数据可视化

基础篇

本章内容

- 基本图形：条形图、饼图、直方图、核密度图、箱线图、点图
- 中级绘图：散点图、气泡图、折线图、相关图、马赛克图

基本图形

- 条形图
- 饼图
- 直方图
- 核密度图
- 箱线图
- 点图

条形图（Bar plots）

- 条形图通过垂直的或水平的条形展示了类别型变量的分布（频数）。函数barplot()的最简单用法是：

```
barplot(height)
```

- 其中的height是一个向量或一个矩阵。
- 数据源：探索类风湿性关节炎新疗法研究的结果。数据已包含在随vcd包分发的Arthritis数据框中。

```
install.packages("vcd")
```

简单的条形图（1/2）

table(): 使用 N 个类别型变量（因子）创建一个 N 维列联表（即频数表）。（注：第5章重点内容）

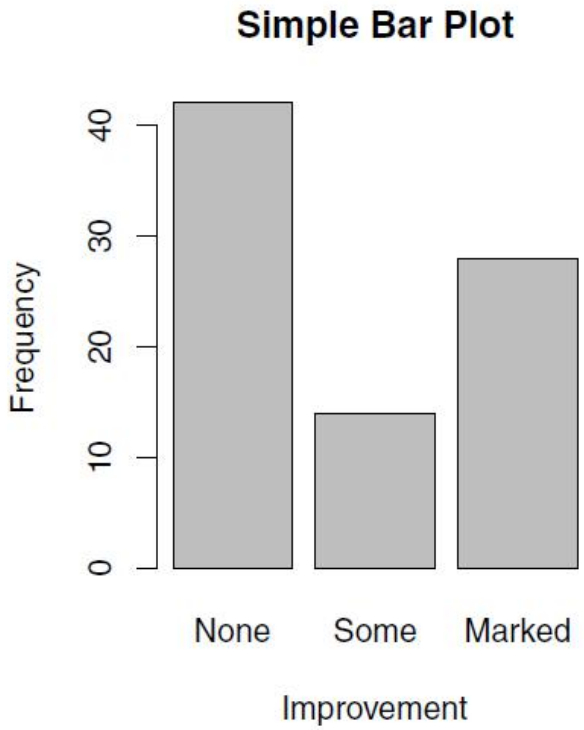
```
library(vcd)
counts <- table(Arthritis$Improved) #准备数据
counts
# vertical barplot
barplot(counts,
        main="Simple Bar Plot",
        xlab="Improvement", ylab="Frequency")
# horizontal bar plot
barplot(counts,
        main="Horizontal Bar Plot",
        xlab="Frequency", ylab="Improvement",
        horiz=TRUE)
```

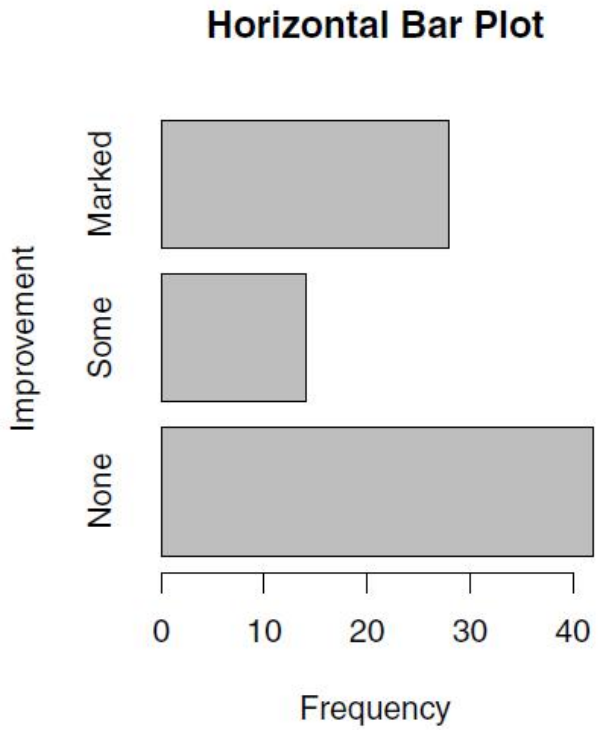
类风湿性关节炎

部分改善	改善	
None	Some	Marked
42	14	28

简单的条形图（2/2）

```
None    Some Marked
  42      14      28
```





堆砌条形图和分组条形图（1/2）

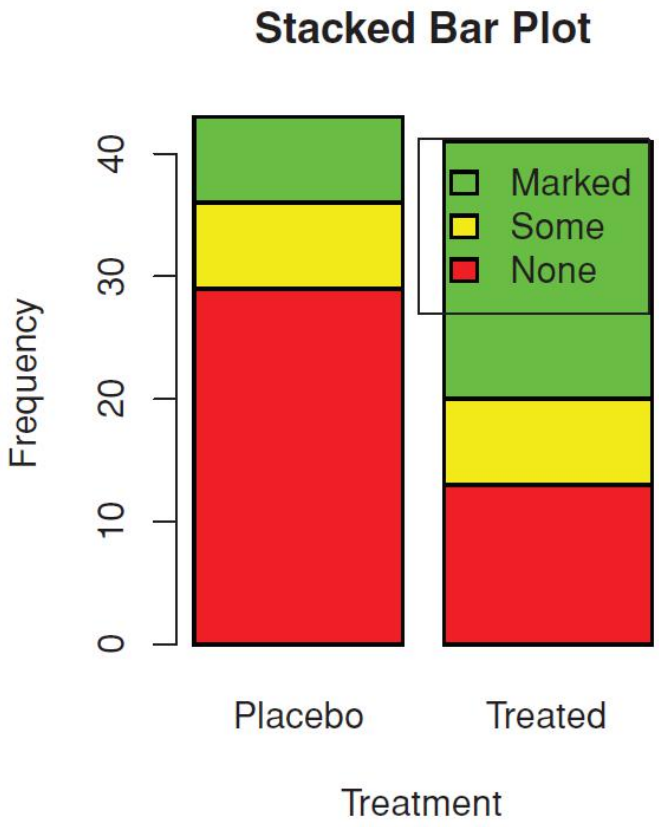
```
library(vcd)
counts <- table(Arthritis$Improved, Arthritis$Treatment)
# stacked barplot
barplot(counts,
        main="Stacked Bar Plot",
        xlab="Treatment", ylab="Frequency",
        col=c("red", "yellow","green"),
        legend=rownames(counts))
# grouped barplot
barplot(counts,
        main="Grouped Bar Plot",
        xlab="Treatment", ylab="Frequency",
        col=c("red", "yellow", "green"),
        legend=rownames(counts), beside=TRUE)
```

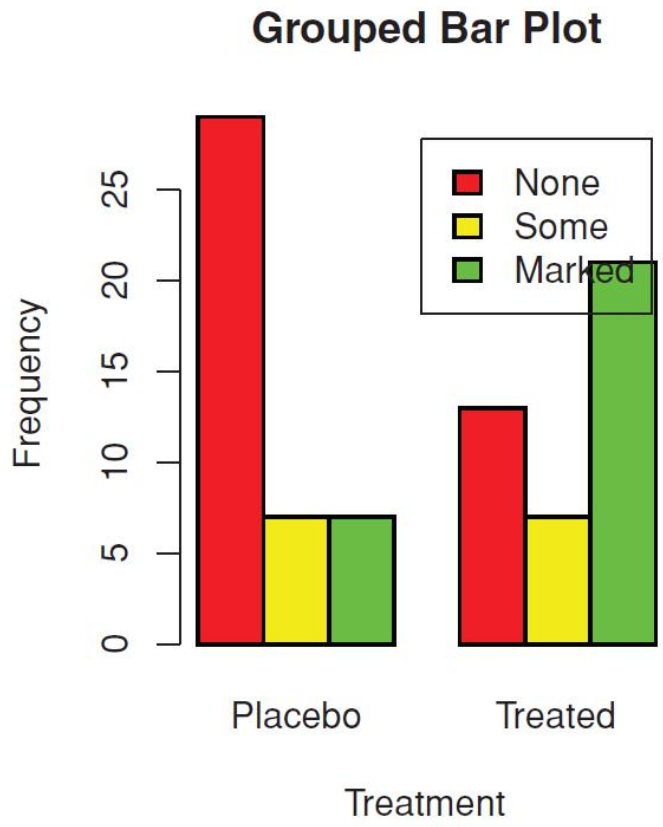
治疗效果 安慰剂 用本方法治疗

Improved	Placebo	Treated
None	29	13
Some	7	7
Marked	7	21

堆砌条形图和分组条形图（2/2）

Improved	Placebo	Treated
None	29	13
Some	7	7
Marked	7	21





均值条形图

- 除了基于计数数据或频率数据，还**可以使用数据整合函数结合barplot()函数，来创建表示均值、中位数、标准差等的条形图**。
- 例，美国各地区平均文盲率排序的条形图

```
states <- data.frame(state.region, state.x77)#R自带数据
means <- aggregate(states$Illiteracy, by=list(state.region), FUN=mean) #求均值
means <- means[order(means$x),]  #排序
barplot(means$x, names.arg=means$Group.1)
title("Mean Illiteracy Rate")
```

aggregate：分类汇总（见第三章）

饼图（Pie charts）

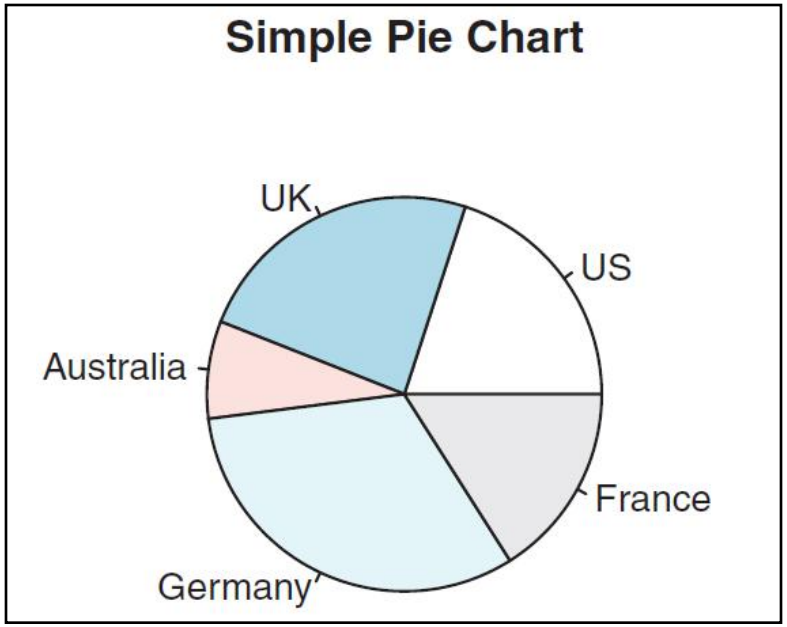
- 饼图在商业世界中无所不在，然而多数统计学家却对它持否定态度。他们更推荐使用条形图或点图。相对于面积，人们对长度的判断更精确。R中饼图的选项十分有限。
- 饼图可由以下函数创建：

  pie(x, labels)

- 其中x是一个非负数值向量，表示每个扇形的面积，而labels则是表示各扇形标签的字符型向量。

饼图 例1

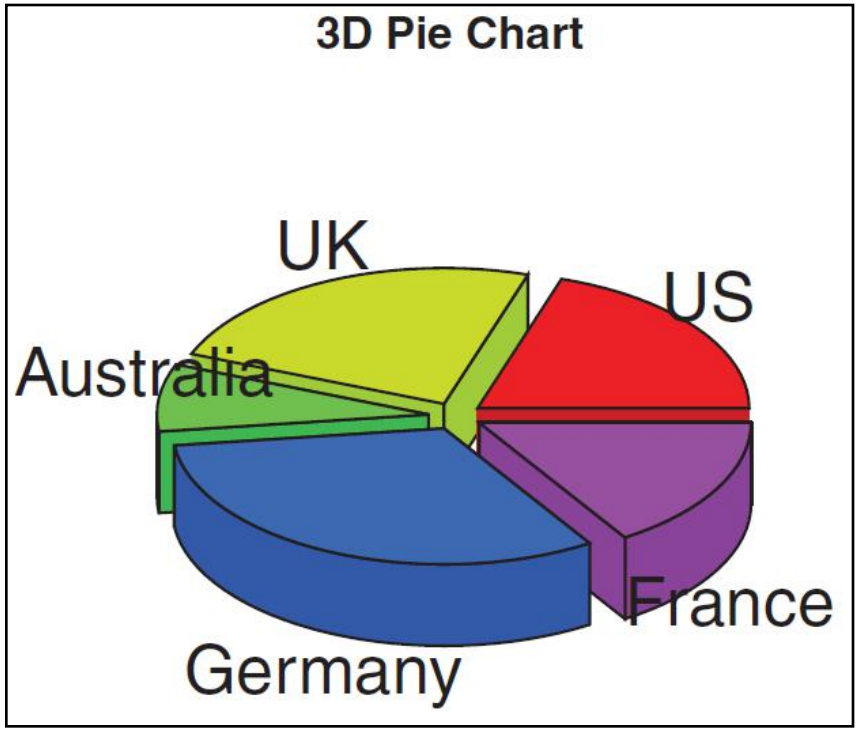
```
slices <- c(10, 12, 4, 16, 8)
lbls <- c("US", "UK", "Australia", "Germany", "France")
pie(slices, labels = lbls, main="Simple Pie Chart")
```



饼图 例2

```
install.packages("plotrix")
library(plotrix)
pie3D(slices, labels=lbls,explode=0.1,
      main="3D Pie Chart ")
```

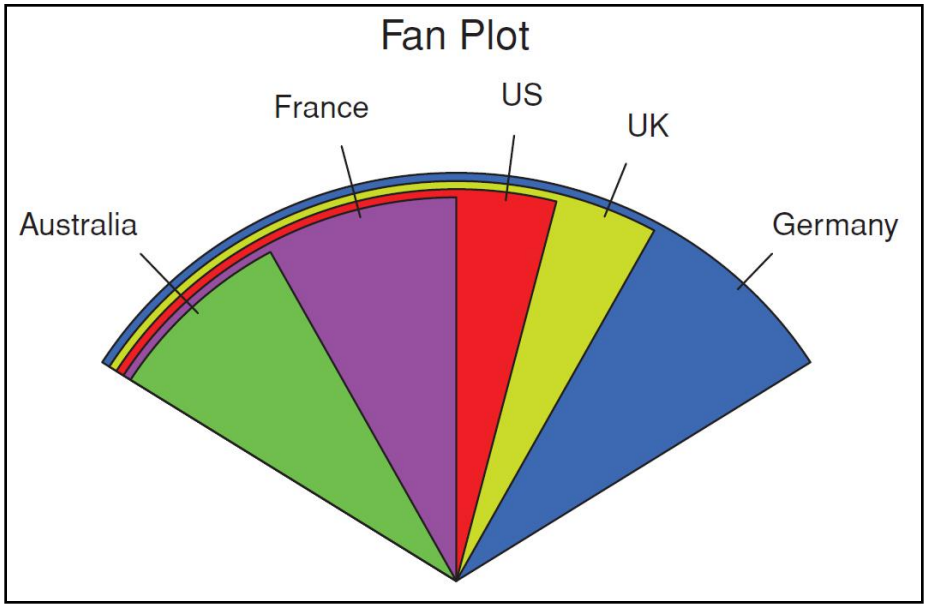
各个“块”之间的间隔，
默认值为**0**



扇形图（Fan plot）

- 饼图很难比较各扇形的值，**扇形图**提供了一种同时展示相对数量和相互差异的方法。

```
library(plotrix)
fan.plot(slices, labels = lbls, main="Fan Plot")
```



直方图（Histograms）

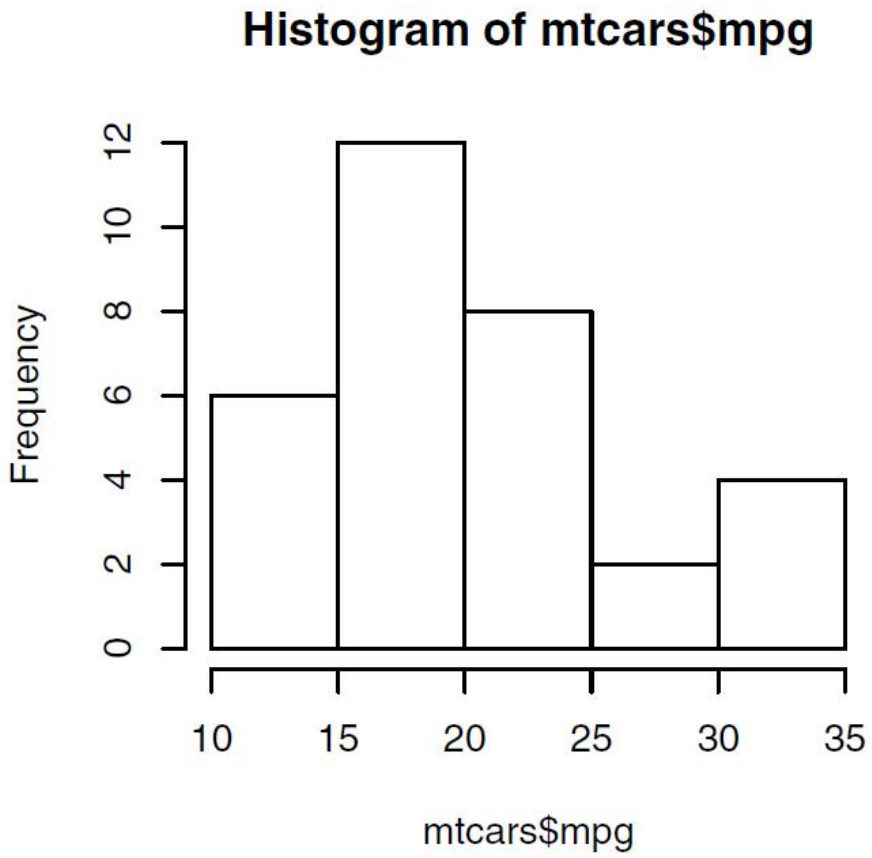
- 直方图通过在X轴上将值域分割为一定数量的组，在Y轴上显示相应值的频数，展示了连续型变量的分布。可以使用如下函数创建直方图：

  hist(x)

- 其中的x是一个由数据值组成的数值向量。参数freq=FALSE表示根据概率密度而不是频数绘制图形。参数breaks用于控制组的数量。在定义直方图中的单元时，默认将生成等距切分。

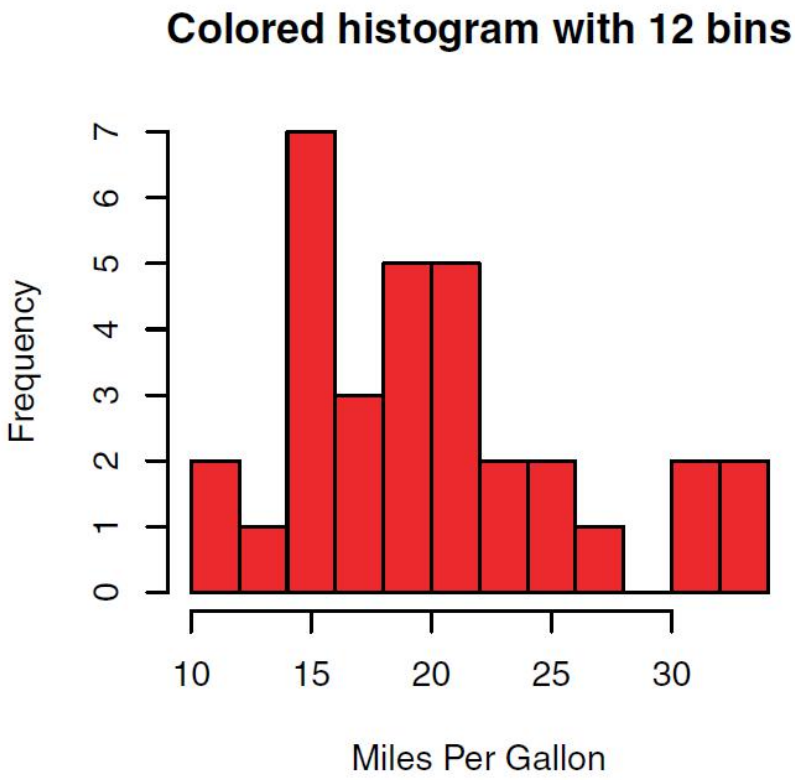
直方图 例1：简单直方图

hist(mtcars$mpg)



直方图 例2：指定组数和颜色

```
hist(mtcars$mpg,
     breaks=12,
     col="red",
     xlab="Miles Per Gallon",
     main="Colored
histogram with 12 bins")
```



核密度图（Kernel density plots）

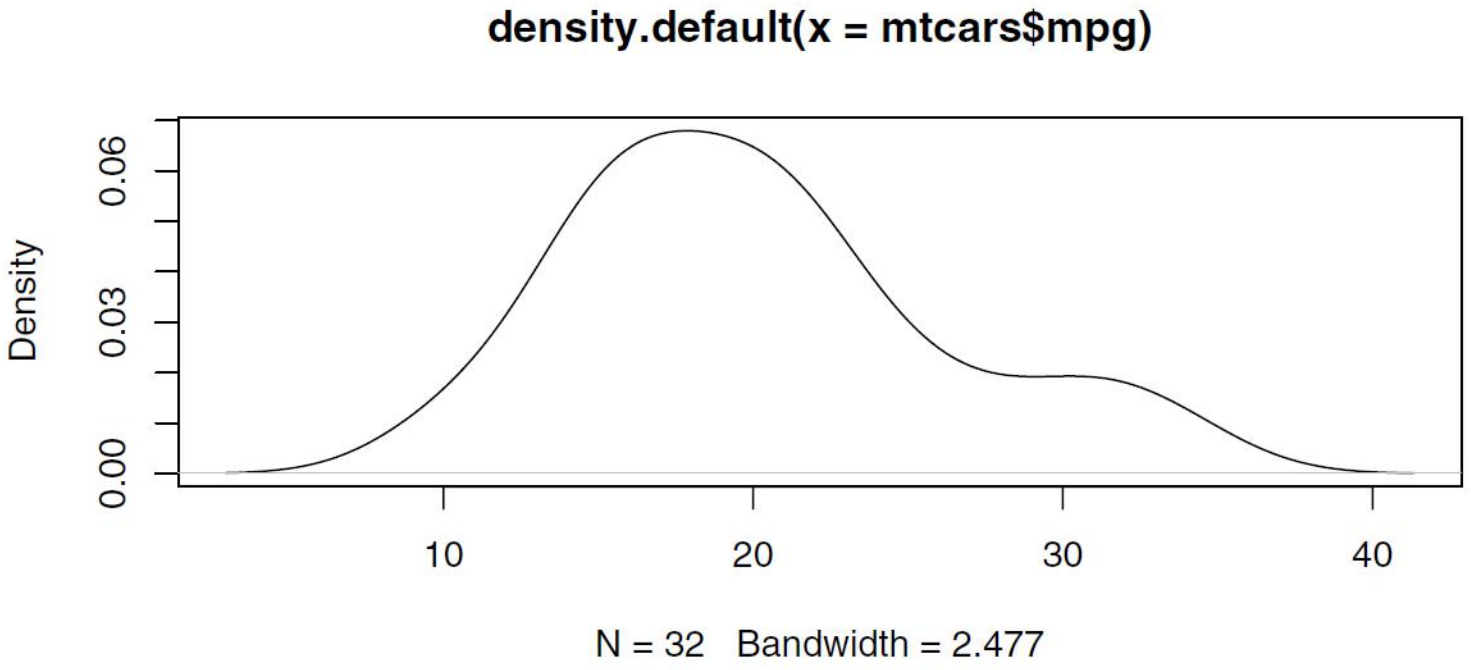
- 核密度估计是用于估计随机变量概率密度函数的一种非参数方法。核密度估计不利用数据分布的先验知识，对数据分布不附加任何假定，是一种从数据样本本身出发研究数据分布特征的方法。
- 绘制密度图的方法为：

  plot(density(x))
- 其中的x是一个数值型向量。
- plot()函数会创建一幅新的图形，所以要向一幅已经存在的图形上叠加一条密度曲线，可以使用lines()函数。

核密度图 例1

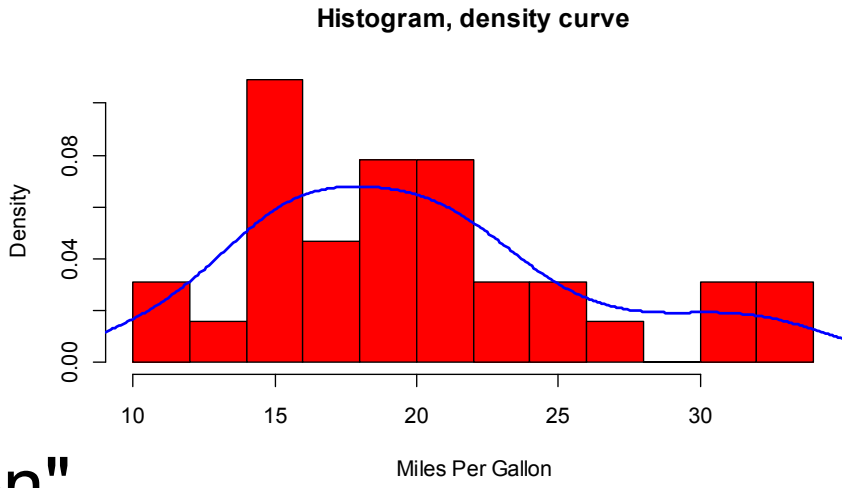
d <- density(mtcars$mpg) #returns the density data

plot(d) #plots the results



核密度图 例2
使用lines()叠加到直方图上

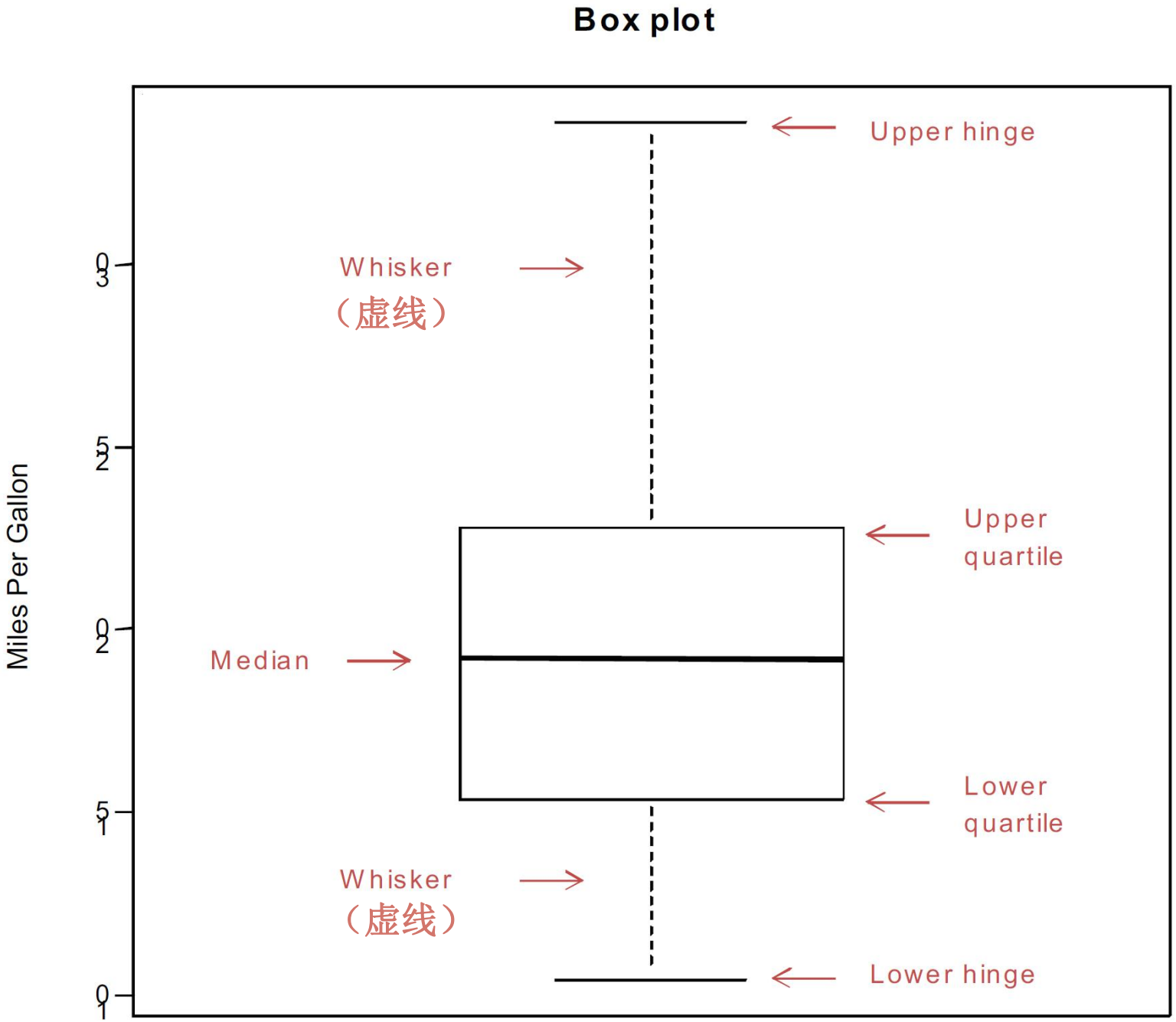
```
hist(mtcars$mpg,
     freq=FALSE,
     breaks=12,
     col="red",
     xlab="Miles Per Gallon",
     main="Histogram, density curve")
lines(density(mtcars$mpg), col="blue", lwd=2)
```



箱线图/盒状图（Box plots）

- 箱线图通过绘制连续型变量的五数总括，即**最小值、下四分位数、中位数、上四分位数以及最大值**，描述了连续型变量的分布。箱线图能够显示出可能为**离群点**的观测。
- 例，

```
boxplot(mtcars$mpg, main="Box plot",
ylab="Miles per Gallon")
```


Box plot
Miles Per Gallon
10
15
20
25
30
Upper hinge
Whisker
（虚线）
Upper
quartile
Median
Lower
quartile
Whisker
（虚线）
Lower hinge

使用并列箱线图进行跨组比较

- 箱线图可以展示单个变量或分组变量。使用格式为：

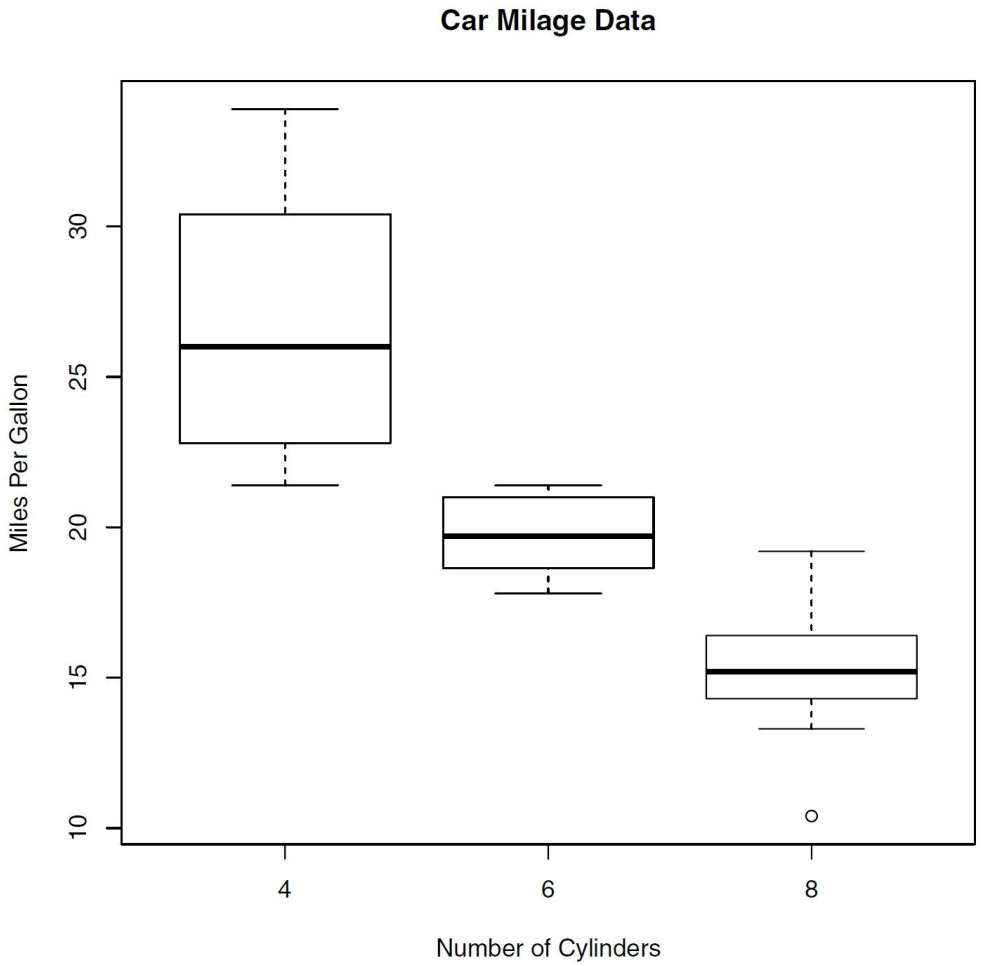
  boxplot(formula, data=dataframe)

- 其中的formula是一个公式，dataframe代表提供数据的数据框（或列表）。一个示例公式为y~A，这将为类别型变量A的每个值并列地生成数值型变量y的箱线图。公式y~A*B则将为类别型变量A和B所有水平的两两组合生成数值型变量y的箱线图。

使用并列箱线图进行跨组比较：例子

```
boxplot(mpg~cyl,data=mtcars,
        main="Car Milage Data",
        xlab="Number of Cylinders",
        ylab="Miles Per Gallon")
```

- cyl：汽缸数
- mpg：每加仑行驶距离
- 公式mpg~cyl表示：针对不同的cyl数值，分别生成mpg的箱线图。
- 通过该图可以得出什么结论？



点图（Dot plots）

- 点图提供了一种在简单水平刻度上绘制大量有标签值的方法。你可以使用dotchart()函数创建点图，格式为：

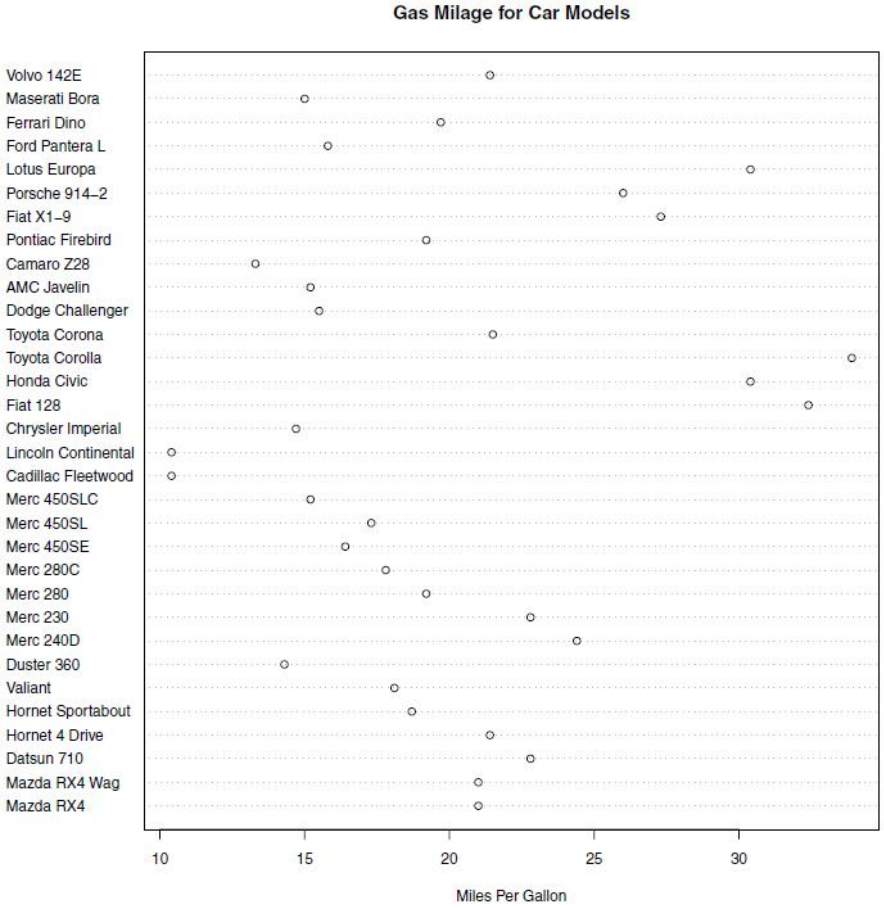
  dotchart(x,labels=)

- 其中的x是一个数值向量，而labels则是由每个点的标签组成的向量。

点图 例1

dotchart(mtcars$mpg, **labels=**row.names(mtcars), cex=.7, main="Gas Mileage for Car Models", xlab="Miles Per Gallon")

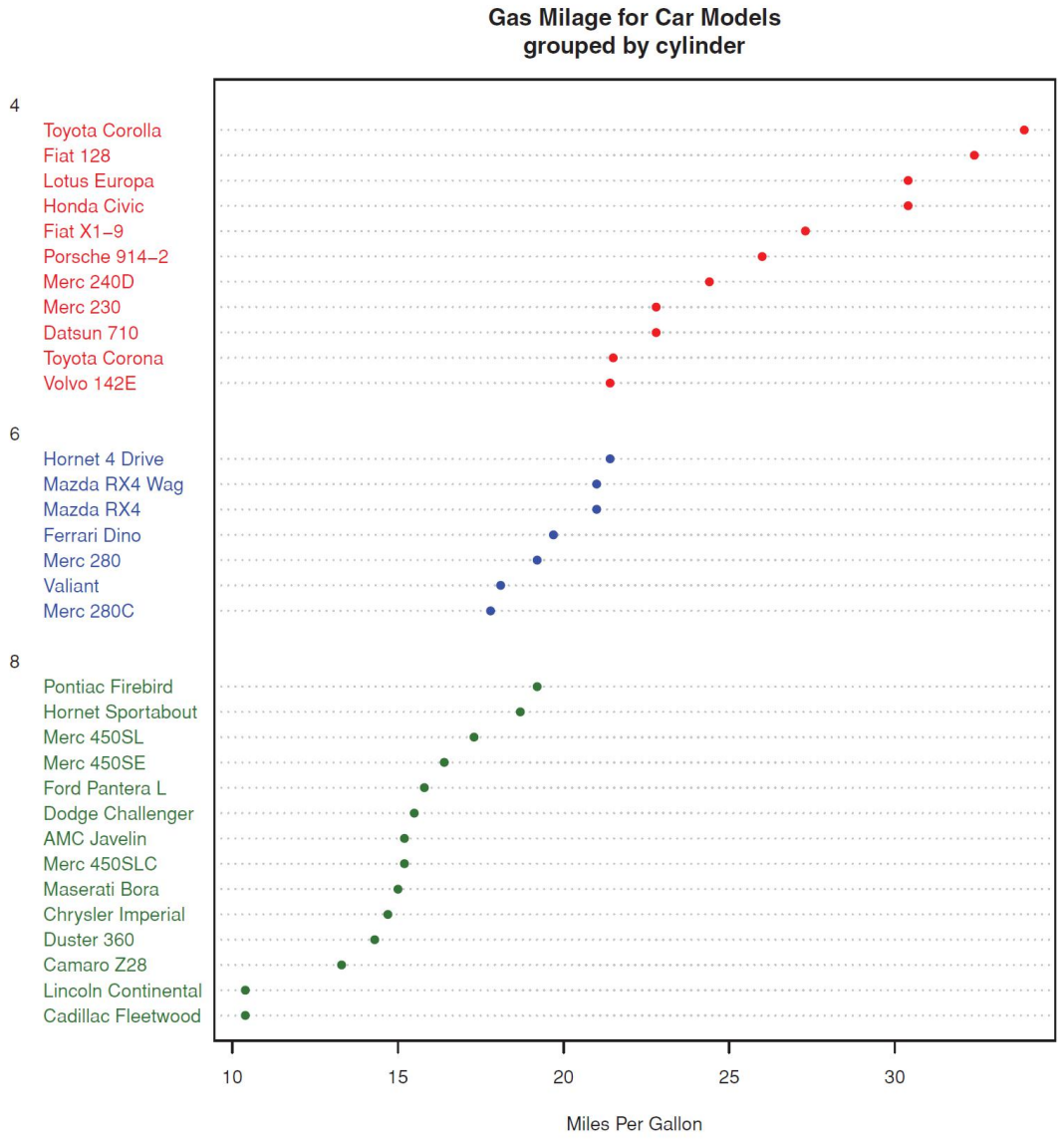
#cex指定字符大小



点图 例2（1/2）

```
x <- mtcars[order(mtcars$mpg),]
x$cyl <- factor(x$cyl)
x$color[x$cyl==4] <- "red"
x$color[x$cyl==6] <- "blue"
x$color[x$cyl==8] <- "darkgreen"
dotchart(x$mpg,
        labels = row.names(x),
        cex=.7,
        pch=19,
        groups = x$cyl,
        gcolor = "black",
        color = x$color,
        main = "Gas Mileage for Car Models\ngrouped by cylinder",
        xlab = "Miles Per Gallon")
```

点图 例2（2/2）


Gas Milage for Car Models
grouped by cylinder
4
Toyota Corolla
Fiat 128
Lotus Europa
Honda Civic
Fiat X1–9
Porsche 914–2
Merc 240D
Merc 230
Datsun 710
Toyota Corona
Volvo 142E
6
Hornet 4 Drive
Mazda RX4 Wag
Mazda RX4
Ferrari Dino
Merc 280
Valiant
Merc 280C
8
Pontiac Firebird
Hornet Sportabout
Merc 450SL
Merc 450SE
Ford Pantera L
Dodge Challenger
AMC Javelin
Merc 450SLC
Maserati Bora
Chrysler Imperial
Duster 360
Camaro Z28
Lincoln Continental
Cadillac Fleetwood
10
15
20
25
30
Miles Per Gallon

中级绘图

- 散点图
- 气泡图
- 折线图
- 相关图
- 马赛克图

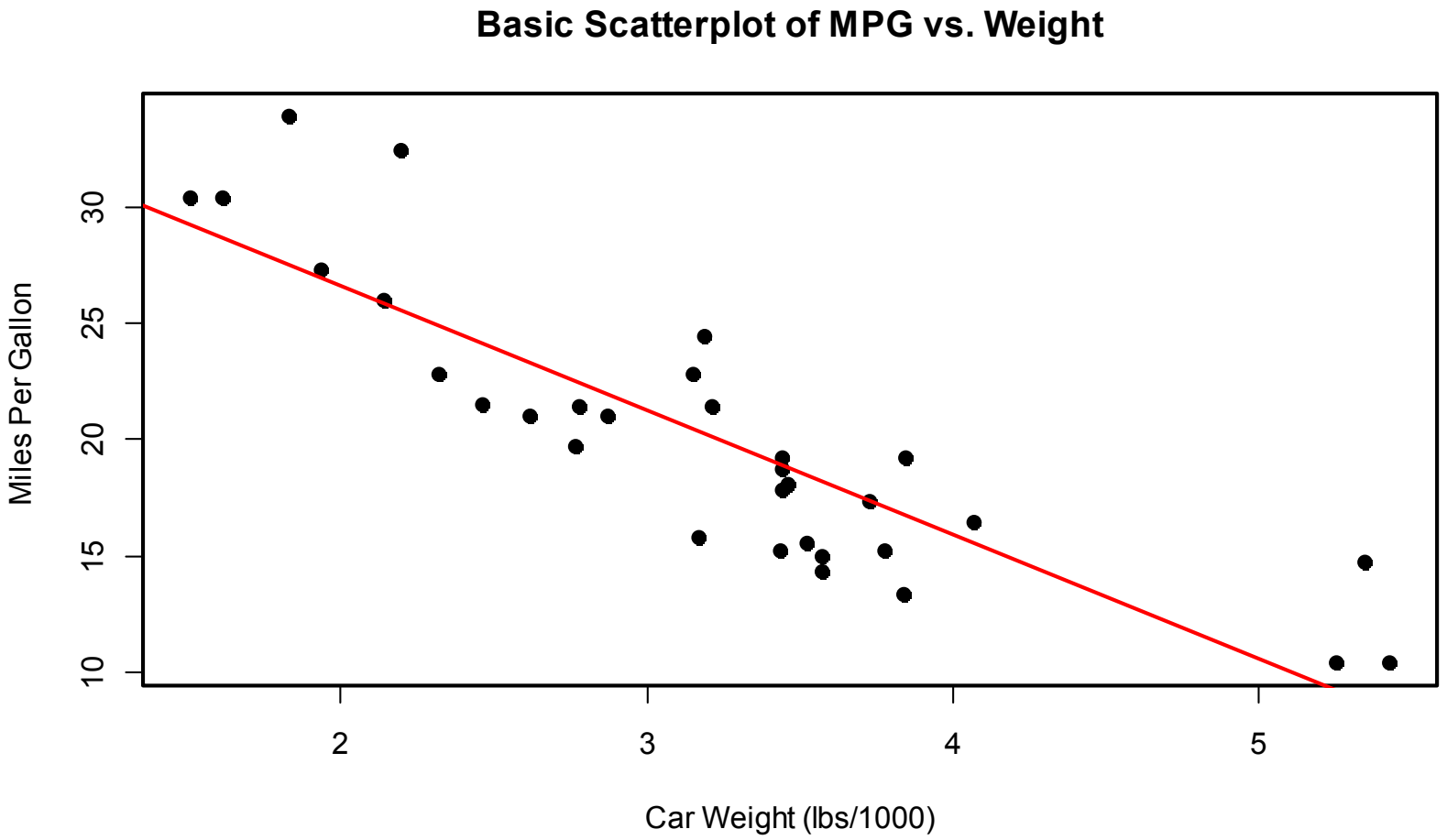
散点图（Scatter plots）

- 散点图可用来描述两个连续型变量间的关系。
- R中创建散点图的基础函数是**plot**(x, y)，其中，x和y是数值型向量，代表着图形中的(x, y)点。
- 通过添加额外信息来增强图形表达功能。

散点图 例（1/2）

```
#探究车重和单位油量行驶公里数的关系
attach(mtcars)
plot(wt, mpg,
    main="Basic Scatterplot of MPG vs. Weight",
    xlab="Car Weight (lbs/1000)",
    ylab="Miles Per Gallon ", pch=19)
abline(lm(mpg ~ wt), col="red", lwd=2, lty=1)
#abline()函数用来添加最佳拟合的线性直线
detach(mtcars)
```

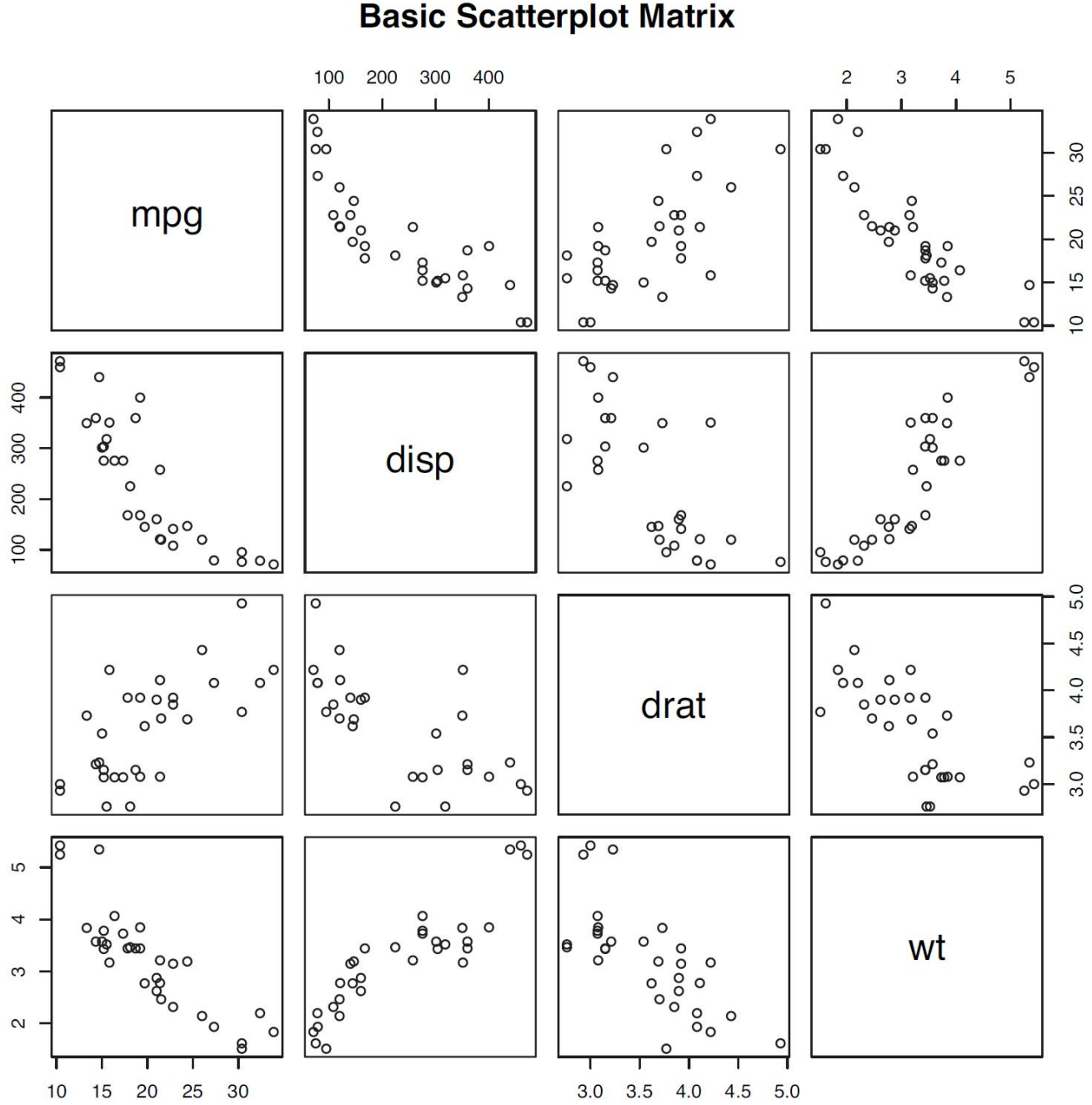
散点图 例（2/2）


Basic Scatterplot of MPG vs. Weight
Miles Per Gallon
10
15
20
25
30
2
3
4
5
Car Weight (lbs/1000)

散点图矩阵（1/2）

- R中有多种函数可以创建散点图矩阵。
- pairs()函数可以创建基础的散点图矩阵。
- 例，

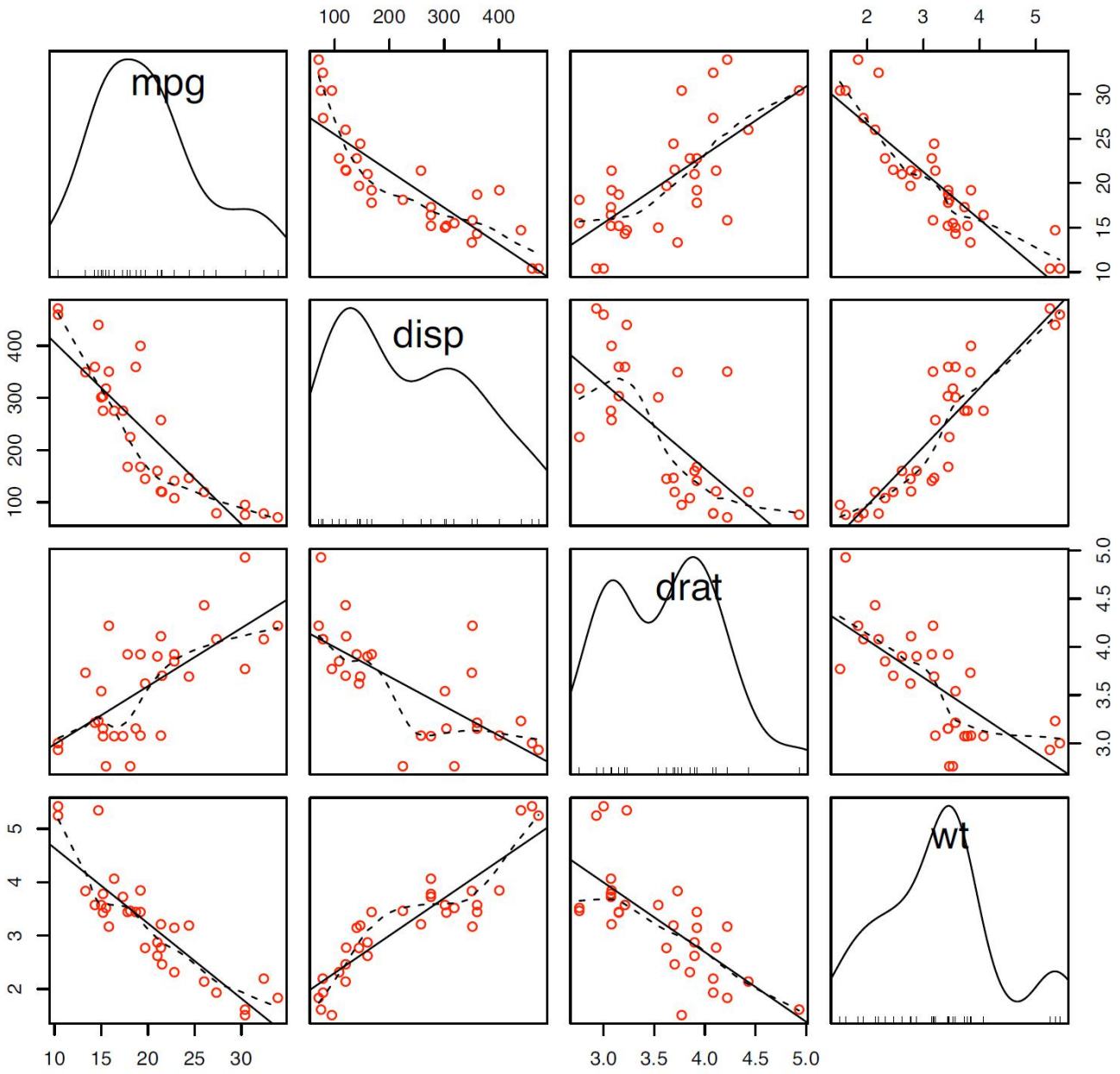
```
pairs(~ mpg + disp + drat + wt, data=mtcars,
      main="Basic Scatterplot Matrix")
```


Basic Scatterplot Matrix
mpg
disp
drat
wt
100 200 300 400
2 3 4 5
10 15 20 25 30
100 200 300 400
3.0 3.5 4.0 4.5 5.0
2 3 4 5
10 15 20 25 30
3.0 3.5 4.0 4.5 5.0

散点图矩阵（2/2）

```
install.packages("car")
library(car)
scatterplotMatrix(~ mpg + disp + drat + wt,
data=mtcars, spread=FALSE,
smoother.args=list(lty=2), main="Scatter Plot
Matrix via car Package")
```

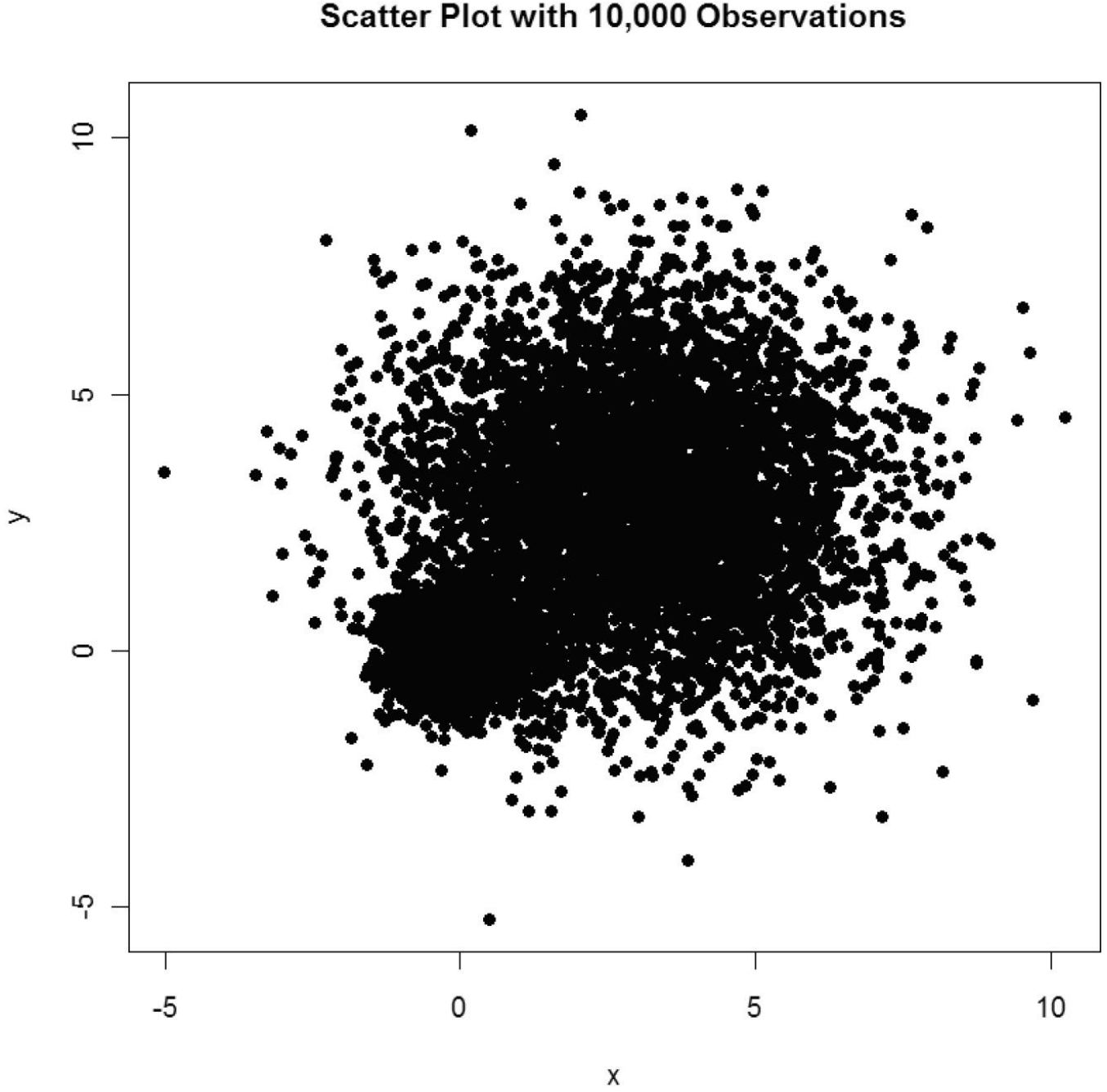
- 线性和平滑拟合曲线被默认添加，主对角线处添加了核密度曲线和轴须图。spread = FALSE选项表示不添加展示分散度和对称信息的直线，lty.smooth =2设定平滑（loess）拟合曲线使用虚线而不是实线。


Scatterplot Matrix via car package
mpg
disp
drat
wt
100 200 300 400
2 3 4 5
10 15 20 25 30
100 200 300 400
3.0 3.5 4.0 4.5 5.0
2 3 4 5
10 15 20 25 30
3.0 3.5 4.0 4.5 5.0

高密度散点图

- 当数据点重叠很严重时，用散点图来观察变量关系就显得“力不从心”了。
- 例（人为设计的例子，10000个观测点）：

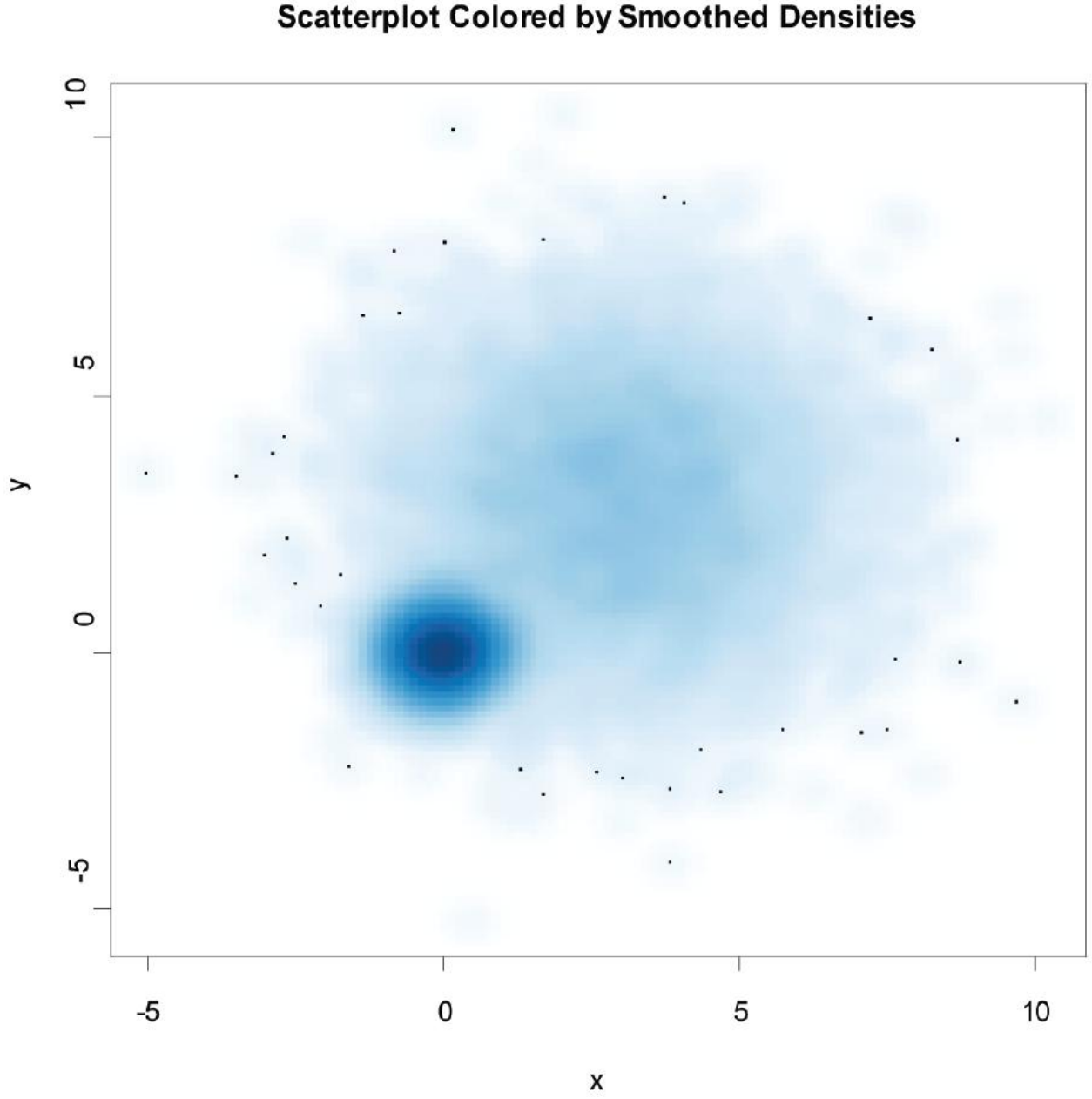
```
set.seed(1234)
n <- 10000
c1 <- matrix(rnorm(n, mean=0, sd=.5), ncol=2)
c2 <- matrix(rnorm(n, mean=3, sd=2), ncol=2)
mydata <- rbind(c1, c2)
mydata <- as.data.frame(mydata)
names(mydata) <- c("x", "y")
with(mydata,
    plot(x, y, pch=19, main="Scatter Plot with 10000 Observations"))
```


Scatter Plot with 10,000 Observations
y
x
10
5
0
-5
-5
0
5
10

高密度散点图：smoothScatter()

- smoothScatter()函数可利用核密度估计生成用颜色密度来表示点分布的散点图。
- 前例可写为：

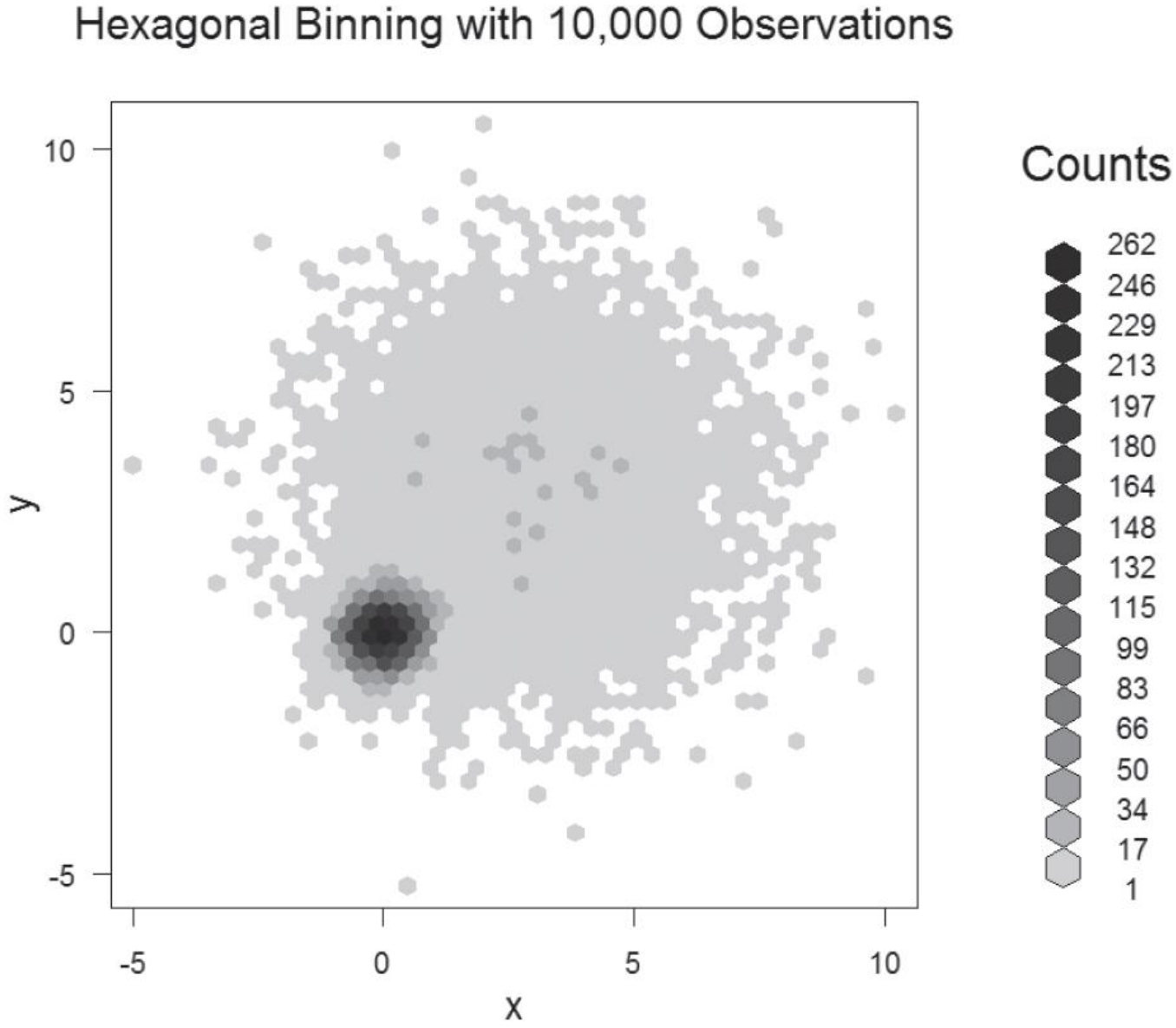
```
with(mydata, smoothScatter(x, y,
main="Scatter Plot colored by Smoothed
Densities"))
```


Scatterplot Colored by Smoothed Densities
y
10
5
0
-5
-5
0
5
10
x

高密度散点图：hexbin()

- hexbin包中的hexbin()函数将二元变量的封箱放到六边形单元格中。例，

```
install.packages("hexbin")
library(hexbin)
with(mydata, {
  bin <- hexbin(x, y, xbins=50)
  plot(bin, main="Hexagonal Binning with 10,000 Observations")
})
```


Hexagonal Binning with 10,000 Observations
10
5
0
-5
y
-5
0
5
10
x
Counts
262
246
229
213
197
180
164
148
132
115
99
83
66
50
34
17
1

高密度散点图：iplot()

- IDPmisc包中的iplot()函数可通过颜色来展示点的密度（在某特定点上数据点的数目）。
- 例，

```
install.packages("IDPmisc")
library(IDPmisc)
with(mydata, iplot(x, y, main="Image Scatter Plot with Color Indicating Density"))
```

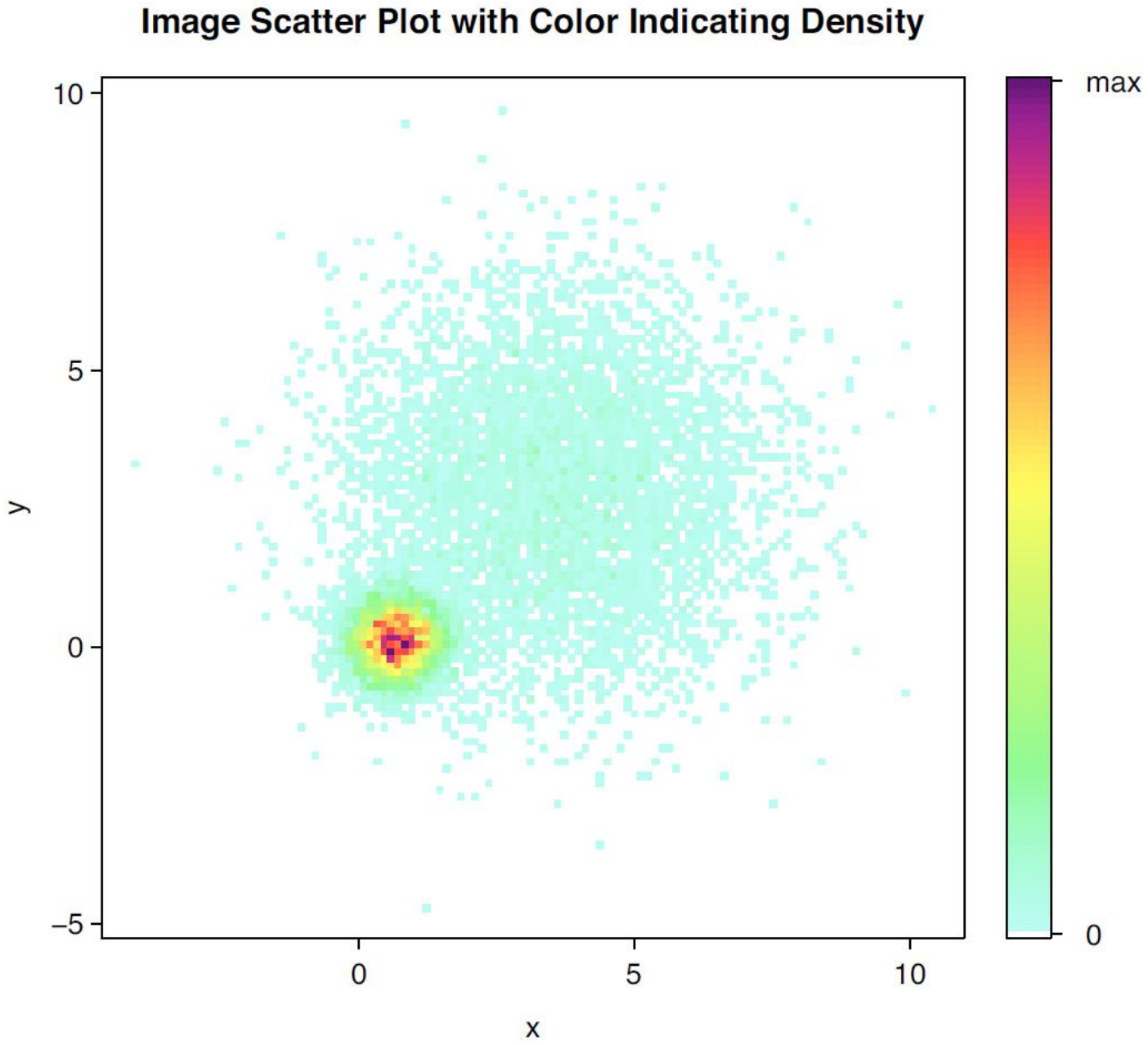


Image Scatter Plot with Color Indicating Density
10
5
0
−5
y
0
5
10
x
max
0

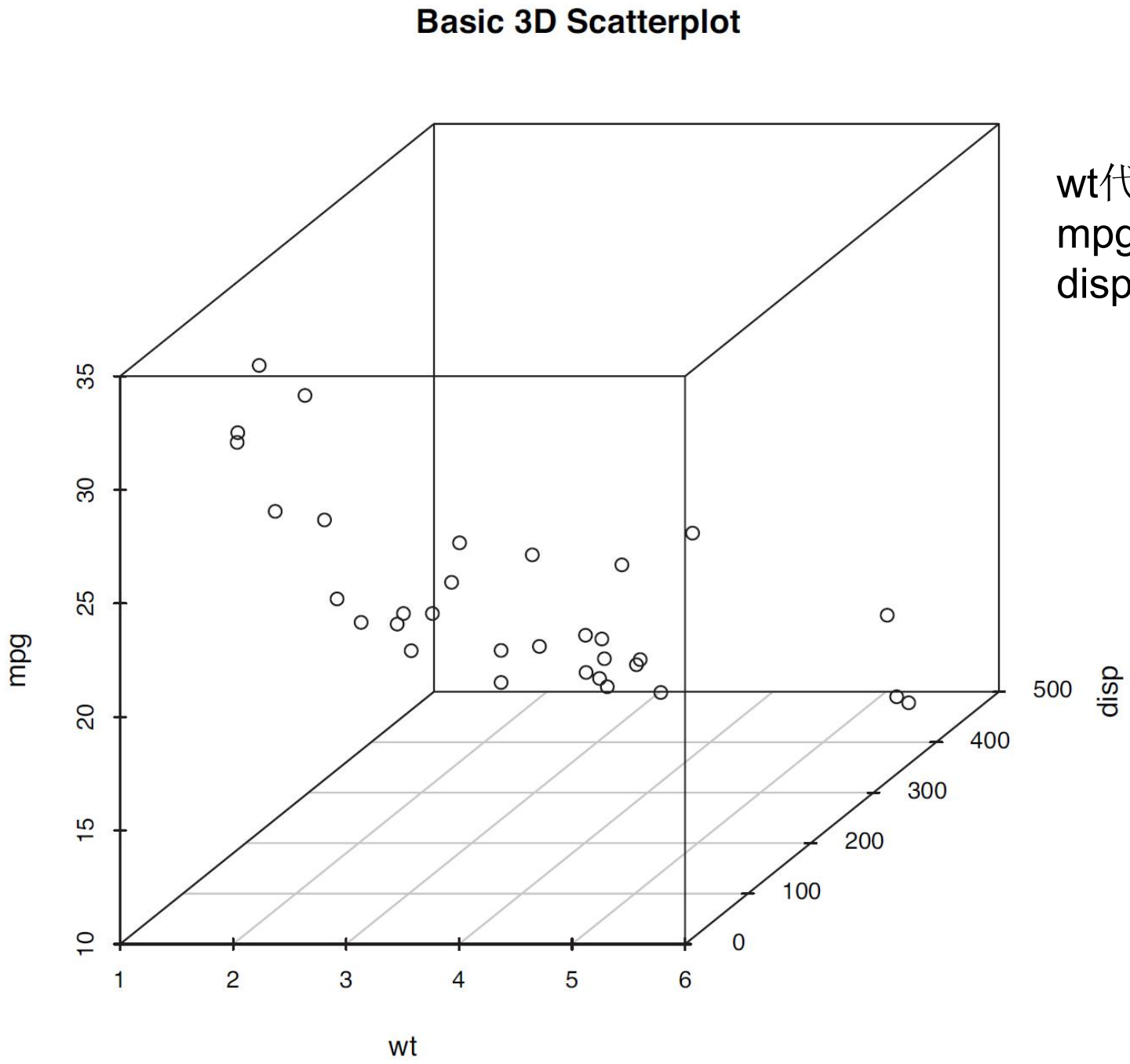
三维散点图

- scatterplot3d中的scatterplot3d()函数可绘制三维散点图。格式：

  scatterplot3d(x, y, z)

- x被绘制在水平轴上，y被绘制在竖直轴上，z被绘制在透视轴上。例，

```
install.packages("scatterplot3d")
library(scatterplot3d)
attach(mtcars)
scatterplot3d(wt, disp, mpg,
              main="Basic 3D Scatter Plot")
```



wt代表车重，
mpg代表每加仑英里数，
disp代表发动机排量。

气泡图（Bubble plots）

- 除了通过三维散点图可以展示三个定量变量间的关系，还可以使用二维散点图，加上用点的大小来代表第三个变量的值。即气泡图。

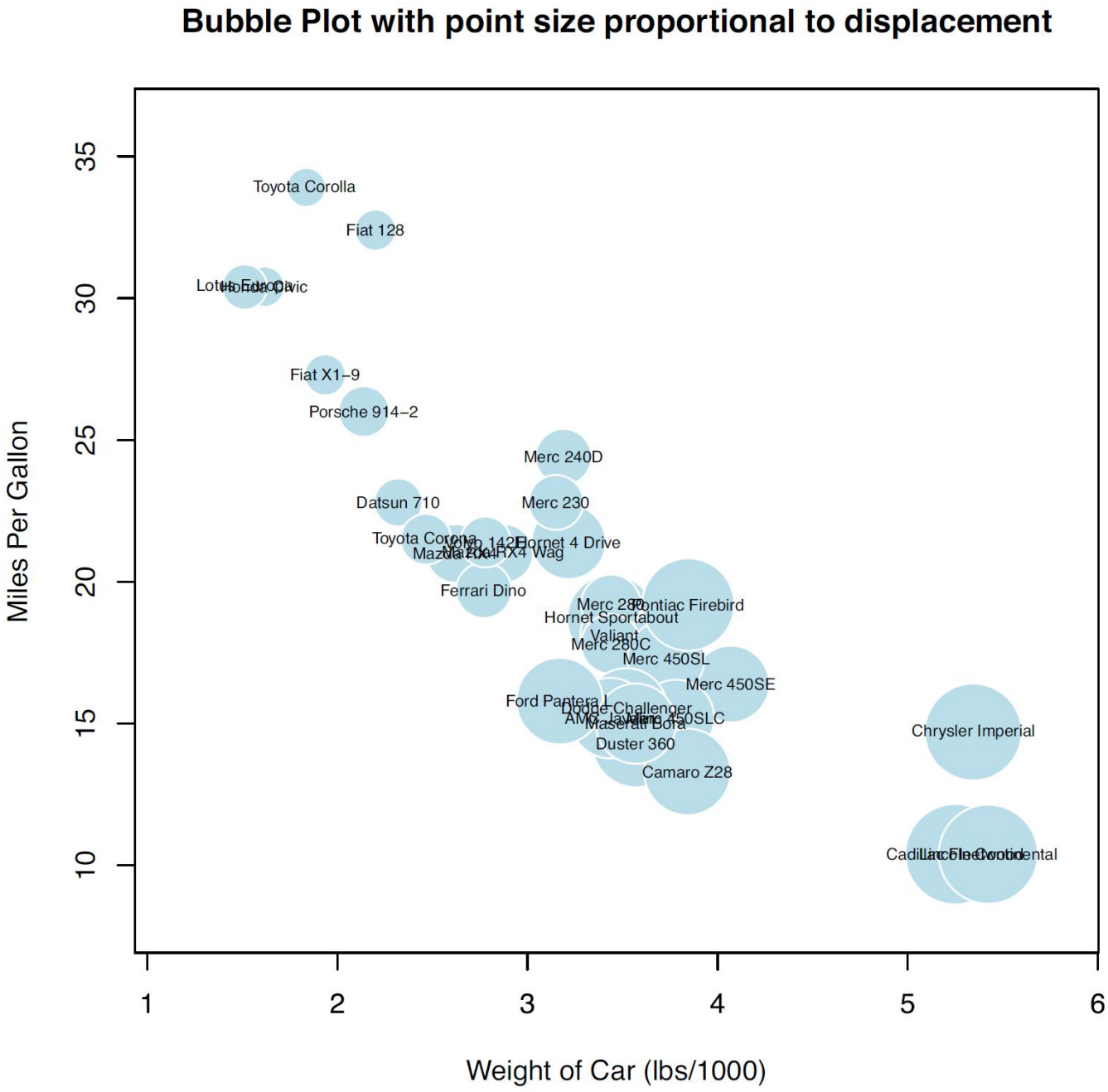
  symbols(x, y, **circle=**radius)

- 其中x、y和radius是需要设定的向量，分别表示x、y坐标和圆圈半径。
- 和饼图一样，统计人员倾向避免使用气泡图。但气泡图在商业中非常受欢迎。

气泡图 例

- mtcars数据集：x轴代表车重，y轴代表每加仑英里数，气泡大小代表发动机排量。

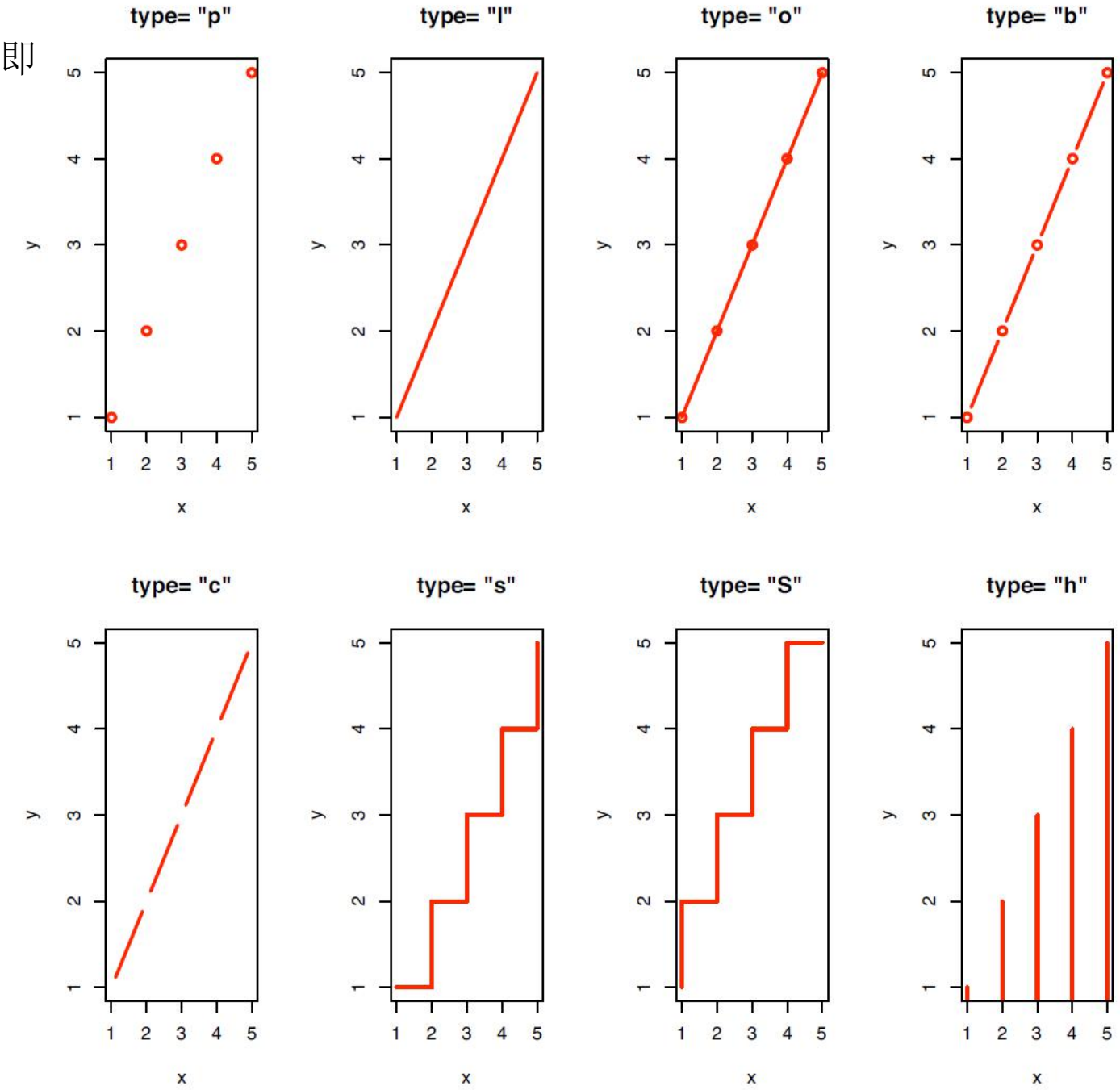
```
attach(mtcars)
r <- sqrt(disp/pi) #根据气泡大小算半径
symbols(wt, mpg, circle=r, inches=0.30,
    fg="white", bg="lightblue",
    main="Bubble Plot with point size proportional to displacement",
    ylab="Miles Per Gallon",
    xlab="Weight of Car (lbs/1000)")
text(wt, mpg, rownames(mtcars), cex=0.6)
#text可选，用来添加各个汽车的名称
detach(mtcars)
```


Bubble Plot with point size proportional to displacement
Miles Per Gallon
Weight of Car (lbs/1000)
10
15
20
25
30
35
1
2
3
4
5
6
Toyota Corolla
Fiat 128
Fiat X1−9
Porsche 914−2
Merc 240D
Datsun 710
Merc 230
Toyota Corona
Hornet 4 Drive
Ferrari Dino
Pontiac Firebird
Hornet Sportabout
Valiant
Merc 280C
Merc 450SL
Merc 450SE
Ford Pantera L
Maserati Bora
Duster 360
Camaro Z28
Chrysler Imperial

折线图（Line charts）

- 如果将散点图上的点从左往右连接起来，那么就会得到一个折线图。折线图是一个刻画变动的优秀工具。
- 折线图一般可用下列两个函数之一来创建：
  - **plot**(x, y, **type=**)
  - **lines**(x, y, **type=**)
- type参数见下页。
- plot()和lines()的区别见大后页。

“p”参数即散点图



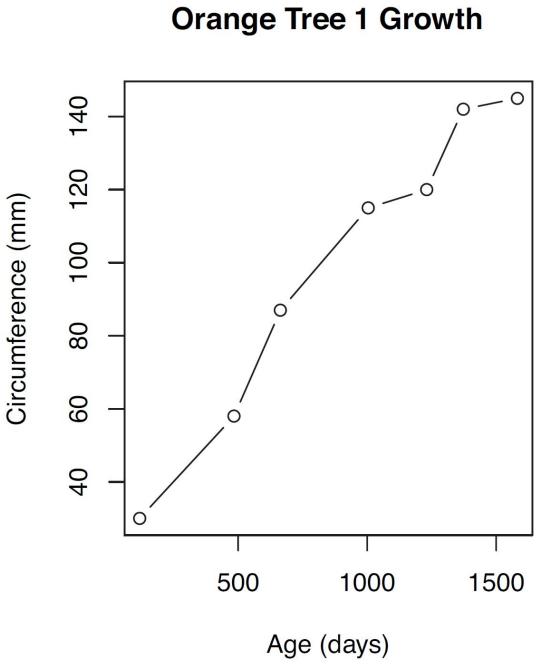
plot()和lines()的区别

- plot()和lines()函数工作原理并不相同。
  - plot()函数是被调用时即创建一幅新图，
  - lines()函数则是在已存在的图形上添加信息，并不能自己生成图形。
- 因此，**lines()函数通常是在plot()函数生成一幅图形后再被调用**。如果对图形有要求，可以先通过plot()函数中的type = n来创建坐标轴、标题和其他图形特征，然后再使用lines()函数添加各种需要绘制的曲线。

折线图 例1

- R自带Orange数据集（包括五种橘树树龄和年轮的数据）。本例考察第一种橘树。

```
t1 <- subset(Orange, Tree==1)
plot(t1$age, t1$circumference,
    xlab="Age (days)",
    ylab="Circumference (mm)",
    main="Orange Tree 1 Growth",
    type="b")
```



折线图 例2：展示5种橘树

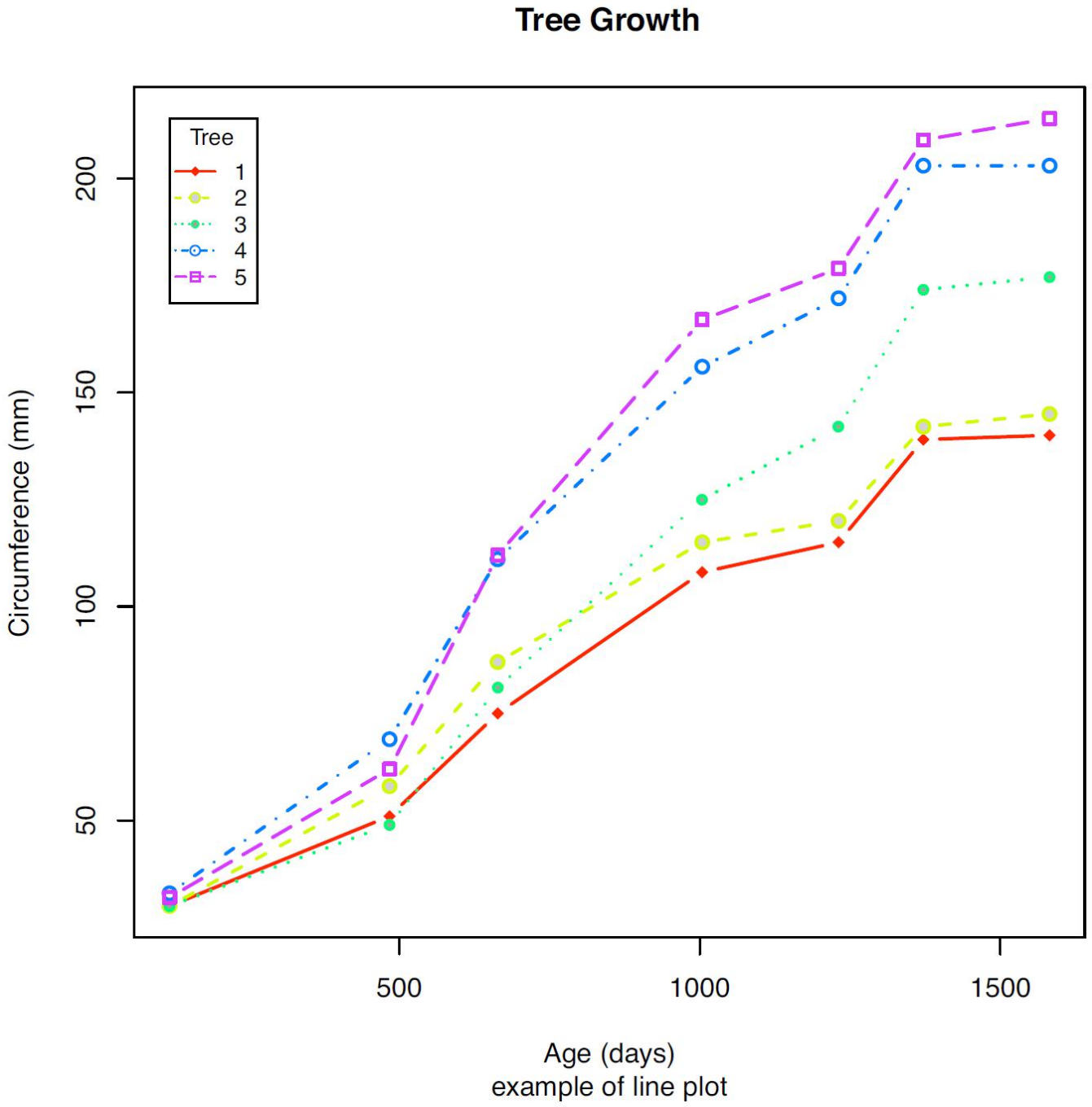
```
Orange$Tree <- as.numeric(Orange$Tree)
ntrees <- max(Orange$Tree)
xrange <- range(Orange$age)
yrange <- range(Orange$circumference)
plot(xrange, yrange,
    type="n",
    xlab="Age (days)",
    ylab="Circumference (mm)"
)
colors <- rainbow(ntrees)
linetype <- c(1:ntrees)
plotchar <- seq(18, 18+ntrees, 1)
for (i in 1:ntrees) {
  tree <- subset(Orange, Tree==i)
  lines(tree$age, tree$circumference,
       type="b",
       lwd=2,
       lty=linetype[i],
       col=colors[i],
       pch=plotchar[i]
  )
}
title("Tree Growth", "example of line plot")
legend(xrange[1], yrange[2],
       1:ntrees,
       cex=0.8,
       col=colors,
       pch=plotchar,
       lty=linetype,
       title="Tree"
)
```

为方便起见，将因子转化为数值型

创建图形

添加线条

添加图例


Tree Growth
Tree
1
2
3
4
5
Circumference (mm)
50
100
150
200
500
1000
1500
Age (days)
example of line plot

相关图（Correlograms）

- 相关系数矩阵是多元统计分析的一个基本方式。哪些被考察的变量与其他变量相关性很强，而哪些并不强？相关变量是否以某种特定的方式聚集在一起？随着变量数的增加，这类问题将变得更难回答。
- 相关图作为一种相对现代的方法，可通过对相关系数矩阵的可视化来回答这些问题。
- 相关图是检验定量变量中众多二元关系的一种有效方式。

相关系数矩阵

- 例：

  options(digits=2)

  cor(mtcars)

- 结果

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
mpg	1.00	-0.85	-0.85	-0.78	0.681	-0.87	0.419	0.66	0.600	0.48	-0.551
cyl	-0.85	1.00	0.90	0.83	-0.700	0.78	-0.591	-0.81	-0.523	-0.49	0.527
disp	-0.85	0.90	1.00	0.79	-0.710	0.89	-0.434	-0.71	-0.591	-0.56	0.395
hp	-0.78	0.83	0.79	1.00	-0.449	0.66	-0.708	-0.72	-0.243	-0.13	0.750
drat	0.68	-0.70	-0.71	-0.45	1.000	-0.71	0.091	0.44	0.713	0.70	-0.091
wt	-0.87	0.78	0.89	0.66	-0.712	1.00	-0.175	-0.55	-0.692	-0.58	0.428
qsec	0.42	-0.59	-0.43	-0.71	0.091	-0.17	1.000	0.74	-0.230	-0.21	-0.656
vs	0.66	-0.81	-0.71	-0.72	0.440	-0.55	0.745	1.00	0.168	0.21	-0.570
am	0.60	-0.52	-0.59	-0.24	0.713	-0.69	-0.230	0.17	1.000	0.79	0.058
gear	0.48	-0.49	-0.56	-0.13	0.700	-0.58	-0.213	0.21	0.794	1.00	0.274
carb	-0.55	0.53	0.39	0.75	-0.091	0.43	-0.656	-0.57	0.058	0.27	1.000

corrgram()函数

- corrgram()函数的格式如下：

  corrgram(x, order=, **panel=**, text.panel=, diag.panel=)

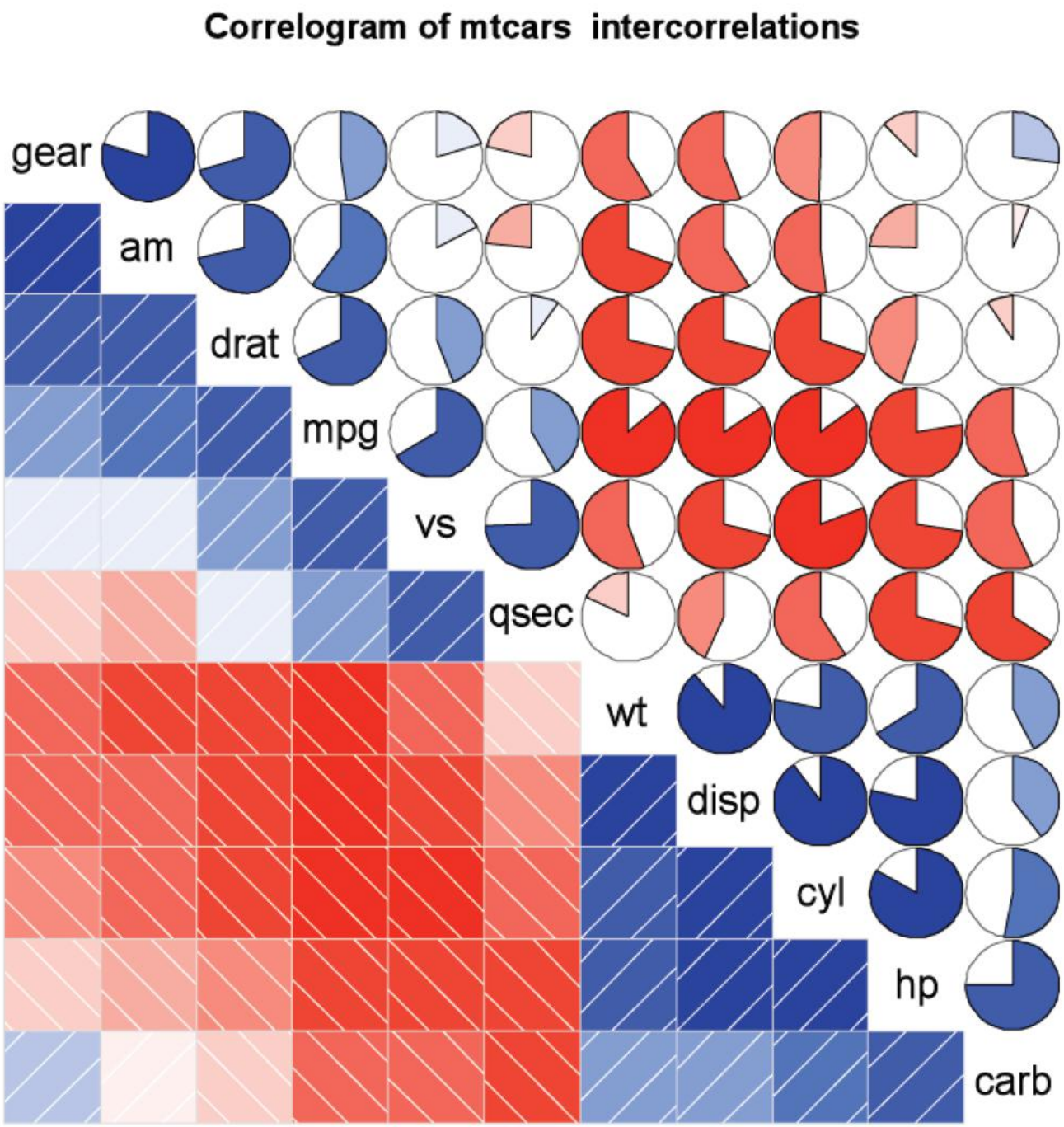
  – 其中，x是一行一个观测的数据框。当order = TRUE时，相关矩阵将使用主成分分析法对变量重新排序，这使得二元变量的关系模式更为明显。选项**panel 设定非对角线面板使用的元素类型**。可通过选项lower.panel和upper.panel来分别设置主对角线下方和上方的元素类型。text.panel和diag.panel选项控制着主对角线元素类型。

corrgram()函数的panel选项

Placement	Panel Option	Description
Off diagonal	`panel.pie`	The filled portion of the pie indicates the magnitude of the correlation.
	`panel.shade`	The depth of the shading indicates the magnitude of the correlation.
	`panel.ellipse`	A confidence ellipse and smoothed line are plotted.
	`panel.pts`	A scatter plot is plotted.
Main diagonal	`panel.minmax`	The minimum and maximum values of the variable are printed.
	`panel.txt`	The variable name is printed.

相关图 例1

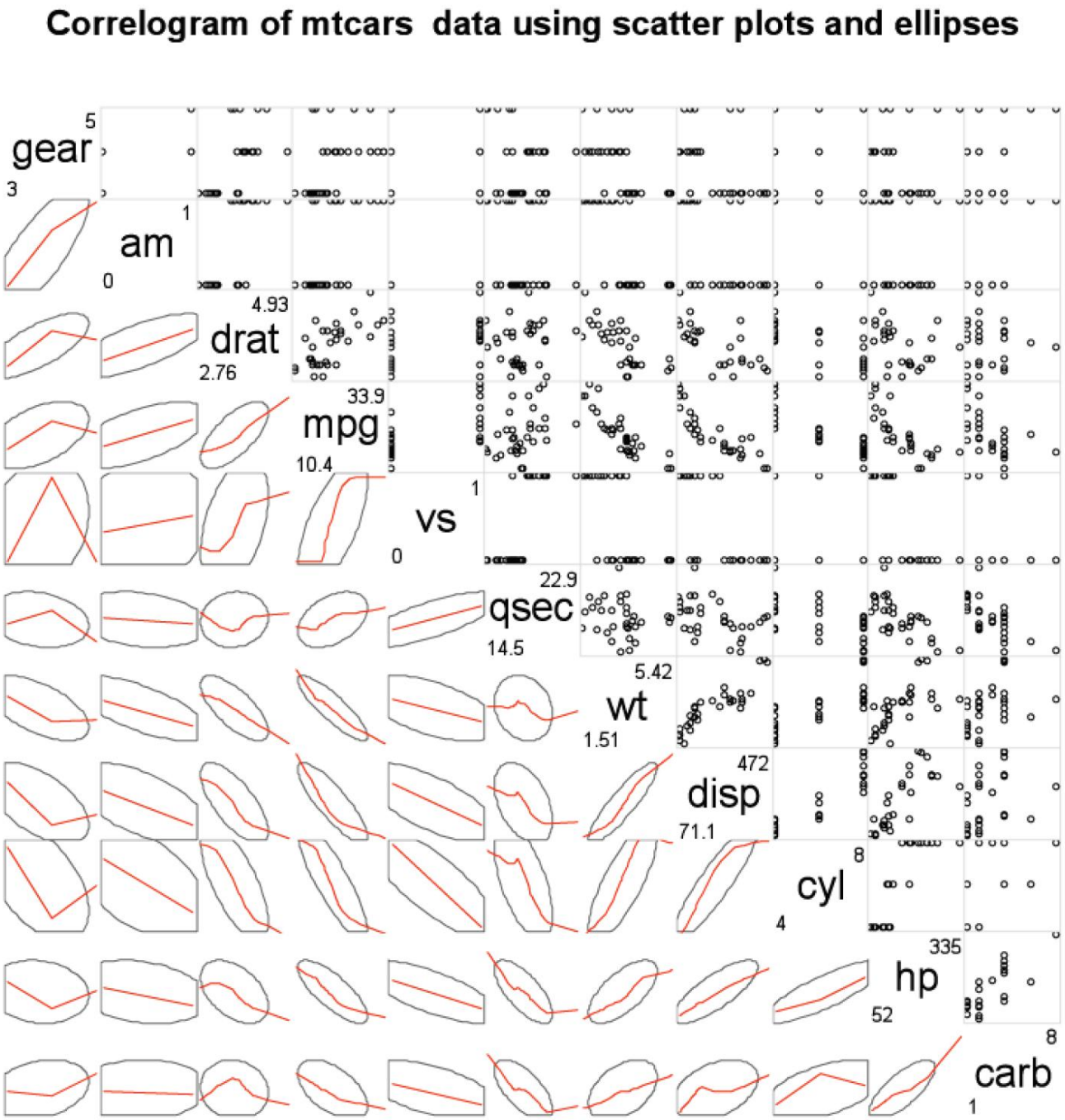
```
install.packages("corrgram")
library(corrgram)
corrgram(mtcars, order=TRUE,
lower.panel=panel.shade,
upper.panel=panel.pie, text.panel=panel.txt,
main="Corrgram of mtcars intercorrelations")
```



- 下三角：蓝色斜杠表示单元格中的两个变量呈正相关。红色斜杠表示变量呈负相关。色彩越深，说明变量相关性越大。
- 上三角：颜色的功能同上，但相关性大小由被填充的饼图块的大小来展示。正相关性将从12点钟处开始顺时针填充饼图，而负相关性则逆时针方向填充饼图。

相关图 例2

```
corrgram(mtcars, order=TRUE,
lower.panel=panel.ellipse,
upper.panel=panel.pts,
text.panel=panel.txt,
diag.panel=panel.minmax, #设置主对角线元素
main="Corrgram of mtcars data using scatter
plotsand ellipses")
```



- 下三角：平滑拟合曲线和置信椭圆。
- 上三角：散点图。散点图限制了一些变量的可用值。例如，挡位数须取3、4或5，气缸数须取4、6或者8。am（传动类型）和vs都是二值型。因此上三角区域的散点图看起来很奇怪。
- 主对角面板包含变量最小和最大值。
- 矩阵的行和列利用主成分分析法进行了重排序。

马赛克图（Mosaic plots）

- 马赛克图用于可视化两个以上的类别型变量（只观察单个类别型变量，可以使用柱状图或者饼图）。
- mosaic()函数调用格式

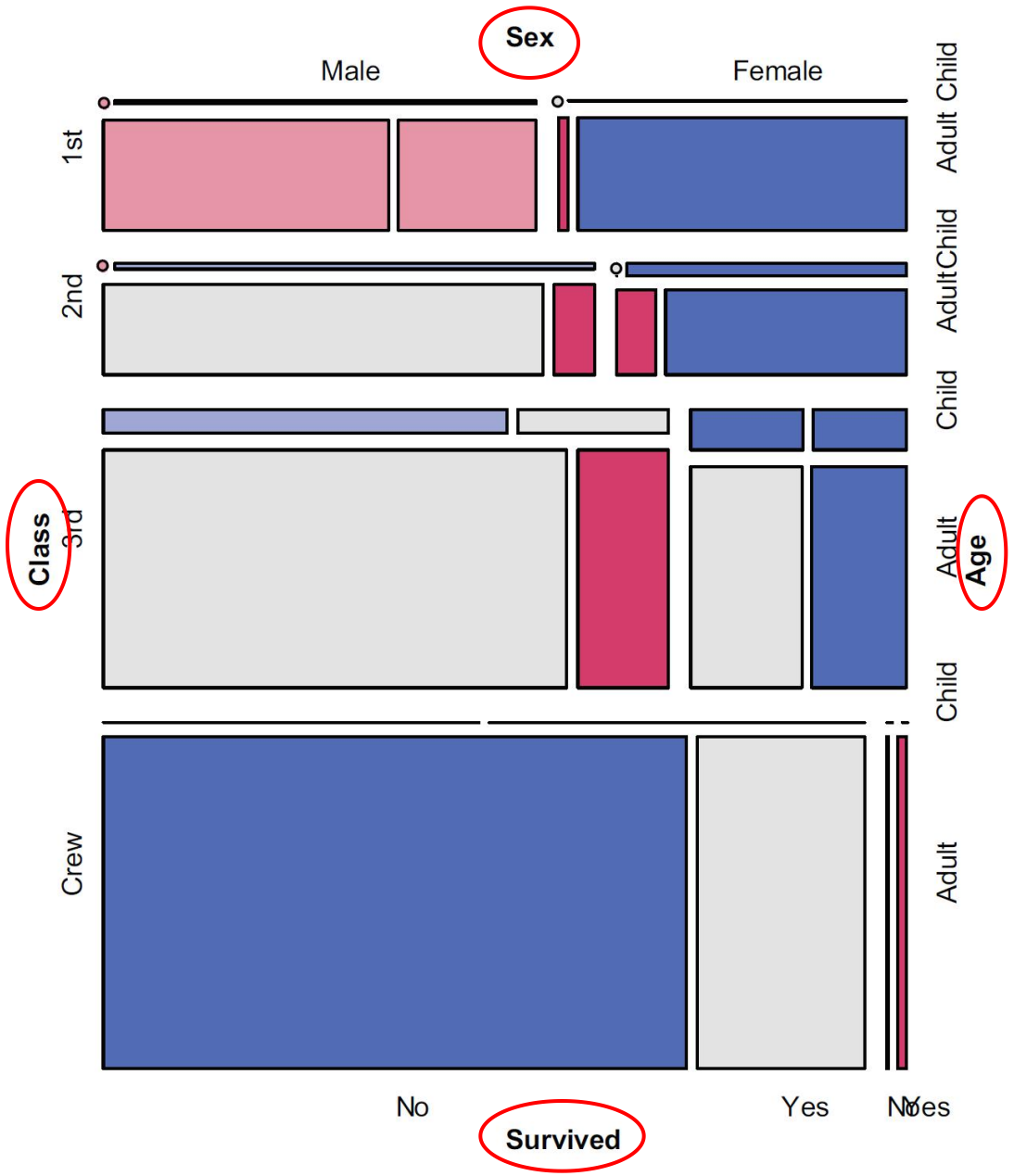
  mosaic(table)

  – 其中table是数组形式的列联表。
  – 添加选项shade = TRUE将根据拟合模型的皮尔逊残差值对图形上色。
  – 添加选项legend = TRUE将展示残差的图例。

马赛克图 例

- 例：基础安装中的Titanic数据集

```
ftable(Titanic)
library(vcd)
mosaic(Titanic, shade=TRUE, legend=TRUE)
```



(1)从船员到头等舱，存活率陡然提高；(2)大部分孩子在三等舱和二等舱；(3)头等舱大部分女性都存活，三等舱仅有一半女性存活；(4)船员女性很少，导致该组的标签重叠。(5)扩展马赛克图添加了颜色和阴影来表示拟合模型的残差值。蓝色阴影表明，在假定生存率与船舱等级、性别和年龄层无关的条件下，该类别下的生存率通常超过预期值。红色阴影则含义相反。在模型的独立条件下，头等舱女性存活数和男性船员死亡数超过模型预期值，三等舱男性的存活数比模型预期值低。