# ggplot2数据可视化

我们先了解下 ggplot2 的格式与术语。

****格式与术语****

### ****数据格式****

对ggplot2来说，数据的结构是一成不变的：它要求是“长”格式的数据框，而不是相反的“宽格式”。

当数据为长格式时，每行表示一个条目。其所属的分组不由它们在矩阵中的位置决定，而是在一个单独的列中指定。

### ****术语****

* 数据是我们想要可视化的对象。它包含了若干变量，变量存储于数据框的每一列。
* 几何对象是用以呈现数据的几何图形对象，如条形、线条和点。
* 图形属性是几何对象的视觉属性，如x坐标和y坐标、线条颜色、点的形状等。
* 数值的值和图形属性之间存在着某类映射。
* 标度控制着数据空间的值到图形属性空间的值的映射。一个连续型的y标度会将较大的数值映射至空间中纵向更高的位置。
* 引导元素向看图者展示了如何将视觉属性映射回数据空间。最常见的元素是坐标轴上的刻度线和标签（还有图例）。

接下来以三个数据集解释ggplot2的使用。第一个是lattice包中的singer数据集，它包括纽约合唱团歌手的高度和语音变量。第二个是mtcars数据集，它包含32辆汽车的详细信息。最后一个是car包中的Salaries数据集，它包含大学教授的收入信息，并用来探索性别差异对它们收入的影响。这些数据集提供了各种可视化的挑战。

## ****ggplot2 初探****

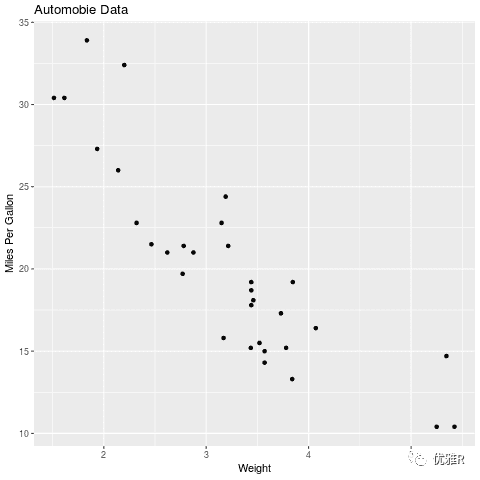
在ggplot2中，图是采用串联起来（+）号函数创建的。每个函数修改属于自己的部分。也就是说，每个函数完成图中各个组件的相应功能，然后通过串联+号将其连接起来，形成一个完整的图形。

> library(ggplot2)

> ggplot(data=mtcars, aes(x=wt, y=mpg)) +

+ geom\_point() +

+ labs(title="Automobie Data", x="Weight", y="Miles Per Gallon")



Scatter Plot between weight and miles of Cars.png

分解上述图形的制作步骤：

ggplot()初始化图形并指定要用到的数据来源和变量。aes()函数的功能是指定每个变量扮演的角色(aes代表aesthetics，即如何用视觉形式呈现信息)。在这里，变量wt的值映射到x轴，mpg的值映射到y轴。

ggplot函数设置图形但没有自己的视觉输出。使用一个或多个几何函数向图中添加了几何对象(简写为geom)，包括点、线、条、箱线图和阴影区域。在上述例子中，geom\_point()函数在图形中画点，创建了一个散点图。labs()函数是可选的，可以添加注释、轴标签、标题等。

ggplot2中有很多函数，并且大多数包含可选的参数。下面我们来看一下相应扩展。

> png('Scatter plot 2.png')

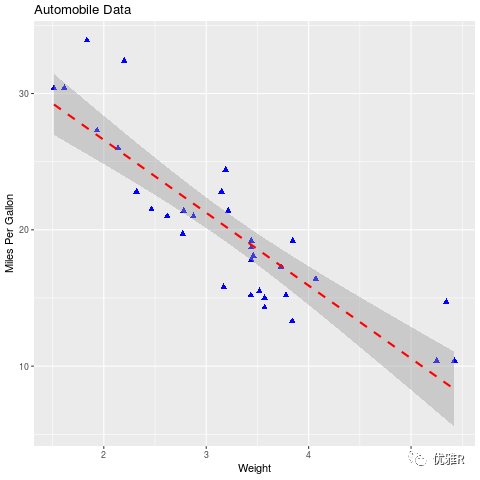
> ggplot(data=mtcars, aes(x=wt, y=mpg)) +

+ geom\_point(pch=17, color='blue', size=2) +

+ geom\_smooth(method='lm', color='red', linetype=2) +

+ labs(title="Automobile Data", x="Weight", y="Miles Per Gallon")

> dev.off()



Scatter plot 2.png

我们依据对最初图形的解释，可以很清晰的观察到不同的函数执行了什么样的功能。

ggplot2包提供了分组和小面化的方法。****分组****指的是在一个图形中显示两组或多组观察结果。****小面化****指的是在单独、并排的图形上显示观察组。需要注意，ggplot2包在定义组或面时使用因子。

这里我们使用mtcars数据集查看分组和面，并进行绘图。

# 将变量转换为因子

mtcars$am <- factor(mtcars$am, levels=c(0,1), labels=c("Automatic", "Manual"))

mtcars$vs <- factor(mtcars$vs, levels=c(0,1), labels=c("V-Engine", "Straight Engine"))

mtcars$cyl <- factor(mtcars$cyl)

library(ggplot2)

# 绘图ggplot(data=mtcars, aes(x=hp, y=mpg, shape=cyl, color=cyl)) +

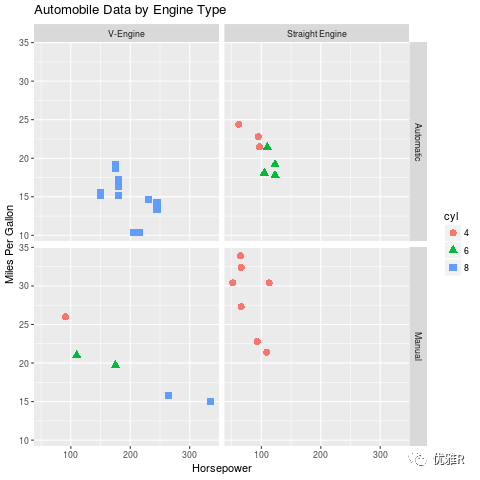
geom\_point(size=3) +

facet\_grid(am~vs) +

labs(title="Automobile Data by Engine Type", x="Horsepower", y="Miles Per Gallon")

复制

在本例中，am和vs是刻度变量，cyl是分组变量。



## ****用几何函数指定图的类型****

ggplot()函数指定要绘制的数据源和变量，几何函数则指定这些变量如何在视觉上进行表示。目前，有37个几何函数可供使用。以下列出常用的函数。

| 函数 | 添加 | 选项 |
| --- | --- | --- |
| geom\_bar() | 条形图 | color, fill, alpha |
| geom\_boxplot() | 箱线图 | color, fill, alpha, notch, width |
| geom\_density() | 密度图 | color, fill, alpha, linetype |
| geom\_histogram() | 直方图 | color, fill, alpha, linetype, binwidth |
| geom\_hline() | 水平线 | color, aplha, linetype, size |
| geom\_jitter() | 抖动点 | color, size, alpha, shape |
| geom\_line() | 线图 | colorvalpha, linetype, size |
| geom\_point() | 散点图 | color, alpha, shape, size |
| geom\_rug() | 地毯图 | color, sides |
| geom\_smooth() | 拟合曲线 | method, formula, color, fill, linetype, size |
| geom\_text() | 文字注解 | 这个非常多,参考相应文档 |
| geom\_violin() | 小提琴图 | color, fill, alpha, linetype |
| geom\_vline() | 垂线 | color, alpha, linetype, size |

关于几何函数的常见选项

| 选项 | 详述 |
| --- | --- |
| color | 对点、线和填充区域的边界进行着色 |
| fill | 对填充区域着色，如条形和密度区域 |
| alpha | 颜色的透明度，从0（完全透明）到1（不透明） |
| linetype | 图案的线条（1=实线，2=虚线，3=点，4=点破折号，5=长破折号，6=双破折号） |
| size | 点的尺寸和线的宽度 |
| shape | 点的形状（和pch一样，0=开放的方形，1=开放的圆形，2=开放的三角形，等等） |
| position | 绘制诸如条形图和点等对象的位置。对条形图来说，'dodge'将分组条形图并排，'stacked'堆叠分组条形图，'fill'垂直地堆叠分组条形图并规范其高度相等。对于点来说，'jitter'减少点重叠。 |
| binwidth | 直方图的宽度 |
| notch | 表示方块图是否应为缺口（TRUE/FALSE) |
| sides | 地毯图的安置（"b"=底部， "l"=左部，"t"=顶部，"r"=右部，"bl"=左下部，等等） |
| width | 箱线图的宽度 |

下面举个例子来验证一下以上参数的使用:

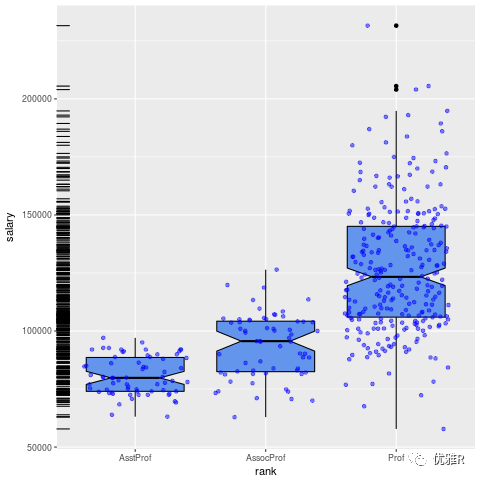
data(Salaries, package='car')library(ggplot2)ggplot(Salaries, aes(x=rank, y=salary)) +

geom\_boxplot(fill="cornflowerblue",

color="black", notch = TRUE) +

geom\_point(position='jitter', color='blue', alpha=0.5) +

geom\_rug(sides='l', color='black')



Salaries by Rank.png

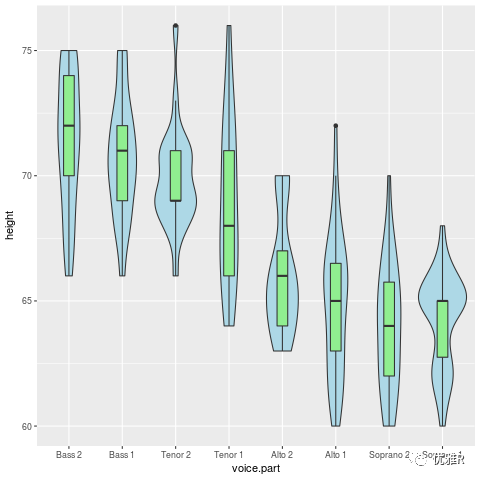
该图显示了不同学术地位对应薪水的缺口箱线图。实际的观察值（教师）是重叠的，因而给予一定的透明度以避免遮挡箱线图。它们还抖动以减少重叠。最后，一个地毯图设置在左侧以指示薪水的一般扩散。

当几何函数组合形成新类型的图时，ggplot2包的真正力量就会得到展示，让我们利用singer数据集再来一探究竟。

library(ggplot2)data(singer, package = "lattice")ggplot(singer, aes(x=voice.part, y=height)) +

geom\_violin(fill="lightblue") +

geom\_boxplot(fill="lightgreen", width=.2)



singer\_combine\_fig.png

箱线图展示了在singer数据框中每个音部的25%,50%,75%分位数得分和任意的异常值。对于每个声部身高范围上的得分分布，小提琴图展示了更多视觉线索。

接下来我们将使用几何函数创建广泛的图表类型。让我们从分组开始吧——在一个图中展示多个分组观察值。

## ****分组****

在R中，组通常用分类变量的水平（因子）来定义。

分组是通过ggplot2图将一个或多个带有诸如颜色、形状、填充、尺寸和线条类型的视觉特征的分组变量来完成的。ggplot()声明中的aes()函数负责分配变量（图形的视觉特征）。

我们依旧以Salaries数据集来进行相关探索。

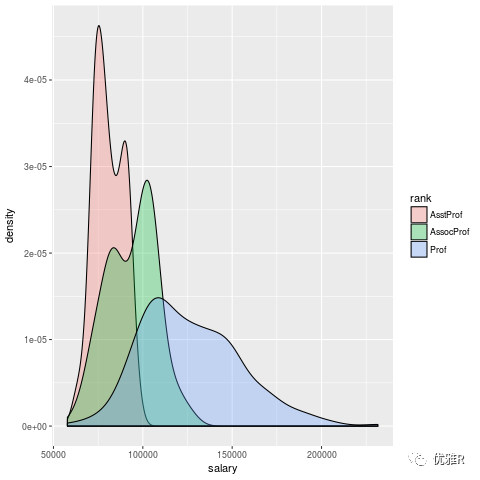
首先，查看薪水是如何随学术等级变化的：

data(Salaries, package='car')

library(ggplot2)

ggplot(data=Salaries, aes(x=salary, fill=rank)) +

geom\_density(alpha=.3)

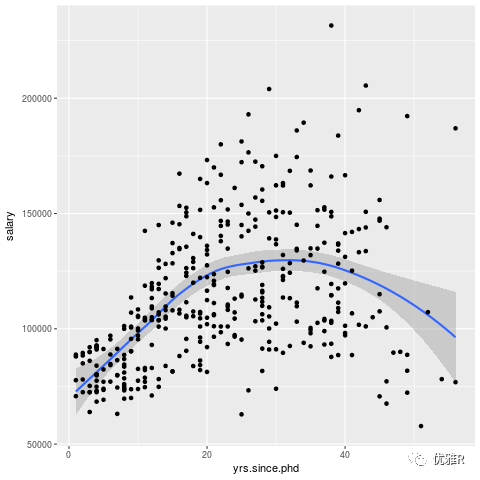


Salaries density by Rank.png

接下来，我们通过性别和学术等级分组，绘制获得博士学位年数和薪水的关系：

ggplot(Salaries, aes(x=yrs.since.phd, y=salary, color=rank, shape=sex)) +

geom\_point()



Salaries by phd.png

****选项可以通过不同的方式使用，这取决于它们发生在**aes()**函数的内部还是外部****。通常来说，****变量应该设在**aes()**函数内，分配常数应该在**aes()**函数外****。

## ****添加光滑曲线****

这一部分我们着重分析一下添加平滑曲线到散点图的方法。

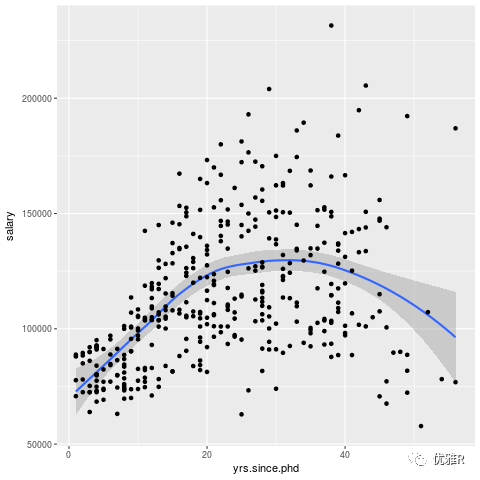
我们可以使用geom\_smooth()函数来添加一系列的平滑曲线和置信区域。函数的参数参考下表：

| 选项 | 描述 |
| --- | --- |
| method= | 使用的平滑函数。允许的值包括lm, glm, smooth, rlm, glm,分别对应线性、广义线性、loess、健壮线和广义相加模型。smooth是默认值 |
| formula= | 在光滑函数中使用的公式。例子包括y~x, y~log(x), y~poly(x,n), y~ns(x) |
| se | 绘制置信区间（TRUE/FALSE）默认为TRUE |
| level | 使用的置信区间水平（默认为95%） |
| fullrange | 指定拟合应涵盖全图(TRUE)，或仅仅是数据(FALSE)。默认为FALSE |

使用Salaries数据集，忽略性别和学术等级，我们先检验博士毕业年数和薪水之间的关系。

data(Salaries, package='car')library(ggplot2)ggplot(data=Salaries, aes(x=yrs.since.phd, y=salary)) +

geom\_smooth() + geom\_point()



Salaries by yrs.png

****统计函数****：

ggplot2包中含有大量统计函数来计算所需的量，从而生产更多的可视化数据。通常情况下，几何函数隐式地调用统计函数，我们不需要直接处理这些问题。不过指导它们的存在是有用的。

## ****修改ggplot2图形的外观****

R的基础绘图中,使用par()函数或特定的画图函数的图形参数来自定义基本函数。遗憾的是，这些对ggplot2图形没有影响，该包提供了特定了函数来改变其图形的外观。

### ****坐标轴****

ggplot2包会自动生成基本所需要的图形参数。当我们需要更大程度定制时，需要了解相应函数的用法。我们已经知道labs()函数可以用来添加标题并改变坐标轴标签，让我们再看看其他的有用函数：

| 函数 | 选项 |
| --- | --- |
| scale\_x\_continuous()和scale\_y\_continuous() | breaks=指定刻度标记、labels=指定刻度标记标签、limits=控制要展示的值的范围 |
| scale\_x\_discrete()和scale\_y\_discrete() | breaks=对因子的水平进行放置和排序，labels=指定这些水平的标签，limits=表示哪些水平应该展示 |
| coord\_filp() | 颠倒x轴和y轴 |

们将这些函数应用一个分组箱线图中，其中包含按学术等级和性别分组的薪资水平，代码如下：

data(Salaries, package='car')library(ggplot2)ggplot(data=Salaries, aes(x=rank, y=salary, fill=sex)) +

geom\_boxplot() +

scale\_x\_discrete(breaks=c('AsstProf', 'AssocProf', 'Prof'),

labels=c('Assistant\nProfessor',

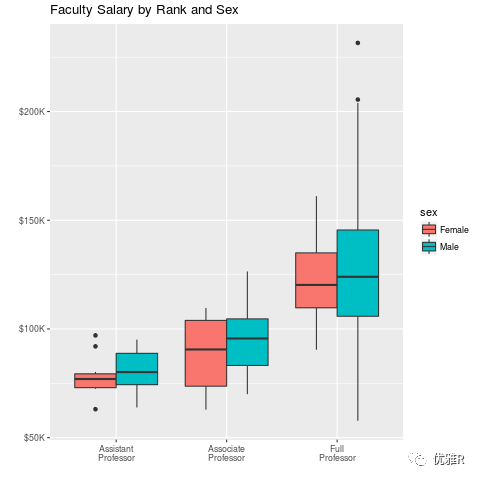
"Associate\nProfessor",

"Full\nProfessor")) +

scale\_y\_continuous(breaks = c(50000, 100000, 150000, 200000),

labels=c('$50K','$100K','$150K','$200K')) +

labs(title="Faculty Salary by Rank and Sex", x='', y='')



Faculty Salary by Rank and Sex.png

### ****图例****

图例是指如何用颜色、形状、尺寸等视觉特征表示数据特征的指南。标题和位置是最常用的定制特征。

当更改图例的标题时，必须综合考虑颜色、填充、尺寸等等。可以通过fill="mytitle"加到labs()函数中来改变标题。

标题的位置由theme()函数中的legen.position选项控制。可能的值包括left, top, right(默认), bottom。我们也可以在图中给定的位置指定一个二元素向量。

使用添加修改上一个图的代码对图形展示效果进行修改：

labs(title="Faculty Salary by Rank and Sex", x='', y='',fill='Gender')

theme(legend.position=c(.1,.8)) # 图例的左上角分别距离左侧边缘10%，底部边缘80%

### ****标尺****

ggplot2包使用标尺把数据空间的观察值映射到可视化的空间中。标尺可以连续也可以离散。

在ggplot2中标尺的概念很普遍，可以通过查看以scale\_开头的函数来了解更多信息。

### ****主题****

主题可以让我们控制这些图的整体外观。theme()函数中的选项可以让我们调整字体、背景、颜色和网格线等。主题可以使用一次，也可以保存起来应用到多个图中。尝试探索以下代码：

data(Salaries, package = 'car')library(ggplot2)

mytheme <- theme(plot.title=element\_text(face="bold.italic",

size = "14", color = "brown"),

axis.title=element\_text(face="bold.italic", size=10,

color="brown"),

axis.text=element\_text(face="bold", size=9,

color="darkblue"),

panel.background = element\_rect(fill="white",

color="darkblue"),

panel.grid.major.y=element\_line(color="grey",

linetype = 2),

panel.grid.minor.y=element\_line(color="grey",

linetype=2),

panel.grid.minor.x=element\_blank(),

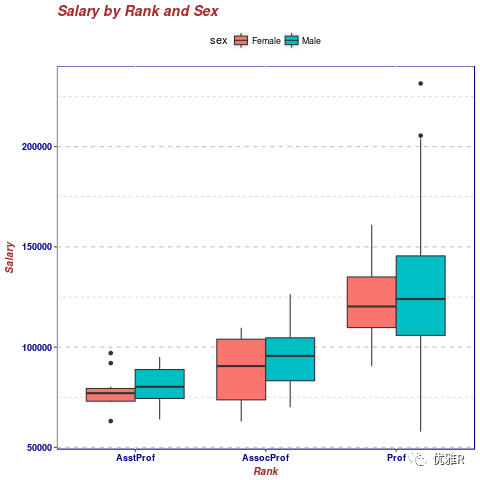
legend.position = "top")

ggplot(Salaries, aes(x=rank,y=salary,fill=sex)) +

geom\_boxplot() +

labs(title="Salary by Rank and Sex", x="Rank", y="Salary") +

mytheme



mytheme.png

### ****多重图****

基础绘图中，我们使用图形参数mfrow和基本函数layout()把两个或多个基本图放到单个图中，同样，这种方法在ggplot2中不适用。将多个ggplot2包的图形放到单个图形中最简单的方式是使用gridExtra包中的grid.arrange()函数。我们需要事先安装这个包。

让我们创建3个ggplot2图并把它放在单个图形中。

data(Salaries, package = 'car')library(ggplot2)

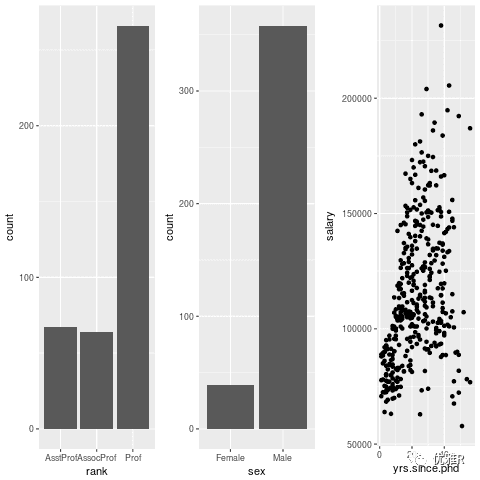
p1 <- ggplot(data=Salaries, aes(x=rank)) + geom\_bar()

p2 <- ggplot(data=Salaries, aes(x=sex)) + geom\_bar()

p3 <- ggplot(data=Salaries, aes(x=yrs.since.phd, y=salary)) + geom\_point()

library(gridExtra)

grid.arrange(p1,p2,p3,ncol=3)



multiple pic.png

注意截面图（刻面图）和多重图的区别。

### ****保存图形****

可以使用标准方法来保存创建的图形，也可以使用ggsave()函数更方便保存它们。它的选项包括保存哪幅图形，保存在哪里和以什么形式保存。例如

myplot <- ggplot(data=mtcars, aes(x=mpg)) + geom\_histogram()ggsave(file="mygraph.png",plot=myplot,width=5,height=4)

复制

将myplot保存为5英寸X4英寸PNG格式。我们可以通过设置文件拓展名为ps, tex, jpeg, pdf, tiff, png, bmp, svg, wmf来保存为不同格式。

如果忽略plot=选项，最近创建的图形会被保存。更多细节参考help(ggsave)。

素材来源：公众号-优雅R