Learn ggplot2 As Kungfu Skills

Given by
Kai Xiao(Data Sciencetist)
Vivian Zhang(Co-founder & CTO)

Contact: vivian.zhang@supstat.com

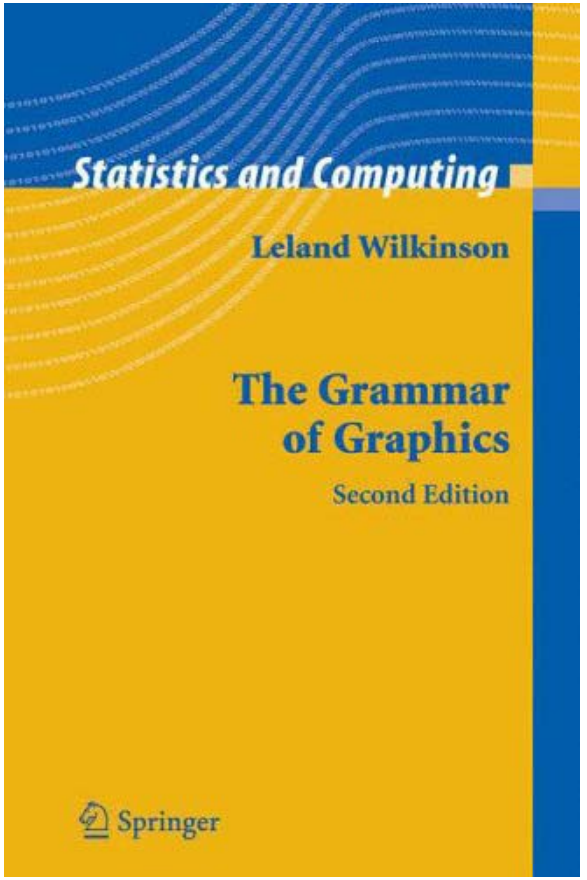
NYC Open Data

SupStat

- I： Point
- II： Bar
- III：Histogram
- IV：Line
- V： Tile
- VI：Map

Introduction

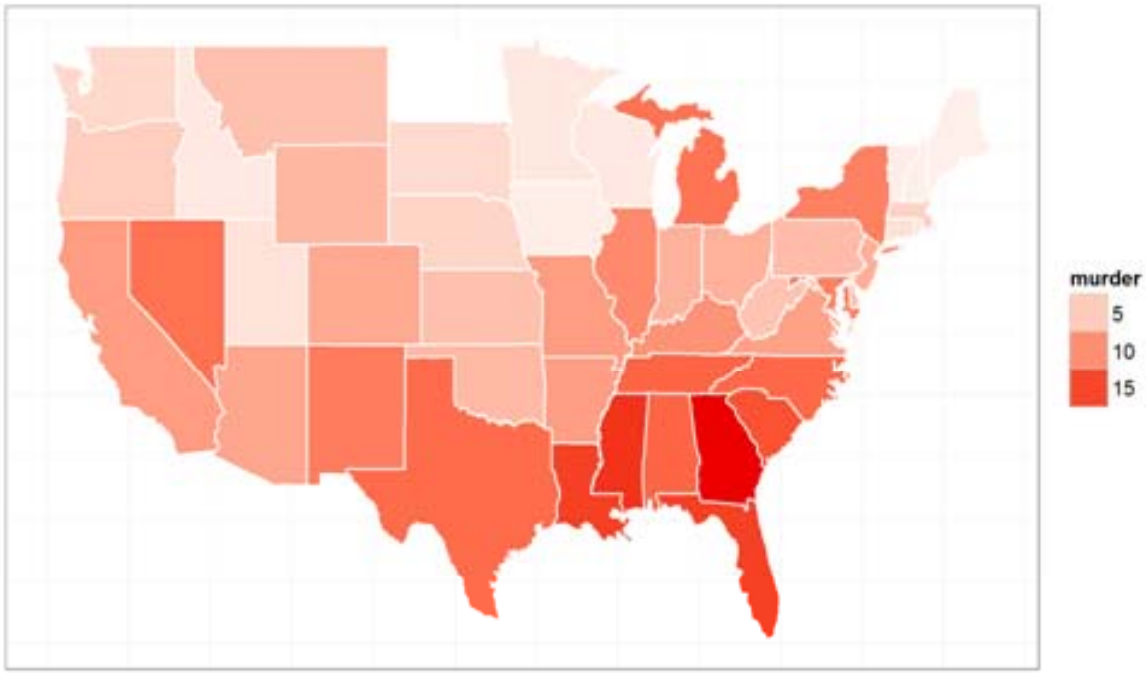
- ggplot2 is a plotting system for R
- based on the《The Grammar of Graphics》
- which tries to take the good parts of base and lattice graphics and none of the bad parts
- It takes care of many of the fiddly details that make plotting a hassle
- It becomes easy to produce complex multi-layered graphics



Why we love ggplot2?

- control the plot as abstract layers and make creativity become reality ;
- get used to structural thinking ;
- get beautiful graphics while avoiding complicated details

1973 murder cases in USA



7 Basic Concepts

- Mapping
- Scale
- Geometric
- Statistics
- Coordinate
- Layer
- Facet

Mapping

Mapping controls relations between variables

length	width	depth	trt
2	3	4	a
1	2	1	a
4	5	15	b
9	10	80	b

x	y	colour
2	3	a
1	2	a
4	5	b
9	10	b

Scale

Scale will present mapping on coordinate scales.

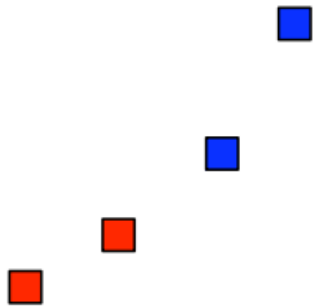
Scale and Mapping is closely related concepts.

x	y	colour
2	3	a
1	2	a
4	5	b
9	10	b

x	y	colour
25	11	red
0	0	red
75	53	blue
200	300	blue

Geometric

Geom means the graphical elements, such as points, lines and polygons.



Geoms

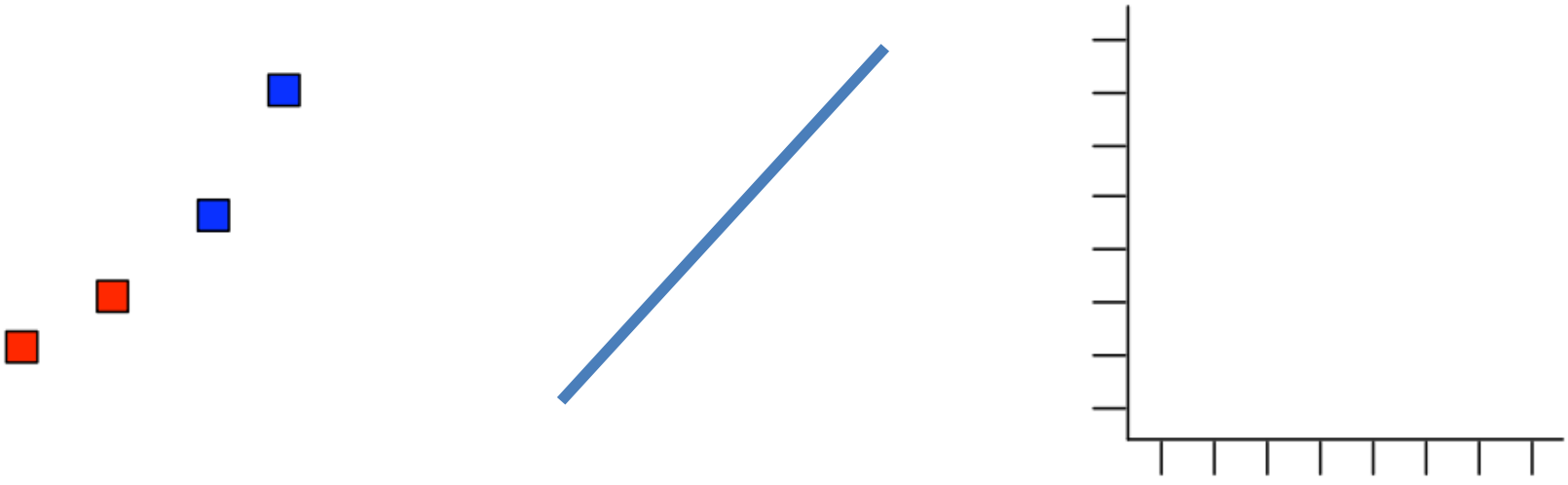
Statistics

Stat enables us to calculate and do statistical analysis based, such as adding a regression line.

Geoms Stat

Coordinate

Cood will affect how we observe graphical elements. Transformation of coordinates is useful.

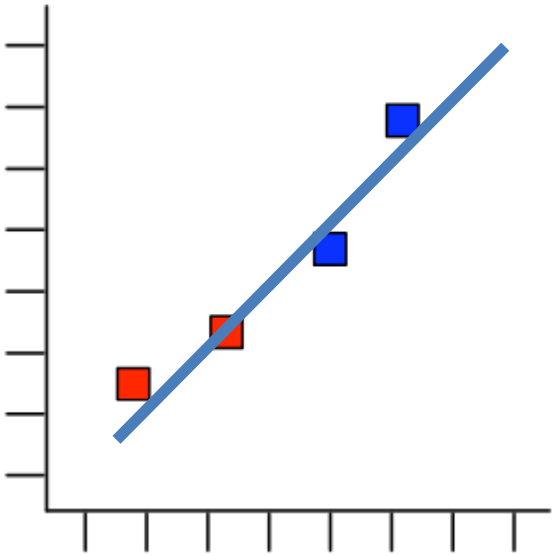


Geoms Stat Coord

Layer

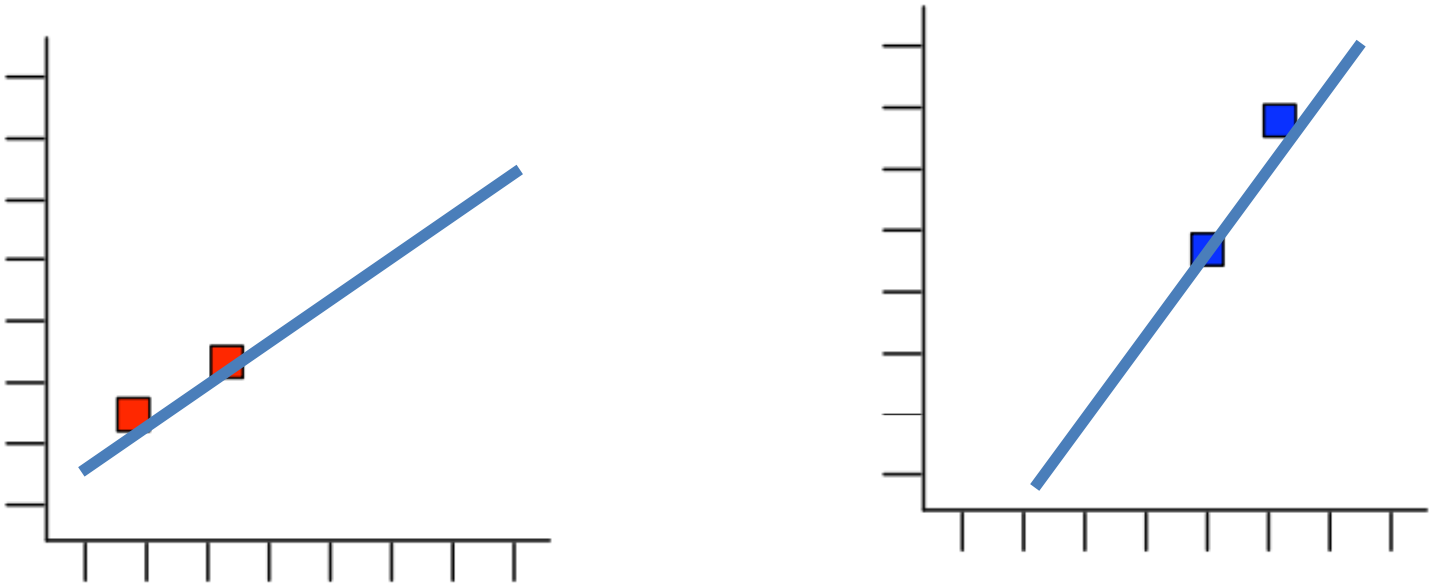
Component: data, mapping, geom, stat

Using layer will allow users to establish plots step by step. It become much easier to modify a plot.



Facet

Facet splits data into groups and draw each group separately. Usually, there is a order.



7 Basic Concepts

- Mapping
- Scale
- Geometric
- Statistics
- Coordinate
- Layer
- Facet

Skill I：Point

Sample data--mpg

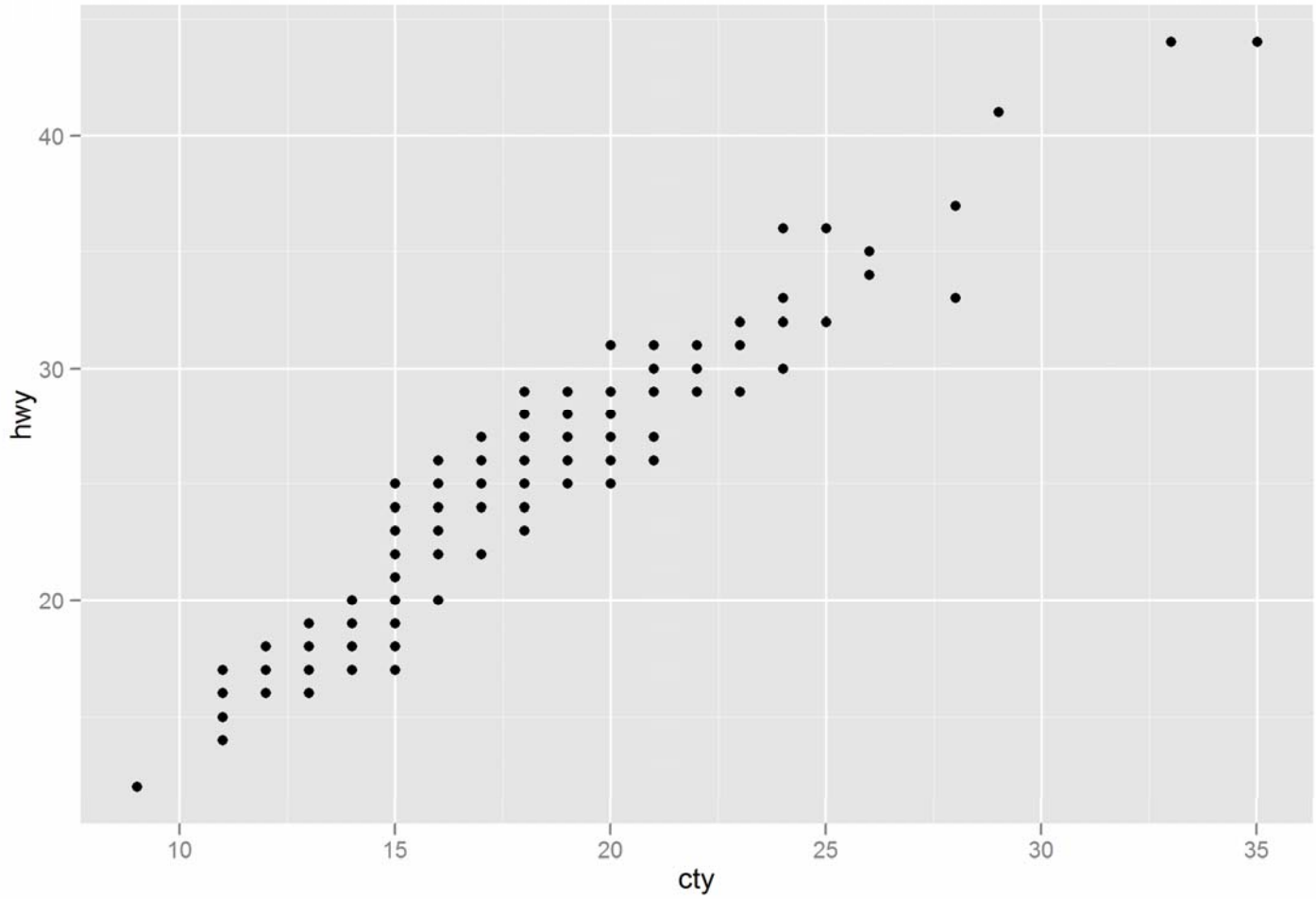
- Fuel economy data from 1999 and 2008 for 38 popular models of car

- Details
- Displ : engine displacement, in litres
- Cyl: number of cylinders
- Trans: type of transmission
- Drv: front-wheel, rear wheel drive, 4wd
- Cty: city miles per gallon
- Hwy: highway miles per gallon

```
> library(ggplot2)
> str(mpg)
'data.frame':	234 obs. of  14 variables:
 $ manufacturer: Factor w/ 15 levels "audi","chevrolet",..:
 $ model       : Factor w/ 38 levels "4runner 4wd",..:
 $ 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
 $ cyl         : int  4 4 4 4 6 6 6 4 4 4 ...
 $ trans       : Factor w/ 10 levels "auto(av)","auto(l3)",..:
 $ drv         : Factor w/ 3 levels "4","f","r":
 $ 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",..:
 $ class       : Factor w/ 7 levels "2seater","compact",..:
```

aesthetics

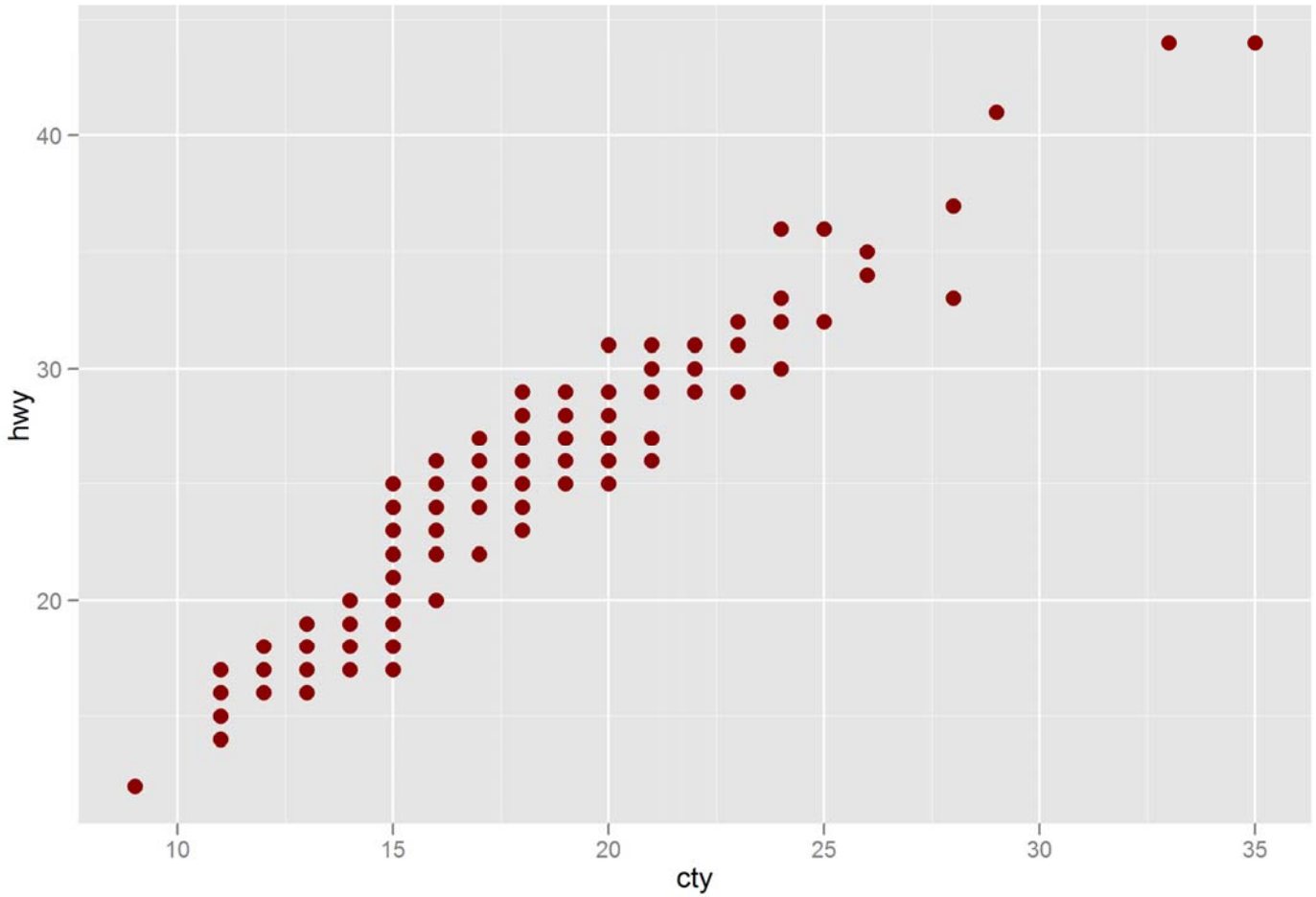
```
p <- ggplot(data=mpg, mapping=aes(x=cty, y=hwy))
p + geom_point()
```



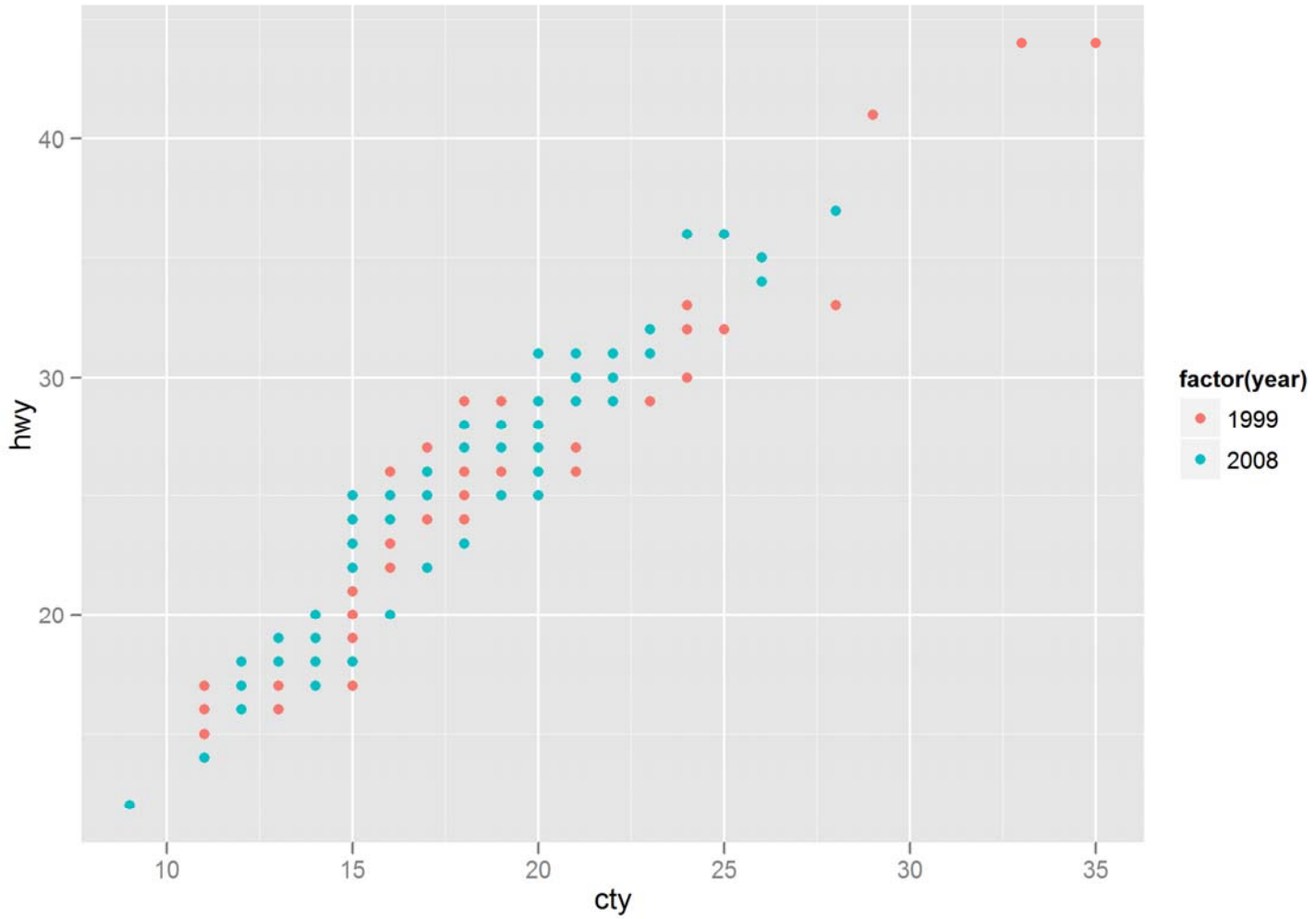
```
> summary(p)
data: manufacturer, model, displ, year, cyl, trans, drv, cty, hwy,
fl, class [234x11]
mapping: x = cty, y = hwy
faceting: facet_null()

> summary(p+geom_point())
data: manufacturer, model, displ, year, cyl, trans, drv, cty, hwy,
  fl, class [234x11]
mapping:  x = cty, y = hwy
faceting: facet_null()
-----------------------------------
geom_point: na.rm = FALSE
stat_identity:
position_identity: (width = NULL, height = NULL)
```

p + geom_point(color='red4',size=3)

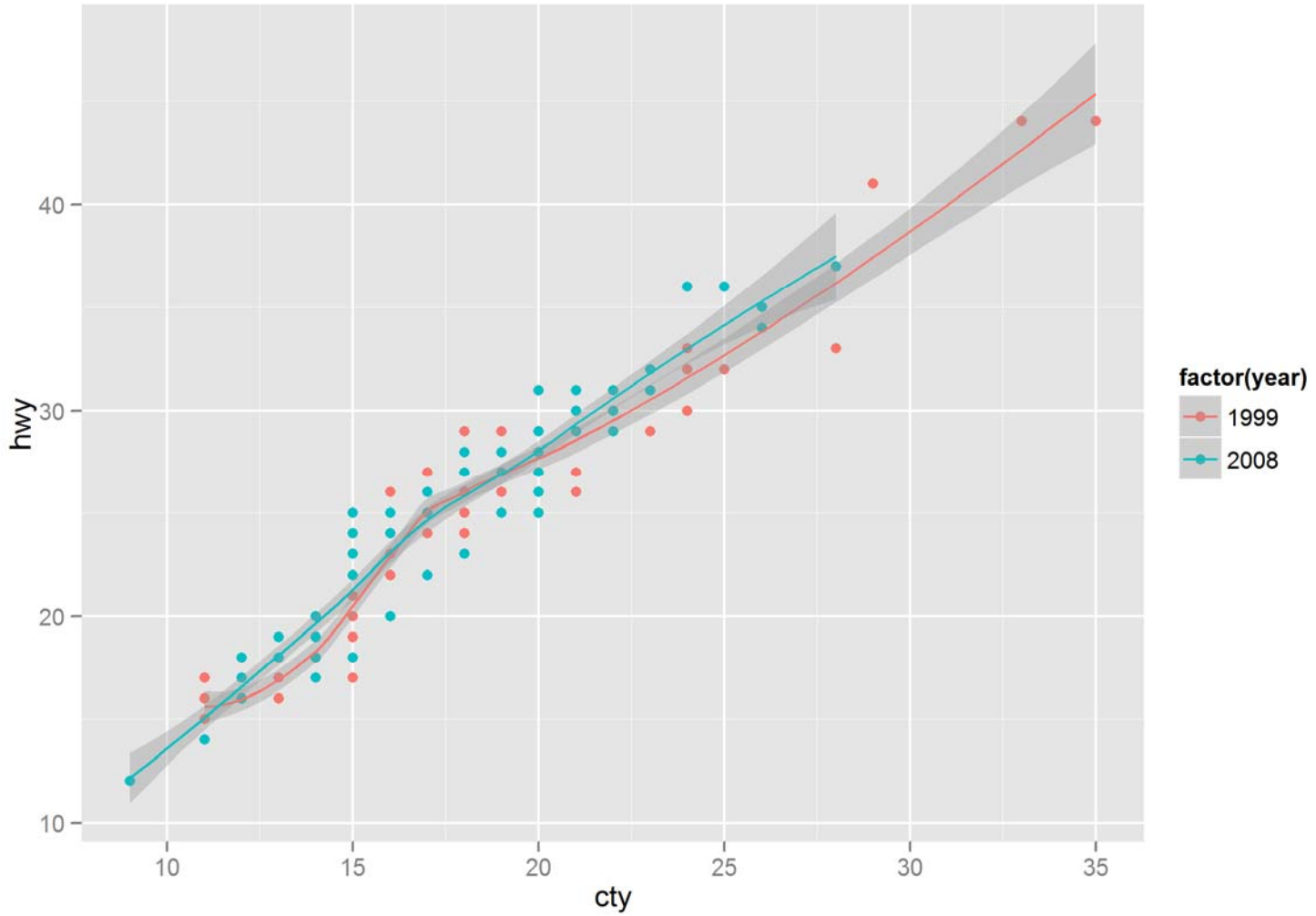


```
# add one more layer--color
p <- ggplot(mpg,aes(x=cty,y=hwy,colour=factor(year)))
p + geom_point()
```

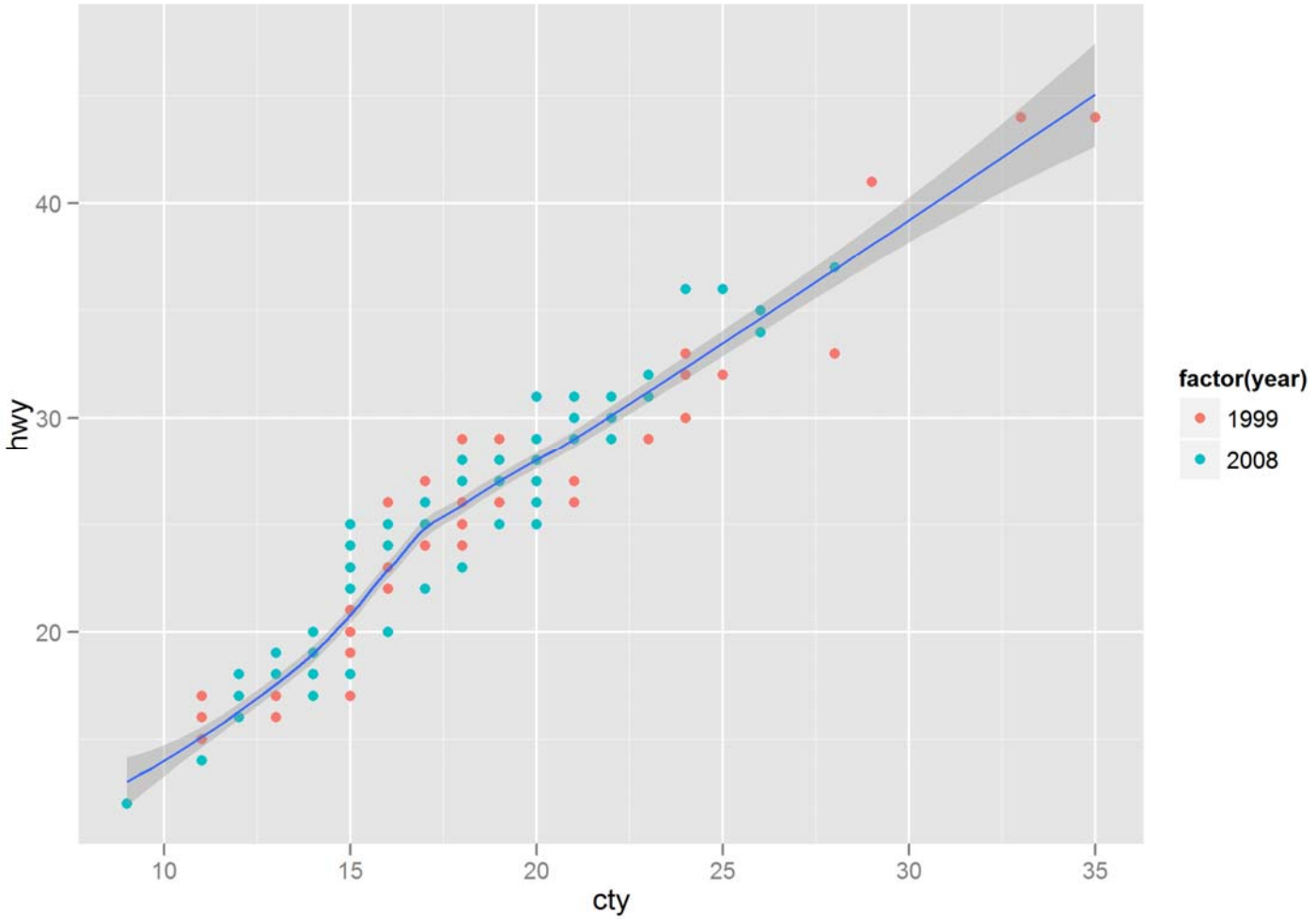


add one more stat (loess: local partial polynomial regression)

> p + geom_point() + stat_smooth()



```
p <- ggplot(data=mpg, mapping=aes(x=cty,y=hwy))
p + geom_point(aes(colour=factor(year)))+
    stat_smooth()
```



Two equally ways to draw

```
p  <-  ggplot(mpg, aes(x=cty,y=hwy))
p  + geom_point(aes(colour=factor(year)))+
       stat_smooth()
```

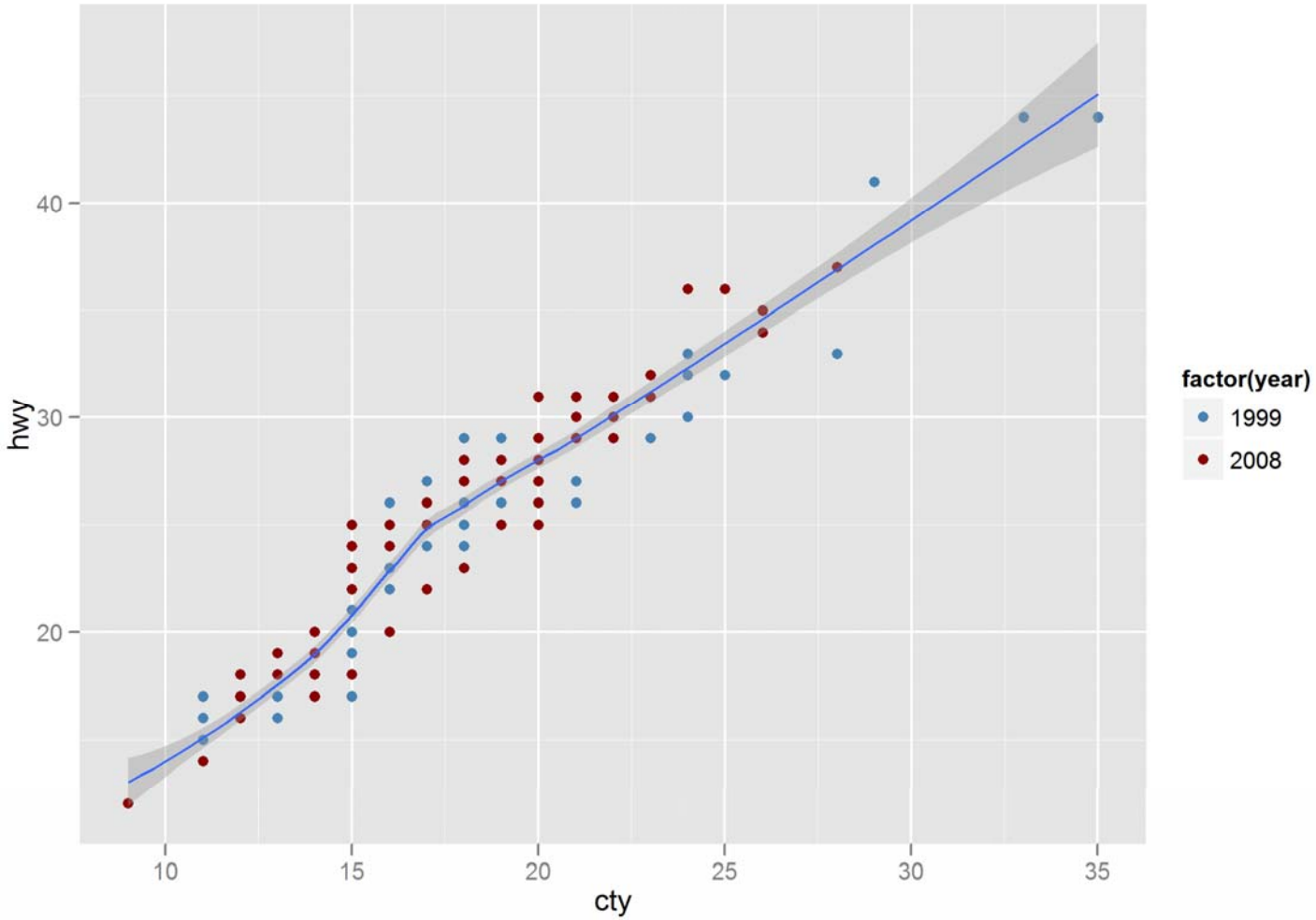
```
d <- ggplot() +
       geom_point(data=mpg, aes(x=cty, y=hwy, colour=factor(year)))+
       stat_smooth(data=mpg, aes(x=cty, y=hwy))
print(d)
```

Beside the “white paper” canvas, we will find geom and stat canvas.

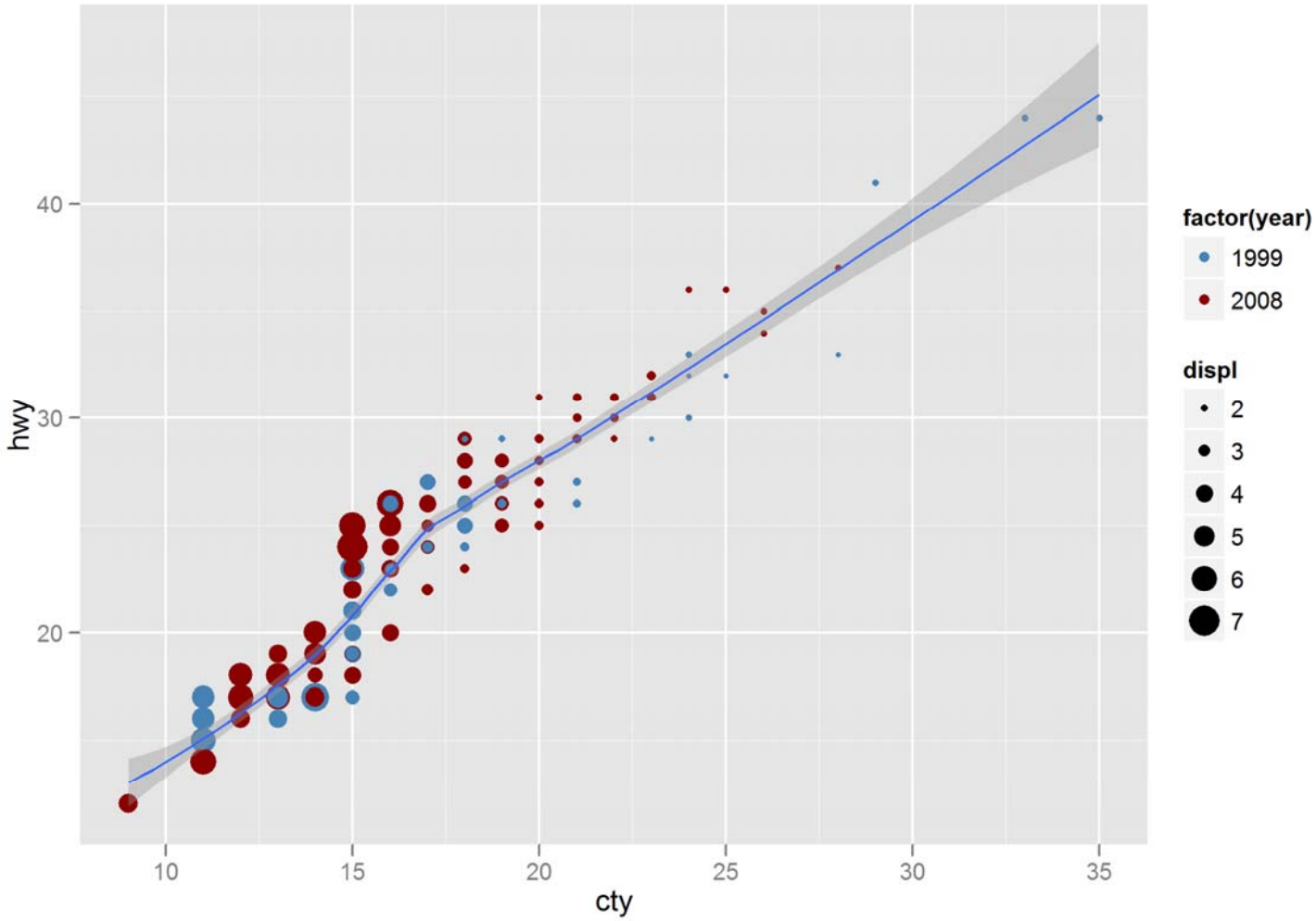
```
> summary(d)
data: [0x0]
faceting: facet_null()
---------------------------------
mapping: x = cty, y = hwy, colour = factor(year)
geom_point: na.rm = FALSE
stat_identity:
position_identity: (width = NULL, height = NULL)

mapping: x = cty, y = hwy
geom_smooth:
stat_smooth: method = auto, formula = y ~ x, se = TRUE,
n = 80, fullrange = FALSE, level = 0.95, na.rm = FALSE
position_identity: (width = NULL, height = NULL)
```

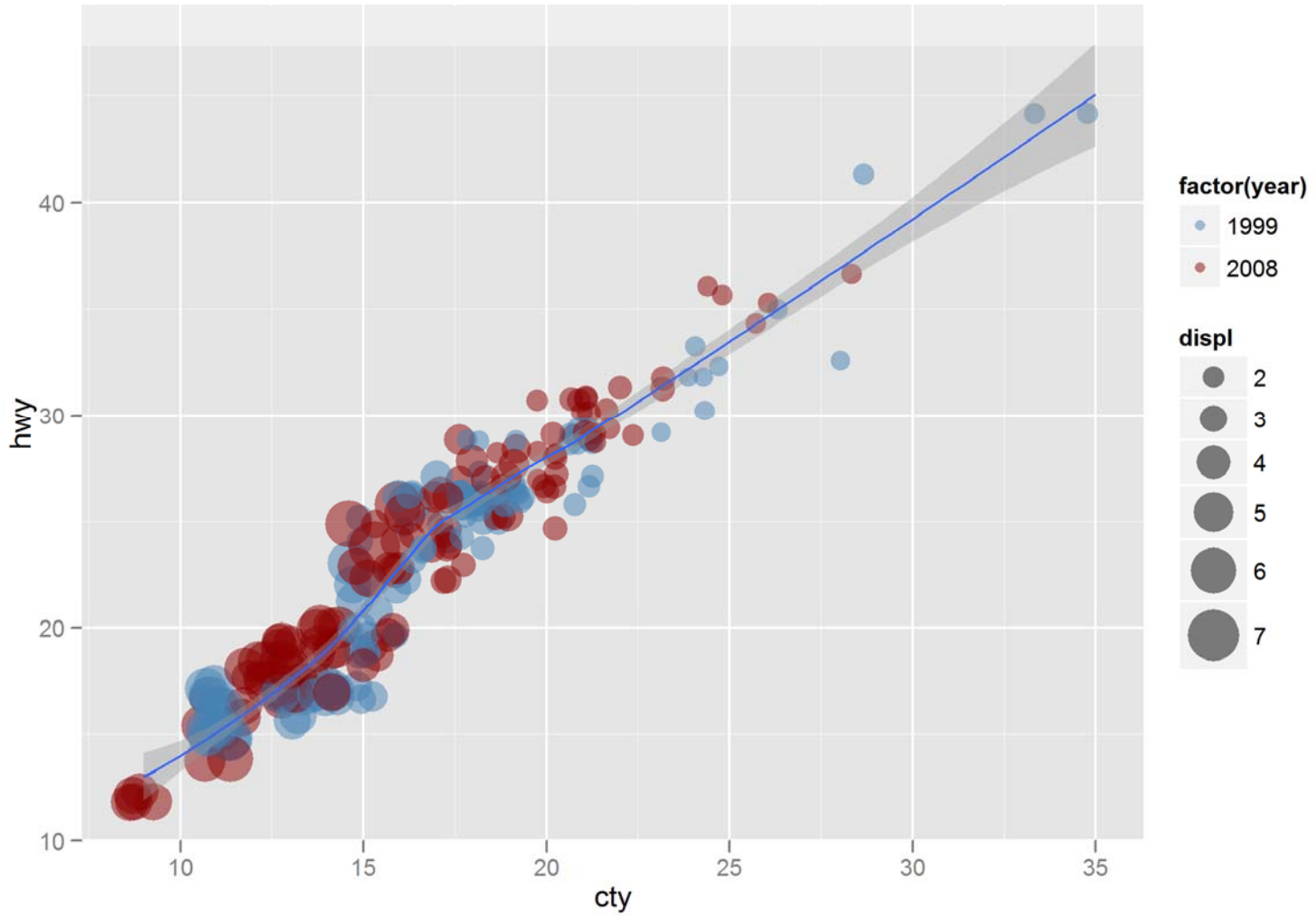
```
# Using scale() function, we can control color of scale.
p + geom_point(aes(colour=factor(year)))+
    stat_smooth()+
    scale_color_manual(values =c('steelblue','red4'))
```



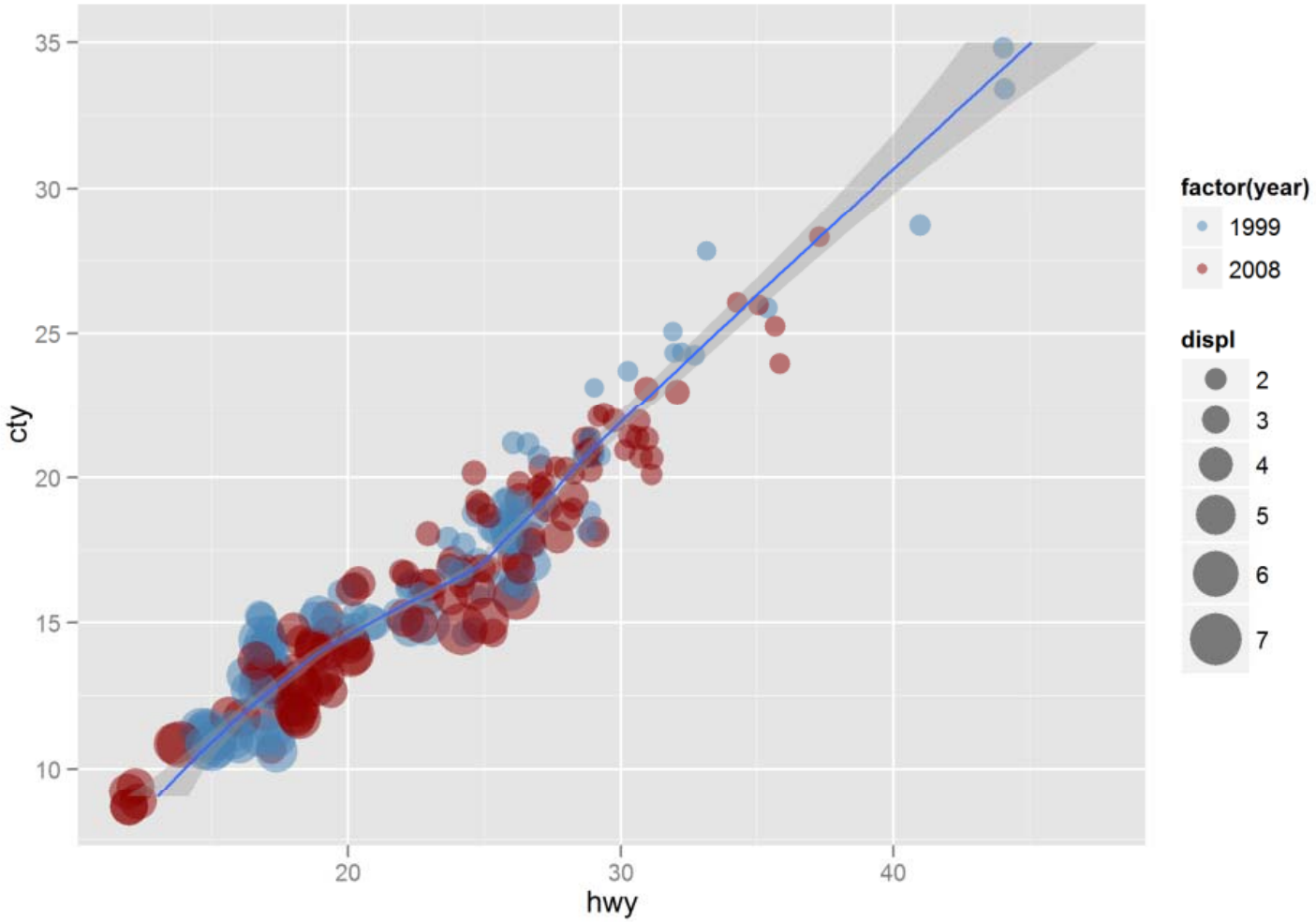
```
# We can map  “displ”  to the size of point
p + geom_point(aes(colour=factor(year),size=displ))+
    stat_smooth()+
    scale_color_manual(values =c('steelblue','red4'))
```



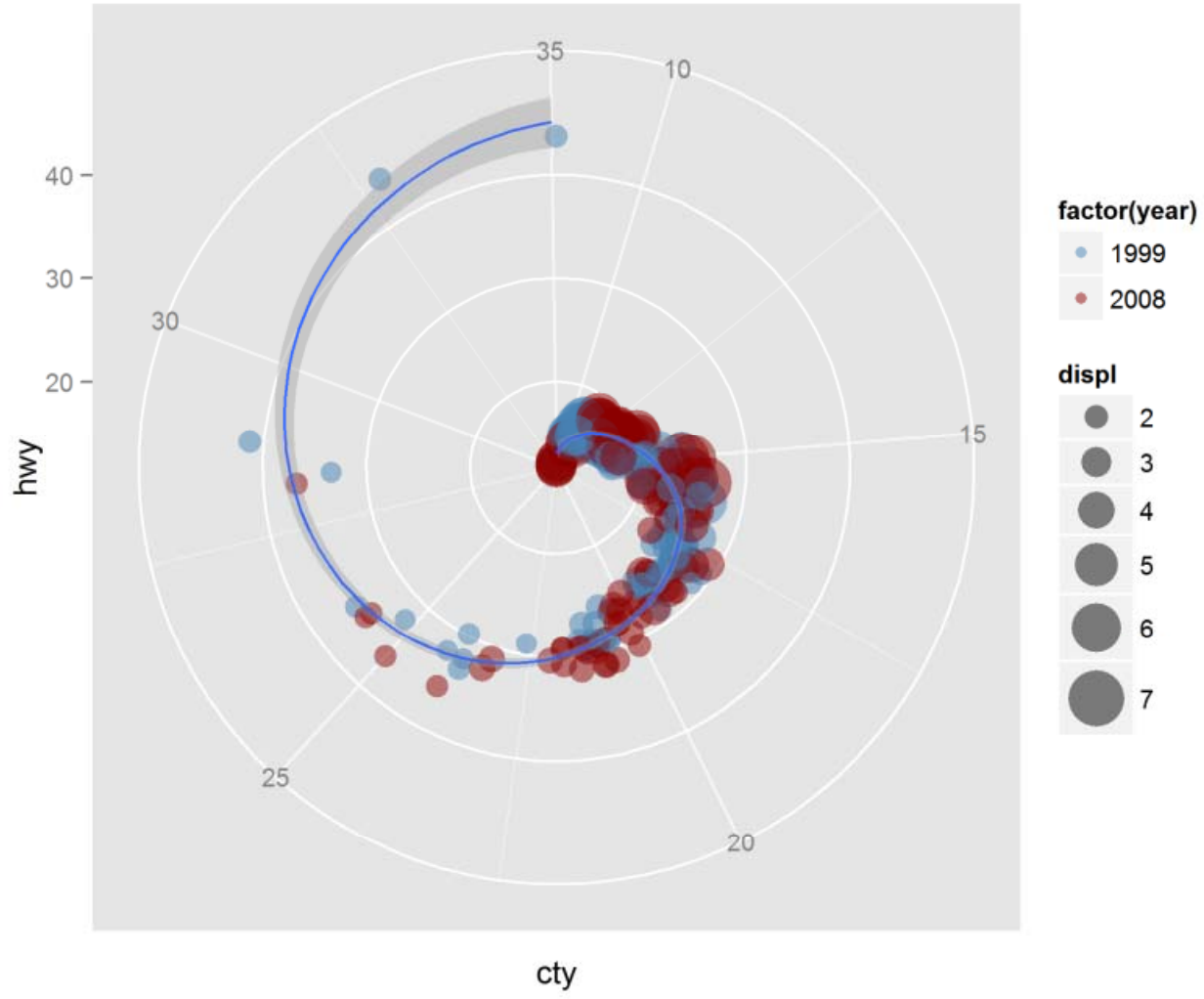
```
# We solve the problem with overlapping and point being too small
p + geom_point(aes(colour=factor(year),size=displ), alpha=0.5,position = "jitter")+
    stat_smooth()+
    scale_color_manual(values =c('steelblue','red4'))+
    scale_size_continuous(range = c(4, 10))
```



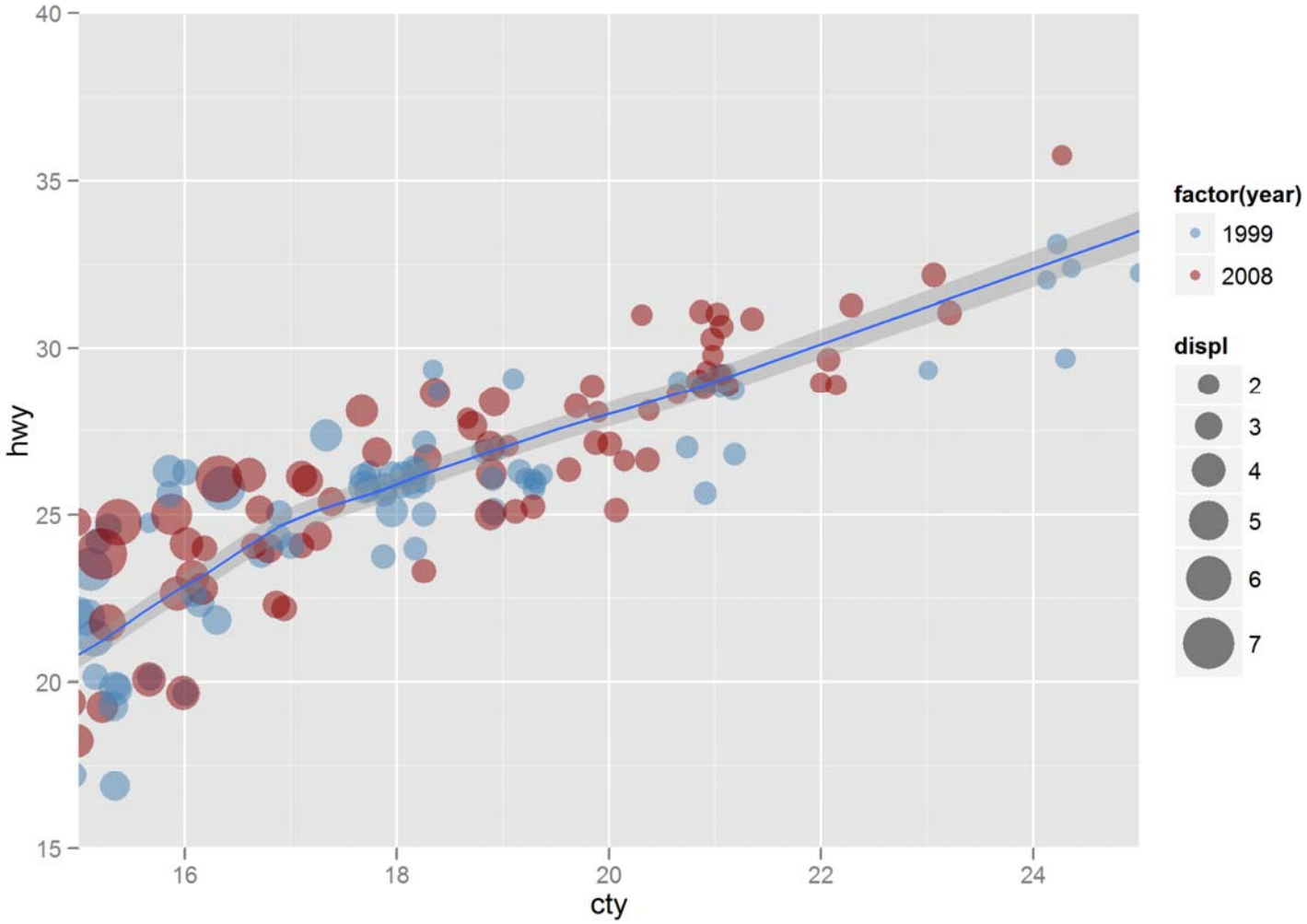
```
# We change the coordinate system.
p + geom_point(aes(colour=factor(year),size=displ), alpha=0.5,position = "jitter")+
    stat_smooth()+
    scale_color_manual(values =c('steelblue','red4'))+
    scale_size_continuous(range = c(4, 10)) +    coord_flip()
```



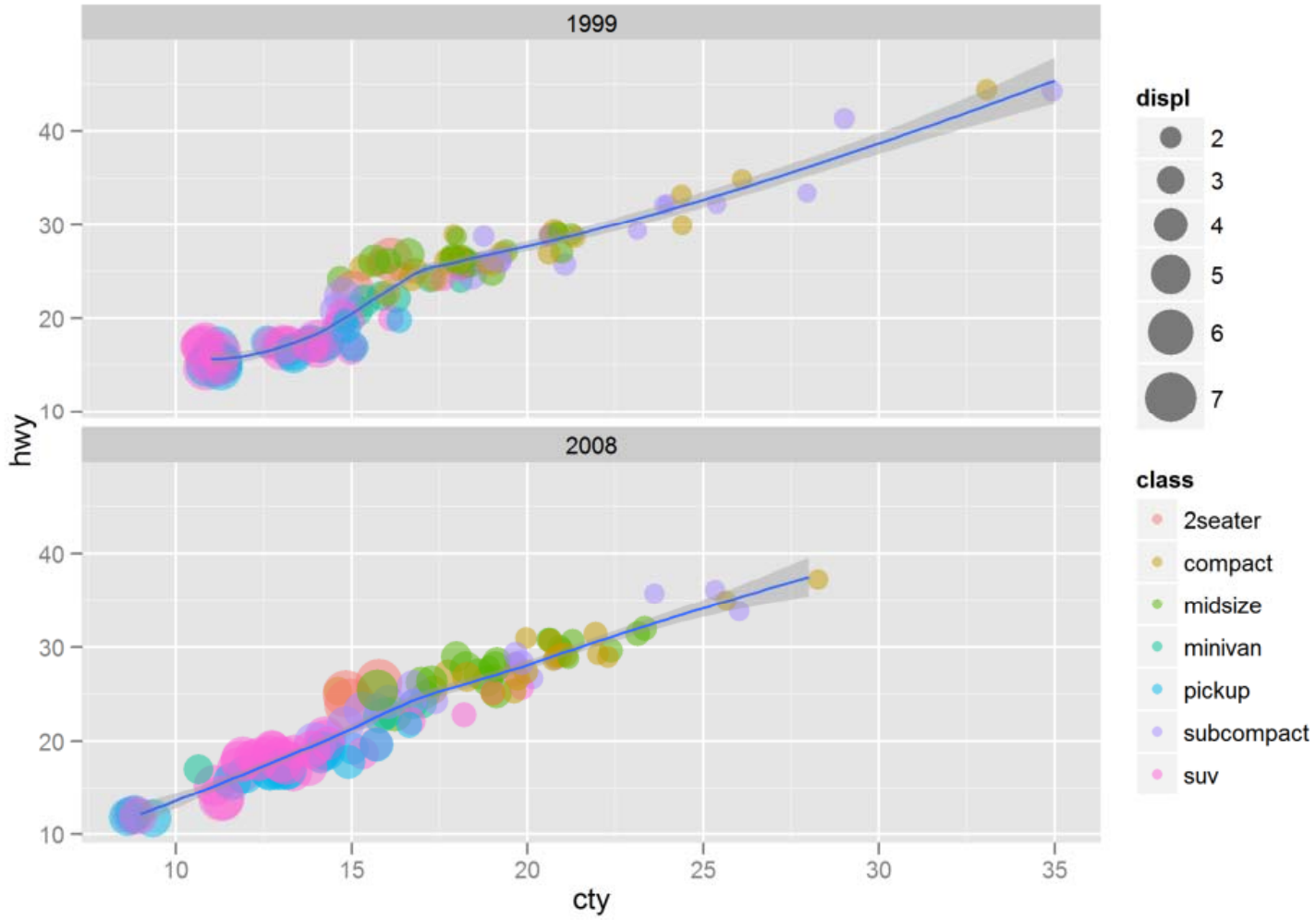
```
p + geom_point(aes(colour=factor(year),size=displ),
               alpha=0.5,position = "jitter")+
   stat_smooth()+
   scale_color_manual(values =c('steelblue','red4'))+
   scale_size_continuous(range = c(4, 10))  +   coord_polar()
```



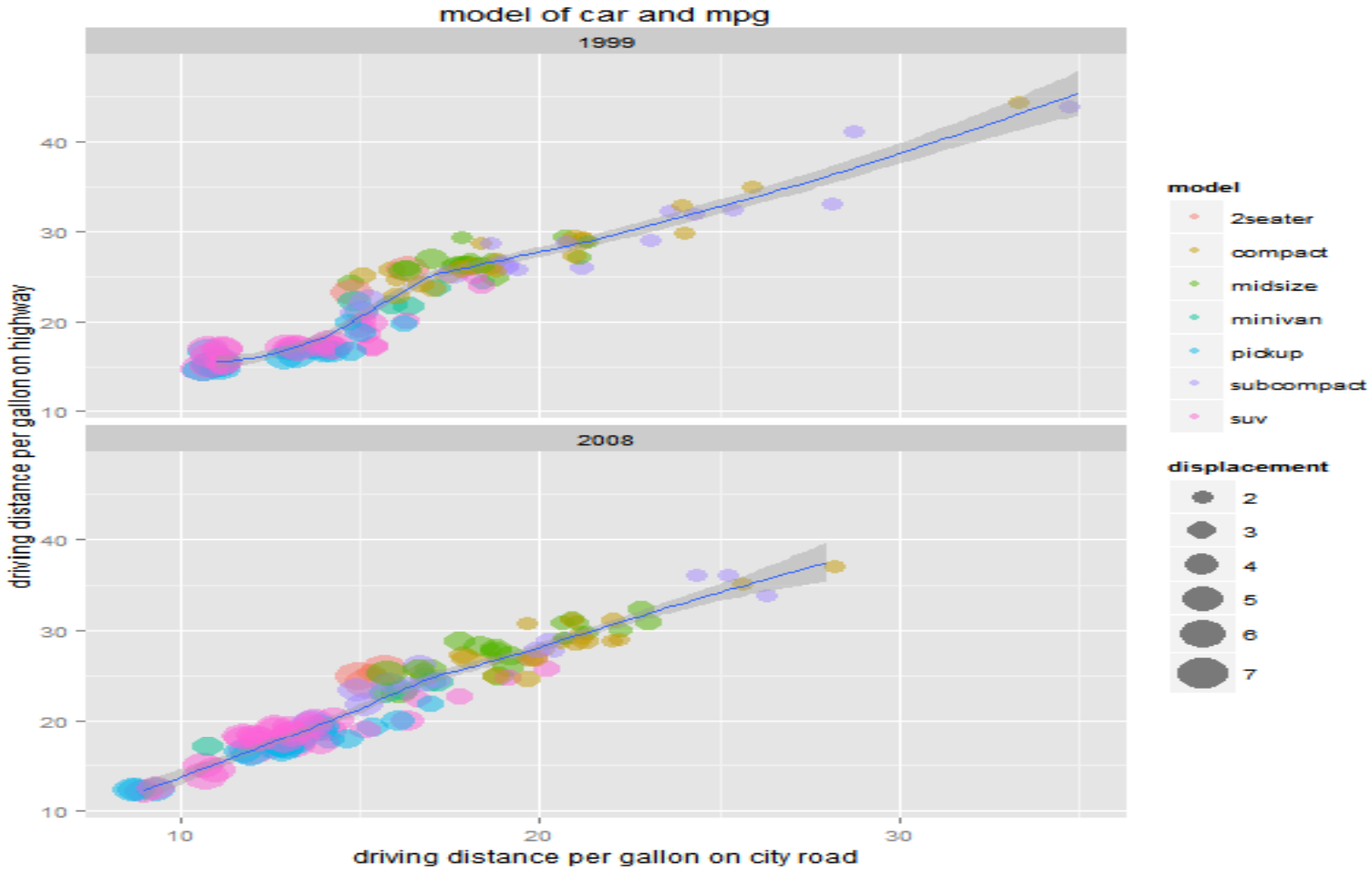
```
p + geom_point(aes(colour=factor(year),size=displ),
               alpha=0.5,position = "jitter")  +    stat_smooth()+
   scale_color_manual(values =c('steelblue','red4'))+
   scale_size_continuous(range = c(4, 10))+
   coord_cartesian(xlim = c(15, 25), ylim=c(15,40))
```



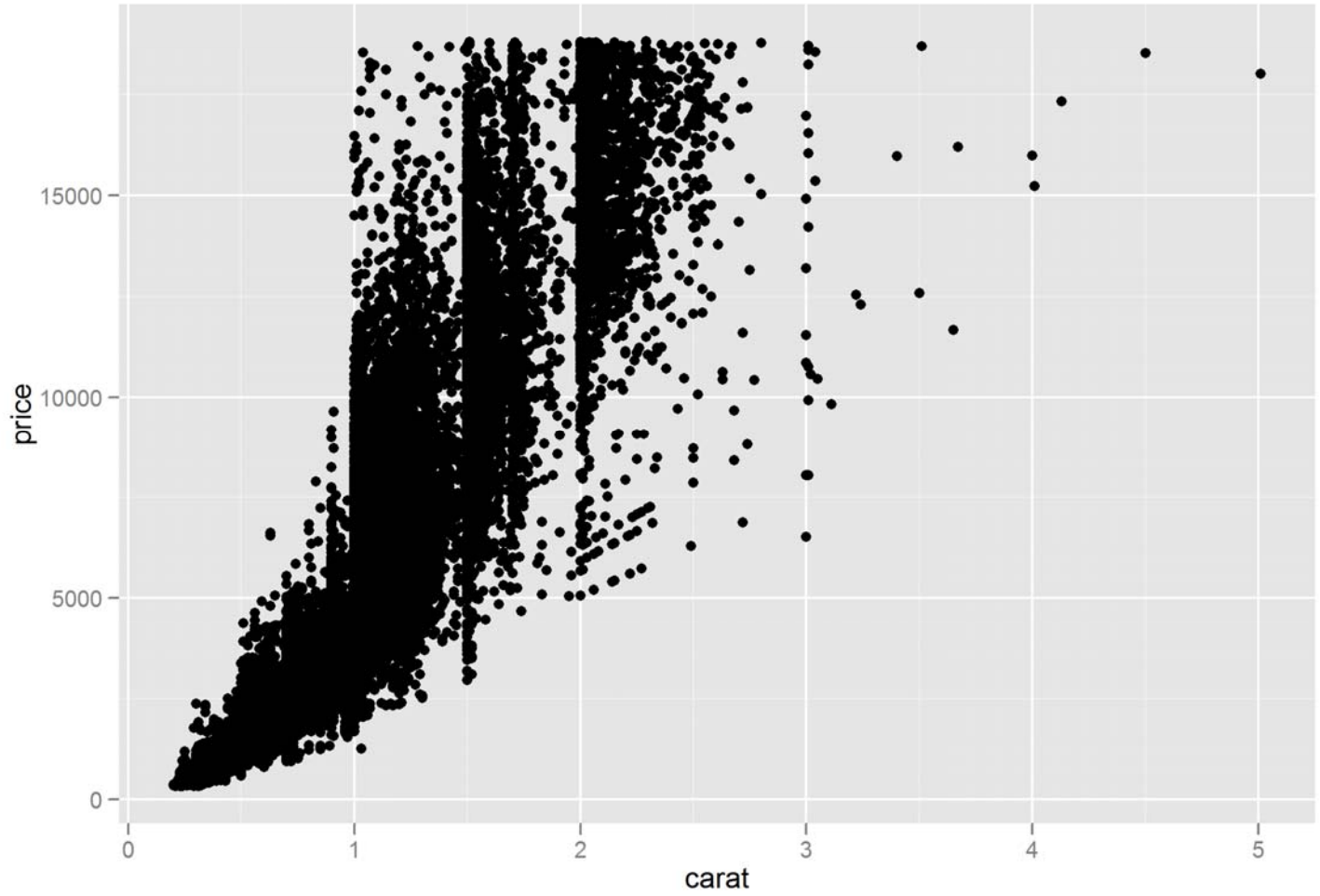
```
# Using facet() function, we now split data and draw them by group
p + geom_point(aes(colour=class,size=displ),
   alpha=0.5,position = "jitter")+
   stat_smooth()+
   scale_size_continuous(range = c(4, 10))+
   facet_wrap(~ year,ncol=1)
```



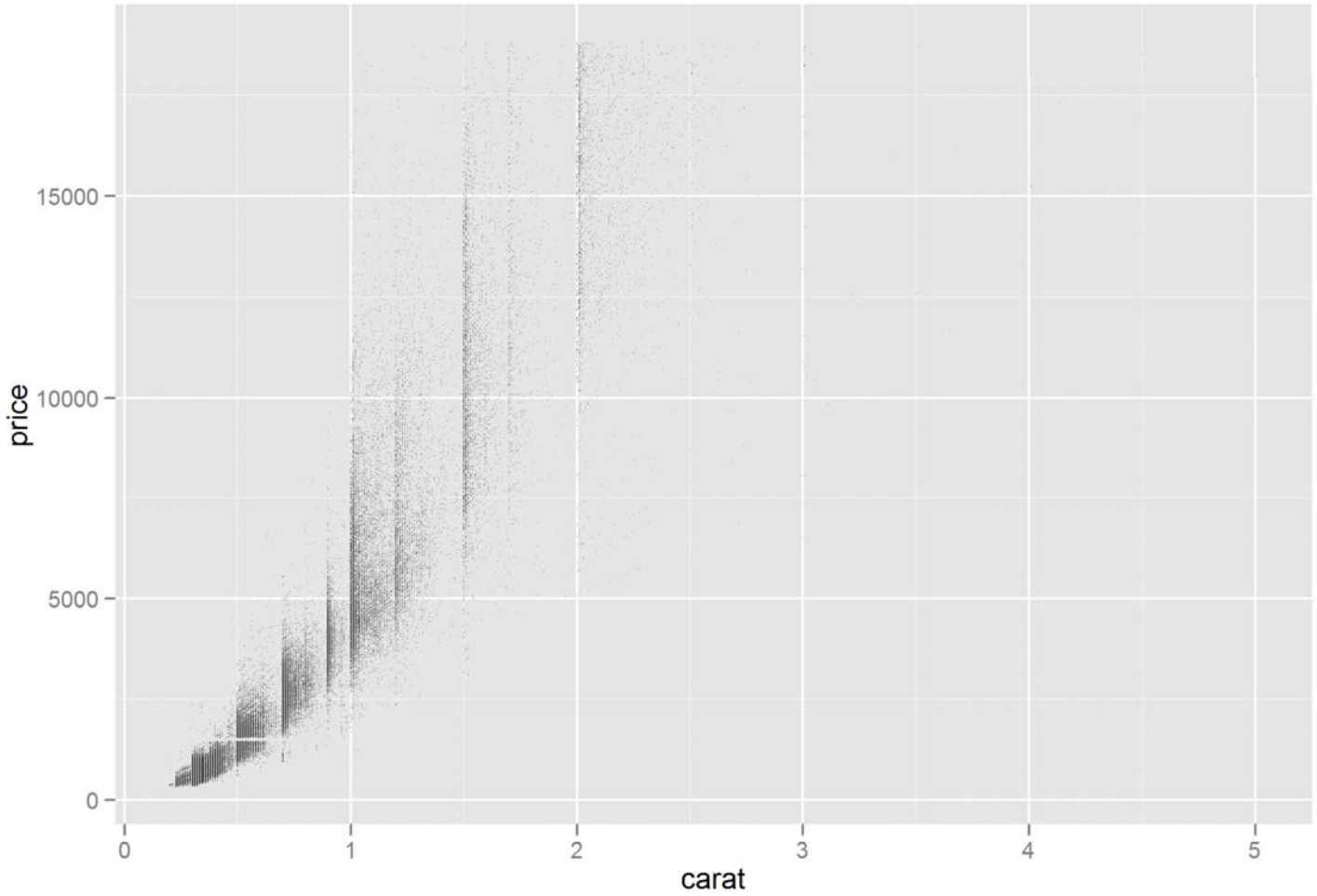
```
# Add plot name and specify all information you want to add
    p <- ggplot(mpg, aes(x=cty,y=hwy))
    p + geom_point(aes(colour=class,size=displ),
              alpha=0.5,position = "jitter")+  stat_smooth()+
    scale_size_continuous(range = c(4, 10))+
    facet_wrap(~ year,ncol=1)  +    opts(title='model of car and mpg')+
    labs(y='driving distance per gallon on highway', x='driving distance per gallon on city road',
          size='displacement', colour ='model')
```



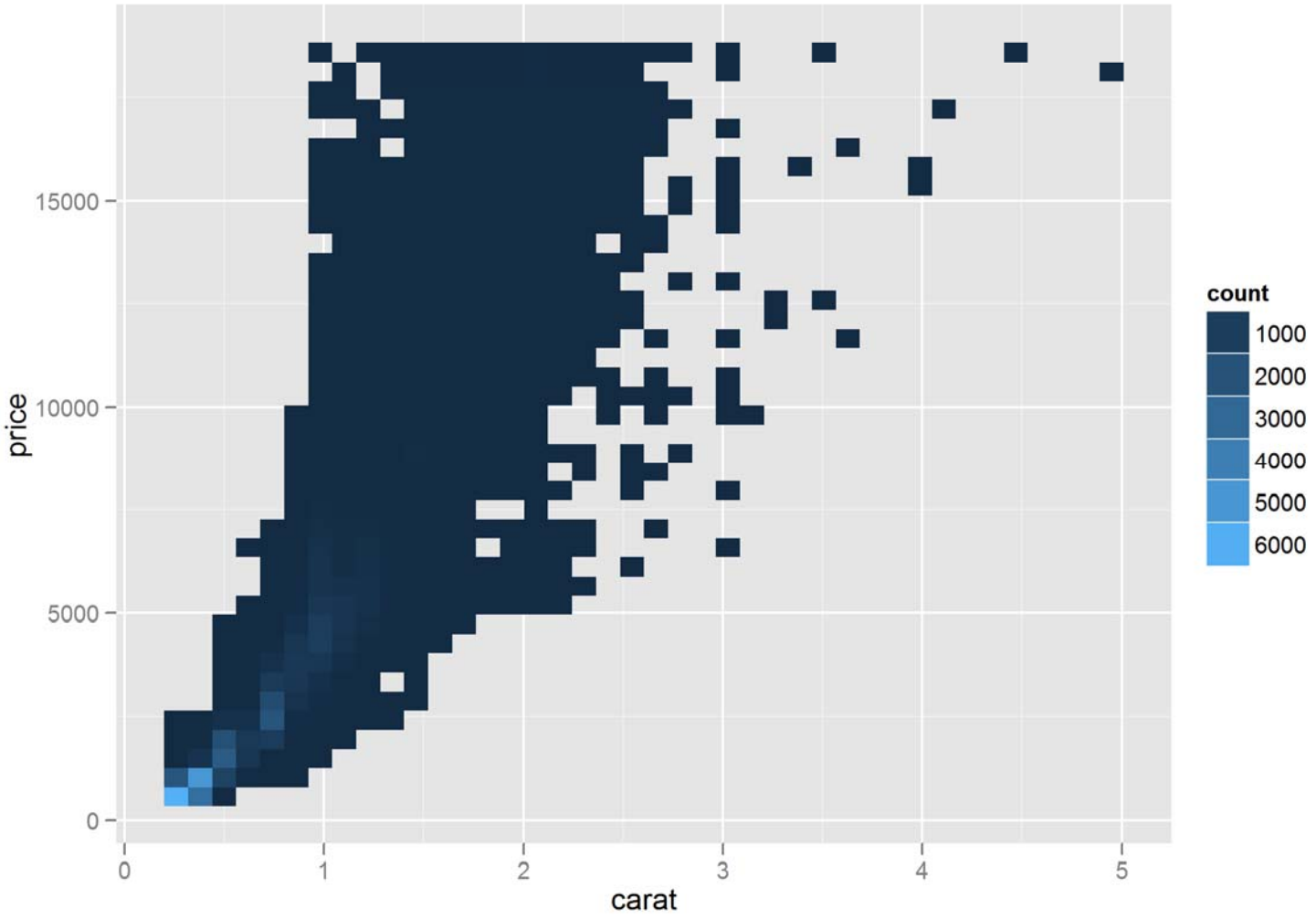
```
# scatter plot for diamond dataset
p <- ggplot(diamonds,aes(carat,price))
p + geom_point()
```



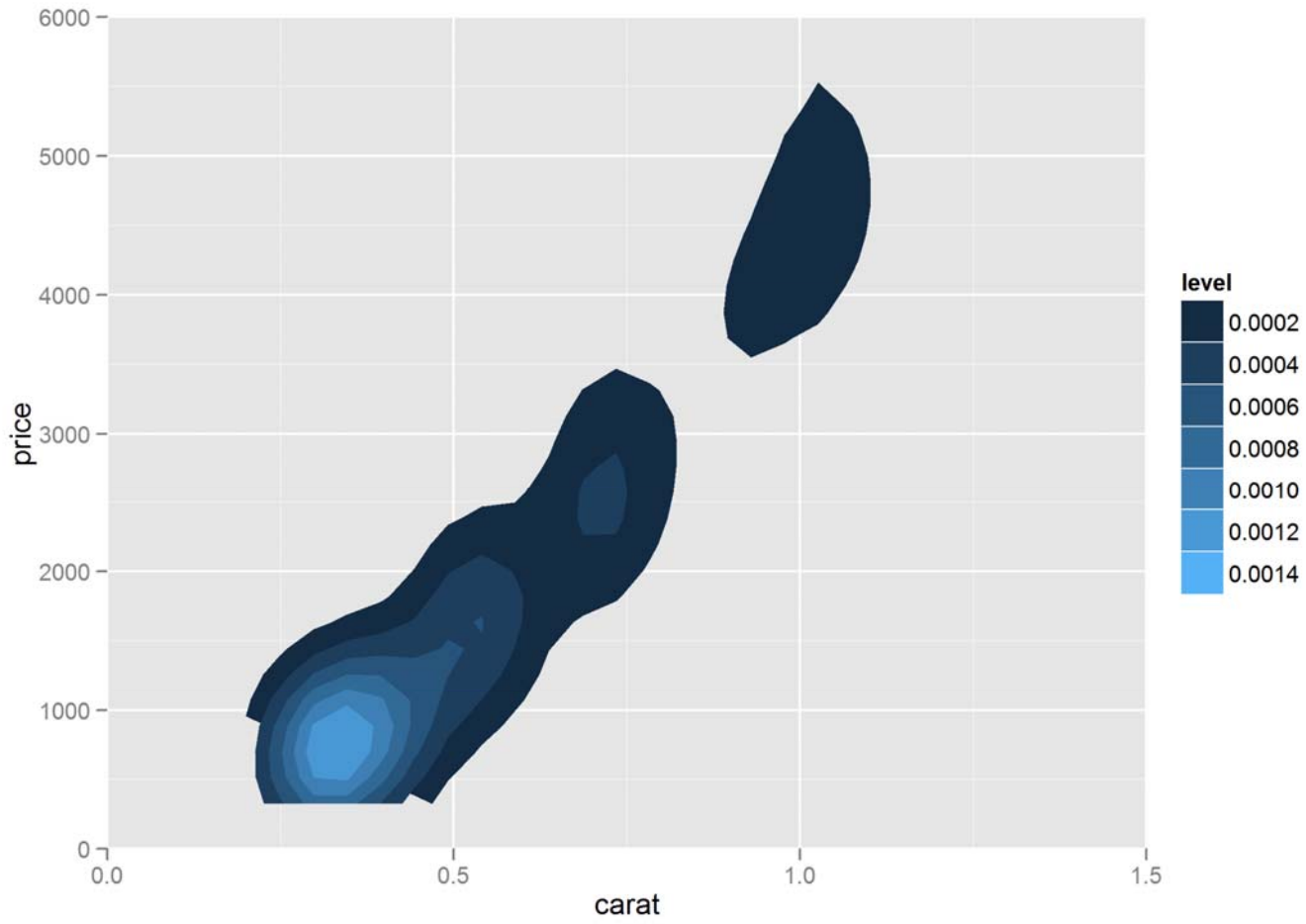
```
# use transparency and small size points
p + geom_point(size=0.1,alpha=0.1)
```



use bin chart to observe intensity of points
p + stat_bin2d(bins = 40)



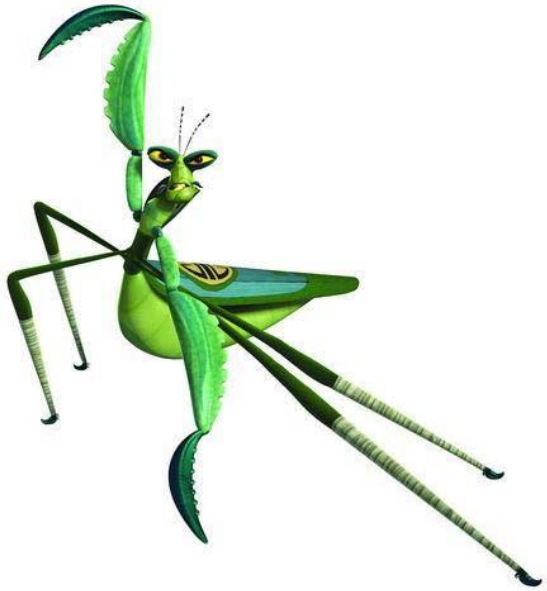
```
# estimate data dentisy
p + stat_density2d(aes(fill = ..level..), geom="polygon") +
    coord_cartesian(xlim = c(0, 1.5),ylim=c(0,6000))
```



Skill II : Bar

Skill III : Histogram

Skill IV：Line



Skill V : Tile

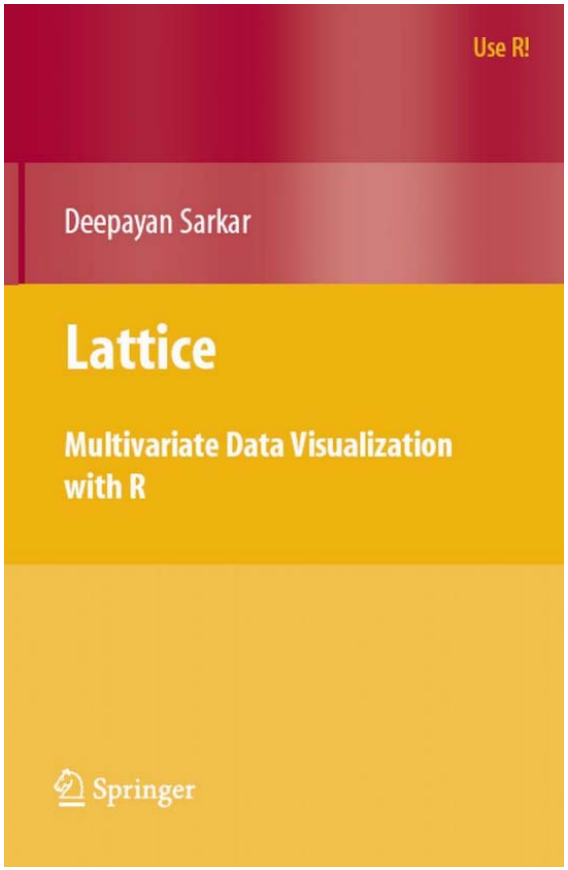
Skill VI : Map

Resources

http://learnr.wordpress.com

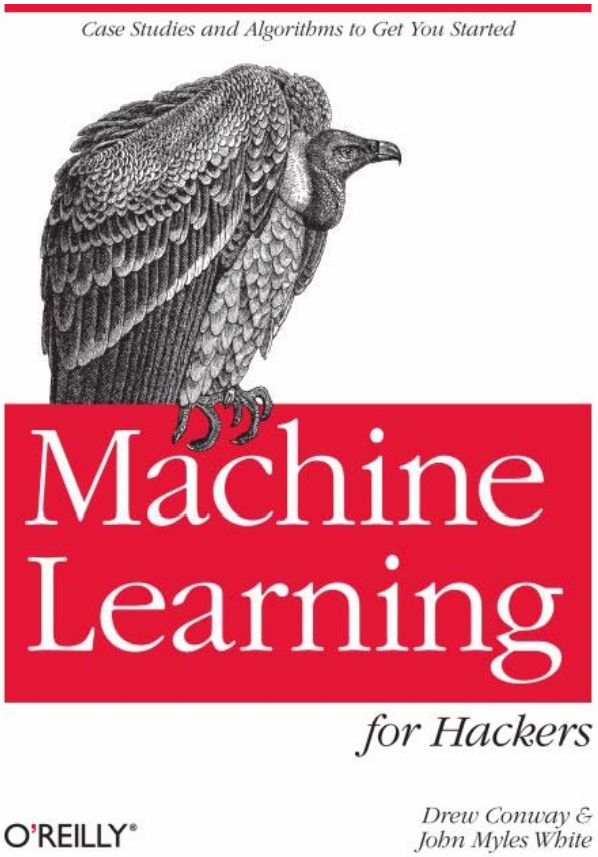
Redraw all the lattice graph

by ggplot2



Resources

All the examples are done by ggplot2.



Resources

- http://wiki.stdout.org/rcookbook/Graphs/
- http://r-blogger.com
- http://Stackoverflow.com
- http://xccds1977.blogspot.com
- http://r-ke.info/
- http://www.youtube.com/watch?v=vnVJJYi1mbw

Thank you! Come back for more!

Sign up at: www.meetup.com/nyc-open-data
Give feedback at: www.bit.ly/nycopen