# محاسبات آماری پیشرفته

## ترم اول سال تحصیلی ۹۳

## جلسه ششم: نمایش گرافیکی داده‌ها

حسین باغیشنی

دانشگاه شاهرود

۱۶ آبان ۱۳۹۳

# مقدمه

بحث این جلسه خیلی گسترده است و نمی‌توان همه چیز را پوشاند.

# مقدمه

بحث این جلسه خیلی گسترده است و نمی‌توان همه چیز را پوشاند.

در واقع همراه با هر روش آماری، نمایش‌های گرافیکی برای بررسی وضعیت موجود، تحلیل نیکویی مدل در نظر گرفته شده، پیدا کردن برخی مشکلات، بهتر کردن مدل قبلی و خیلی موارد دیگر، به کار می‌روند.

# مقدمه

بحث این جلسه خیلی گسترده است و نمی‌توان همه چیز را پوشاند.

در واقع همراه با هر روش آماری، نمایش‌های گرافیکی برای بررسی وضعیت موجود، تحلیل نیکویی مدل در نظر گرفته شده، پیدا کردن برخی مشکلات، بهتر کردن مدل قبلی و خیلی موارد دیگر، به کار می‌روند.

نمایش گرافیکی، بیشتر مربوط به تحلیل اکتشافی داده‌ها ($ExploratoryDataAnalysis$) ($EDA$) و نمودارهای آماری است.

# مقدمه

بحث این جلسه خیلی گسترده است و نمی‌توان همه چیز را پوشاند.

در واقع همراه با هر روش آماری، نمایش‌های گرافیکی برای بررسی وضعیت موجود، تحلیل نیکویی مدل در نظر گرفته شده، پیدا کردن برخی مشکلات، بهتر کردن مدل قبلی و خیلی موارد دیگر، به کار می‌روند.

نمایش گرافیکی، بیشتر مربوط به تحلیل اکتشافی داده‌ها ($ExploratoryDataAnalysis$) ($EDA$) و نمودارهای آماری است.

اصطلاح اکتشافی در مقابل تاییدی است که روش‌های تاییدی را می‌توان به آزمون فرضیه‌ها منتسب کرد.

# مقدمه

بحث این جلسه خیلی گسترده است و نمی‌توان همه چیز را پوشاند.

در واقع همراه با هر روش آماری، نمایش‌های گرافیکی برای بررسی وضعیت موجود، تحلیل نیکویی مدل در نظر گرفته شده، پیدا کردن برخی مشکلات، بهتر کردن مدل قبلی و خیلی موارد دیگر، به کار می‌روند.

نمایش گرافیکی، بیشتر مربوط به تحلیل اکتشافی داده‌ها ($ExploratoryDataAnalysis$) ($EDA$) و نمودارهای آماری است.

اصطلاح اکتشافی در مقابل تاییدی است که روش‌های تاییدی را می‌توان به آزمون فرضیه‌ها منتسب کرد.

در واقع توکی، معتقد بود قبل از آزمون فرضیه، بهتر است تحلیل اکتشافی بر روی داده‌ها انجام شود تا سوالات مناسب برای پرسیدن و روش‌های مناسب برای پاسخ به آن‌ها را بیابیم.

# مقدمه

بحث این جلسه خیلی گسترده است و نمی‌توان همه چیز را پوشاند.

در واقع همراه با هر روش آماری، نمایش‌های گرافیکی برای بررسی وضعیت موجود، تحلیل نیکویی مدل در نظر گرفته شده، پیدا کردن برخی مشکلات، بهتر کردن مدل قبلی و خیلی موارد دیگر، به کار می‌روند.

نمایش گرافیکی، بیشتر مربوط به تحلیل اکتشافی داده‌ها ($ExploratoryDataAnalysis$) ($EDA$) و نمودارهای آماری است.

اصطلاح اکتشافی در مقابل تاییدی است که روش‌های تاییدی را می‌توان به آزمون فرضیه‌ها منتسب کرد.

در واقع توکی، معتقد بود قبل از آزمون فرضیه، بهتر است تحلیل اکتشافی بر روی داده‌ها انجام شود تا سوالات مناسب برای پرسیدن و روش‌های مناسب برای پاسخ به آن‌ها را بیابیم.

در این جلسه، قسمتی از موارد مهم و شاید کمی جذاب‌تر را (از دیدگاه مدرس) مطرح می‌کنیم.

# نمودارهای چندک ـ چندک

یکی از مواردی که در مورد داده‌ها مایلیم بدانیم، این است که آیا آن‌ها از توزیع نرمال پیروی می‌کنند؟ آیا از یک توزیع شناخته‌شده پیروی می‌کنند؟

# نمودارهای چندک ـ چندک

یکی از مواردی که در مورد داده‌ها مایلیم بدانیم، این است که آیا آن‌ها از توزیع نرمال پیروی می‌کنند؟ آیا از یک توزیع شناخته‌شده پیروی می‌کنند؟

در بعضی از موارد، واضح است که داده‌ها مثلا از نرمال پیروی نمی‌کنند. زیرا به عنوان مثال چگالی داده‌ها دو مدی است. اما مواردی هم وجود دارند که چنین آگاهی واضح و روشن نیست.

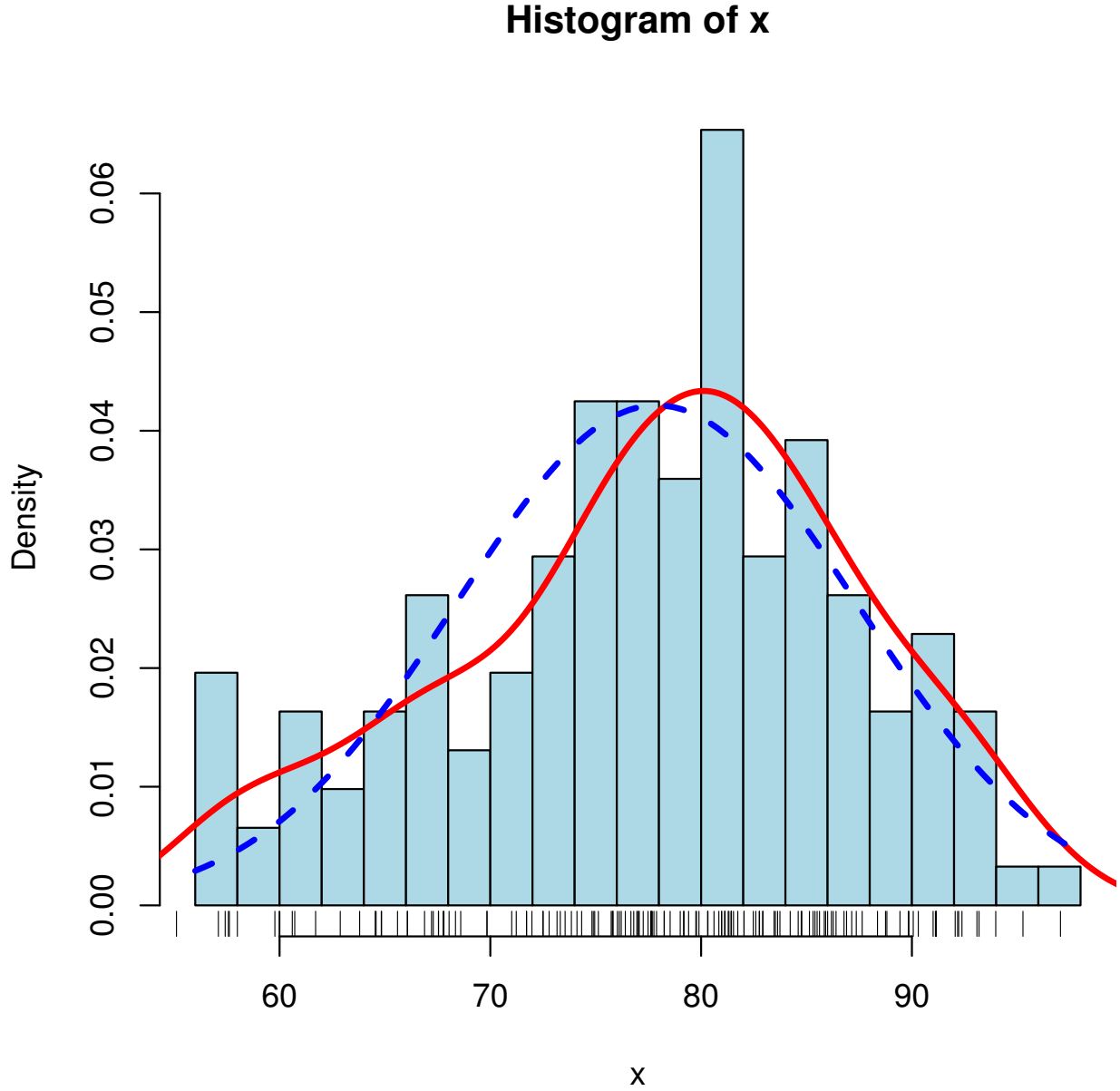
# نمودارهای چندک ـ چندک

یکی از مواردی که در مورد داده‌ها مایلیم بدانیم، این است که آیا آن‌ها از توزیع نرمال پیروی می‌کنند؟ آیا از یک توزیع شناخته‌شده پیروی می‌کنند؟

در بعضی از موارد، واضح است که داده‌ها مثلا از نرمال پیروی نمی‌کنند. زیرا به عنوان مثال چگالی داده‌ها دو مدی است. اما مواردی هم وجود دارند که چنین آگاهی واضح و روشن نیست.

یک راه برای بررسی این موضوع، مقایسه چگالی برآوردشده با چگالی توزیع نرمال است.

```
data(airquality)
x <- airquality[,4]
hist(x, probability=TRUE, breaks=20, col="light blue")
rug(jitter(x, 5))
points(density(x), type='l', lwd=3, col='red')
f <- function(t) {
  dnorm(t, mean=mean(x), sd=sd(x) )
}
curve(f, add=T, col="blue", lwd=3, lty=2)
```


Histogram of x
Density
0.00
0.01
0.02
0.03
0.04
0.05
0.06
60
70
80
90
x

# راه حل دوم

روش دیگر که یک روش بصری است، رسم چندک‌های نمونه‌ای در مقابل چندک‌های نرمال است. به این نمودار، چندک‌ـ‌چندک گویند.

# راه حل دوم

روش دیگر که یک روش بصری است، رسم چندک‌های نمونه‌ای در مقابل چندک‌های نرمال است. به این نمودار، چندک‌ـ‌چندک گویند.

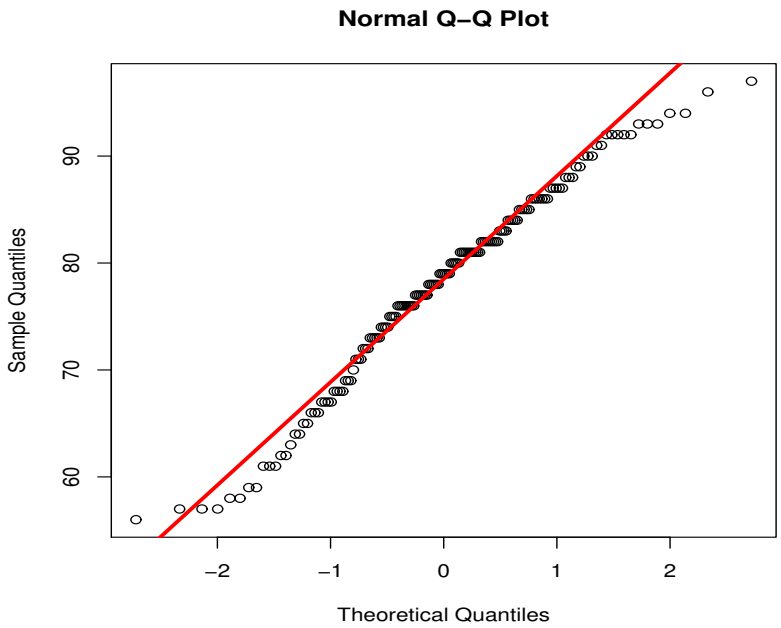
در نمودار زیر به نظر می‌رسد داده‌ها تقریبا نرمال هستند، اما می‌توان دید که داده‌ها گسسته‌اند!!

# راه حل دوم

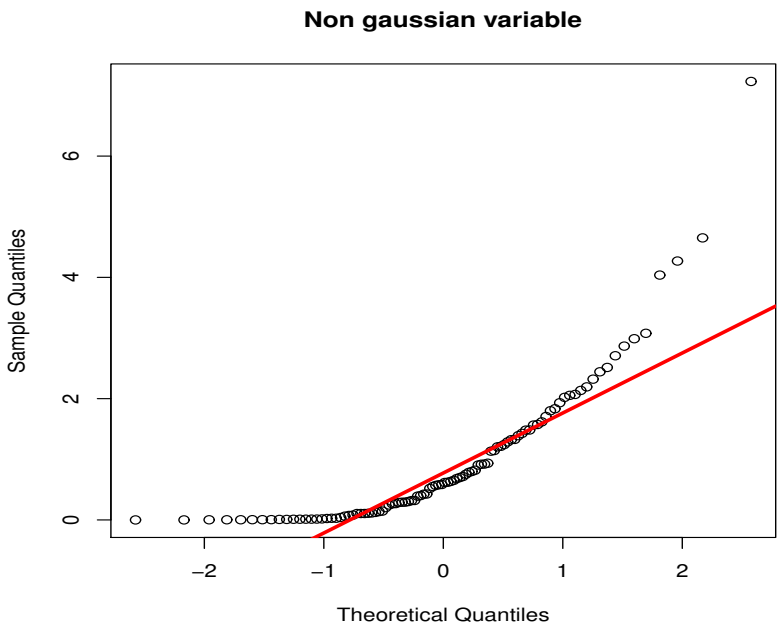
روش دیگر که یک روش بصری است، رسم چندک‌های نمونه‌ای در مقابل چندک‌های نرمال است. به این نمودار، چندک‌ـ‌چندک گویند.

در نمودار زیر به نظر می‌رسد داده‌ها تقریبا نرمال هستند، اما می‌توان دید که داده‌ها گسسته‌اند!!

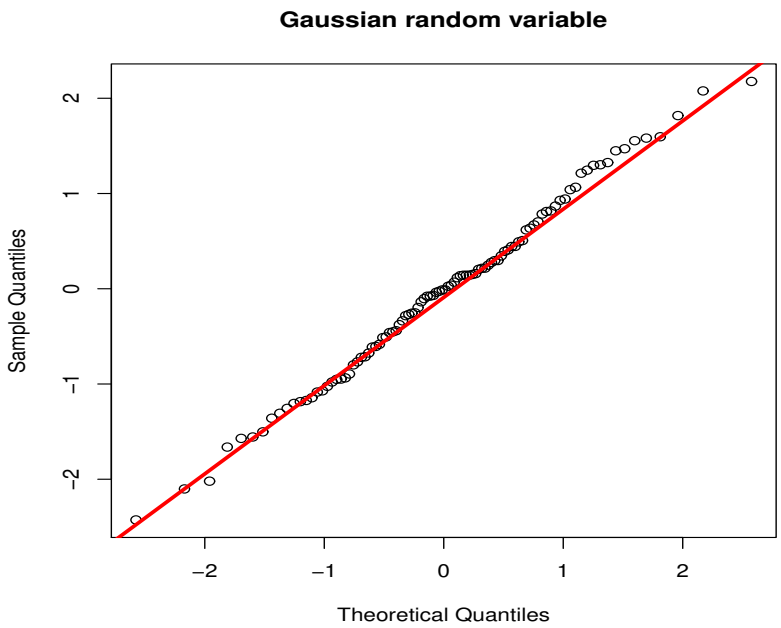
```
x <- airquality[,4]
qqnorm(x)
qqline(x, col="red", lwd=3)
```



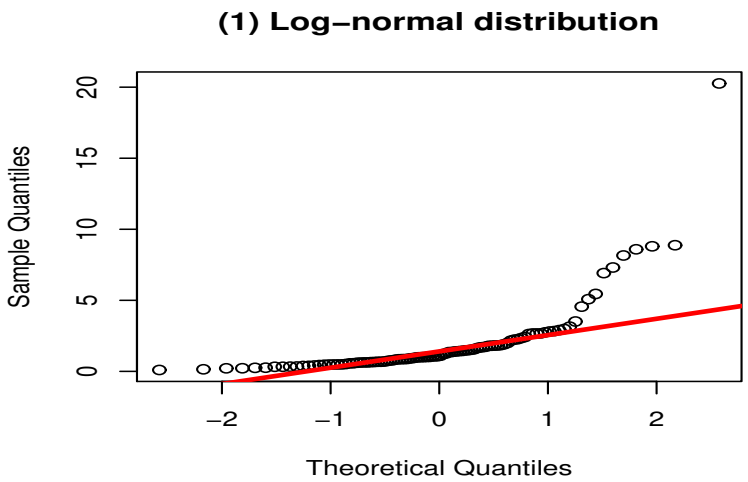
```
y <- rnorm(100)^2
qqnorm(y, main="Non gaussian variable")
qqline(y, col="red", lwd=3)
```

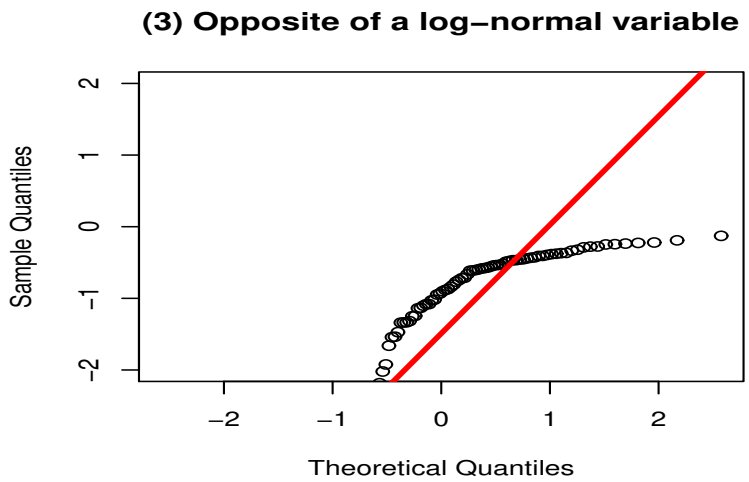


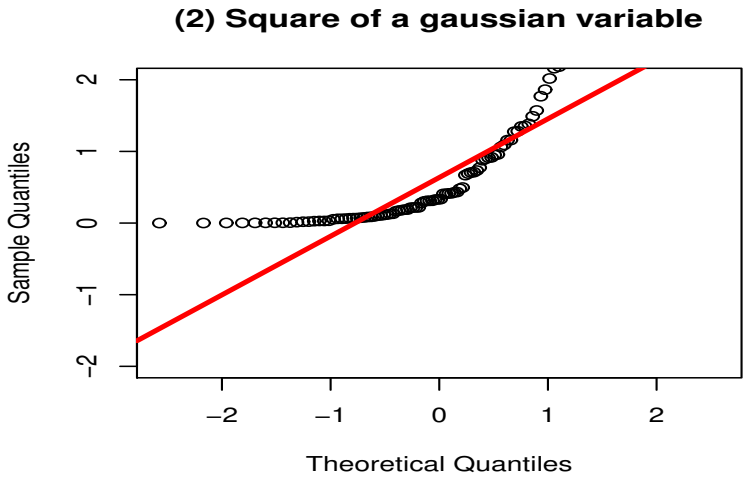

```
y <- rnorm(100)
qqnorm(y, main="Gaussian random variable")
qqline(y, col="red", lwd=3)
```

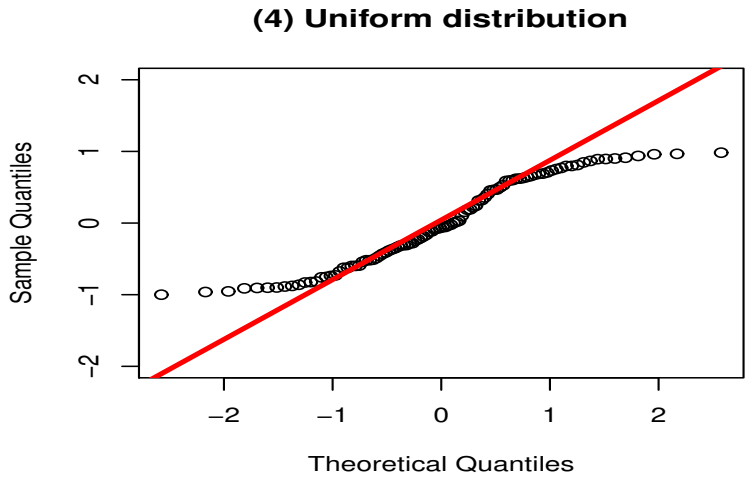


# سایر مثال‌ها

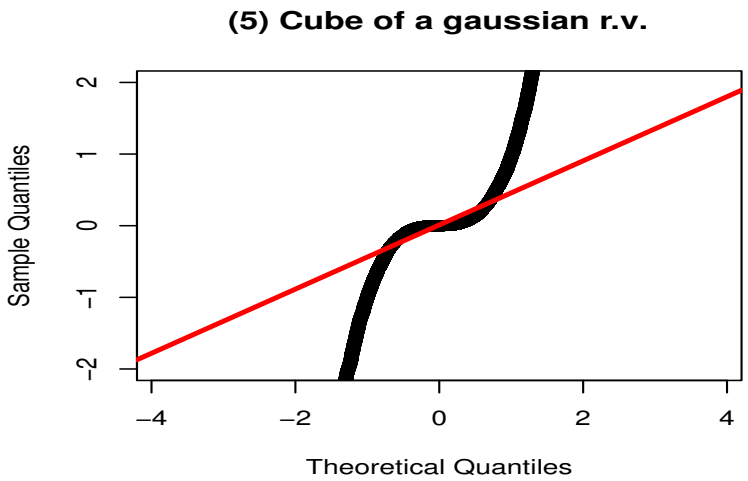

(1) Log−normal distribution
Sample Quantiles
Theoretical Quantiles
0
5
10
15
20
−2
−1
0
1
2

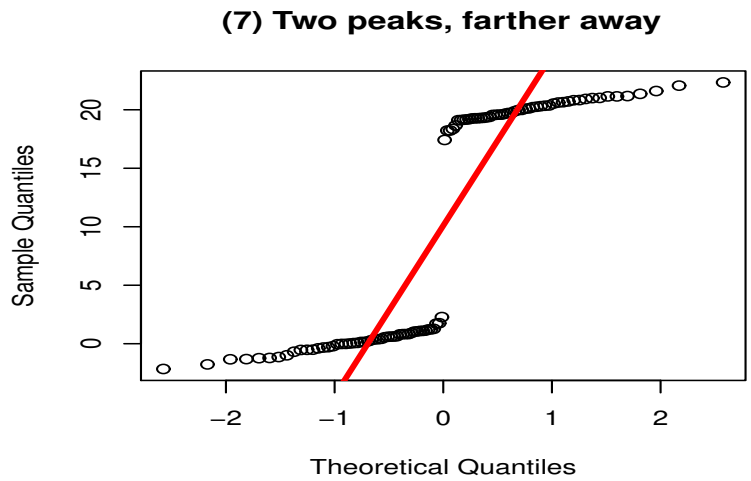

(3) Opposite of a log−normal variable
Sample Quantiles
Theoretical Quantiles
−2
−1
0
1
2

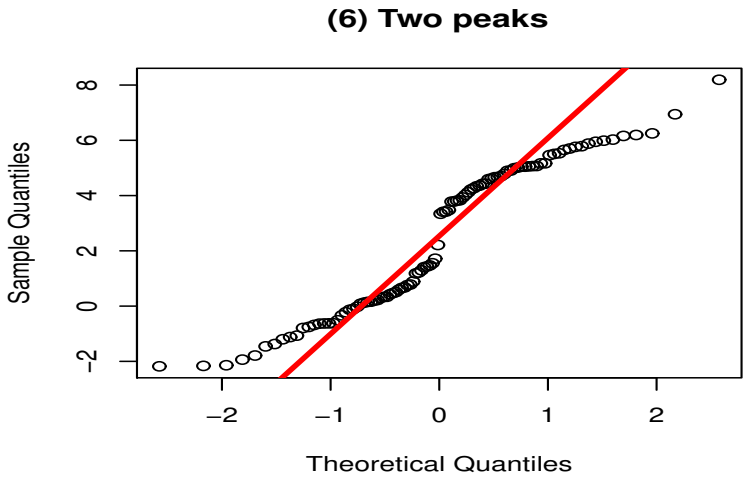

(2) Square of a gaussian variable
Sample Quantiles
Theoretical Quantiles
−2
−1
0
1
2

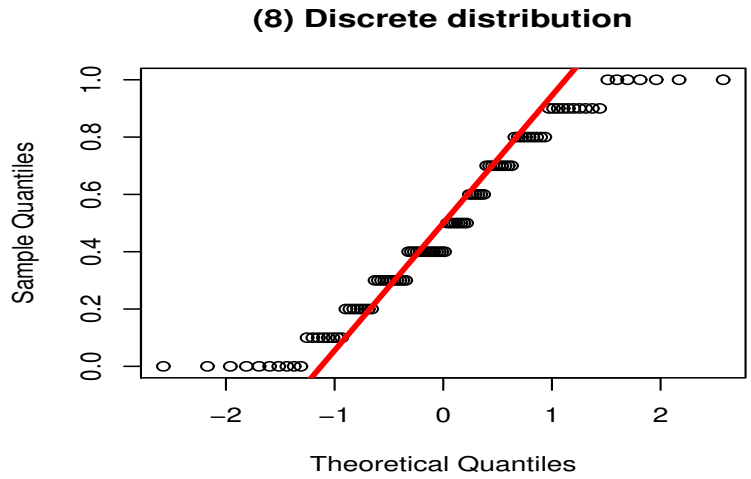

(4) Uniform distribution
Sample Quantiles
Theoretical Quantiles
−2
−1
0
1
2

# سایر مثال‌ها


(5) Cube of a gaussian r.v.
Sample Quantiles
Theoretical Quantiles
−4
−2
0
2
4
−2
−1
0
1
2

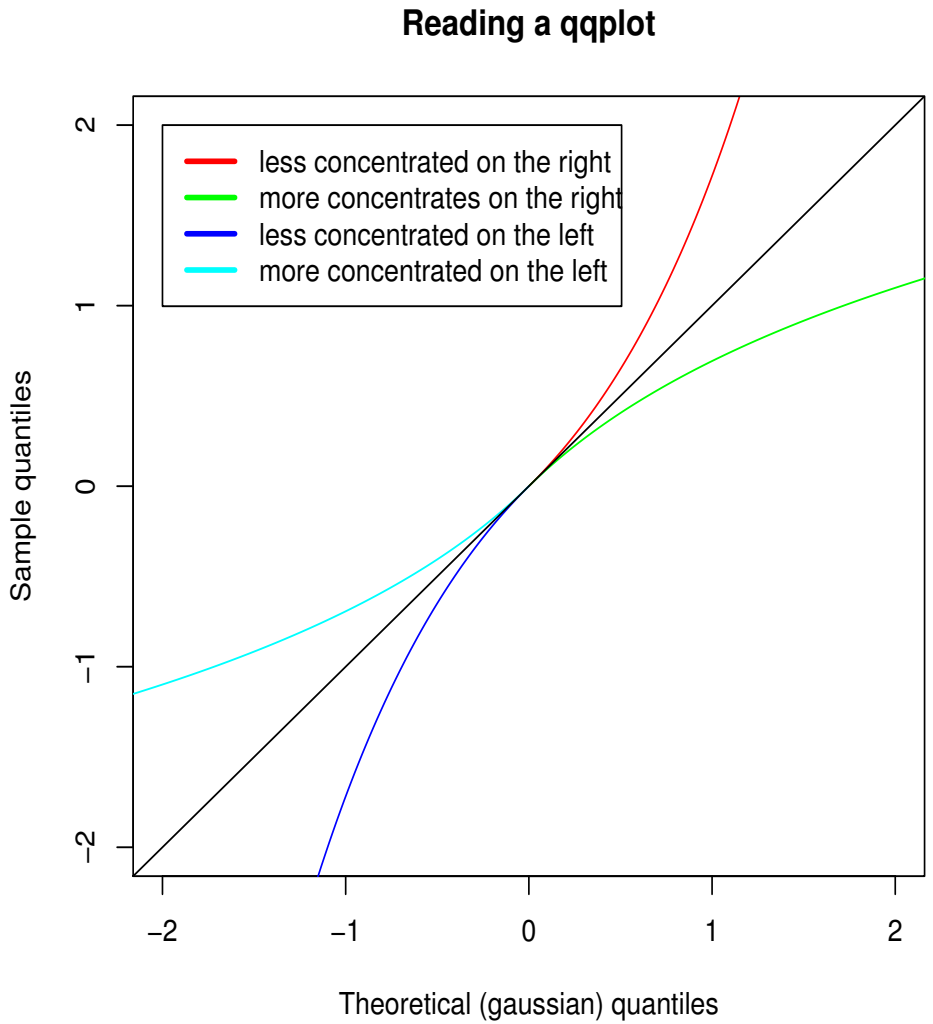

(7) Two peaks, farther away
Sample Quantiles
Theoretical Quantiles
−2
−1
0
1
2
0
5
10
15
20


(6) Two peaks
Sample Quantiles
Theoretical Quantiles
−2
−1
0
1
2
−2
0
2
4
6
8

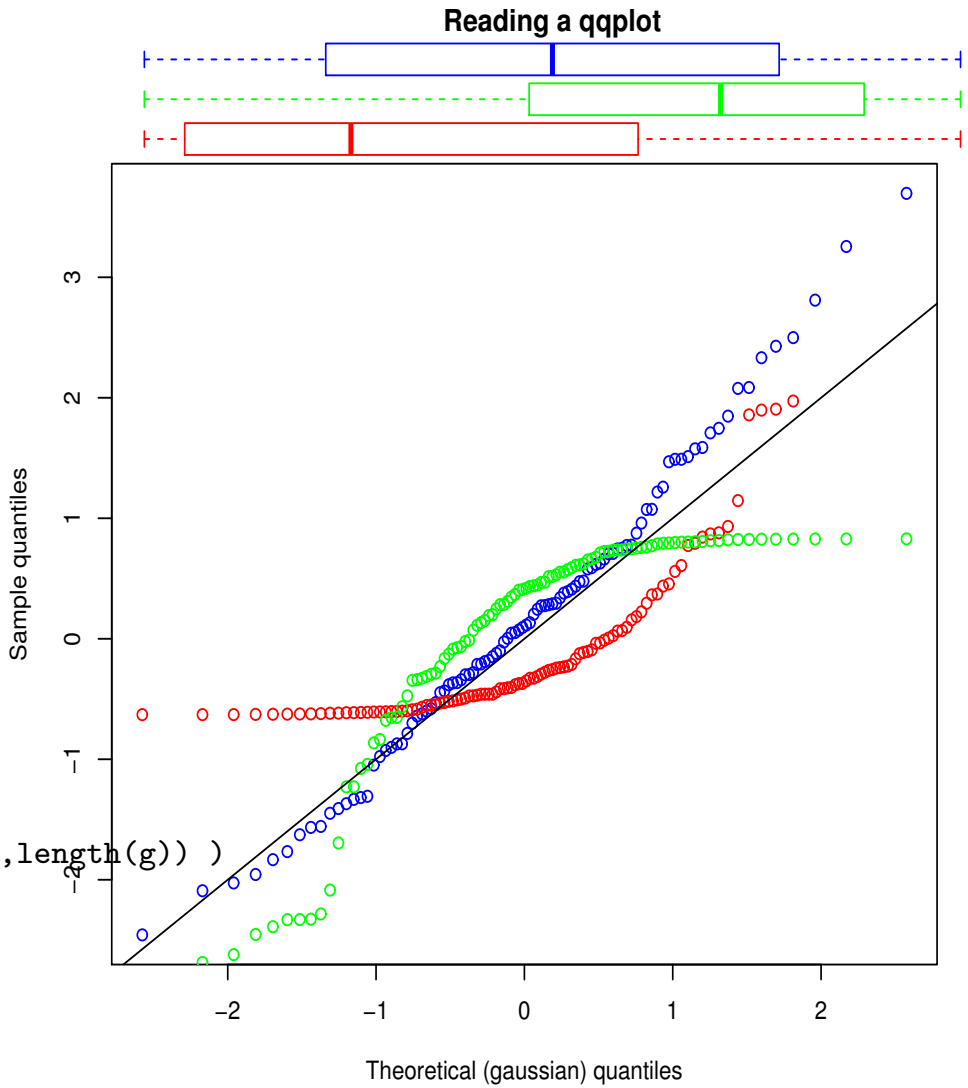

(8) Discrete distribution
Sample Quantiles
Theoretical Quantiles
−2
−1
0
1
2
0.0
0.2
0.4
0.6
0.8
1.0

# دستورالعمل کلی

```
x <- seq(from=0, to=2, length=100)
y <- exp(x)-1
plot( y ~ x, type = 'l', col = 'red',
      xlim = c(-2,2), ylim = c(-2,2),
      xlab = "Theoretical (gaussian) quantiles",
      ylab = "Sample quantiles")
lines( x~y, type='l', col='green')
x <- -x
y <- -y
lines( y~x, type='l', col='blue', )
lines( x~y, type='l', col='cyan')
abline(0,1)
legend( -2, 2,
        c( "less concentrated on the right",
           "more concentrates on the right",
           "less concentrated on the left",
           "more concentrated on the left"
         ),
        lwd=3,
        col=c("red", "green", "blue", "cyan")
      )
title(main="Reading a qqplot")
```



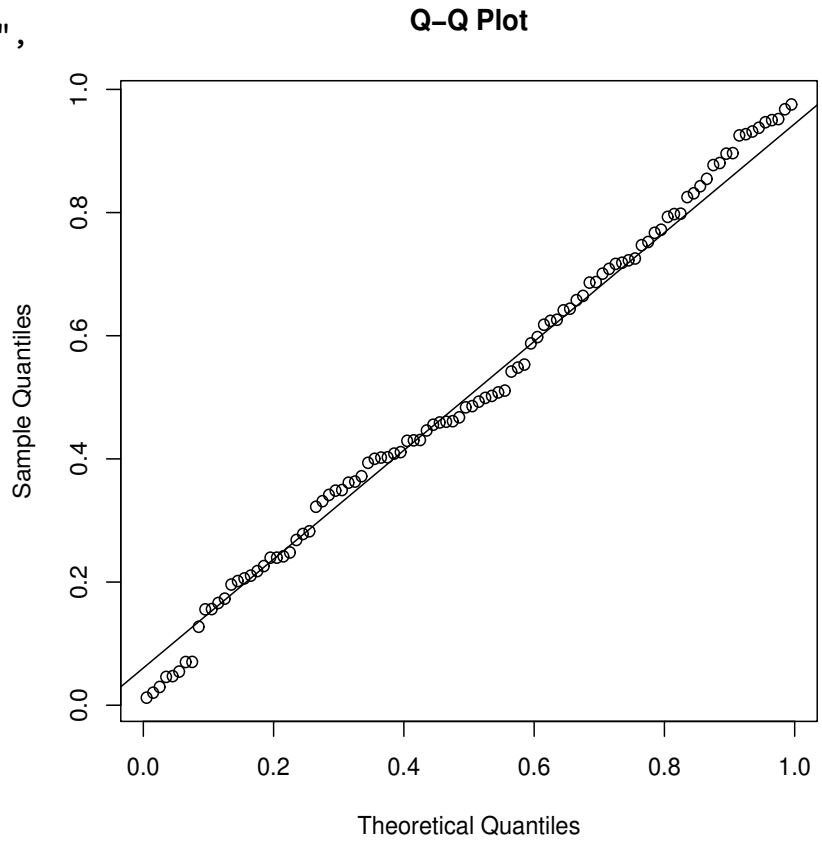
```
op <- par()
layout( matrix( c(2,2,1,1), 2, 2, byrow=T ),
        c(1,1), c(1,6),
      )
# The plot
n <- 100
y <- rnorm(n)
x <- qnorm(ppoints(n))[order(order(y))]
par(mar=c(5.1,4.1,0,2.1))
plot( y ~ x, col = "blue",
      xlab = "Theoretical (gaussian) quantiles",
      ylab = "Sample quantiles" )
y1 <- scale( rnorm(n)^2 )
x <- qnorm(ppoints(n))[order(order(y1))]
lines(y1~x, type="p", col="red")
y2 <- scale( -rnorm(n)^2 )
x <- qnorm(ppoints(n))[order(order(y2))]
lines(y2~x, type="p", col="green")
abline(0,1)
# The legend
par(bty='n', ann=F)
g <- seq(0,1, length=10)
e <- g^2
f <- sqrt(g)
h <- c( rep(1,length(e)), rep(2,length(f)), rep(3,length(g)) )
par(mar=c(0,4.1,1,0))
boxplot( c(e,f,g) ~ h, horizontal=T,
         border=c("red", "green", "blue"),
         col="white", # Something prettier?
         xaxt='n',
         yaxt='n',
         )
title(main="Reading a qqplot")
par(op)
```



# چندک_چندک برای سایر توزیع‌ها

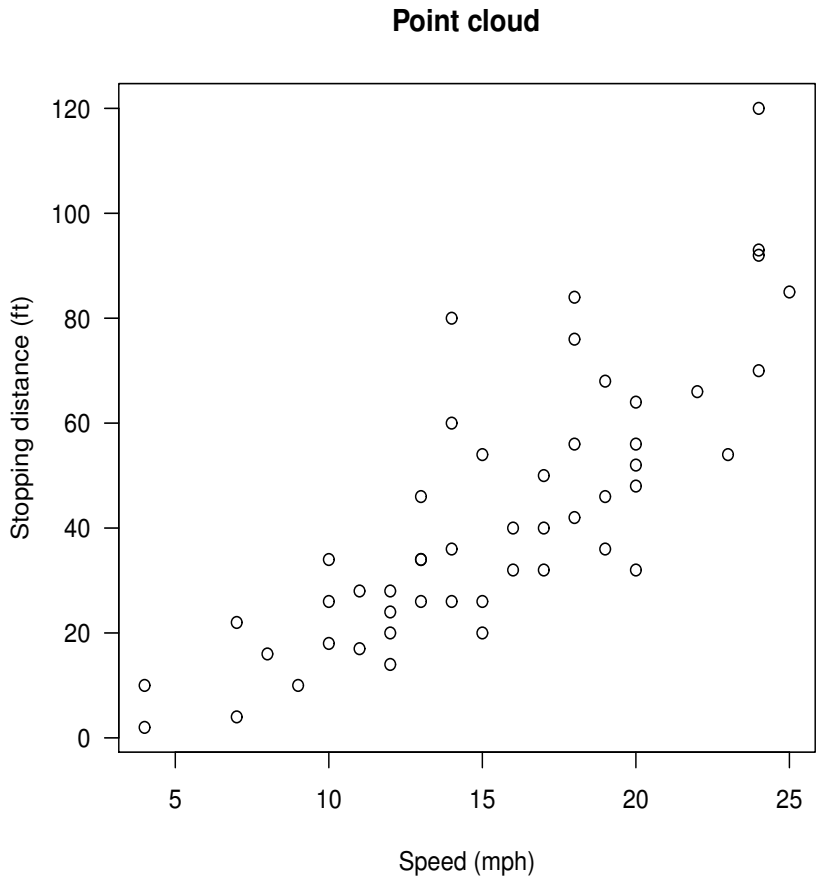
می‌توان برای سایر توزیع‌ها از همین ایده نمودار چندک_چندک نرمال استفاده کرد

```
qq <- function (y, ylim, quantiles=qnorm,
    main = "Q-Q Plot", xlab = "Theoretical Quantiles",
    ylab = "Sample Quantiles", plot.it = TRUE, ...)
{
  y <- y[!is.na(y)]
  if (0 == (n <- length(y)))
    stop("y is empty")
  if (missing(ylim))
    ylim <- range(y)
  x <- quantiles(ppoints(n))[order(order(y))]
  if (plot.it)
    plot(x, y, main = main, xlab = xlab,
    ylab = ylab, ylim = ylim, ...)
  # From qqline
  y <- quantile(y, c(0.25, 0.75))
  x <- quantiles(c(0.25, 0.75))
  slope <- diff(y)/diff(x)
  int <- y[1] - slope * x[1]
  abline(int, slope, ...)
  invisible(list(x = x, y = y))
}
y <- runif(100)
qq(y, quantiles=qunif)
```



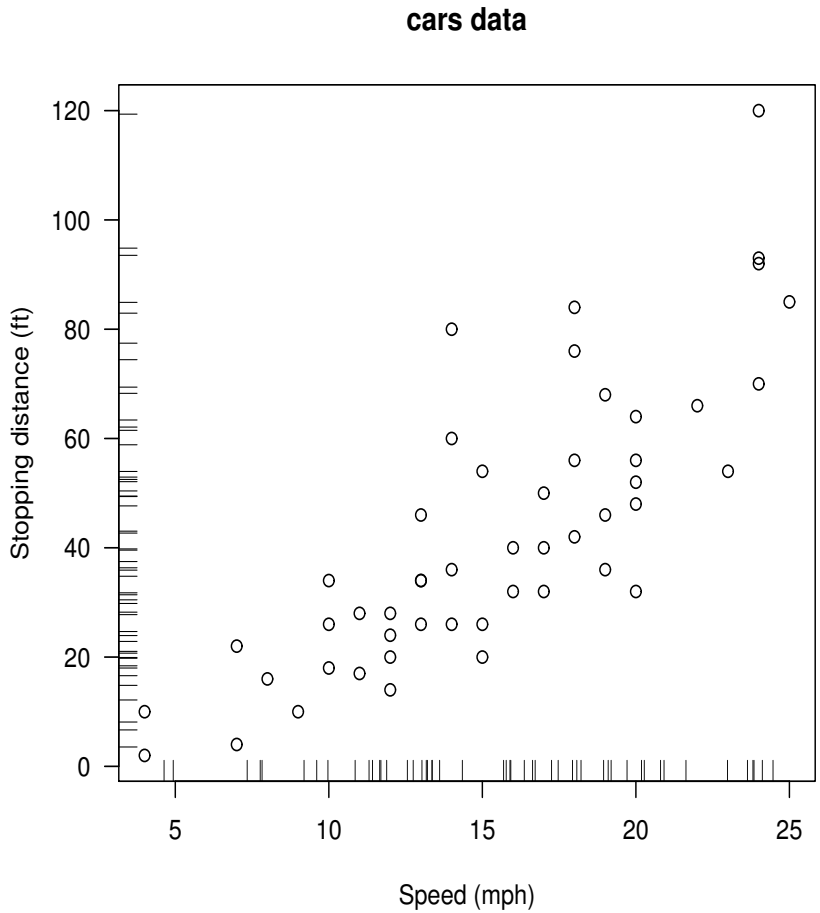
# نمودارهای پراکنش

```
data(cars)
plot(cars$dist ~ cars$speed,
     xlab = "Speed (mph)",
     ylab = "Stopping distance (ft)",
     las = 1)
title(main = "Point cloud")
```



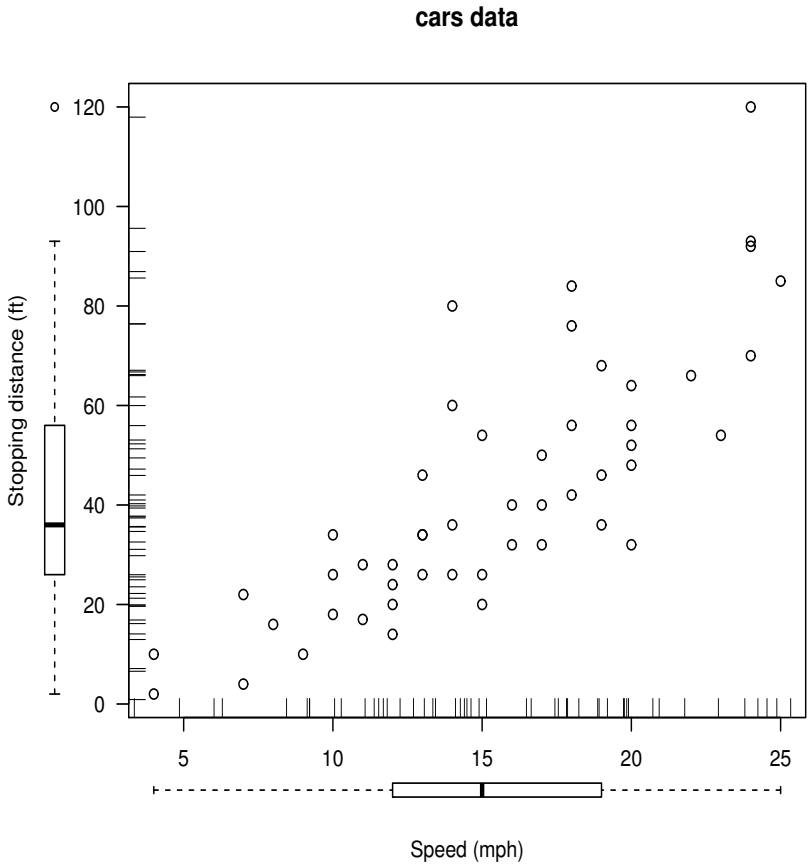
می‌توان نمودارهای پراکنش تک‌بعدی را هم به حاشیه نمودار اضافه کرد.

```
plot(cars$dist ~ cars$speed,
     xlab = "Speed (mph)",
     ylab = "Stopping distance (ft)",
     las = 1)
title(main = "cars data")
rug(side=1, jitter(cars$speed, 5))
rug(side=2, jitter(cars$dist, 20))
```



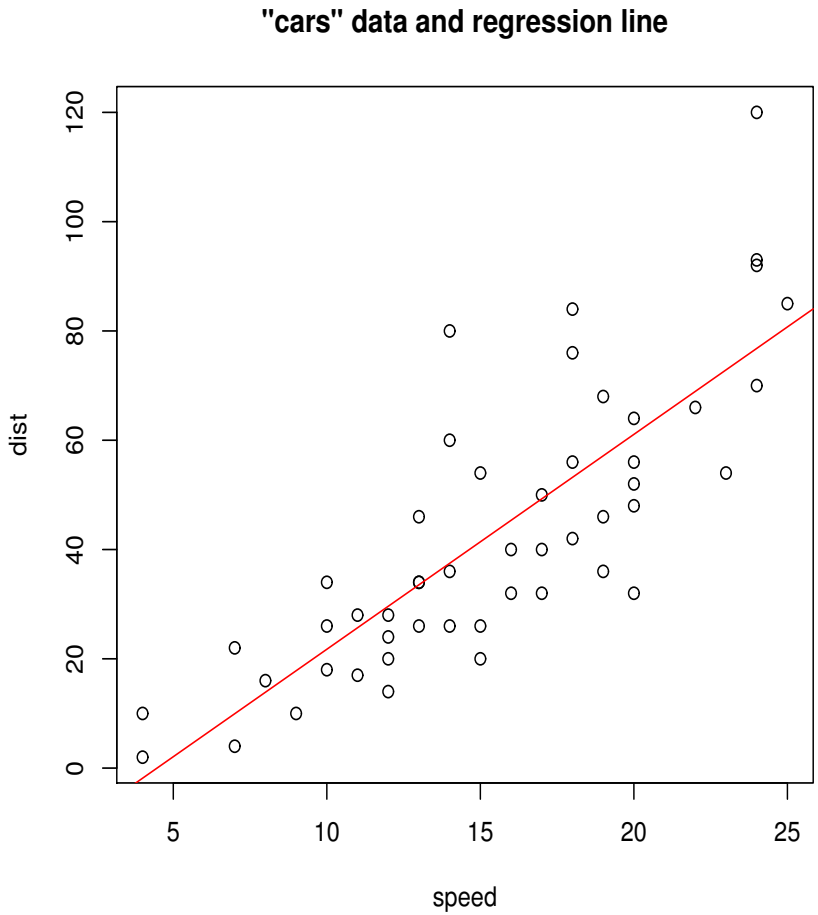
حتی می‌توان نمودارهای جعبه‌ای حاشیه‌ای هر متغیر را نیز همراه با نمودار پراکنش رسم کرد.

```
layout( matrix( c(2,1,0,3), 2, 2, byrow=T ),
        c(1,6), c(4,1),
      )

par(mar=c(1,1,5,2))
plot(cars$dist ~ cars$speed,
     xlab='', ylab='',
     las = 1)
rug(side=1, jitter(cars$speed, 5) )
rug(side=2, jitter(cars$dist, 20) )
title(main = "cars data")

par(mar=c(1,2,5,1))
boxplot(cars$dist, axes=F)
title(ylab='Stopping distance (ft)', line=0)

par(mar=c(5,1,1,2))
boxplot(cars$speed, horizontal=T, axes=F)
title(xlab='Speed (mph)', line=1)
```



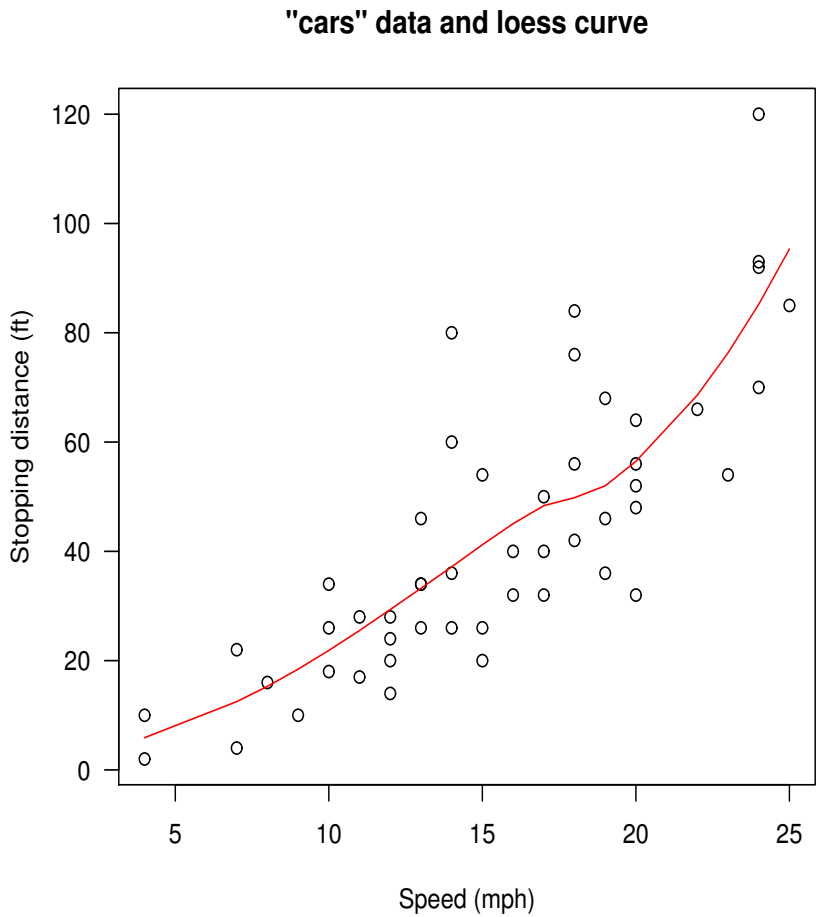
می‌توان یک تقریب خطی برای رابطه بین دو متغیر به نمودار اضافه کرد.

```
plot(dist ~ speed, data = cars,
     main = "\"cars\" data and regression line")
abline(lm( dist ~ speed, data = cars), col = 'red')
```



می‌توان مثلا از تابع *loess* برای تقریب یک منحنی (نه لزوما خطی) برای رابطه دو متغیر استفاده کرد.

```
plot(cars, xlab = "Speed (mph)",
     ylab = "Stopping distance (ft)",
     las = 1)
r <- loess(dist ~ speed, data=cars)
lines(r$x, r$fitted, col="red")
title(main = "\"cars\" data and loess curve")
```

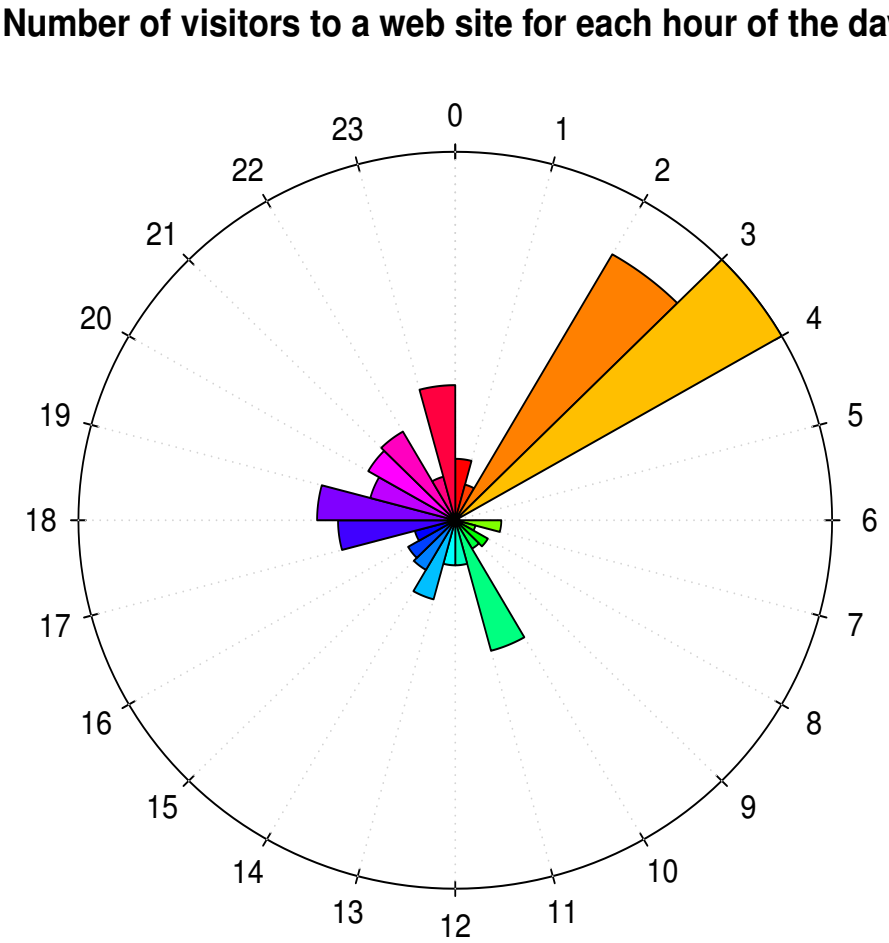


# نمودارهای مرتبط با یک متغیر دوره‌ای

اگر یک متغیر، دوره‌ای باشد، مانند ساعت‌های یک روز، نمایش آن‌ها به عنوان یک دایره گاهی مواقع نمودار خیلی مفیدی است. به عنوان مثال، تعداد بازدید از یک سایت را به عنوان تابعی از زمان روز در نظر بگیرید:

```
x <- c(15, 9, 75, 90, 1, 1, 11, 5, 9, 8, 33, 11, 11,
       20, 14, 13, 10, 28, 33, 21, 24, 25, 11, 33)
clock.plot <- function (x, col = rainbow(n), ...) {
  if( min(x)<0 ) x <- x - min(x)
  if( max(x)>1 ) x <- x/max(x)
  n <- length(x)
  if(is.null(names(x))) names(x) <- 0:(n-1)
  m <- 1.05
  plot(0,
       type = 'n', # do not plot anything
       xlim = c(-m,m), ylim = c(-m,m),
       axes = F, xlab = '', ylab = '', ...)
  a <- pi/2 - 2*pi/200*0:200
  polygon( cos(a), sin(a) )
  v <- .02
  a <- pi/2 - 2*pi/n*0:n
  segments( (1+v)*cos(a), (1+v)*sin(a),
            (1-v)*cos(a), (1-v)*sin(a) )
  segments( cos(a), sin(a),
            0, 0,
            col = 'light grey', lty = 3)
  ca <- -2*pi/n*(0:50)/50
  for (i in 1:n) {
    a <- pi/2 - 2*pi/n*(i-1)
    b <- pi/2 - 2*pi/n*i
    polygon( c(0, x[i]*cos(a+ca), 0),
             c(0, x[i]*sin(a+ca), 0),
             col=col[i] )
    v <- .1
    text((1+v)*cos(a), (1+v)*sin(a), names(x)[i])
  }
}
clock.plot(x,
main = "No. of visitors to a web site for each hour")
```

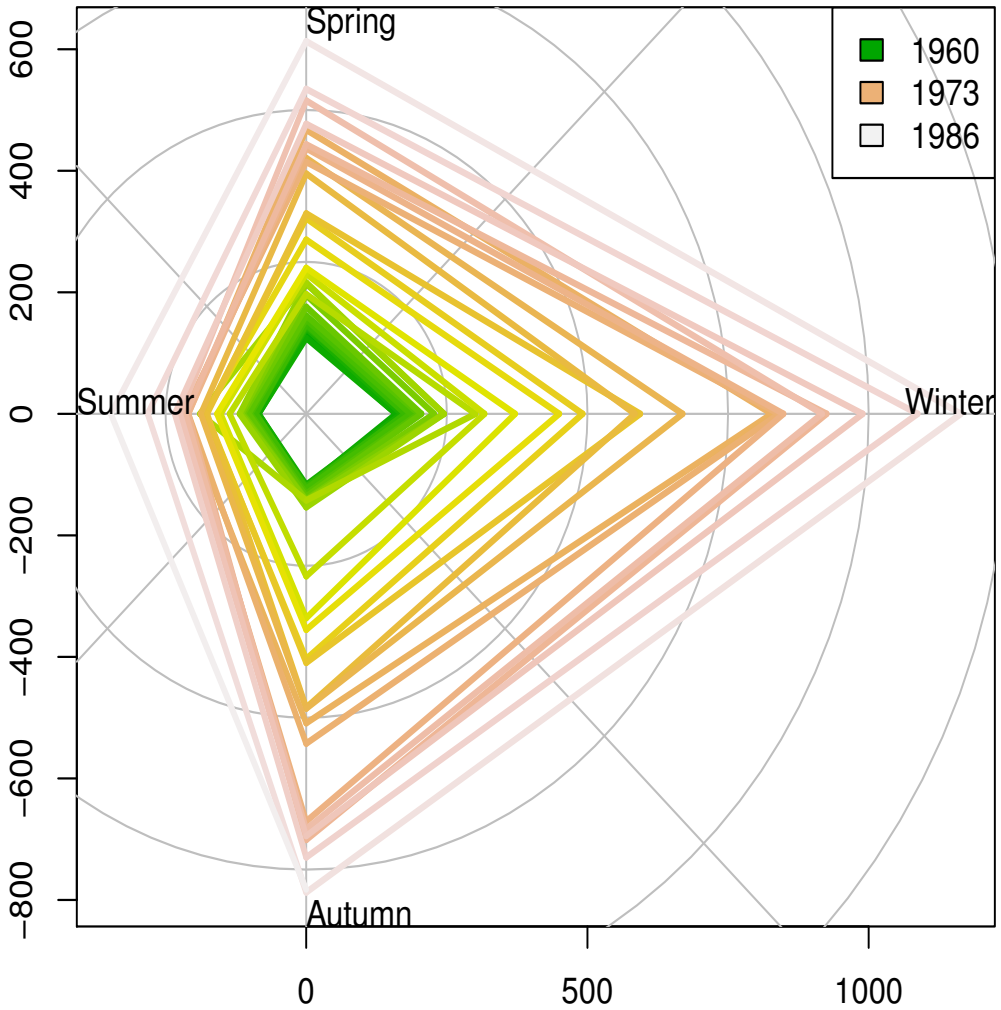
در مورد نوع داده‌های این نمودار، می‌توانید به راهنمای بسته $circular$ مراجعه کنید. همچنین نمودار مشابهی را می‌توان با استفاده از بسته $plotrix$ تولید کرد.



# نمودار قطبی

یک نمودار مفید مشابه نمودار دوره‌ای بالا، نمودار قطبی است:

```
# Polar plot to spot seasonal patterns
x <- as.vector(UKgas)
n <- length(x)
theta <- seq(0, by=2*pi/4, length=n)
plot(x * cos(theta), x * sin(theta),
     type = "l",
     xlab = "", ylab = "",
     main = "UK gas consumption")
abline(h=0, v=0, col="grey")
abline(0, 1, col="grey")
abline(0, -1, col="grey")
circle <- function (x, y, r, N=100, ...) {
  theta <- seq(0, 2*pi, length=N+1)
  lines(x + r * cos(theta), y + r * sin(theta), ...)
}
circle(0,0, 250, col="grey")
circle(0,0, 500, col="grey")
circle(0,0, 750, col="grey")
circle(0,0, 1000, col="grey")
circle(0,0, 1250, col="grey")
segments( x[-n] * cos(theta[-n]),
          x[-n] * sin(theta[-n]),
          x[-1] * cos(theta[-1]),
          x[-1] * sin(theta[-1]),
          col = terrain.colors(length(x)), lwd = 3)
text(par("usr")[2], 0, "Winter", adj=c(1,0))
text(0, par("usr")[4], "Spring", adj=c(0,1))
text(par("usr")[1], 0, "Summer", adj=c(0,0))
text(0, par("usr")[3], "Autumn", adj=c(0,0))
legend("topright", legend = c(1960, 1973, 1986), fill = terrain.colors(3))
```


UK gas consumption
Spring
Summer
Winter
Autumn
1960
1973
1986
600
400
200
0
−200
−400
−600
−800
0
500
1000

# نمودارهای ماتریسی

یکی از مهم‌ترین کاربردهای ابزار گرافیکی، پیدا کردن ساختارها یا گروه‌هایی در داده‌هاست.
نمودارهای ماتریسی یا نمایش پانلی قسمتی از این هدف را می‌توانند انجام دهند.

# نمودارهای ماتریسی

یکی از مهم‌ترین کاربردهای ابزار گرافیکی، پیدا کردن ساختارها یا گروه‌هایی در داده‌هاست. نمودارهای ماتریسی یا نمایش پانلی قسمتی از این هدف را می‌توانند انجام دهند.

به عنوان مثال یک ماتریس پراکنش، نمودارهای پراکنش جفت متغیرهای یک آرایه را نمایش می‌دهد. این نمودارها می‌توانند در شناخت وجود رابطه بین متغیرها (به عنوان یافتن ساختار) مفید باشند.

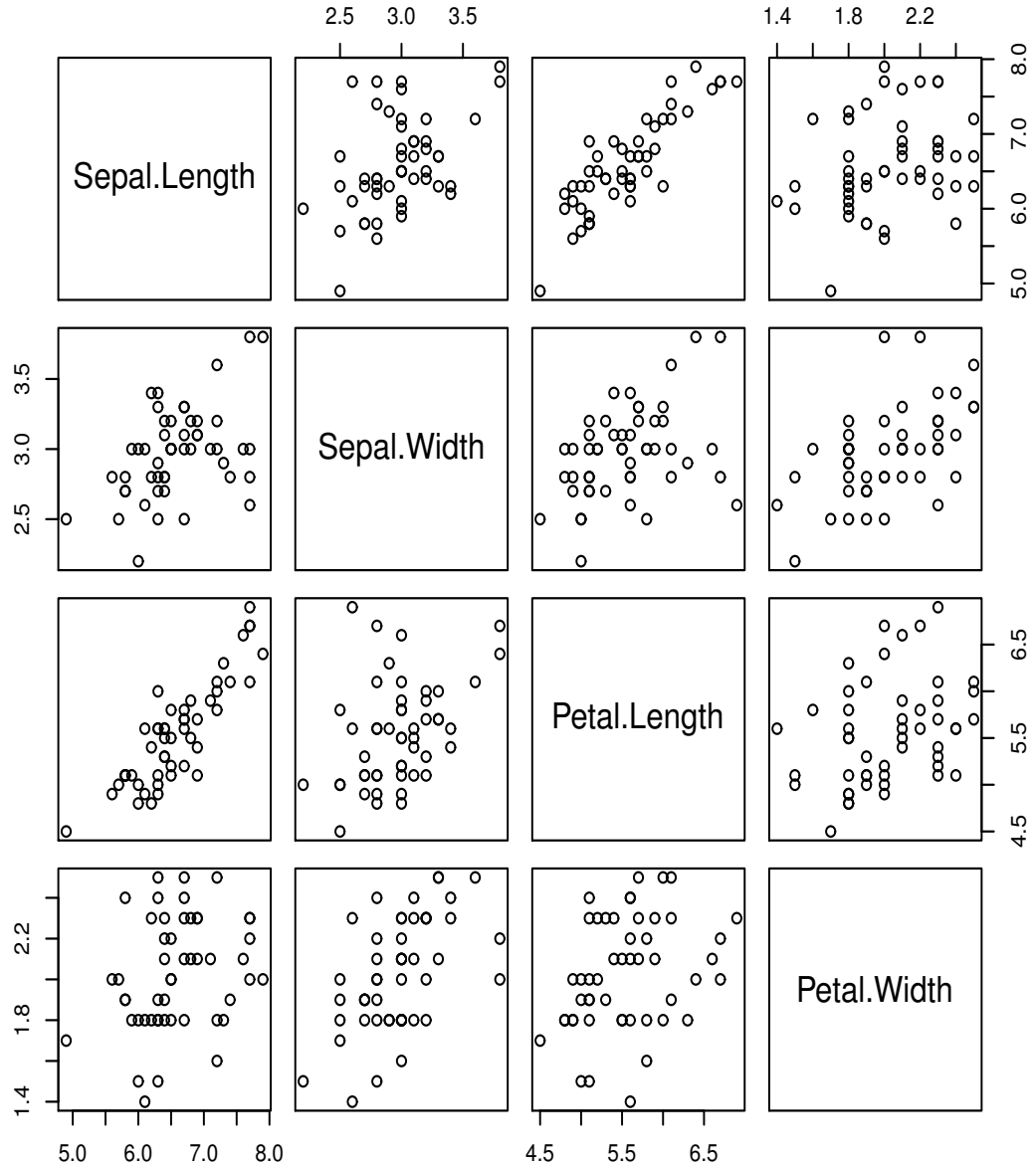
# نمودارهای ماتریسی

یکی از مهم‌ترین کاربردهای ابزار گرافیکی، پیدا کردن ساختارها یا گروه‌هایی در داده‌هاست. نمودارهای ماتریسی یا نمایش پانلی قسمتی از این هدف را می‌توانند انجام دهند.

به عنوان مثال یک ماتریس پراکنش، نمودارهای پراکنش جفت متغیرهای یک آرایه را نمایش می‌دهد. این نمودارها می‌توانند در شناخت وجود رابطه بین متغیرها (به عنوان یافتن ساختار) مفید باشند.

تابع *pairs* در بسته *graphics* چنین ماتریسی را تولید می‌کند

```
data(iris)
pairs(iris[101:150,1:4])
```


Sepal.Length
Sepal.Width
Petal.Length
Petal.Width
2.5 3.0 3.5
1.4 1.8 2.2
5.0 6.0 7.0 8.0
4.5 5.5 6.5

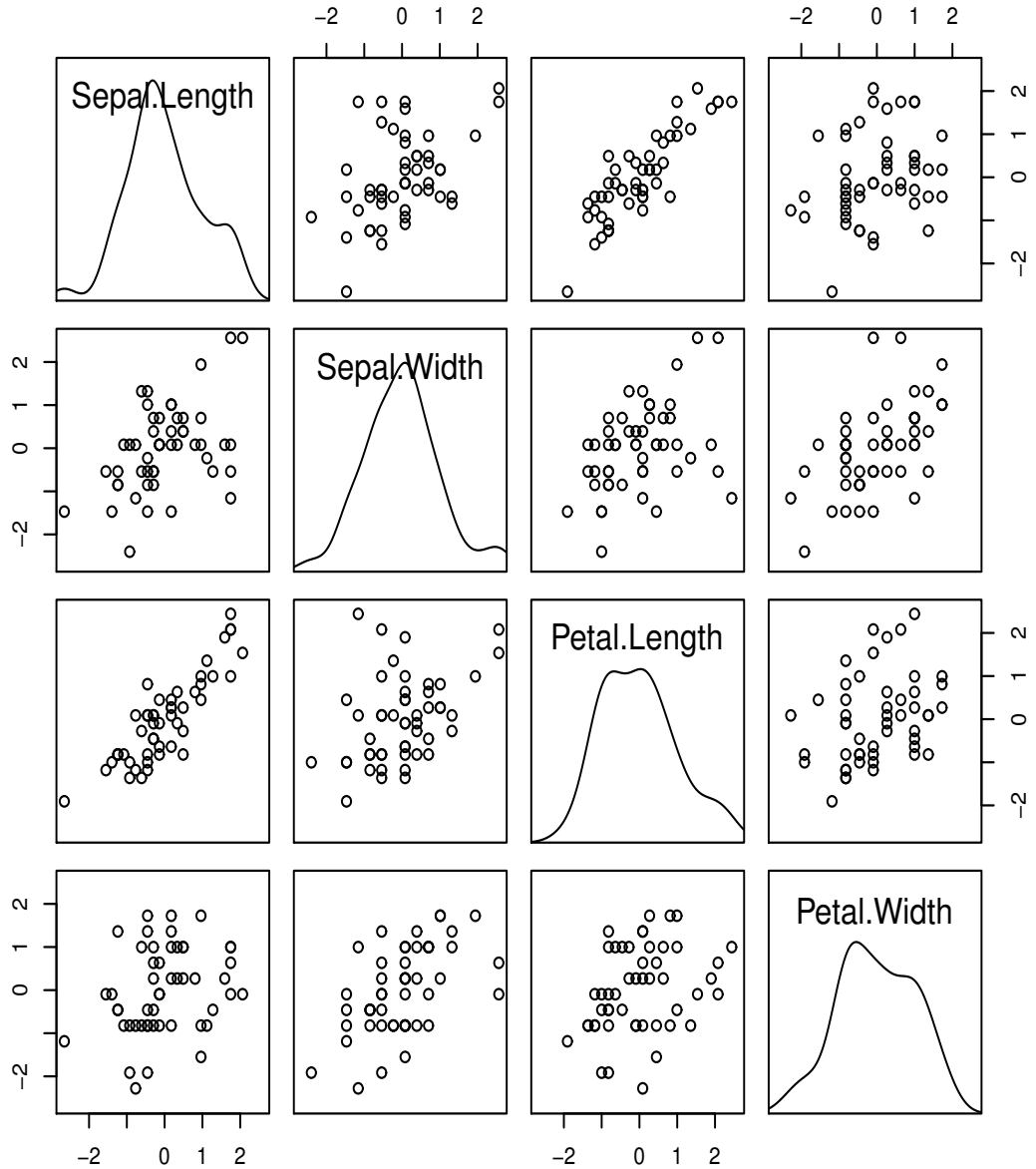
تابع $pairs$ گزینه‌ای دارد با نام $diag.panel$ که تابعی است برای مشخص کردن آن که در روی قطر چه چیزی نمایش داده شود. پیش‌فرض آن نام متغیرهاست.

تابع *pairs* گزینه‌ای دارد با نام *diag.panel* که تابعی است برای مشخص کردن آن که در روی قطر چه چیزی نمایش داده شود. پیش‌فرض آن نام متغیرهاست.

به عنوان مثال، می‌توان تابعی را تعریف کرد تا روی قطر، نمودار توابع چگالی برآوردشده هر متغیر را نمایش دهد:

```
panel.d <- function(x, ...) {
        usr <- par("usr")
        on.exit(par(usr))
        par(usr = c(usr[1:2], 0, .5))
        lines(density(x))
    }
x <- scale(iris[101:150, 1:4])
pairs(x, diag.panel = panel.d, xlim = range(x),
ylim = range(x))
```

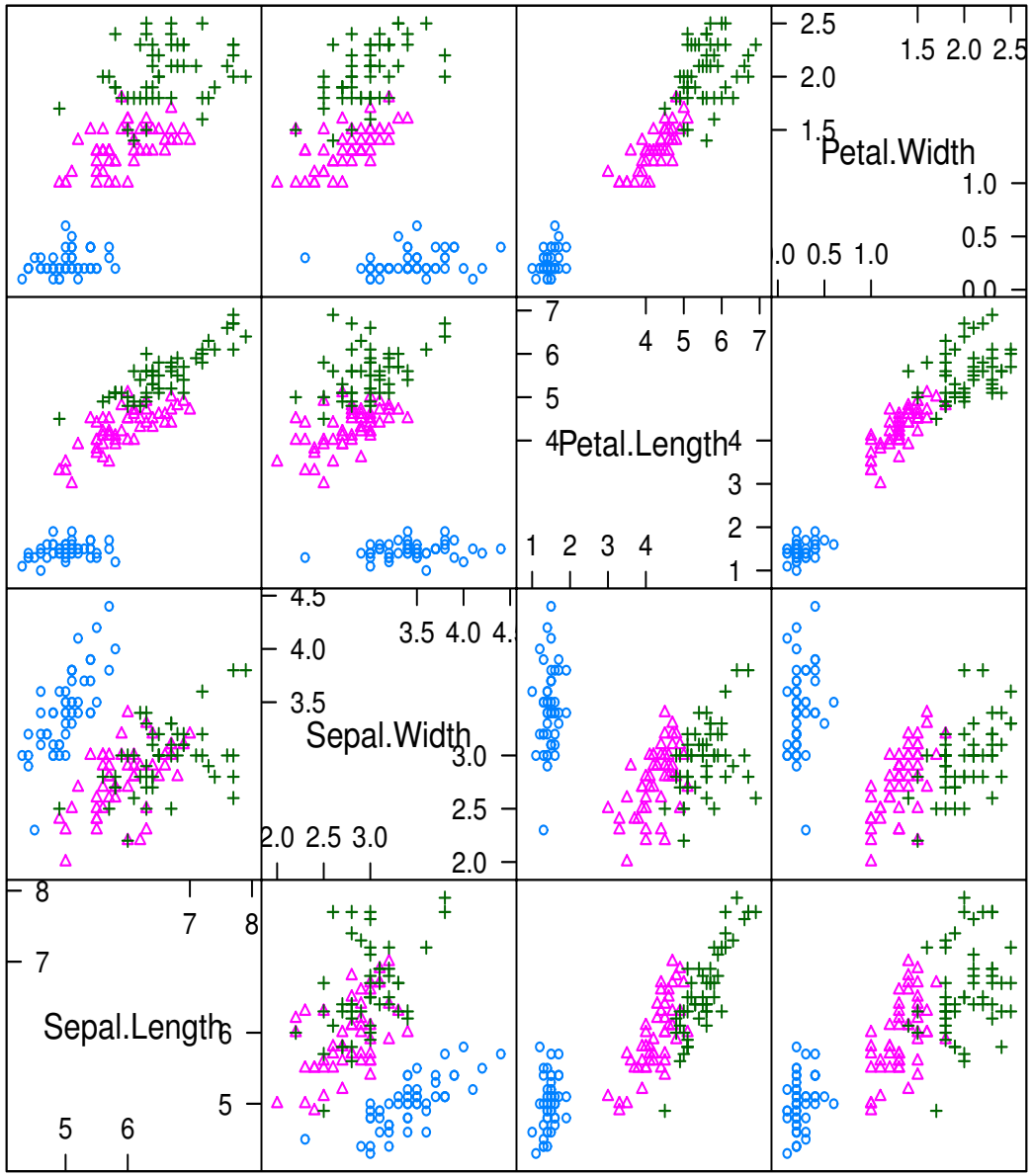
پارامتر *usr* مشخص می‌کند که مقادیر فرین محورها را خود کاربر مشخص می‌کند و دستور *scale* نمونه‌های مربوط به هر متغیر (تکی) را استاندارد می‌کند.


Sepal.Length
Sepal.Width
Petal.Length
Petal.Width
−2
0
1
2

# نمودارهای ماتریسی با استفاده از بسته *lattice*

بسته *lattice* توابعی را برای ساخت نمودارهای ماتریسی و پانلی، فراهم آورده است. به عنوان مثالی قابل مقایسه با ماتریس پراکنش، به مثال زیر توجه کنید:

```
library(lattice)
splom(iris[101:150, 1:4])    # plot 1
splom(iris[,1:4], groups = iris$Species) # plot 2
splom(~iris[1:4], groups = Species, data = iris,
pch = c(1, 2, 3),  cex = c(.5,.5,.5)) #plot 3
```




Scatter Plot Matrix

# سایر موارد استفاده از $lattice$

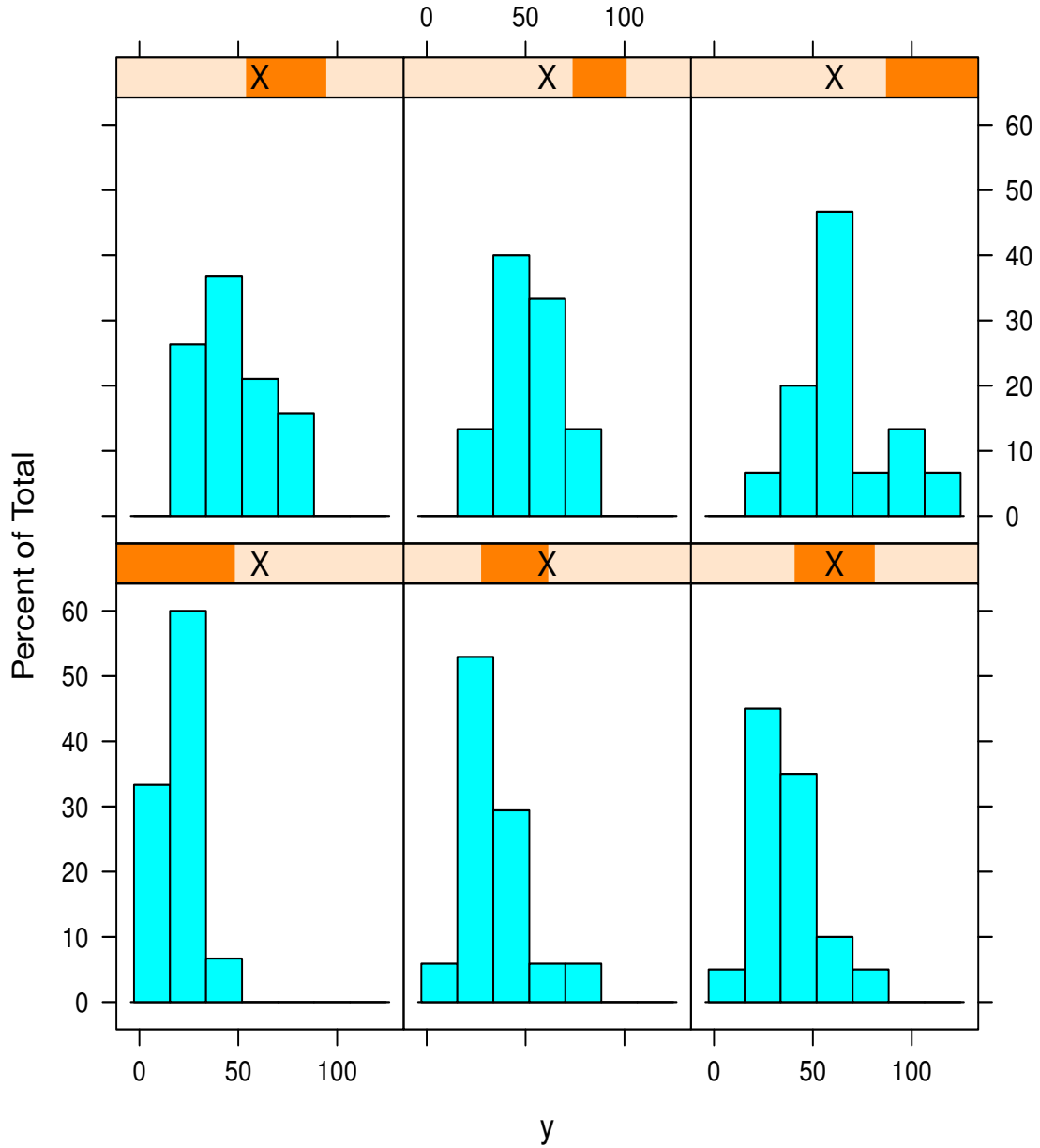
یکی دیگر از موارد استفاده نمودارهای $lattice$، تقسیم کردن مقادیر یک متغیر مانند $y$ و رسم نمودارهای جعبه‌ای، بافت‌نگار و چگالی یک متغیر دیگر مانند $x$ در طبقات حاصل از تقسیم‌بندی است.

# سایر موارد استفاده از *lattice*

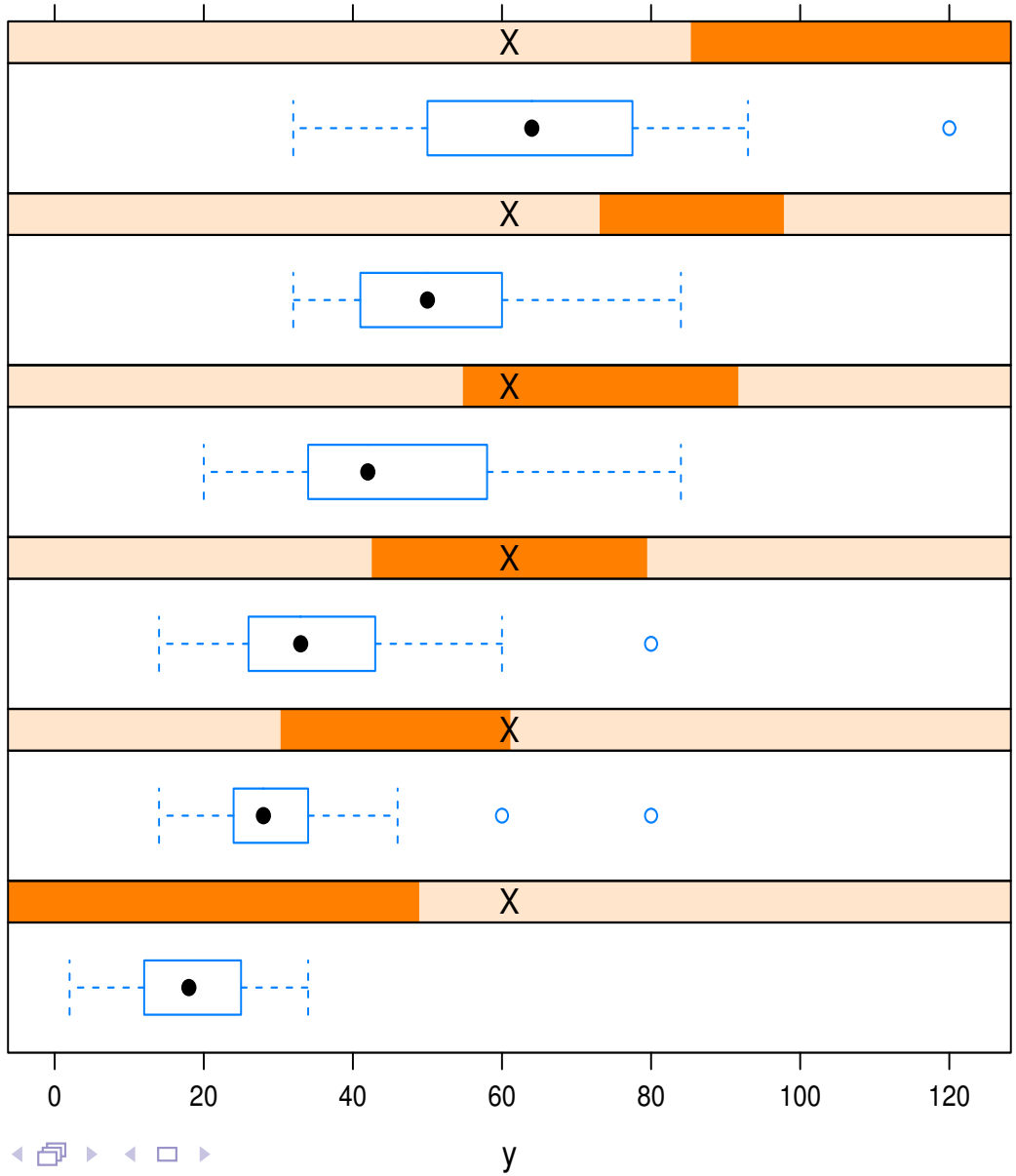
یکی دیگر از موارد استفاده نمودارهای *lattice*، تقسیم کردن مقادیر یک متغیر مانند $y$ و رسم نمودارهای جعبه‌ای، بافت‌نگار و چگالی یک متغیر دیگر مانند $x$ در طبقات حاصل از تقسیم‌بندی است.

این نمودارها در بعضی از موارد بسیار مفید هستند.

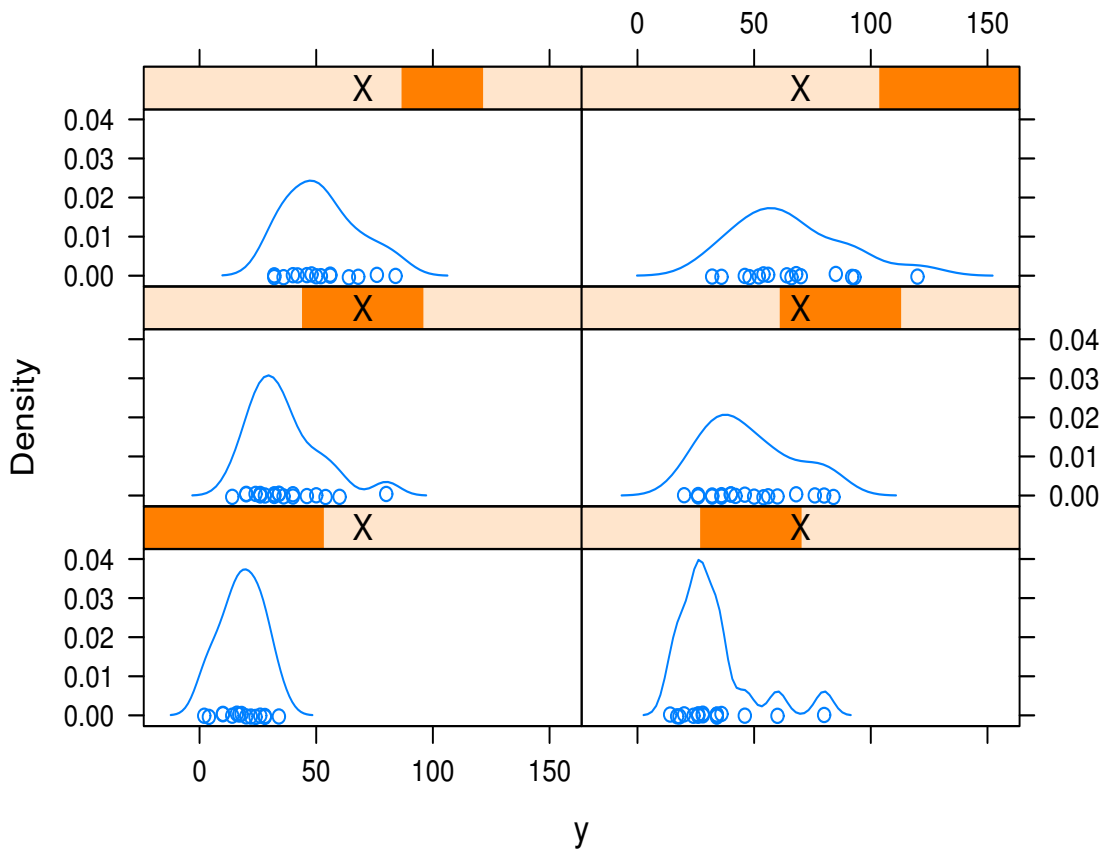
```
library(lattice)
y <- cars$dist
x <- cars$speed
X <- equal.count(x)
histogram(~ y | X)
```


X
X
X
X
X
X
Percent of Total
y
0
50
100
10
20
30
40
60

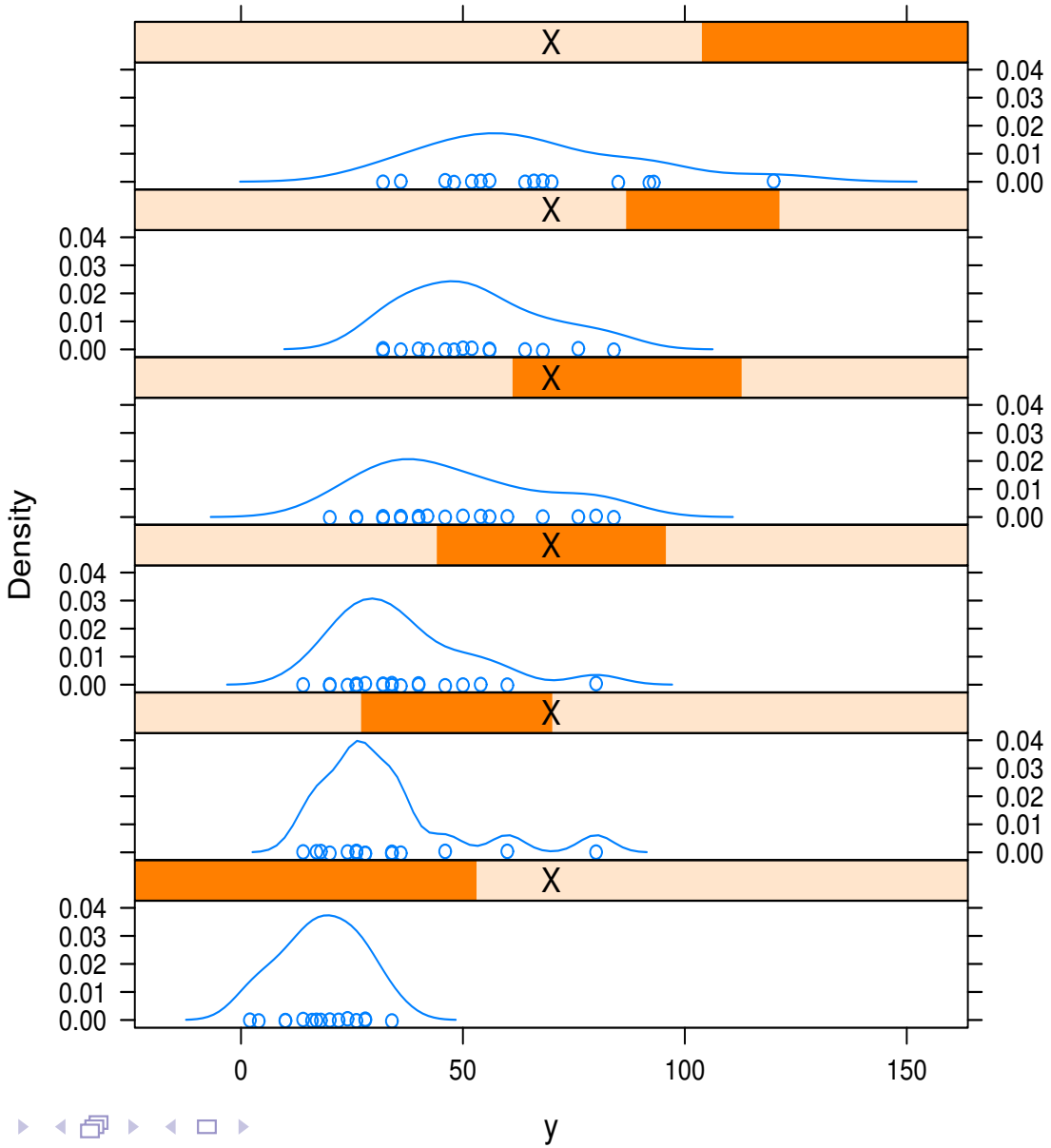
```
bwplot(~ y | vitesse, layout=c(1,6))
```



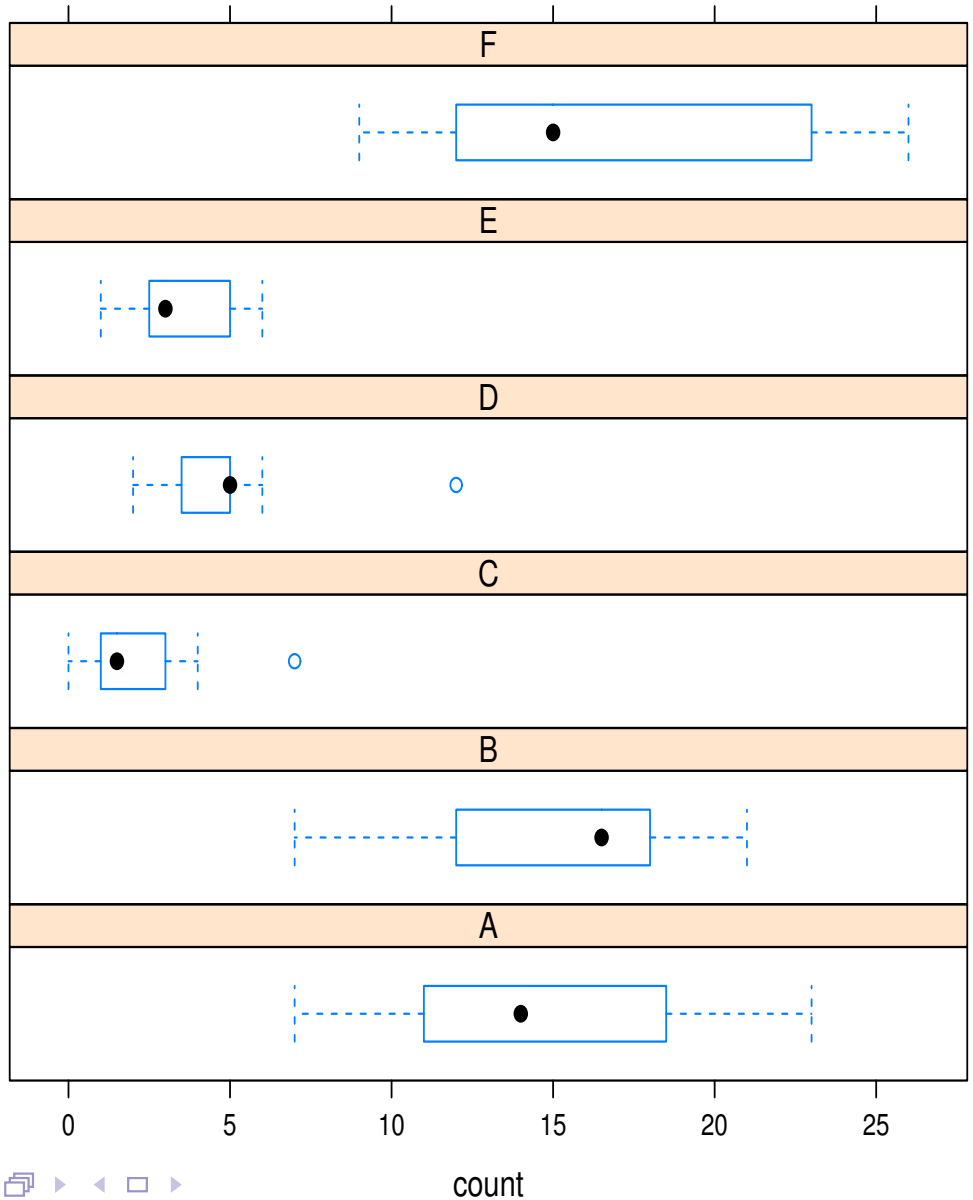
```
densityplot(~ y | X, aspect='xy')
```



```
densityplot(~ y | X, layout=c(1,6))
```



```
bwplot( ~ count | spray, data = InsectSprays, layout=c(1,6))
```



# رسم رویه‌ها

چندین بسته در $R$ وجود دارند که رسم رویه‌ها و همچنین نمودارهای تراز را فراهم می‌آورند. با کمک تابع $persp$ می‌توان نمودارهای رویه را روی یک صفحه رسم کرد.

# رسم رویه‌ها

چندین بسته در $R$ وجود دارند که رسم رویه‌ها و همچنین نمودارهای تراز را فراهم می‌آورند. با کمک تابع $persp$ می‌توان نمودارهای رویه را روی یک صفحه رسم کرد.

برای مشاهده قابلیت‌های جالب و متنوع این تابع، دستور $demo(persp)$ را در $R$ اجرا کنید.

# رسم رویه‌ها

چندین بسته در $R$ وجود دارند که رسم رویه‌ها و همچنین نمودارهای تراز را فراهم می‌آورند. با کمک تابع $persp$ می‌توان نمودارهای رویه را روی یک صفحه رسم کرد.

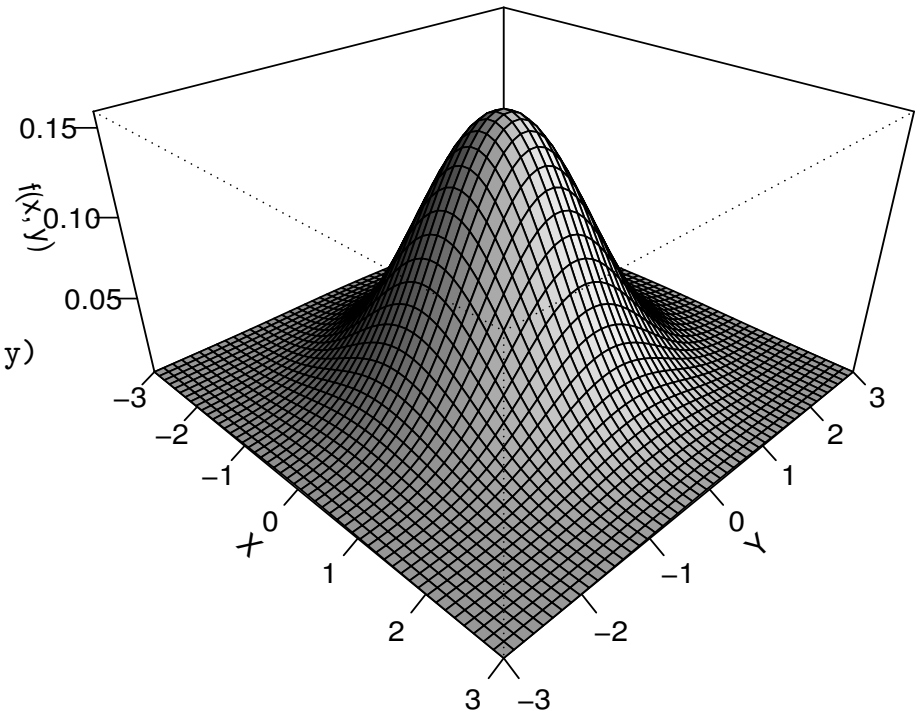
برای مشاهده قابلیت‌های جالب و متنوع این تابع، دستور $demo(persp)$ را در $R$ اجرا کنید.

به عنوان مثال، فرض کنید بخواهیم رویه تابع چگالی نرمال استاندارد دومتغیره را رسم کنیم:

$$f(x,y) = \frac{۱}{۲\pi} e^{-\frac{۱}{۲}(x^۲+y^۲)},\ (x,y) \in \Re^۲.$$

دقت کنید در کد زیر $z_{ij} = f(x_i, y_j)$ با دستور $outer$ محاسبه می‌شود.

```
f <- function(x,y) {
        z <- (1/(2*pi)) * exp(-.5 * (x^2 + y^2))
}
y <- x <- seq(-3, 3, length= 50)
z <- outer(x, y, f)   # compute density for all (x,y)
persp(x, y, z)        # the default plot
persp(x, y, z, theta = 45, phi = 30, expand = 0.6,
ltheta = 120, shade = 0.75, ticktype = "detailed",
xlab = "X", ylab = "Y", zlab = "f(x, y)")
```



## رویه‌ها در *lattice*

رویه‌ها را با تابع *wireframe* در بسته *lattice* هم می‌توانید رسم کنید.

# رویه‌ها در *lattice*

رویه‌ها را با تابع *wireframe* در بسته *lattice* هم می‌توانید رسم کنید.

برای این کار، باید فرمولی به صورت $z \sim x * y$ به همراه یک ساختار داده یا ماتریس شامل نقاط $(x, y, z)$ آماده کنید.

# رویه‌ها در *lattice*

رویه‌ها را با تابع *wireframe* در بسته *lattice* هم می‌توانید رسم کنید.

برای این کار، باید فرمولی به صورت $z \sim x * y$ به همراه یک ساختار داده یا ماتریس شامل نقاط $(x, y, z)$ آماده کنید.

با استفاده از بسته *rgl* می‌توانید یک نمایش سه بعدی تعاملی ایجاد کنید، به طوری که توانایی چرخش نمودار رویه به هر جهت فراهم می‌شود. برای دیدن این قابلیت، از دو خط زیر استفاده کنید

```
library(rgl)
demo(bivar)
```

# رویه‌ها در *lattice*

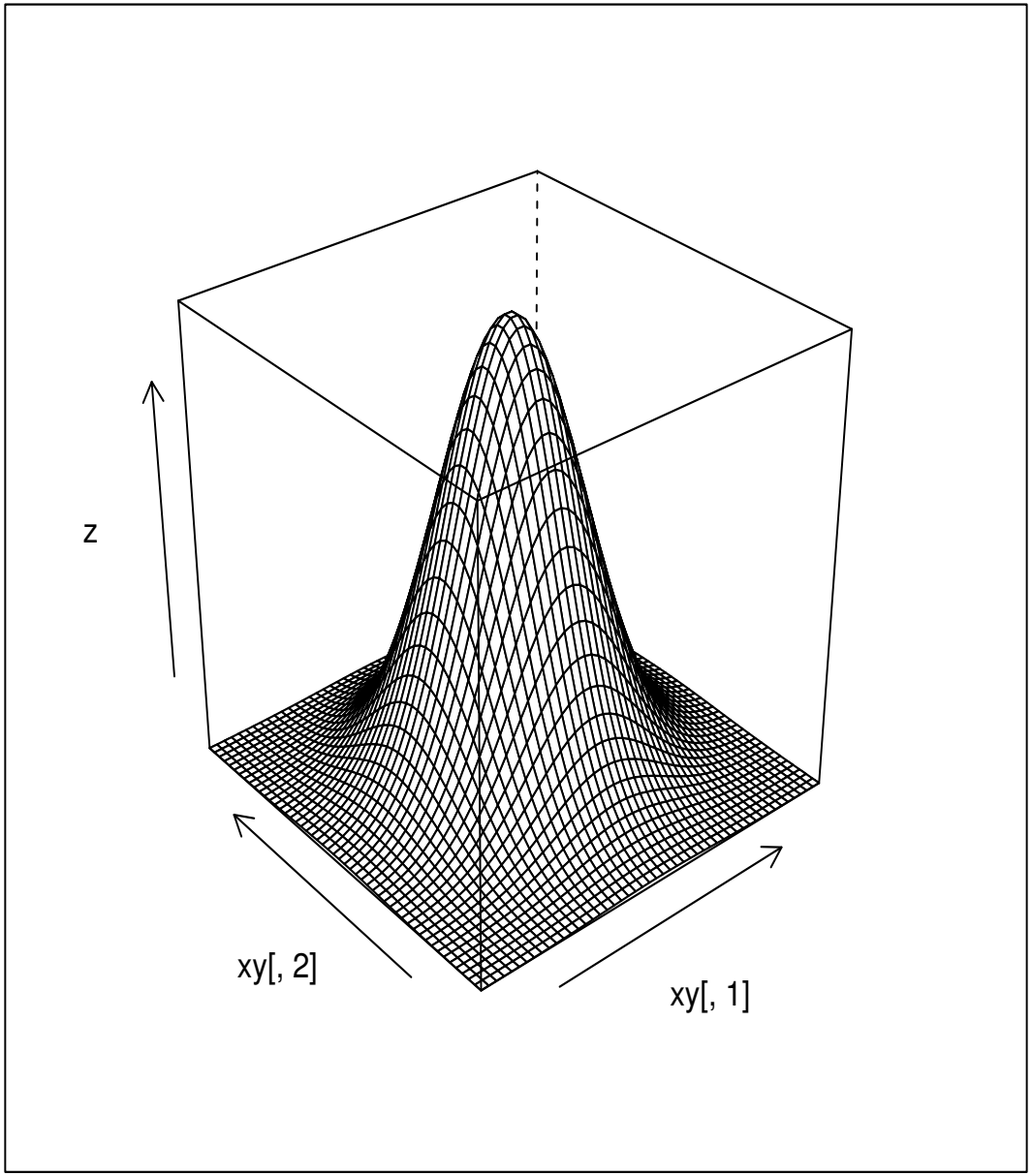
رویه‌ها را با تابع *wireframe* در بسته *lattice* هم می‌توانید رسم کنید.

برای این کار، باید فرمولی به صورت $z \sim x * y$ به همراه یک ساختار داده یا ماتریس شامل نقاط $(x, y, z)$ آماده کنید.

با استفاده از بسته *rgl* می‌توانید یک نمایش سه بعدی تعاملی ایجاد کنید، به طوری که توانایی چرخش نمودار رویه به هر جهت فراهم می‌شود. برای دیدن این قابلیت، از دو خط زیر استفاده کنید

```
library(rgl)
demo(bivar)
```

```
library(lattice)
x <- y <- seq(-3, 3, length= 50)
xy <- expand.grid(x, y)
z <- (1/(2*pi)) * exp(-.5 * (xy[,1]^2 + xy[,2]^2))
wireframe(z ~ xy[,1] * xy[,2])
```


z
xy[, 2]
xy[, 1]

# نمودارهای تراز

نمودارهای تراز ($contourplot$) خطوط تراز یک شکل سه‌بعدی را رسم می‌کند. به عبارت دیگر این نمودارها، همان رویه سه بعدی $(x, y, f(x, y))$ را در یک صفحه، با منحنی‌های تراز $f(x, y) = c$ برای ثابت‌های انتخاب شده $c$، تصویر می‌کند.

# نمودارهای تراز

نمودارهای تراز ($contourplot$) خطوط تراز یک شکل سه‌بعدی را رسم می‌کند. به عبارت دیگر این نمودارها، همان رویه سه بعدی $(x, y, f(x, y))$ را در یک صفحه، با منحنی‌های تراز $f(x, y) = c$ برای ثابت‌های انتخاب شده $c$، تصویر می‌کند.

دو تابع $contour$ در بسته $graphics$ که در خود هسته $R$ وجود دارند و تابع $contourplot$ در بسته $lattice$ این نمودارها را تولید می‌کنند.

# نمودارهای تراز

نمودارهای تراز ($contourplot$) خطوط تراز یک شکل سه‌بعدی را رسم می‌کند. به عبارت دیگر این نمودارها، همان رویه سه بعدی $(x, y, f(x, y))$ را در یک صفحه، با منحنی‌های تراز $f(x, y) = c$ برای ثابت‌های انتخاب شده $c$، تصویر می‌کند.

دو تابع $contour$ در بسته $graphics$ که در خود هسته $R$ وجود دارند و تابع $contourplot$ در بسته $lattice$ این نمودارها را تولید می‌کنند.

تابع $image$ تابعی هم‌جنس $contour$ است که از رنگ‌ها برای تشخیص ترازها استفاده می‌کند.

# نمودارهای تراز

نمودارهای تراز ($contourplot$) خطوط تراز یک شکل سه‌بعدی را رسم می‌کند. به عبارت دیگر این نمودارها، همان رویه سه بعدی $(x, y, f(x, y))$ را در یک صفحه، با منحنی‌های تراز $f(x, y) = c$ برای ثابت‌های انتخاب شده $c$، تصویر می‌کند.

دو تابع $contour$ در بسته $graphics$ که در خود هسته $R$ وجود دارند و تابع $contourplot$ در بسته $lattice$ این نمودارها را تولید می‌کنند.

تابع $image$ تابعی هم‌جنس $contour$ است که از رنگ‌ها برای تشخیص ترازها استفاده می‌کند.

یک مثال مناسب برای نمایش این نمودارها در $R$ بر اساس داده‌های $volcano$ ارایه شده است. برای اطلاع از این داده‌ها می‌توانید از $help$ نرم‌افزار $R$ کمک بگیرید.

# نمودارهای تراز

نمودارهای تراز ($contourplot$) خطوط تراز یک شکل سه‌بعدی را رسم می‌کند. به عبارت دیگر این نمودارها، همان رویه سه بعدی $(x, y, f(x, y))$ را در یک صفحه، با منحنی‌های تراز $f(x, y) = c$ برای ثابت‌های انتخاب شده $c$، تصویر می‌کند.

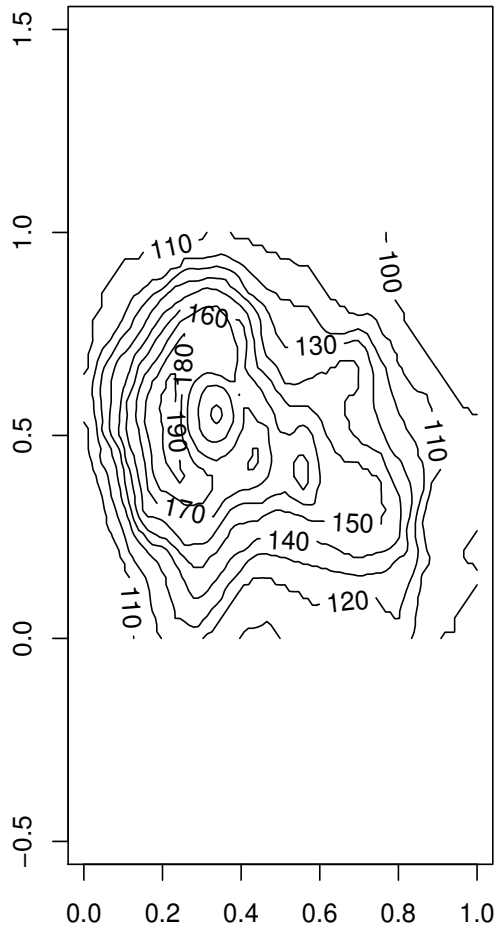
دو تابع $contour$ در بسته $graphics$ که در خود هسته $R$ وجود دارند و تابع $contourplot$ در بسته $lattice$ این نمودارها را تولید می‌کنند.

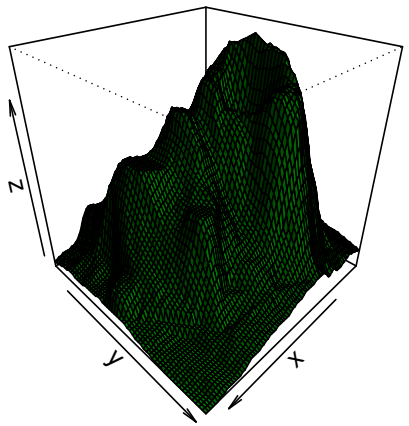
تابع $image$ تابعی هم‌جنس $contour$ است که از رنگ‌ها برای تشخیص ترازها استفاده می‌کند.

یک مثال مناسب برای نمایش این نمودارها در $R$ بر اساس داده‌های $volcano$ ارایه شده است. برای اطلاع از این داده‌ها می‌توانید از $help$ نرم‌افزار $R$ کمک بگیرید.

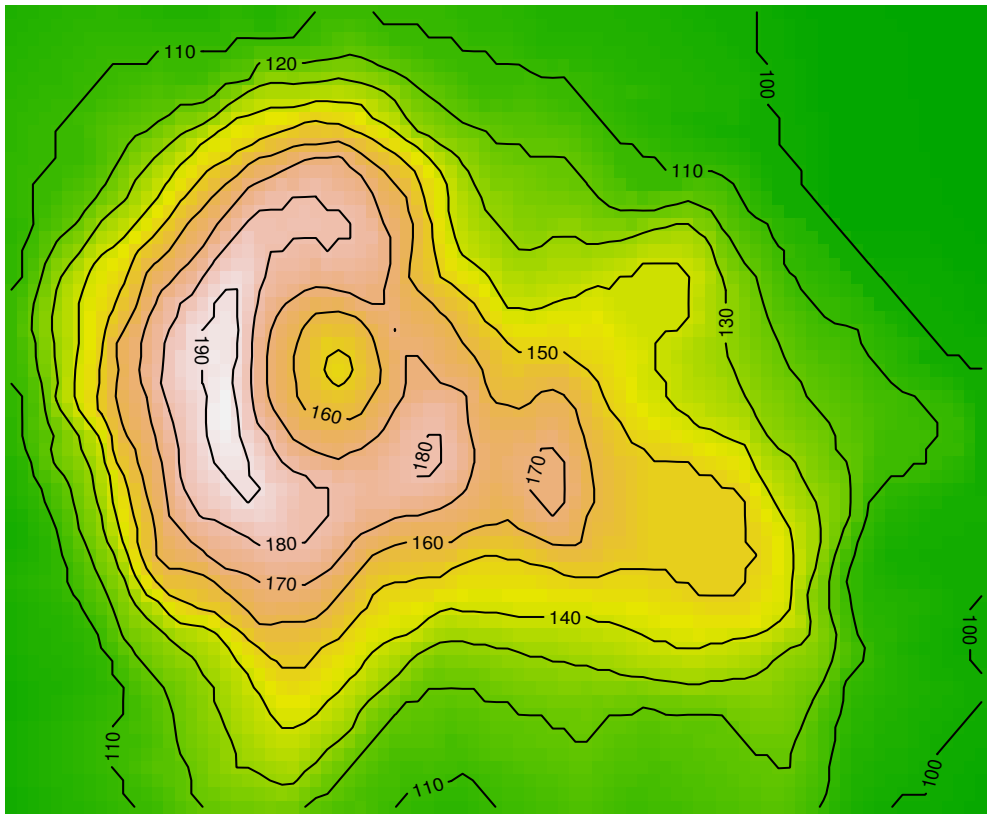
توابع $filled.contour$ در $graphics$ و $levelplot$ در $lattice$، نمودارهای تراز پرشده (رنگی) را تولید می‌کنند. طیف رنگی موجود در این نمودارها، مشابه $image$، ارتفاع نمودار $f(x, y)$ را نمایش می‌دهند.

```
# contour plot with labels
contour(volcano, asp = 1, labcex = 1)
# another version from lattice package
library(lattice)
contourplot(volcano) # similar to above
```

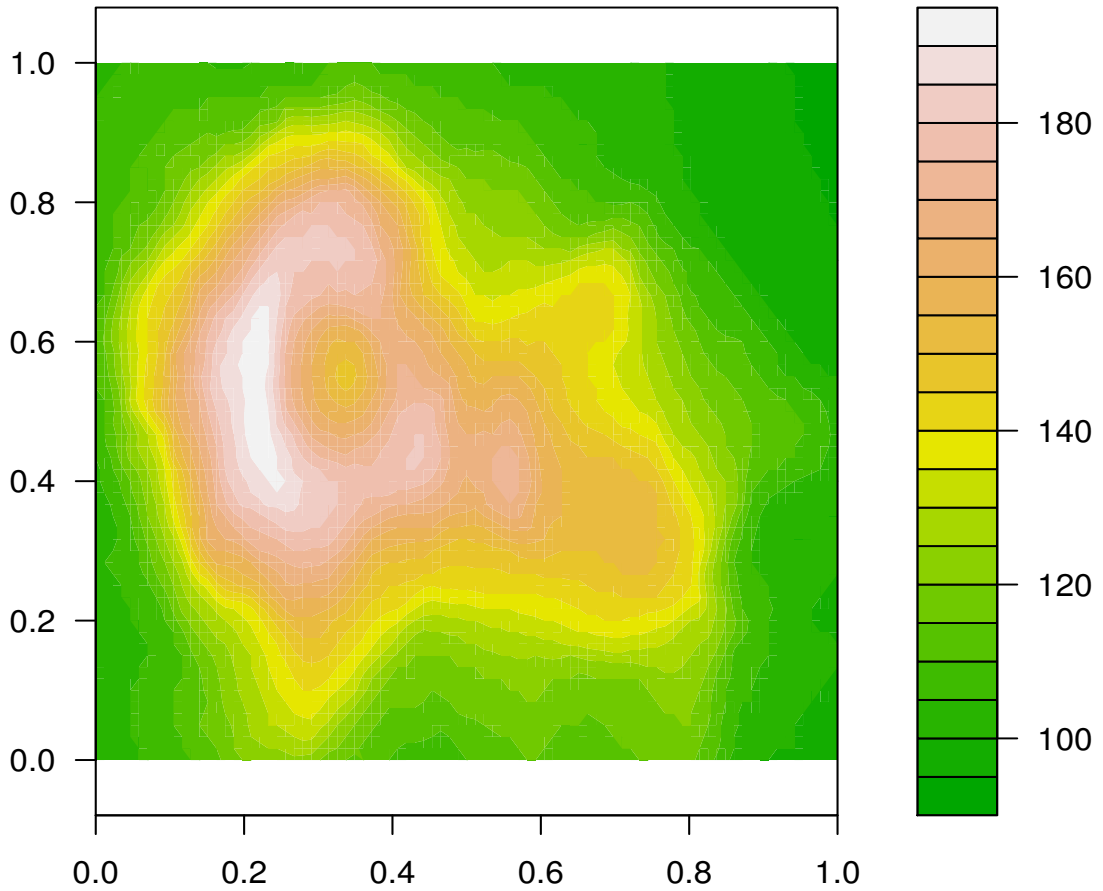


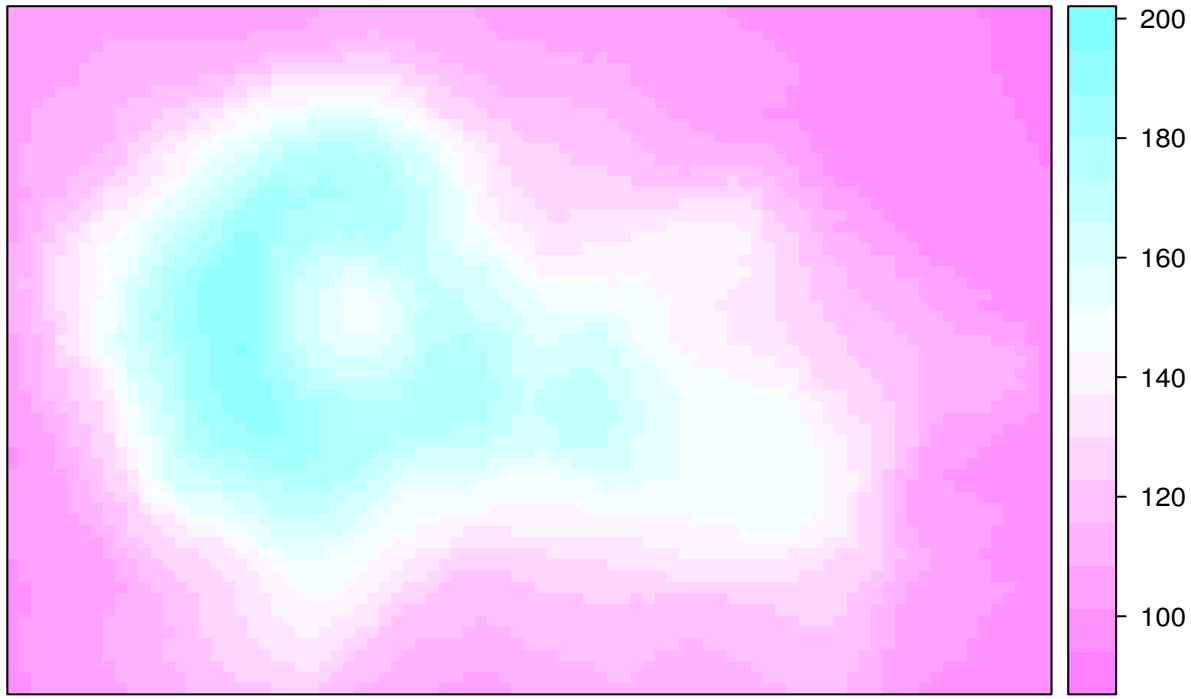
```
image(volcano, col = terrain.colors(100), axes = FALSE)
contour(volcano, levels = seq(100,200,by = 10), add = TRUE)
```



```
filled.contour(volcano, color = terrain.colors, asp = 1)
```



```
levelplot(volcano, scales = list(draw = FALSE),
                   xlab = "", ylab = "")
```



# تابع *image*

از این نوع توابع، می‌توان برای بررسی وجود گروه‌هایی در داده‌ها نیز استفاده کرد.

# تابع *image*

از این نوع توابع، می‌توان برای بررسی وجود گروه‌هایی در داده‌ها نیز استفاده کرد.

از آن‌ها در خوشه‌بندی داده‌ها و متغیرها نیز استفاده می‌شود.

# تابع *image*

از این نوع توابع، می‌توان برای بررسی وجود گروه‌هایی در داده‌ها نیز استفاده کرد.

از آن‌ها در خوشه‌بندی داده‌ها و متغیرها نیز استفاده می‌شود.

مشابه تابع *image*، تابع *image.plot* در بسته *fields* وجود دارد با این مزیت که طیف رنگی را نیز به همراه نمودار، بر اساس مقیاس داده‌ها، مدرج‌بندی می‌کند.

# تابع *image*

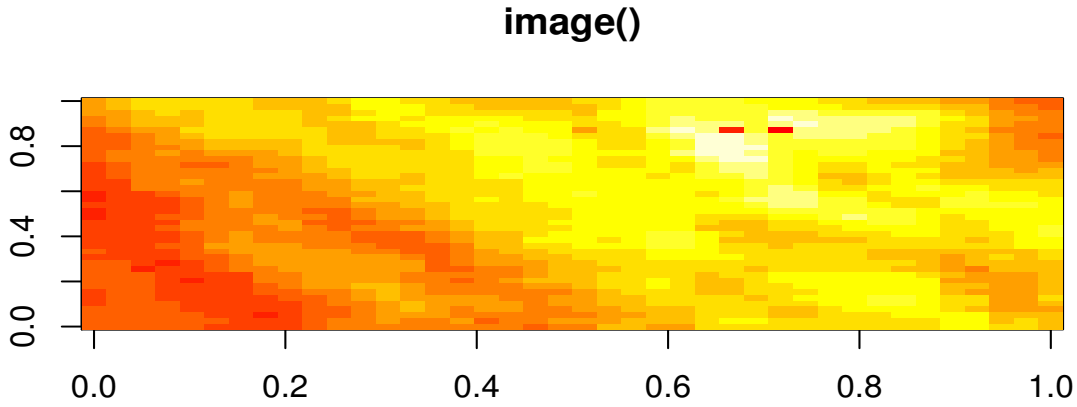
از این نوع توابع، می‌توان برای بررسی وجود گروه‌هایی در داده‌ها نیز استفاده کرد.

از آن‌ها در خوشه‌بندی داده‌ها و متغیرها نیز استفاده می‌شود.

مشابه تابع *image*، تابع *image.plot* در بسته *fields* وجود دارد با این مزیت که طیف رنگی را نیز به همراه نمودار، بر اساس مقیاس داده‌ها، مدرج‌بندی می‌کند.

از این تابع به ویژه در کاربردهای آمار فضایی، برای مشاهده همگنی متغیرهای اندازه‌گیری شده در نواحی جغرافیایی استفاده می‌شود.

```
library(fields)
data(lennon)
x <- lennon[201:240,201:240]
par(mfrow=c(2,1))
image(x, main="image()")
image.plot(x, main="image.plot()")
```


image()
0.0
0.4
0.8
0.0
0.2
0.4
0.6
0.8
1.0

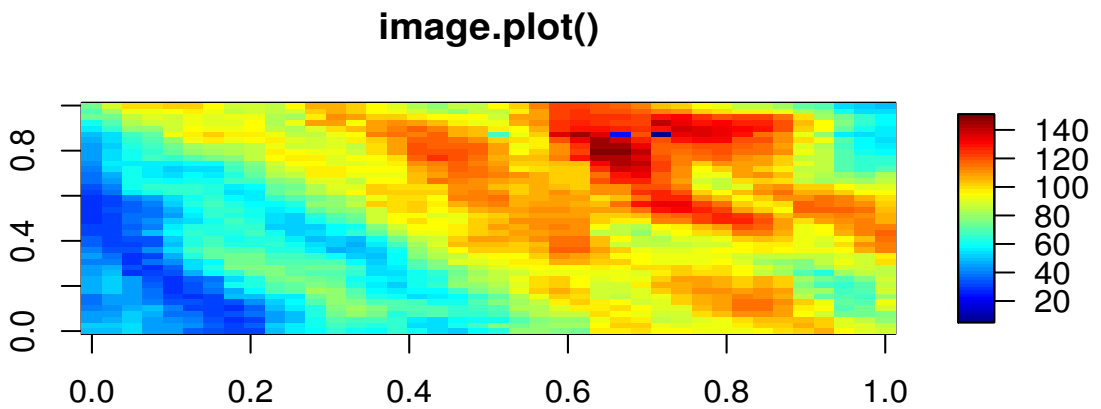


image.plot()
0.0
0.4
0.8
0.0
0.2
0.4
0.6
0.8
1.0
140
120
100
80
60
40
20

# نمودارهای موازی

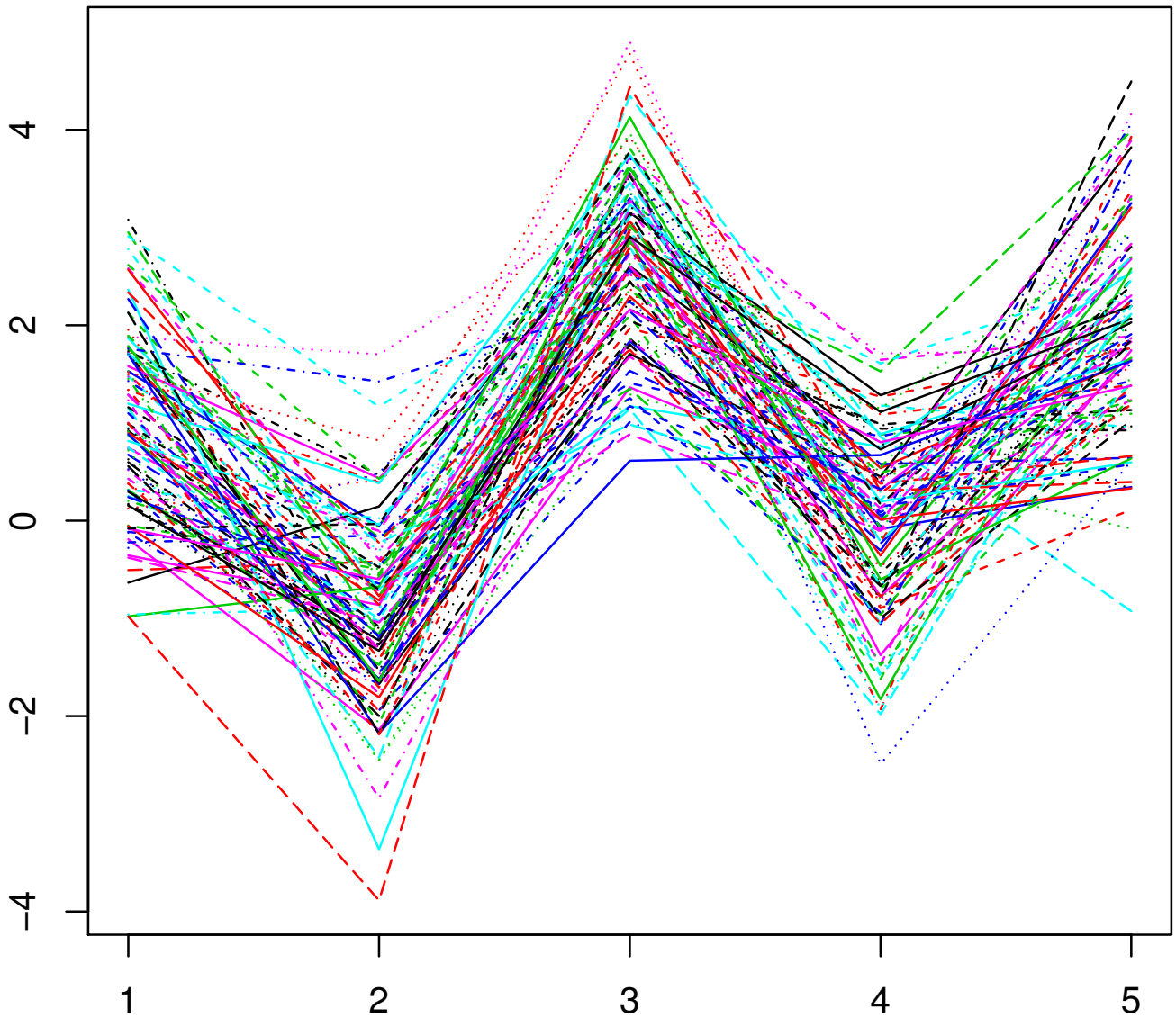
از این نوع نمودارها نیز می‌توان برای پی بردن به وجود گروه‌هایی در داده‌ها، بهره برد.

در سیستم مختصات موازی، به جای نمایش محورهای مختصات به صورت عمود بر هم، آن‌ها را به صورت محورهایی با خطوط موازی و فاصله یکسان نمایش می‌دهند.

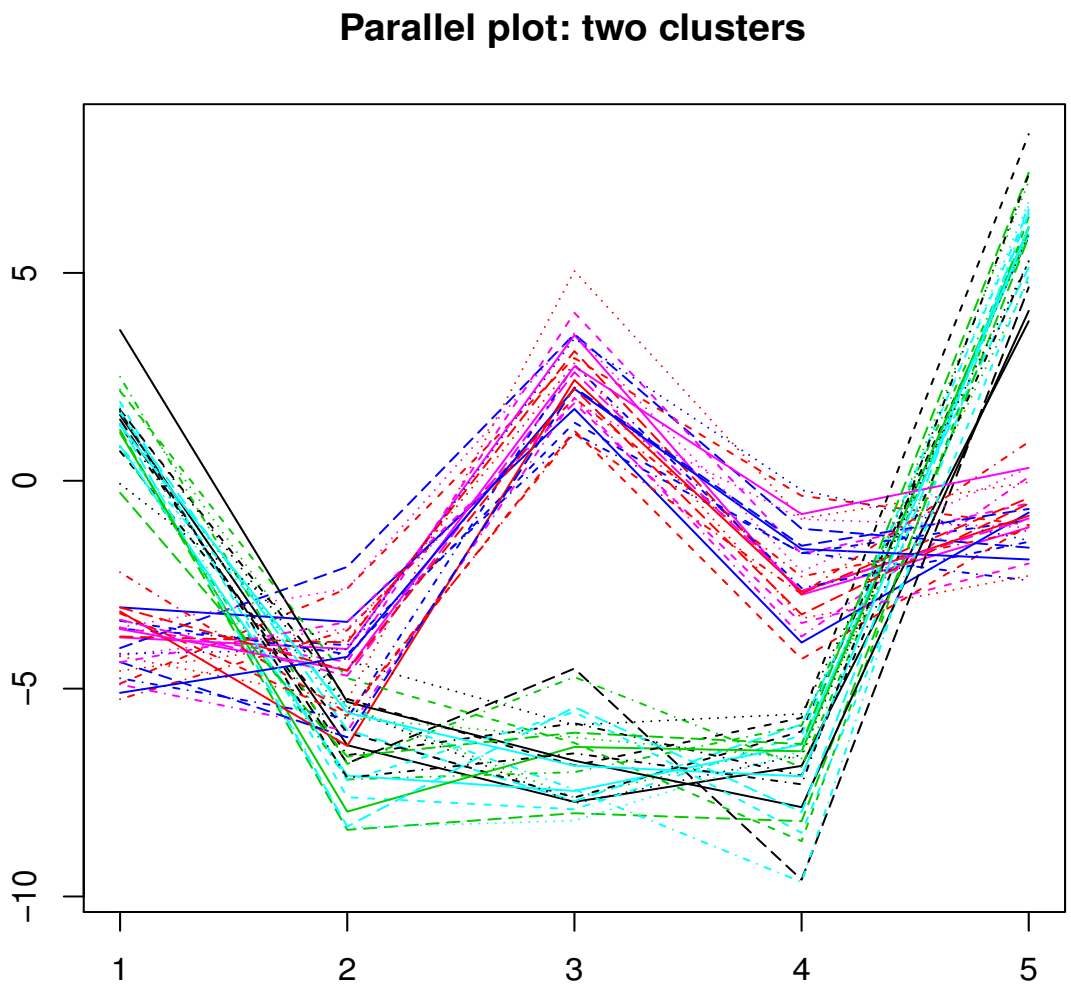
بنابراین در مختصات موازی به جای نمایش بردارها در فضای $\Re^d$، یک بردار را به صورت $d$ خط حقیقی (با طول‌های برابر) نشان می‌دهند و آن‌ها را به هم وصل می‌کنند.

```
n <- 100
m <- matrix( rnorm(5*n)+c(1,-1,3,0,2),
             nr = n, nc = 5, byrow = TRUE )
matplot(1:5, t(m), type = 'l',
        xlab = "", ylab = "")
title(main = "Parallel plot: Homogeneous cloud")
```

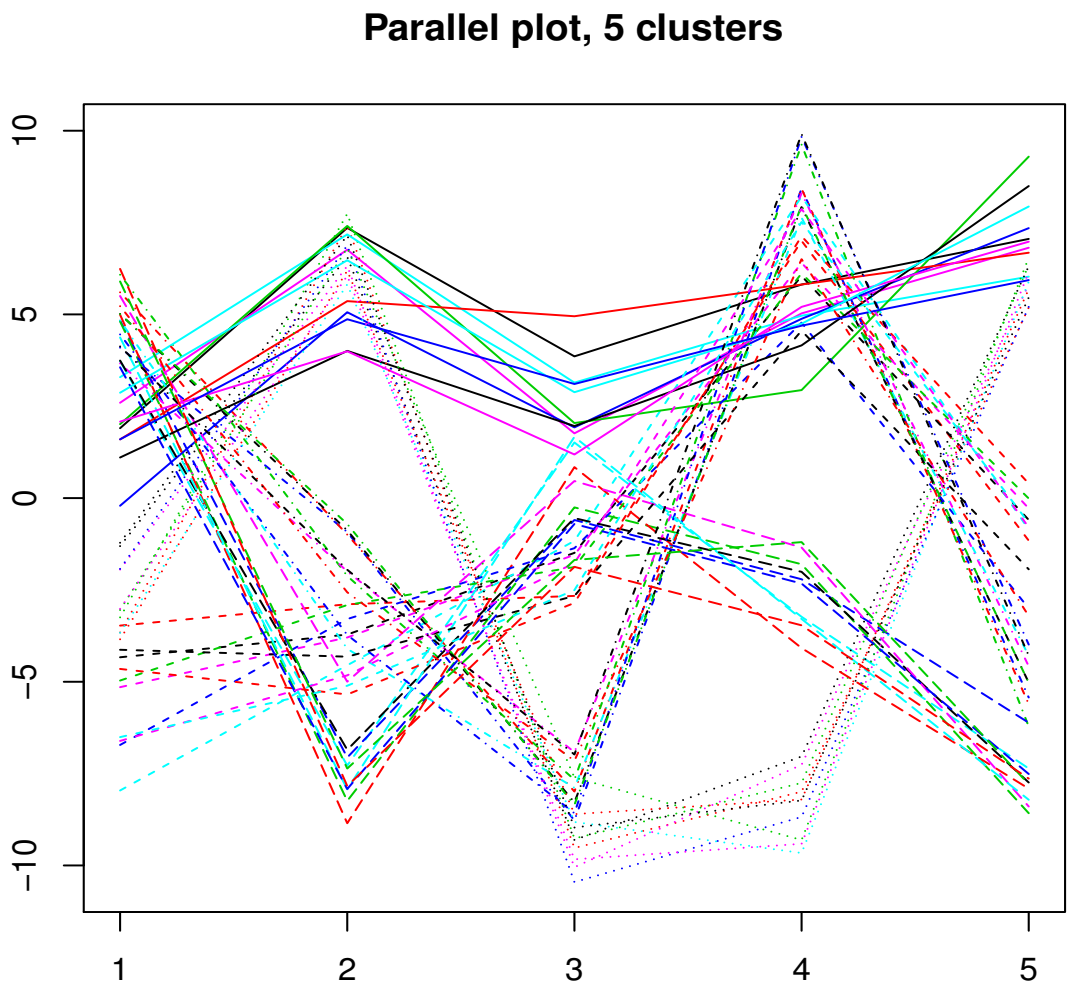
## Parallel plot: Homogeneous cloud



```
n <- 50
k <- 2
m <- matrix( rnorm(5*k*n) +
               runif(5*k, min = -10, max = 10),
             nr = n, nc = 5, byrow = TRUE )
matplot(1:5, t(m), type = 'l', xlab = "", ylab = "")
title(main = "Parallel plot: two clusters")
```

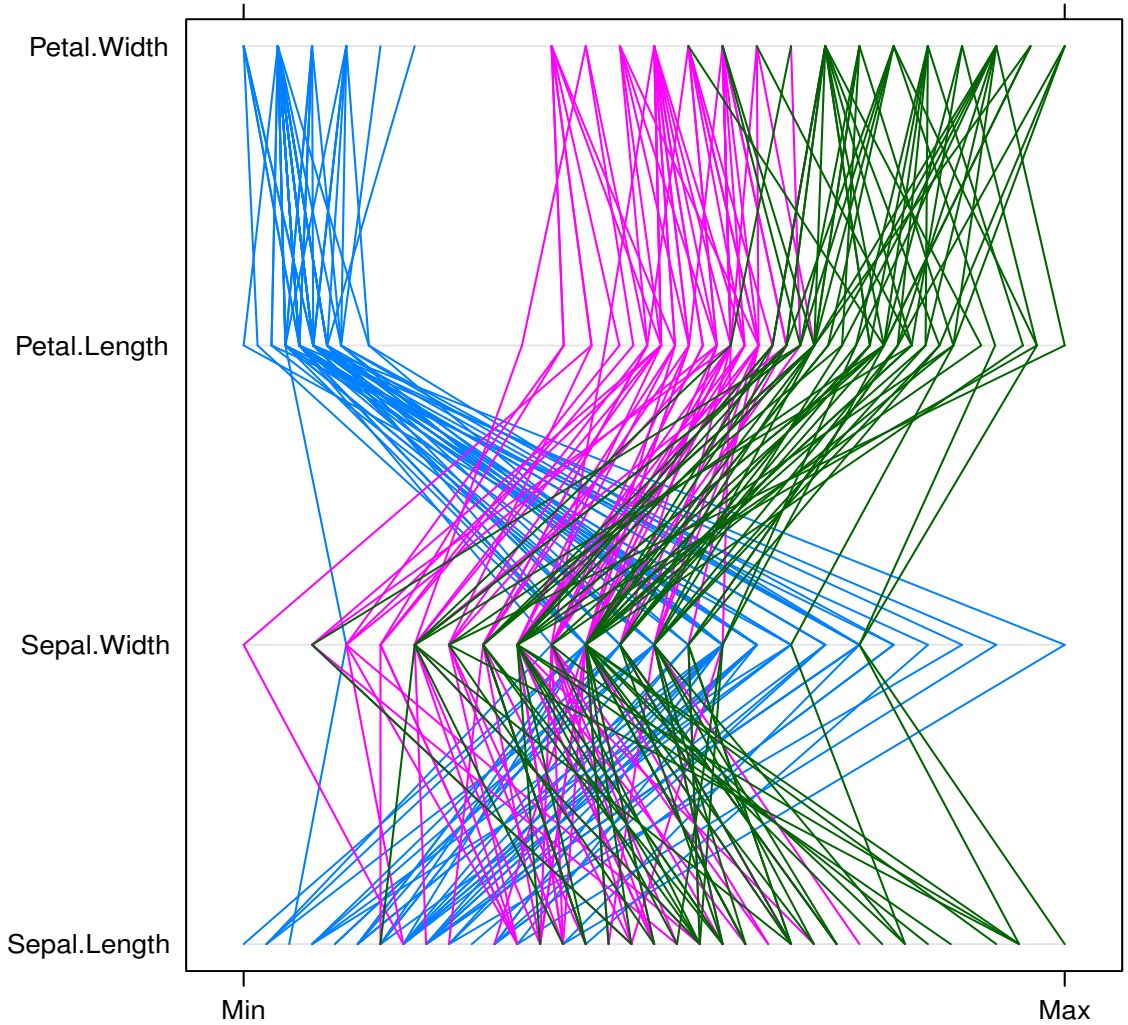


```
n <- 50
k <- 5
m <- matrix( rnorm(5*k*n) +
               runif(5*k, min = -10, max = 10),
             nr = n, nc = 5, byrow = TRUE )
matplot(1:5, t(m), type = 'l', xlab = "", ylab = "")
title(main = "Parallel plot, 5 clusters")
```



از توابع *parcoord* در *MASS* و *parallel* در *lattice* هم می‌توان برای تولید این نمودارها استفاده کرد.

```
library(lattice)
parallel(~iris[1:4], groups = Species, iris)
```



# منحنی‌های اندروز

یک رهیافت دیگر برای نمایش داده‌های چندبعدی، مانند $X_1, X_2, \ldots, X_n \in \Re^d$، در یک شکل دوبعدی، نگاشتن هر بردار به یک تابع حقیقی مقدار است.

منحنی‌های اندروز $(Andrews)$ هر مشاهده $x_i = (x_{i1}, \ldots, x_{id})$ را به تابع زیر تصویر می‌کند:

$$f_i(t) = \frac{x_{i1}}{\sqrt{2}} + x_{i2} \sin t + x_{i3} \cos t + x_{i4} \sin 2t + x_{i5} \cos 2t + \cdots$$

یا به عبارت خلاصه‌تر:

$$f_i(t) = \frac{x_{i1}}{\sqrt{2}} + \sum_{1 \le k \le d/2} x_{i,2k} \sin 2k + \sum_{1 \le k \le d/2} x_{i,2k+1} \cos 2k, \quad -\pi \le t \le \pi.$$

# منحنی‌های اندروز

یک رهیافت دیگر برای نمایش داده‌های چندبعدی، مانند $X_1, X_2, \ldots, X_n \in \Re^d$، در یک شکل دوبعدی، نگاشتن هر بردار به یک تابع حقیقی مقدار است.

منحنی‌های اندروز ($Andrews$) هر مشاهده $x_i = (x_{i1}, \ldots, x_{id})$ را به تابع زیر تصویر می‌کند:

$$f_i(t) = \frac{x_{i1}}{\sqrt{2}} + x_{i2} \sin t + x_{i3} \cos t + x_{i4} \sin 2t + x_{i5} \cos 2t + \cdots$$

یا به عبارت خلاصه‌تر:

$$f_i(t) = \frac{x_{i1}}{\sqrt{2}} + \sum_{1 \le k \le d/2} x_{i,2k} \sin 2k + \sum_{1 \le k \le d/2} x_{i,2k+1} \cos 2k, \quad -\pi \le t \le \pi.$$

در واقع هر مشاهده توسط تصویرش بر مجموعه توابع پایه متعامد $\{2^{-1/2}, \{\sin kt\}_{k=1}^{\infty}, \{\cos kt\}_{k=1}^{\infty}\}$ نمایش داده می‌شود.

```
x <- seq(-pi, pi, length=100)
f <-  function (u) u[1]/sqrt(2) + u[2]*cos(x) +
u[3] * sin(x) + u[4] * cos(2*x)
y <- apply(as.matrix(iris[,1:4]), 1, f)
matplot(x, y,
        type = "l",
        lty = 1,
        col = as.numeric(iris[,5]),
        xlab = "", ylab = "",
        main = "Fourier (Andrew) curves")
```