有名，在研究中有广泛的应用。接着，将回顾用于相关性可视化的相关图，以及用于类别型变量中多元关系可视化的马赛克图。这些方法也非常实用，不过了解这些方法的研究人员和数据分析师并不多。通过这些绘图方法的示例，你将能更好地理解数据，并将你的发现展示给其他人。

11.1 散点图

在之前各章中，我们了解到散点图可用来描述两个连续型变量间的关系。本节，我们首先描述一个二元变量关系（*x*对*y*），然后探究各种通过添加额外信息来增强图形表达功能的方法。接着，我们将学习如何把多个散点图组合起来形成一个散点图矩阵，以便可以同时浏览多个二元变量关系。我们还将回顾一些数据点重叠的特殊案例，由于重叠将会削弱图形描述数据的能力，所以我们将围绕该难点讨论多种解决途径。最后，通过添加第三个连续型变量，我们将把二维图形扩展到三维，包括三维散点图和气泡图。它们都可帮助你更好地迅速理解三变量间的多元关系。

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。代码清单11-1展示了一个例子。

代码清单11-1 添加了最佳拟合曲线的散点图

```
attach(mtcars)
plot(wt, mpg,
     main="Basic Scatter plot 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)

lines(lowess(wt,mpg), col="blue", lwd=2, lty=2)
```

图形结果参见图11-1。

代码清单11-1中的代码加载了`mtcars`数据框，创建了一幅基本的散点图，图形的符号[①]是实心圆圈。与预期结果相同，随着车重的增加，每加仑英里数减少，虽然它们不是完美的线性关系。`abline()`函数用来添加最佳拟合的线性直线，而`lowess()`函数则用来添加一条平滑曲线。该平滑曲线拟合是一种基于局部加权多项式回归的非参数方法。算法细节可参见Cleveland(1981)。

注意 R有两个平滑曲线拟合函数：`lowess()`和`loess()`。`loess()`是基于`lowess()`表达式版本的更新和更强大的拟合函数。这两个函数的默认值不同，因此要小心使用，不要把它们弄混淆了。

① 即指`plot()`函数中的`pch`参数。——译者注

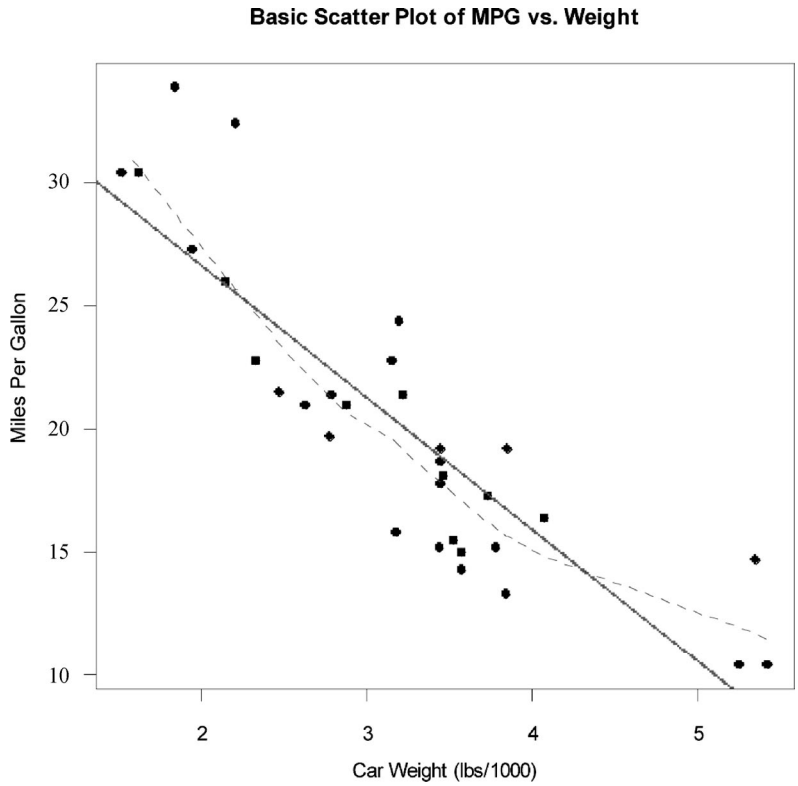

图11-1　汽车英里数对车重的散点图，添加了线性拟合直线和lowess拟合曲线

car包中的scatterplot()函数增强了散点图的许多功能，它可以很方便地绘制散点图，并能添加拟合曲线、边界箱线图和置信椭圆，还可以按子集绘图和交互式地识别点。例如，以下代码可生成一个比之前图形更复杂的版本：

```
library(car)
scatterplot(mpg ~ wt | cyl, data=mtcars, lwd=2,
            main="Scatter Plot of MPG vs. Weight by # Cylinders",
            xlab="Weight of Car (lbs/1000)",
            ylab="Miles Per Gallon",
            legend.plot=TRUE,
            id.method="identify",
            labels=row.names(mtcars),
            boxplots="xy"
)
```

此处，scatterplot()函数用来绘制有四个、六个和八个气缸的汽车每加仑英里数对车重的图形。表达式mpg ~ wt | cyl表示按条件绘图（即按cyl的水平分别绘制mpg和wt的关系图）。结果见图11-2。

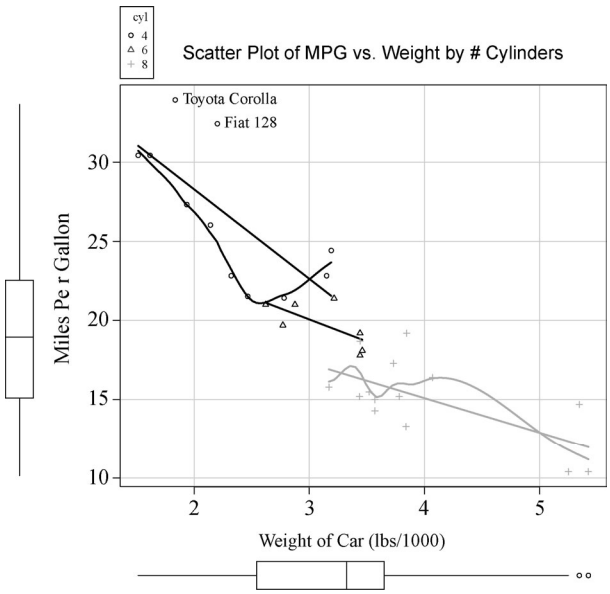

图11-2 各子集的散点图与其相应的拟合曲线

默认地，各子集会通过颜色和图形符号加以区分，并同时绘制线性拟合和平滑拟合曲线。平滑拟合默认需要五个单独的数据点，因此六缸车型的平滑曲线无法绘制。`id.method`选项的设定表明可通过鼠标单击来交互式地识别数据点，直到用户选择Stop（通过图形或者背景菜单）或者敲击Esc键。`labels`选项的设定表明可通过点的行名称来识别点。此图中可以看到，给定Toyata Corolla和Fiat 128的车重，通常每加仑燃油可行驶得更远。`legend.plot`选项表明在左上边界添加图例，而`mpg`和`weight`的边界箱线图可通过`boxplots`选项来绘制。总之，`scatterplot()`函数还有许多特性值得探究，比如本节未讨论的稳健性选项和数据集中度椭圆选项。更多细节可参见`help(scatterplot)`。

散点图可以一次对两个定量变量间的关系进行可视化。但是如果想观察下汽车里程、车重、排量（立方英寸）和后轴比间的二元关系，该怎么做呢？一种途径就是将六幅散点图绘制到一个矩阵中，这便是下节即将介绍的散点图矩阵。

11.1.1 散点图矩阵

R中至少有四种创建散点图矩阵的实用函数。相信数据分析师一定很喜爱散点图矩阵吧？`pairs()`函数可以创建基础的散点图矩阵。下面的代码生成了一个散点图矩阵，包含`mpg`、`disp`、`drat`和`wt`四个变量：

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

图中包含~右边的所有变量，参见图11-3。

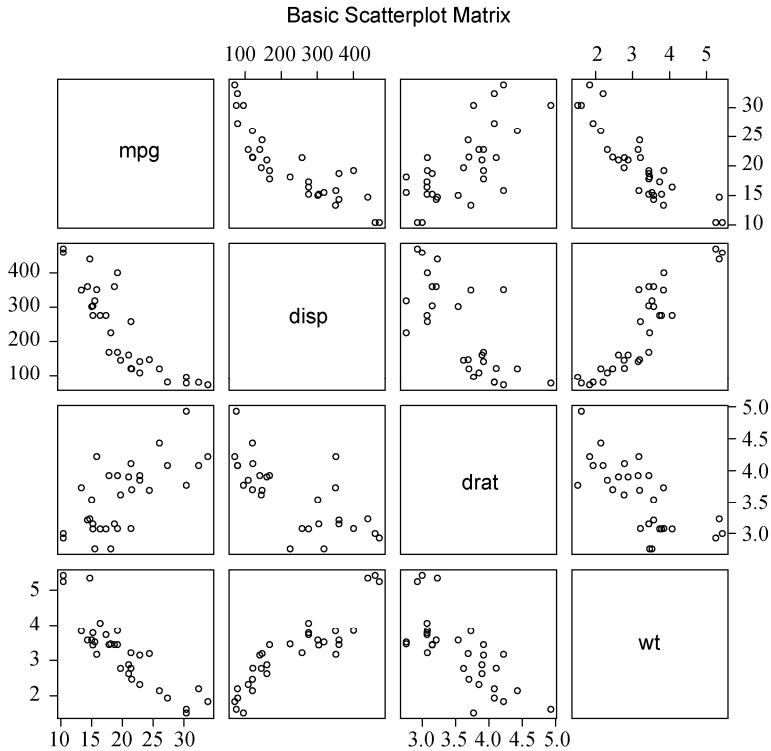

图11-3　pairs()函数创建的散点图矩阵

在图11-3中，你可以看到所有指定变量间的二元关系。例如，mpg和disp的散点图可在两变量的行列交叉处找到。值得注意的是，主对角线的上方和下方的六幅散点图是相同的，这也是为了方便摆放图形的缘故。通过调整参数，可以只展示下三角或者上三角的图形。例如，选项upper.panel = NULL将只生成下三角的图形。

car包中的scatterplotMatrix()函数也可以生成散点图矩阵，并有以下可选操作：

- ❑ 以某个因子为条件绘制散点图矩阵；
- ❑ 包含线性和平滑拟合曲线；
- ❑ 在主对角线放置箱线图、密度图或者直方图；
- ❑ 在各单元格的边界添加轴须图。

例如：

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

结果见图11-4。可以看到线性和平滑（loess）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。

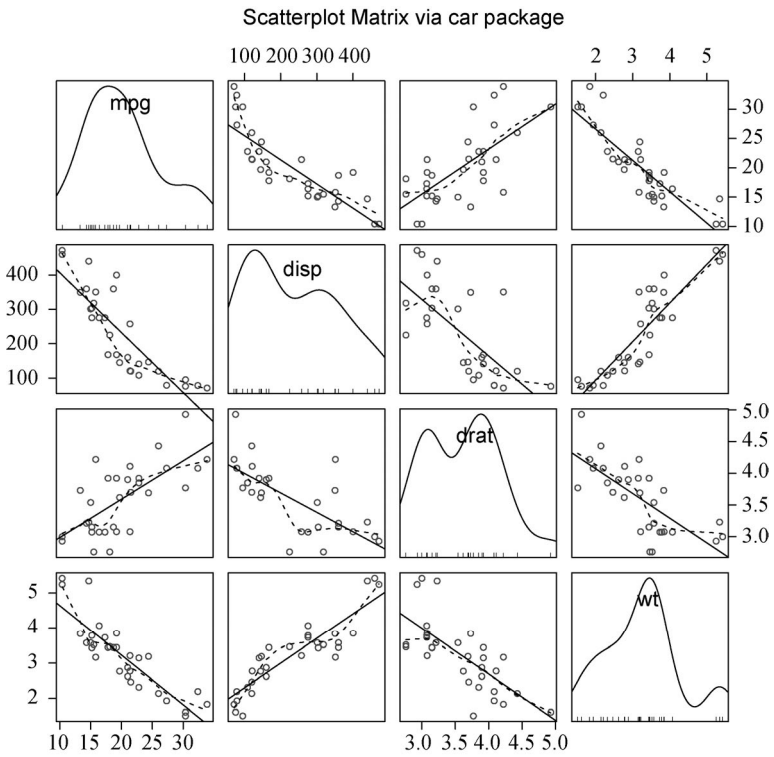

图11-4 scatterplotMatrix()函数创建的散点图矩阵。主对角线上有核密度曲线和轴须图，其余图形都含有线性和平滑拟合曲线

下面的代码展示了scatterplotMatrix()函数的另一个使用示例：

```
library(car)
scatterplotMatrix(~ mpg + disp + drat + wt | cyl, data=mtcars,
                  spread=FALSE, diagonal="histogram",
                  main="Scatter Plot Matrix via car Package")
```

这里，我们将主对角线的核密度曲线改成了直方图，并且直方图是以各车的气缸数为条件绘制的。结果见图11-5。

默认地，回归直线拟合整个样本，包含选项by.groups = TRUE将可依据各子集分别生成拟合曲线。

gclus包中的cpairs()函数提供了一个有趣的散点图矩阵变种。它含有可以重排矩阵中变量位置的选项，可以让相关性更高的变量更靠近主对角线。该函数还能对各单元格进行颜色编码来展示变量间的相关性大小。下面考虑mpg、wt、disp和drat间的相关性：

```
> cor(mtcars[c("mpg", "wt", "disp", "drat")])

           mpg      wt   disp   drat
mpg    1.000 -0.868 -0.848  0.681
```

```
wt   -0.868  1.000  0.888 -0.712
disp -0.848  0.888  1.000 -0.710
drat  0.681 -0.712 -0.710  1.000
```

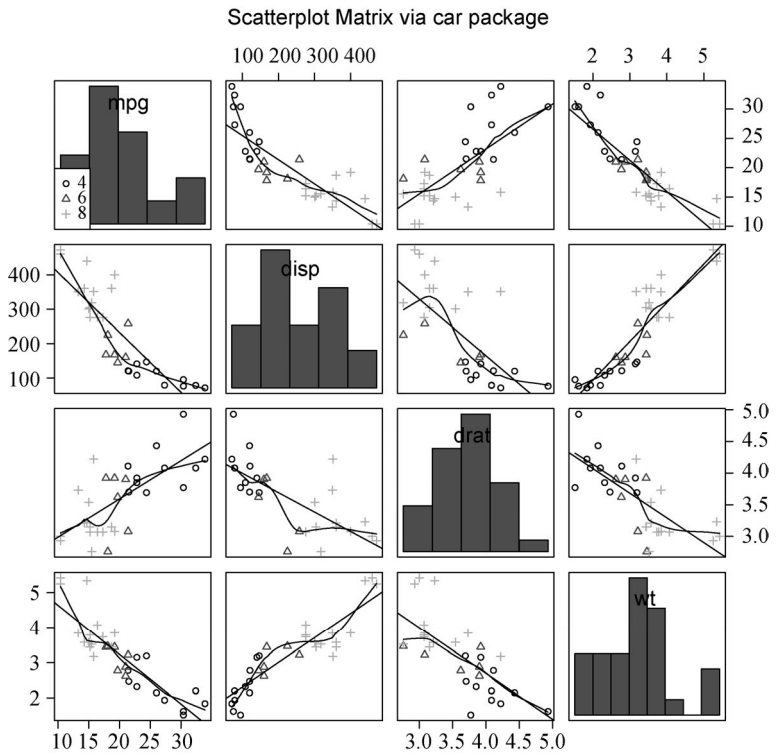

图11-5　scatterplotMatrix()函数生成的散点图矩阵。图形包含主对角线中的直方图以及其他部分的线性和平滑拟合曲线。另外，子群（根据气缸数）通过符号类型和颜色来区分标注

可以看到相关性最高（0.89）的是车重（wt）与排量（disp），以及车重（wt）与每加仑英里数（mpg；−0.87）。相关性最低（0.68）的是每加仑英里（mpg）与后轴比（drat）。参照代码清单11-2，你可以对散点图矩阵中的这些变量重新排序并添加颜色。

代码清单11-2　gclus包生成的散点图矩阵

```
library(gclus)
mydata <- mtcars[c(1, 3, 5, 6)]
mydata.corr <- abs(cor(mydata))

mycolors <- dmat.color(mydata.corr)

myorder <- order.single(mydata.corr)

cpairs(mydata,
```

```
        myorder,
        panel.colors=mycolors,
        gap=.5,
        main="Variables Ordered and Colored by Correlation"
)
```

代码清单11-2中使用的dmat.color()、order.single()和cpairs()函数都来自于gclus包。第一步，从mtcars数据框中选择所需的变量，并计算它们相关系数的绝对值。第二步，使用dmat.color()函数获取绘图的颜色。给定一个对称矩阵（本例中是相关系数矩阵），dmat.color()将返回一个颜色矩阵。第三步，可对图中变量进行排序。通过order.single()函数重排对象，可使得相似的对象更为靠近。本例中变量排序的基础是相关系数的相似性。最后，散点图矩阵将根据新的变量顺序（myorder）和颜色列表（mycolors）绘图、上色，gap选项使矩阵各单元格间的间距稍微增大一点。结果图形见图11-6。

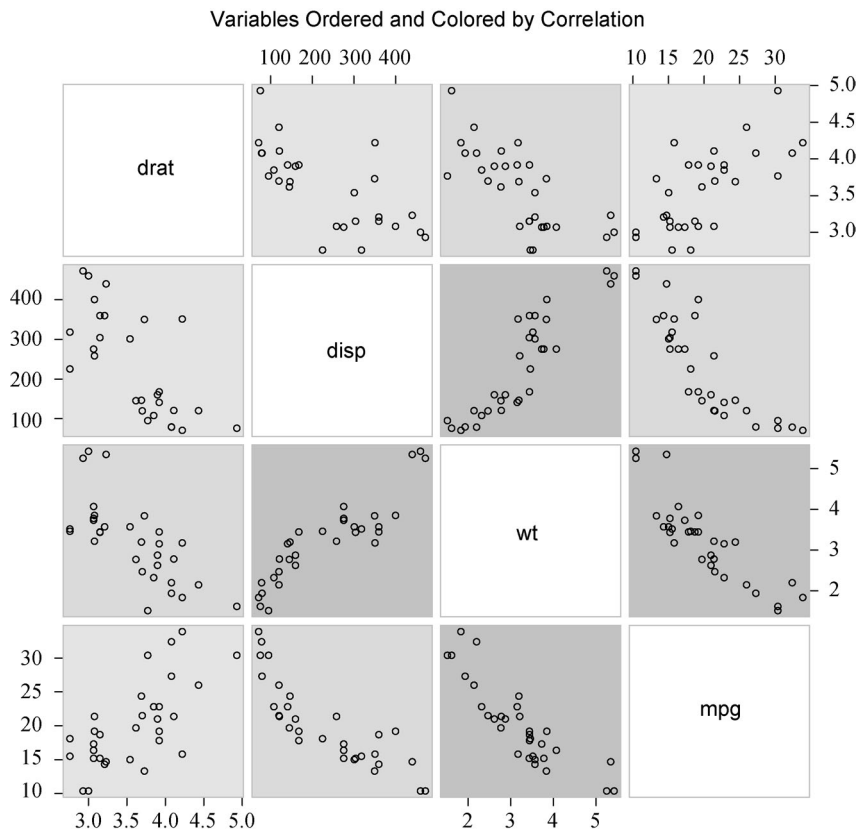

图11-6 gclus包中的cpairs()函数生成的散点图矩阵。变量离主对角线越近，相关性越高（另见彩插图11-6）

从图11-6中可以看到，相关性最高的变量对是车重与排量，以及每加仑英里数与车重（标了红色，并且离主对角线最近）。相关性最低的是后轴比与每加仑英里数（标了黄色，并且离主对