

第九章 ggplot2

ggplot2 简介

- ◆ ggplot2最早出现于2005年，相对于lattice而言，ggplot2的语法更加简洁；
- ◆ ggplot2包是基于Wilkinson在《Grammar of Graphics》一书中所提出的**图形语法**的具体实现；
- ◆ ggplot2的核心理念是将绘图与数据分离，数据相关的绘图与数据无关的绘图分离；
- ◆ ggplot的绘图有以下几个特点：第一，有明确的起始（以ggplot函数开始）与终止（一句语句一幅图）；其二，图层之间的叠加是靠“+”号实现的，越后面其图层越高。

图形语法涉及到的一些概念如下：

- **几何对象 (geom)**：表示图形中我们实际看到的图形元素，如各种点、线等元素。
- **标度 (scale)**：标度的作用是将数据映射到图形空间，比如用颜色、大小或形状来表示不同的数据。通过自定义标度，可以更加精确地控制图形的外观。
- **坐标系 (coord)**：描述了数据如何映射到图形所在的平面，最常用的是直角坐标轴，坐标轴可以进行变换以满足不同的需要，如对数坐标、极坐标和地图投影。
- **图层 (layer)**：如果你用过photoshop，那么对于图层一定不会陌生。一个图层好比是一张玻璃纸，包含有各种图形元素，你可以分别建立图层然后叠放在一起，组合成图形的最终效果。图层可以允许用户一步步的构建图形，方便单独对图层进行修改、增加统计量、甚至改动数据。
- **分面 (facet)**：很多时候需要将数据按某种方法分组，分别进行绘图，分面就是控制分组绘图的方法和排列形式。通过坐标系和分面，用户可以控制图形元素的位置。

9.1 快速作图qplot()

```
qplot(x, y = NULL, ..., data, facets = NULL, margins = FALSE,  
      geom = "auto", xlim = c(NA, NA), ylim = c(NA, NA), log = "",  
      main = NULL, xlab = deparse(substitute(x)),  
      ylab = deparse(substitute(y)), asp = NA, stat = NULL, position = NULL)
```

参数

x, y,

传递到每一图层中的图形属性，简而言之就是分别代表所画图层的X坐标和Y坐标。

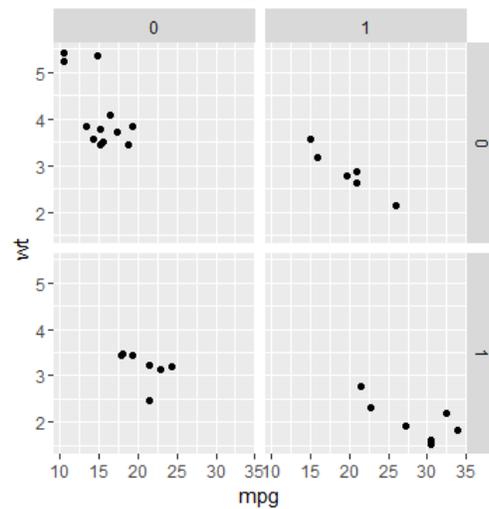
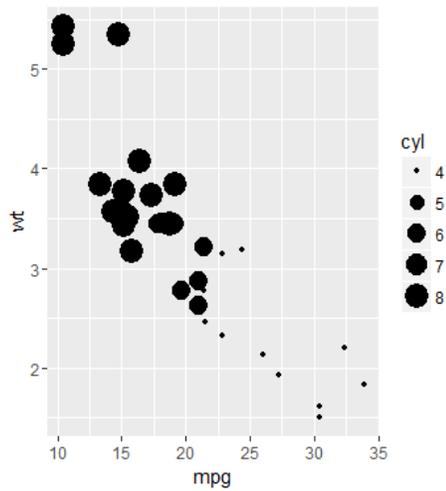
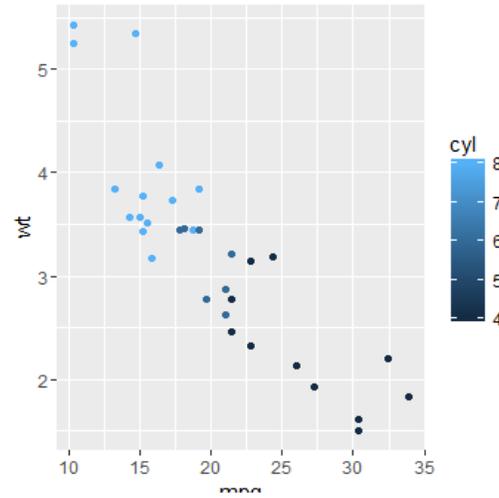
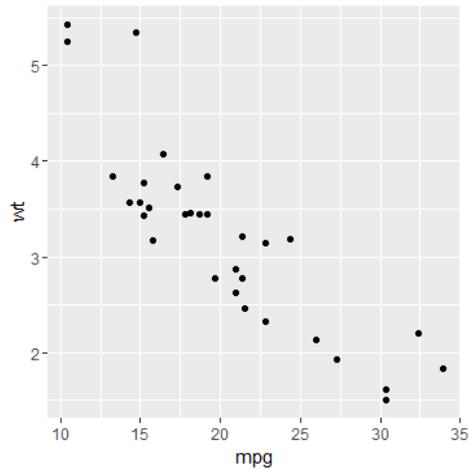
data

作图使用的数据框（可选），如果指定了数据框，**qplot()**会首先在数据框内查找变量名；如果没有指定数据框，R就会在当前环境中尝试提取向量创建一个数据框。

facets

图形/数据的分面。这是**ggplot2**作图比较特殊的一个概念，它把数据按某种规则进行分类，每一类数据做一个图形，所以最终效果就是一页多图。

```
qplot(mpg, wt, data = mtcars)
qplot(mpg, wt, data = mtcars, colour = cyl)
qplot(mpg, wt, data = mtcars, size = cyl)
qplot(mpg, wt, data = mtcars, facets = vs ~ am)
```



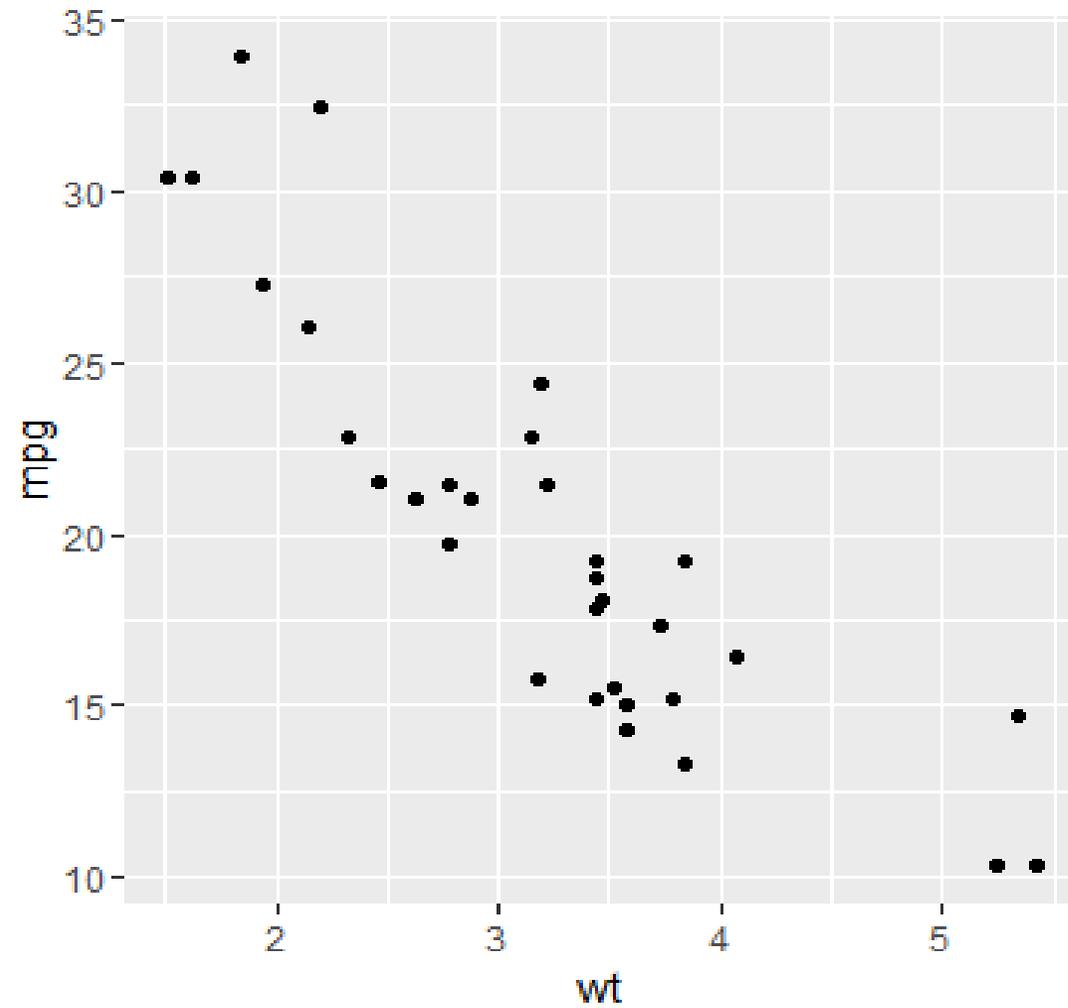
- ◆ 在上述4个例子中，还有**colour**、**size**、**shape**等其他图形属性参数，分别用点的颜色、点的大小以及点的形状表示不同的数值。
- ◆ 可以通过**geom**参数来指定绘制的几何对象类型，如**geom="smooth"**将拟合一条平滑曲线，**geom="boxplot"**将绘制箱线胡须图，**geom="histogram"**绘制直方图，**geom="freqpoly"**绘制频率多边形，**geom="density"**绘制密度曲线

9.3 散点图

- ◆ 散点图一般用来描述两个连续变量之间的关系，图中的每一个点表示一个观测值，根据散落在图中点的位置，可以判断数据的大概趋势。另外，人们也会在散点图中添加基于统计模型的趋势线来更直观地显示数据趋势。

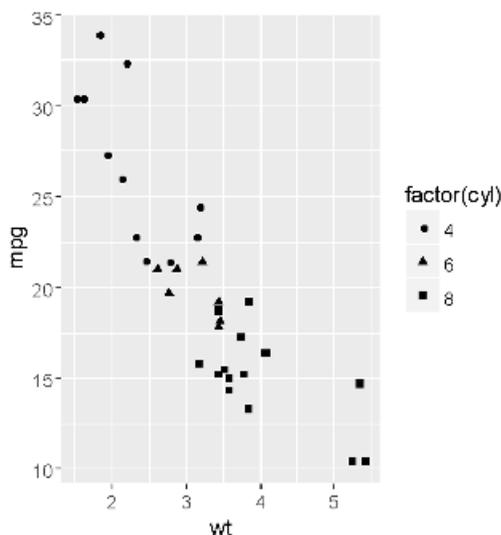
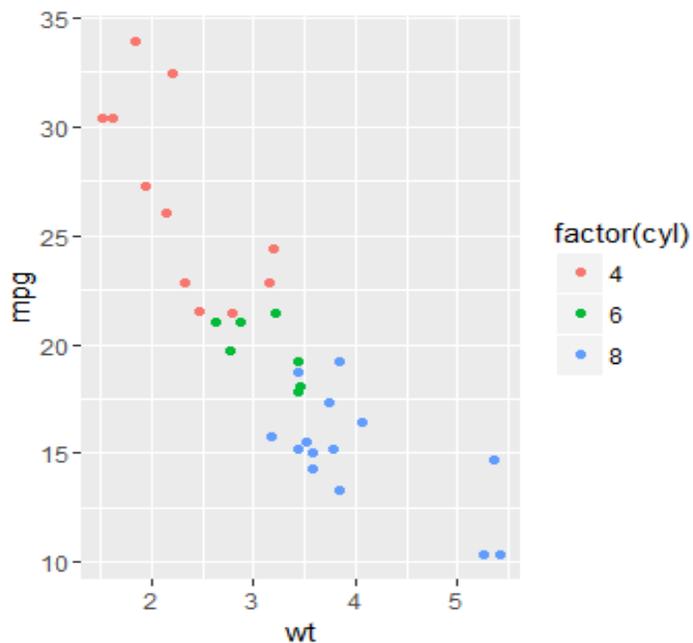
```
geom_point(mapping = NULL, data = NULL, stat = "identity",  
position = "identity", ..., na.rm = FALSE, show.legend = NA,  
inherit.aes = TRUE)
```

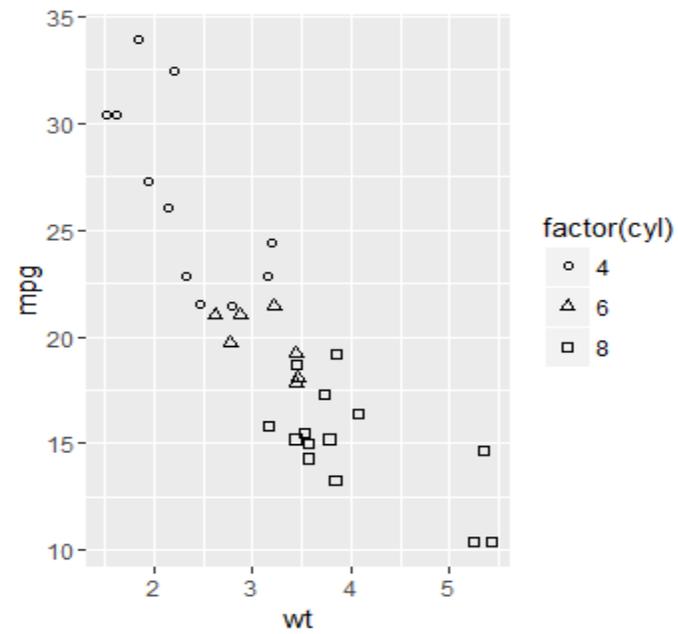
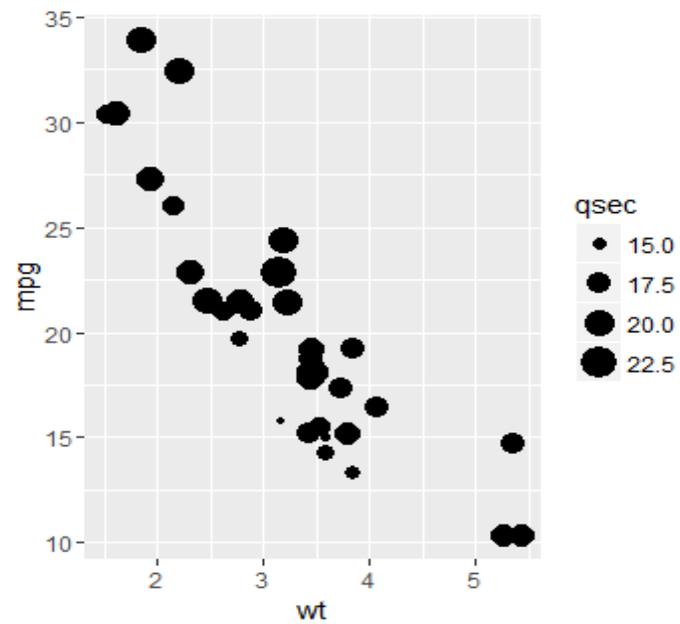
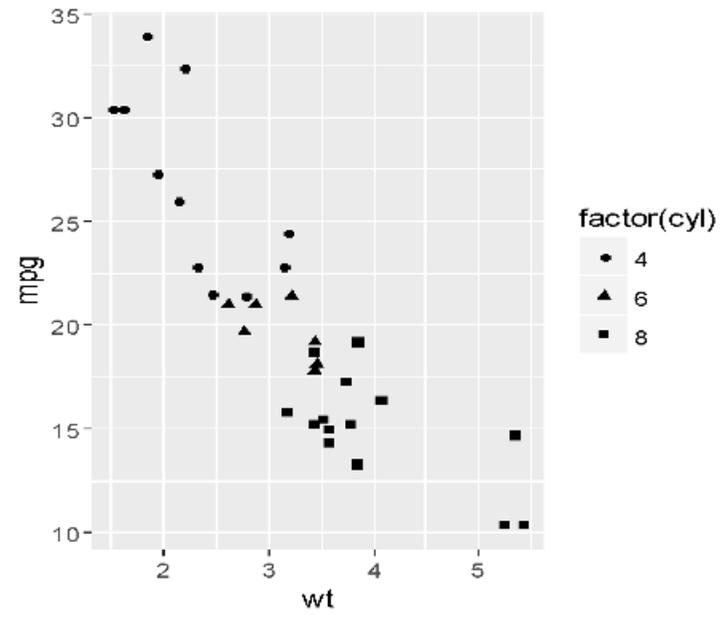
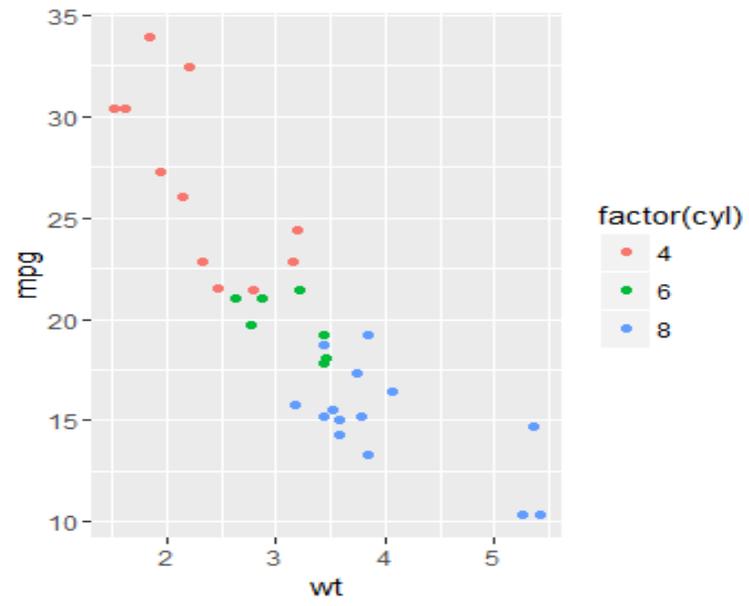
```
ggplot(mtcars, aes(wt, mpg))+geom_point()
```



再来看看图形属性是如何通过[aes](#)来进行设定的，为了简化输入，我们将[ggplot\(\)](#)初始化的图层赋值给变量p，然后分别在P上叠加其他散点图层：

```
p <- ggplot(mtcars, aes(wt, mpg))  
p + geom_point(aes(colour = factor(cyl)))  
p + geom_point(aes(shape = factor(cyl)))  
p + geom_point(aes(size = qsec))  
p + geom_point(aes(shape = factor(cyl))) + scale_shape(solid = FALSE)
```



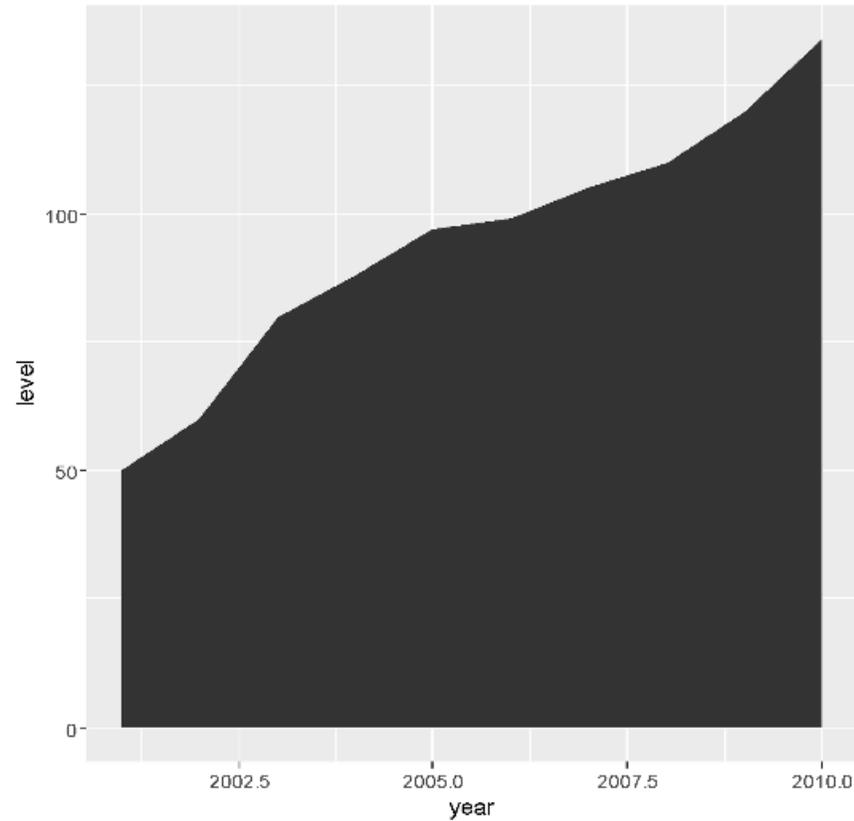


9.4 面积图

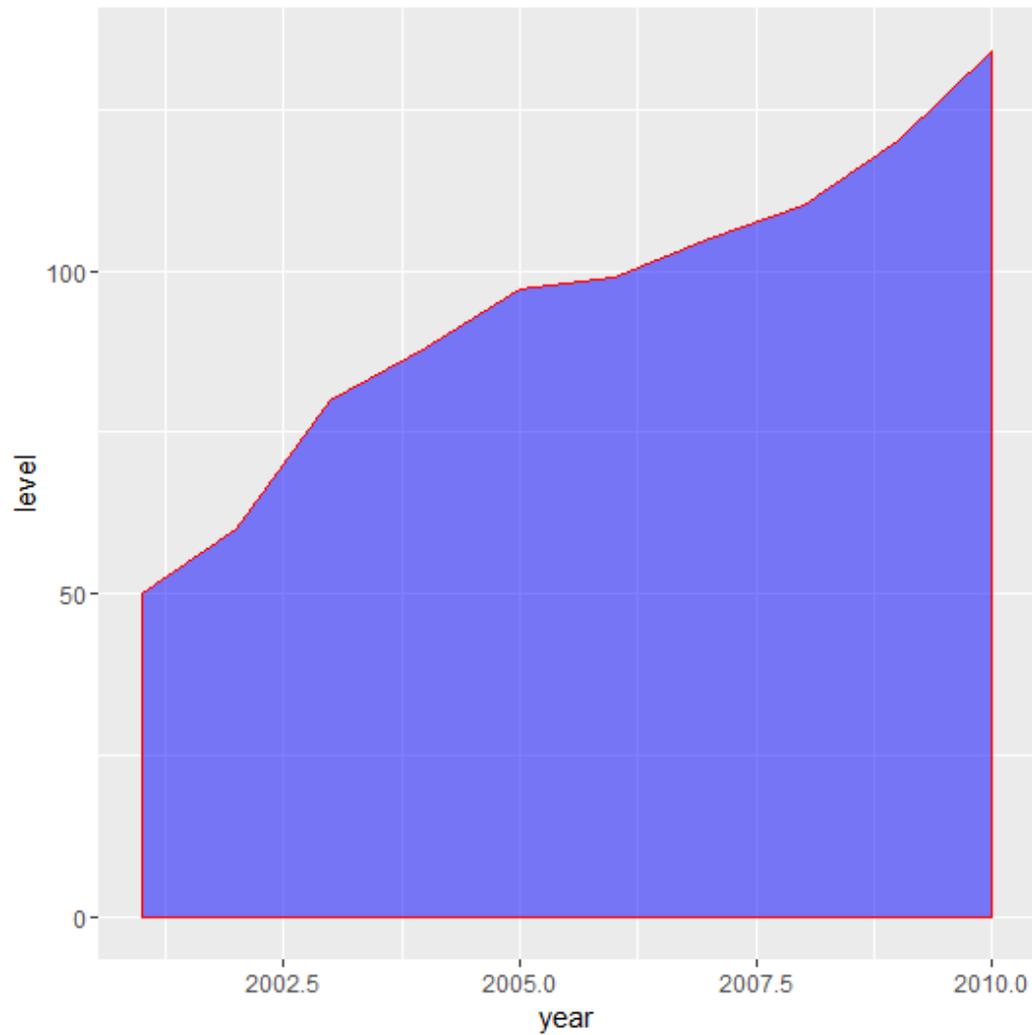
- ◆ 面积图就是折线图的一种特例，它强调因变量随自变量而变化的程度

```
geom_area(mapping = NULL, data = NULL, stat = "identity",  
          position = "stack", na.rm = FALSE, show.legend = NA,  
          inherit.aes = TRUE, ...)
```

```
huron <- data.frame(year = 2001:2010, level =  
c(50,60,80,88,97,99,105,110,120,134))  
h <- ggplot(huron, aes(year))  
h + geom_area(aes(y = level))
```

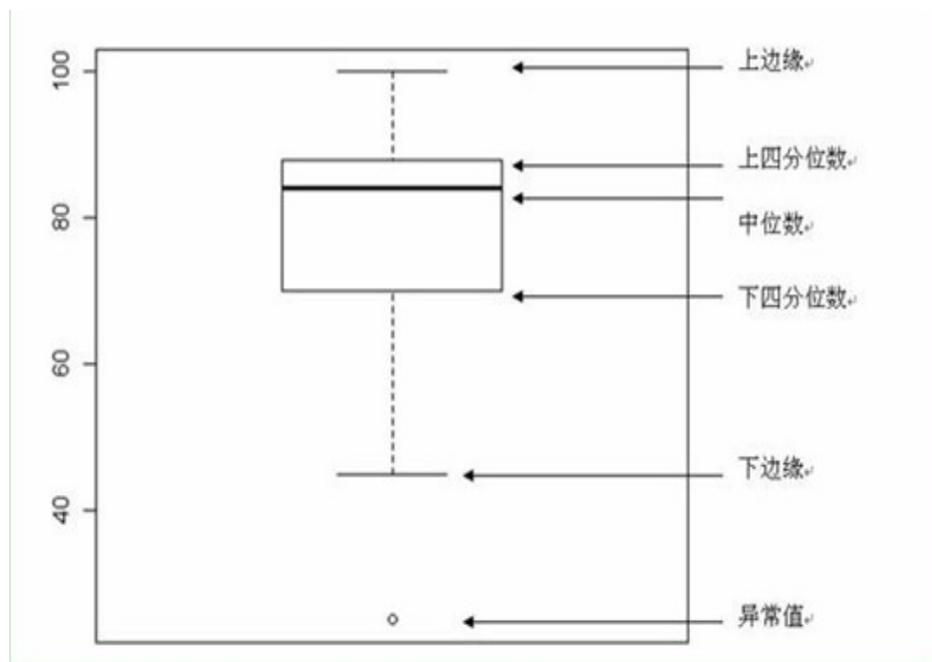


```
ggplot(huron, aes(year)) + geom_area(aes(y = level),  
fill="blue", colour="red")
```



9.5 箱形图

- ◆ 箱形图（**Box-plot**）又称为盒须图、盒式图或箱线图，是一种用作显示一组数据分散情况资料的统计图，每组数据图形由一个矩形及上下两条边缘线组成，因形状如箱子而得名。箱形图主要包含六个数据节点，将一组数据从大到小排列，分别计算出他的上边缘，上四分位数 Q_3 ，中位数，下四分位数 Q_1 ，下边缘，以及异常值。（但`ggplot2`中箱形图分位数的计算方法稍有不同，上下边缘都为四分位数的1.5倍，读者可以通过`?geom_boxplot()`查看计算方法的差异。）



```
geom_boxplot(mapping = NULL, data = NULL, stat = "boxplot",  
  position = "dodge", ..., outlier.colour = NULL, outlier.color = NULL,  
  outlier.shape = 19, outlier.size = 1.5, outlier.stroke = 0.5,  
  notch = FALSE, notchwidth = 0.5, varwidth = FALSE, na.rm = FALSE,  
  show.legend = NA, inherit.aes = TRUE)
```



示例

```
p <- ggplot(mpg, aes(class, hwy))  
p + geom_boxplot()  
p + geom_boxplot() + coord_flip() #坐标轴翻转
```

```
p + geom_boxplot(fill="red", colour="#3366FF", alpha=0.5)  
p + geom_boxplot(outlier.colour = "red", outlier.shape = 1)
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：
<https://d.book118.com/655301120201011203>