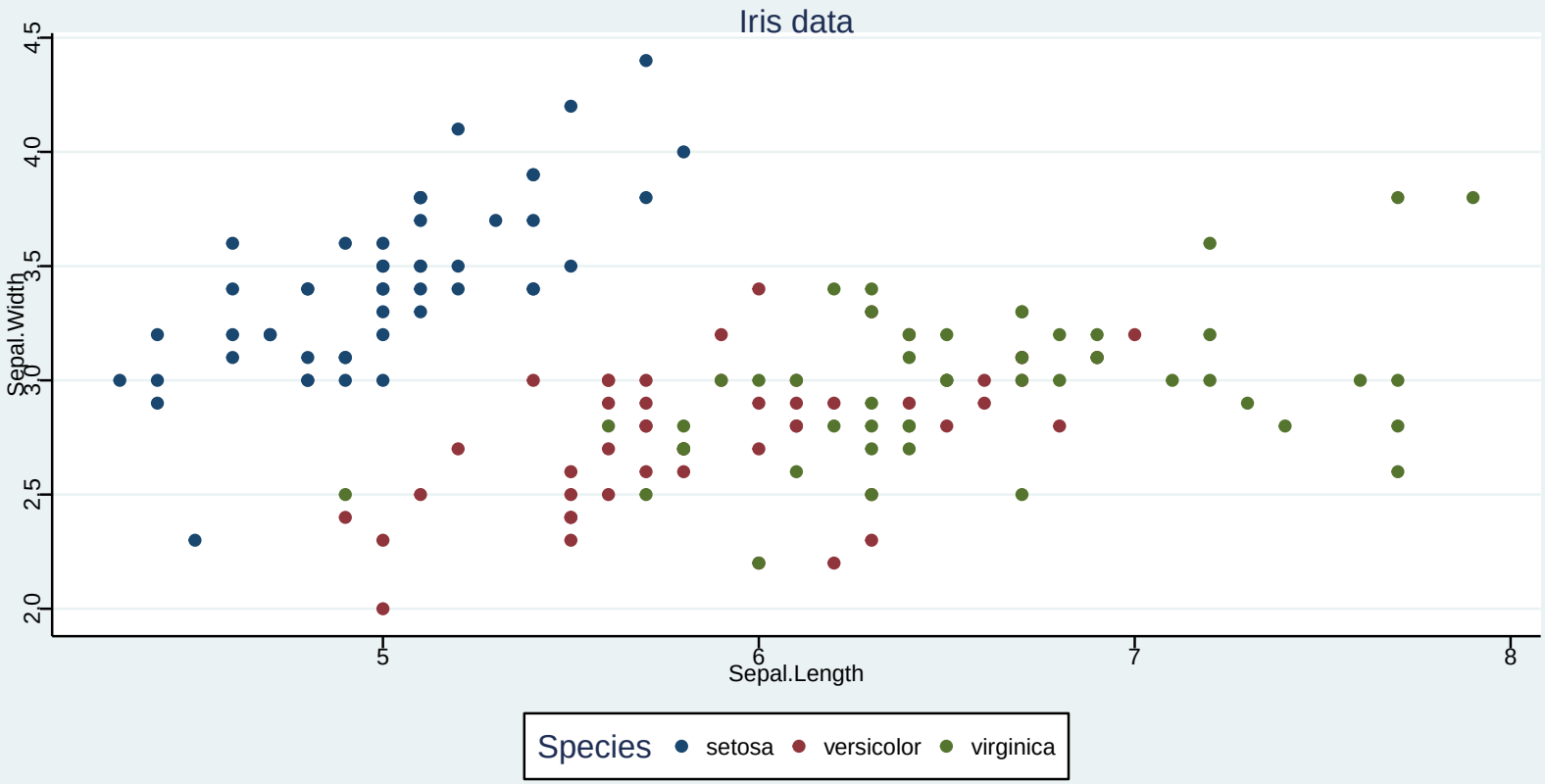
Highcharts JS

```
p<-ggplot(iris,aes(Sepal.Length,Sepal.Width,colour
= Species))+ geom_point(size=2)
p + theme_hc()+ scale_colour_hc()
```



Stata template

```
p<-ggplot(iris,aes(Sepal.Length,Sepal.Width,colour= Species
p + theme_stata() + scale_color_stata() +
ggtitle("Iris data")
```



```
myplot<-ggplot(data=mtcars,aes(x=mpg))+
                    geom_histogram()
ggsave(file='mygraph.png',plot=myplot,width=5,height=4)
ggsave(file='mygraph.png',width=5,height=4)
```

summary

- Plot = data + Aesthetics + Geometry.
- grouping (aes)
- facet (facet_wrap, facet_grid)
- appearance of ggplot2 graphs (axes,legend,scale, theme)
- combine and save the figures

ggplot2 books

- R Graphics Cookbook (Winston Chang)
- ggplot2: Elegant Graphics for Data Analysis (Hadley Wickham)