数据可视化及 ggplot2 绘图 (初阶)

左敏

1 绘图简介

1.1 基础图形系统与 ggplot2 包

基础图形系统类似于“艺术家的调色板”，以空白画布为基础，首先利用高级函数 (如，*plot( )*、*hist( )*、*boxplot( )* 等) 进行绘图，随后通过低级函数 (如，*abline( )*、*axis( )* 等) 添加 / 修改文本、线、点、轴等。

ggplot2 包由 Hadley Wickham (2009a) 编写，它提供了一种基于 Wilkinson (2005) 所描述的图形语法的图形系统，Wickham (2009b) 还对该语法进行了扩展。ggplot2 包的目标是提供一个全面的、基于语法的、连贯一致的图形生成系统，允许用户创建新颖的、有创新性的数据可视化图形。其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。此外，ggplot2 包按图层作图；保有命令式作图的调整函数，使其更具灵活性；并且将常见的统计变换融入到了绘图中。

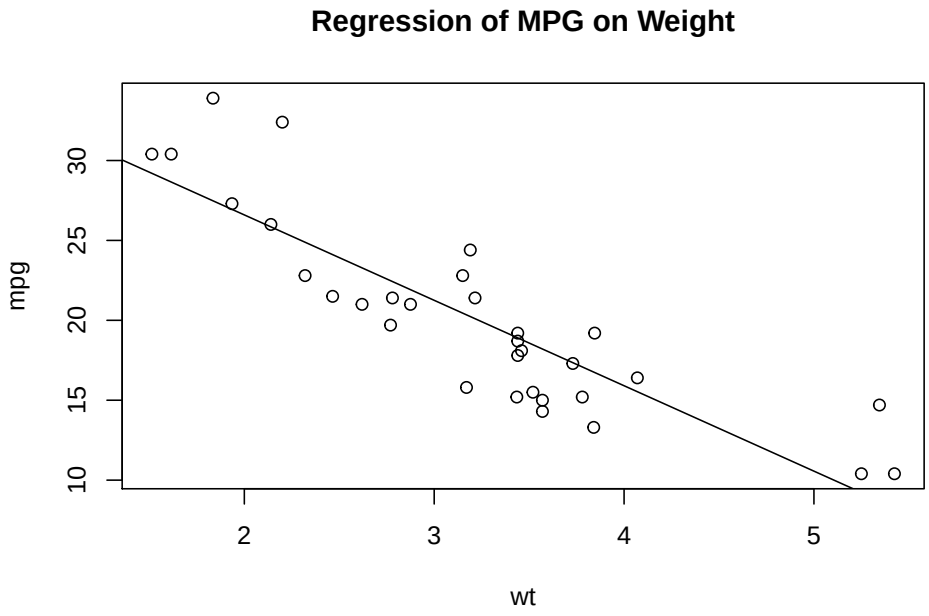
请看如下代码：

```
library(ggplot2)
data("mtcars")
attach(mtcars)
```

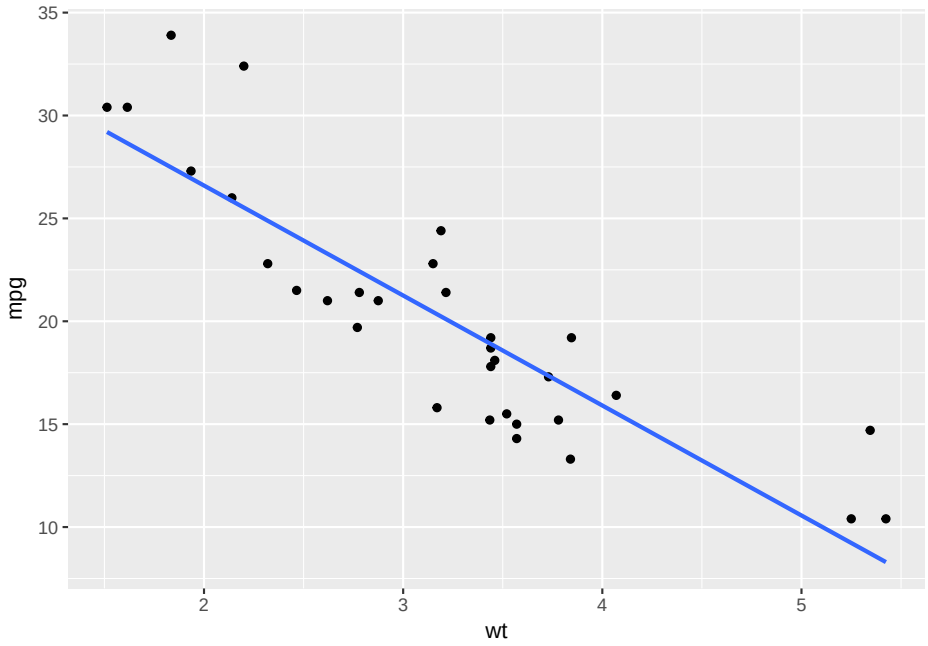
```
## The following object is masked from package:ggplot2:
##
##     mpg
```

```
## Base 版
# wt<-mtcars$wt mpg<-mtcars$mpg
plot(wt, mpg)
```

```
abline(lm(mpg~wt))
title("Regression of MPG on Weight")
```




```
# 图的标题应该在下方 R 默认主标题在上方 注意一下
#base 版通过改变 type 的数字改变得到的图形种类 1 为折线图
#ggplot2 通过 geom 后面的参数进行变换

## ggplot2 版
ggplot(mtcars, aes(wt,mpg)) +
        geom_point() +
        geom_smooth(method = "lm", se = FALSE)
```




```
# ?geom_method
# ?geom_abline
detach(mtcars)
# View(mtcars$wt)
# View(mtcars)
```

1.2 图片保存

在 R 中，我们可以通过 *pdf( )*、*win.metafile( )*、*png( )*、*jpeg( )*、*bmp( )*、*tiff( )*、*xfig( )* 以及 *postscript( )* 函数对绘制图形进行保存。需要注意的是，Windows 图元文件格式仅在 Windows 系统中可用。

以 *pdf( )* 为例的图片保存代码，如下所示：

```
pdf("mygraph.pdf")
attach(mtcars)
```

```
## The following object is masked from package:ggplot2:
##
##     mpg
```

```
plot(wt, mpg)
abline(lm(mpg~wt))
title("Regression of MPG on Weight")
detach(mtcars)
dev.off()
```

```
## pdf
##   2
```

```
# ?dev.off()
```

2 基础统计图形绘制

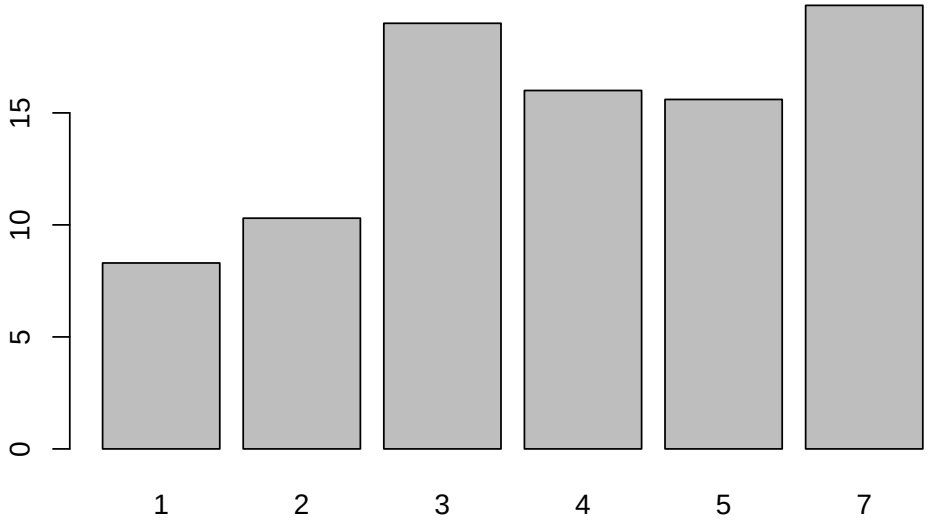
无论是在看报告还是看论文时，我们都会不可避免地遇到一些统计图表。常见的统计图表有柱状图、直方图、箱线图等。本次课程，我们将以柱状图、直方图以及箱线图为例，给大家展示这些图形的 R 语言实现。

2.1 柱状图

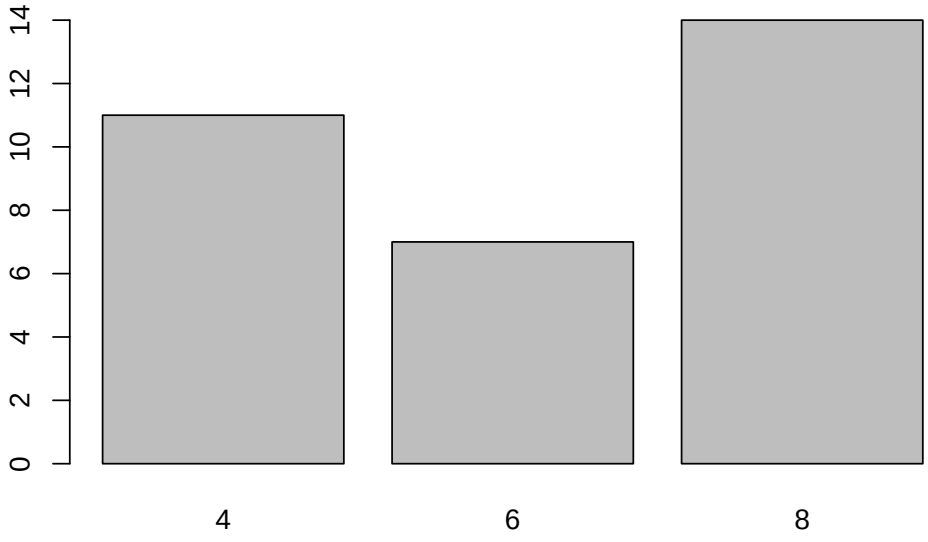
柱状图的特点及用途：

- 柱状图针对离散型变量
- 柱状图展示频数、展示常用的统计量，展示回归系数估计值。

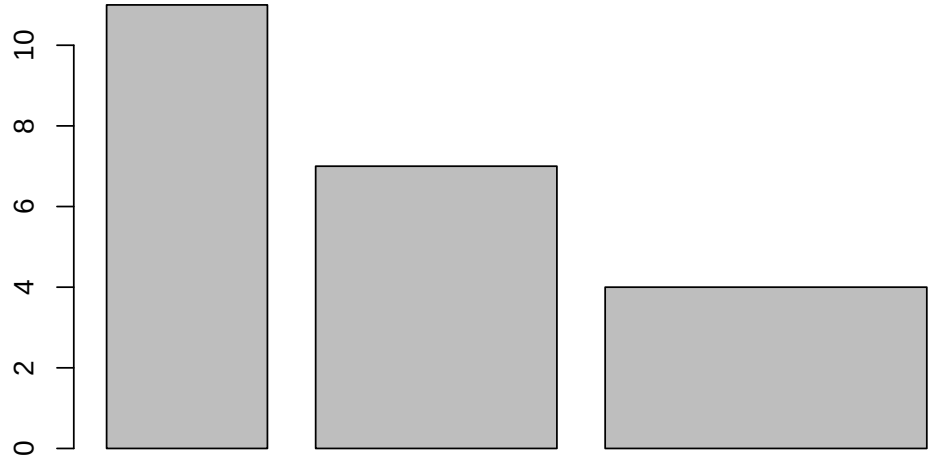
```
# ?barplot

# 1
barplot(BOD$demand, names.arg = BOD$Time)
```




```
# 2
#ggplot(BOD, aes(x = Time, y = demand)) +geom_bar(stat = "identity")

# 3
barplot(table(mtcars$cyl))
```



```r
# 4
#ggplot(BOD, aes(x = factor(Time), y = demand)) +geom_bar(stat = "identity")

# 5
width <- c(4, 6, 8)
height<- c(11, 7, 4)
barplot(height, width)
```



2.2 直方图

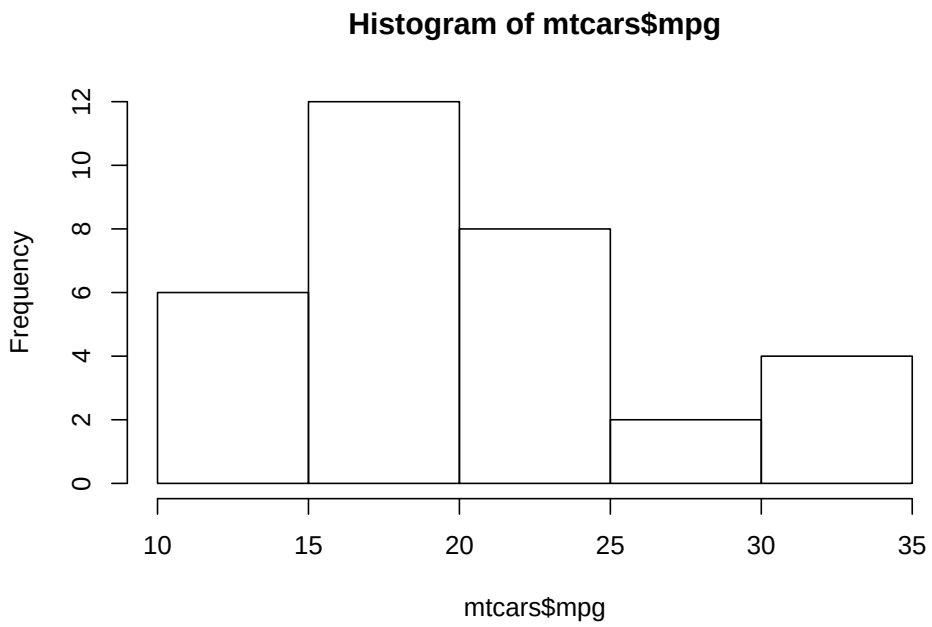
直方图的特点及用途：

- 直方图是针对连续型变量所做的统计图。
- 直方图是表示分布情况的图形，也称为数据探索性分析。

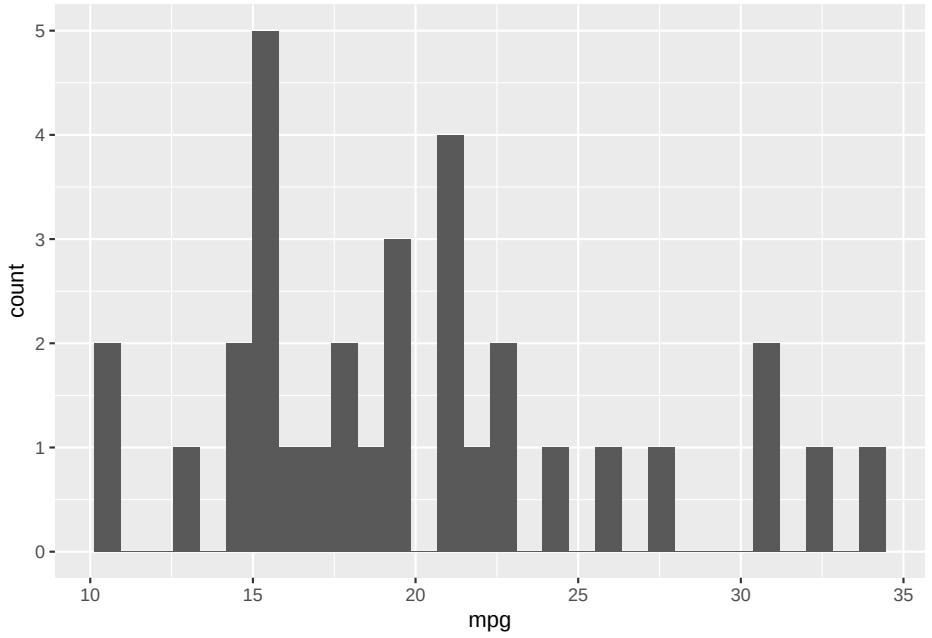
注意事项：柱状图与直方图的区别：柱状图针对离散型变量，直方图针对连续型变量

直方图：JASA 文章回顾统计学思想，进行有趣的调查美国统计协会的理事以及在澳大利亚统计局工作的朋友，列举 3 个他们认为最简单实用的统计思想或者方法。排在第一位的，是直方图/作图/探索性分析（histograms/plot the data/exploratory analysis）。直方图是被提及最多的统计。虽然这项调查有偏差，但直方图确实是一种应用广泛的统计图。那么今天我们来讲一下这个格外受统计学家欢迎的直方图。

```
hist(mtcars$mpg,breaks = 4)
```



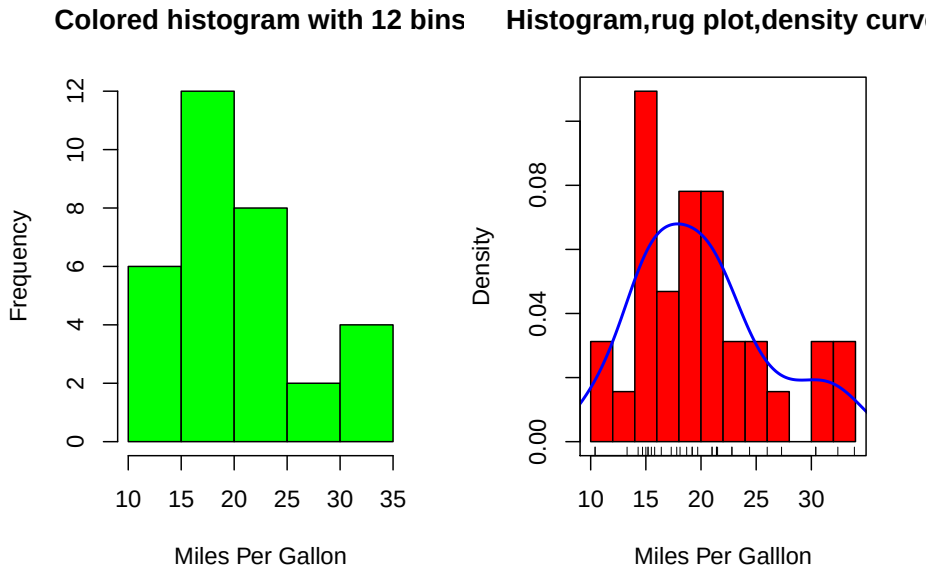

```
ggplot(mtcars, aes(x = mpg))+
        geom_histogram(bins = 30)
```



```
#View(mpg)
#?geom_histogram
#binwidth 组距 bin 可翻译为直条
# 按照经验，其实 `geom_histogram`中最常用的参数是 `bins = 30`
# 而不是 `bindwidth = some_number`，因为直方图一般是做探索性分析的，
# 你一般是一开始完全不知道数据的分布的，所以也很难确定 `bindwidth`参数，
# 很有可能画出来图很奇怪；但 `bins`参数取 30 ~ 50 一般总归没错。
attach(mtcars)
```

```
## The following object is masked from package:ggplot2:
##
##     mpg
```

```
par(mfrow = c(1, 2))
hist(mpg, breaks = 7, col = "green", xlab = "Miles Per Gallon",
     main = "Colored histogram with 12 bins")
hist(mpg, freq = FALSE, breaks = 12, col = "red",
     xlab = "Miles Per Galllon",
     main = "Histogram,rug plot,density curve")
#?hist
#?rug
rug(mpg)
#rug 是在坐标轴上标出元素出现的频数。出现一次，就会画一个小竖杠。
#rug 越密，density 越高。
lines(density(mpg), col = "blue", lwd = 2)
box()
```



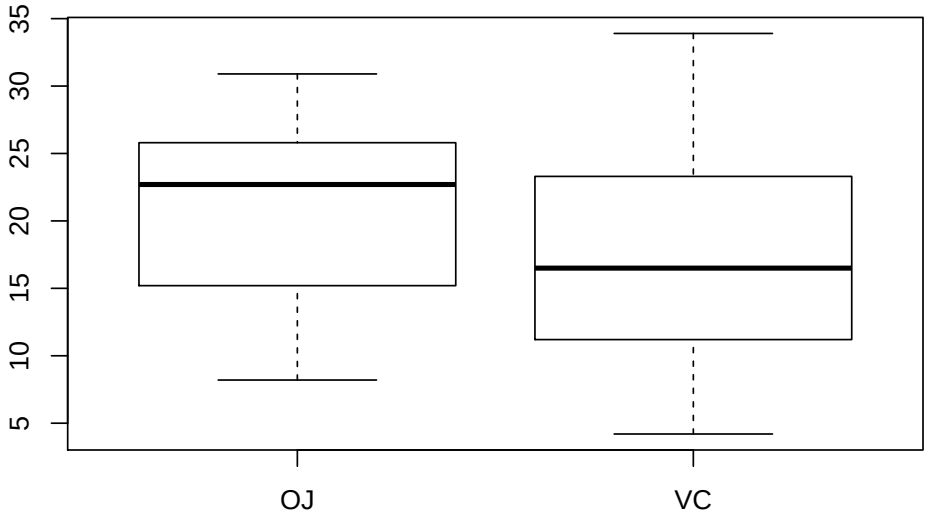

```
par(mfrow = c(1,1))
detach(mtcars)
```

2.3 箱线图

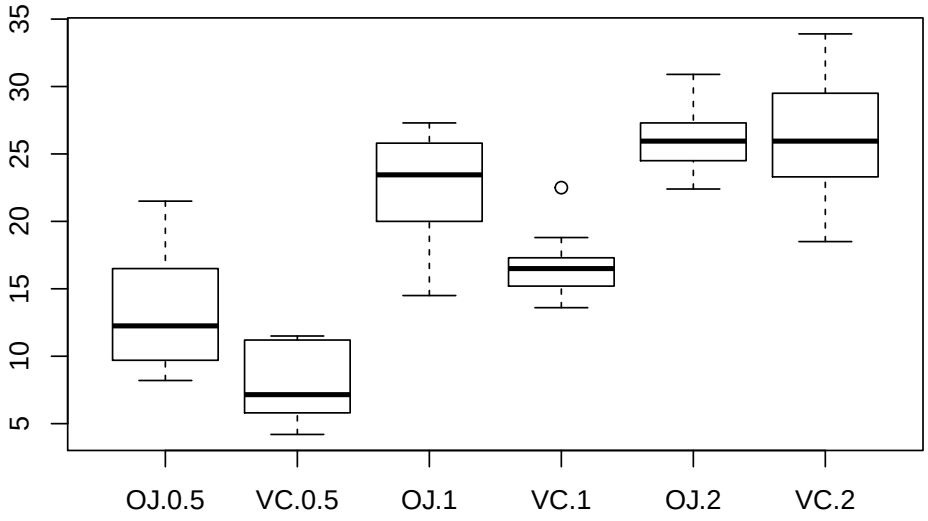
箱线图的特点及用途：

- 箱线图是针对连续型变量的，解读时候重点关注平均水平、波动程度和异常值。
- 箱线图一般按照中位数从小到大排列，箱体的宽窄表示样本量的大小。
- 箱子的上下限，分别是数据的上四分位数和下四分位数。这意味着箱子包含了 50% 的数据。因此，箱子的高度在一定程度上反映了数据的波动程度。
- 当箱子被压得很扁，或者有很多异常的时候，试着做对数变换。
- 当只有一个连续型变量时，并不适合画箱线图，直方图是更常见的选择。
- 箱线图最有效的使用途径是作比较，配合一个或者多个定性数据，画分组箱线图。

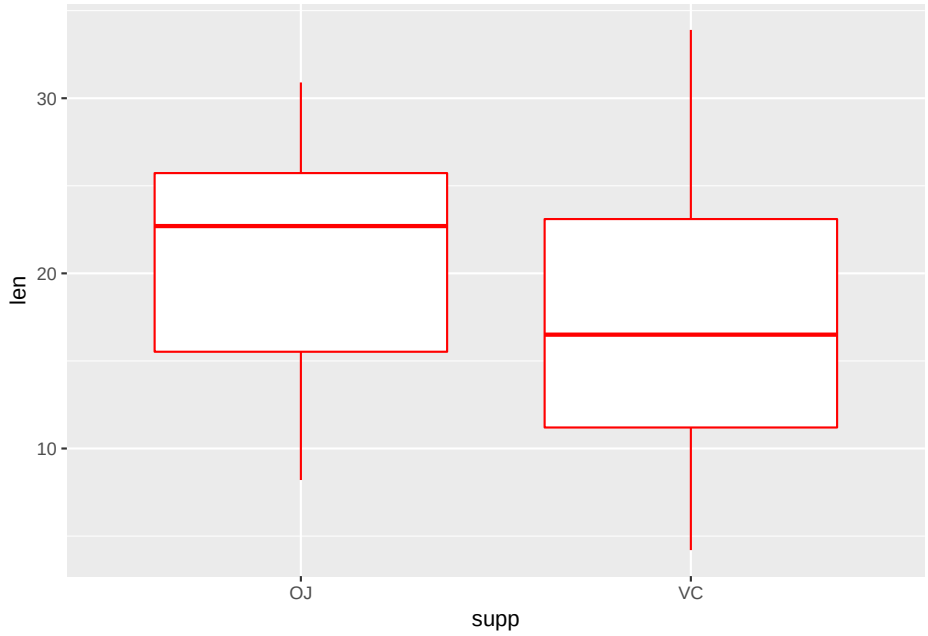
```
rm(list=ls())
boxplot(len~supp, data = ToothGrowth)
```



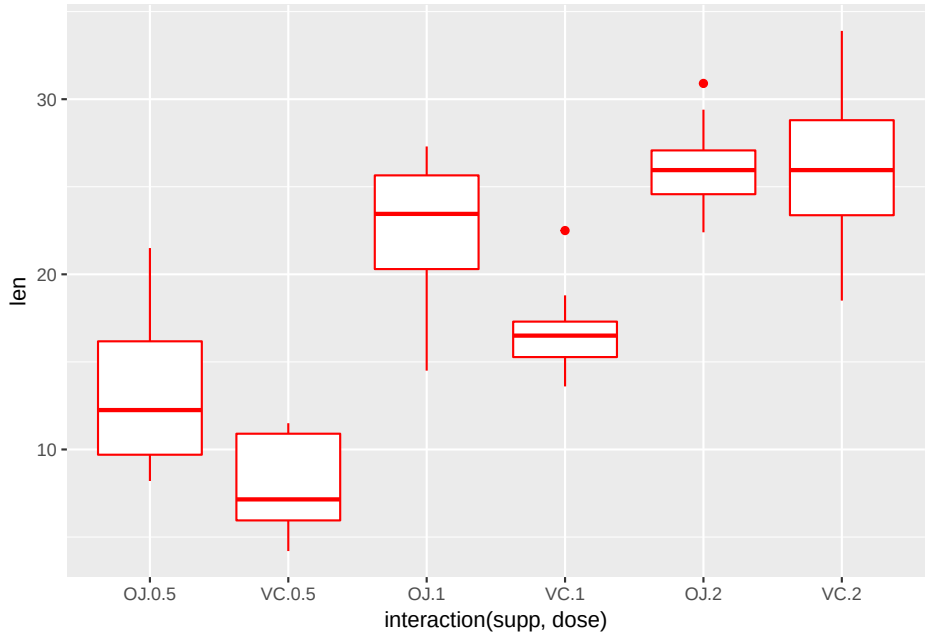

```
boxplot(len~supp+dose, data = ToothGrowth)
```




```
ggplot(ToothGrowth, aes(supp, len))+
        geom_boxplot(color = 'red')
```




```
ggplot(ToothGrowth, aes(x = interaction(supp, dose), len))+
        geom_boxplot(color = 'red')
```



3 参数调整

3.1 作图说明，调整与实例

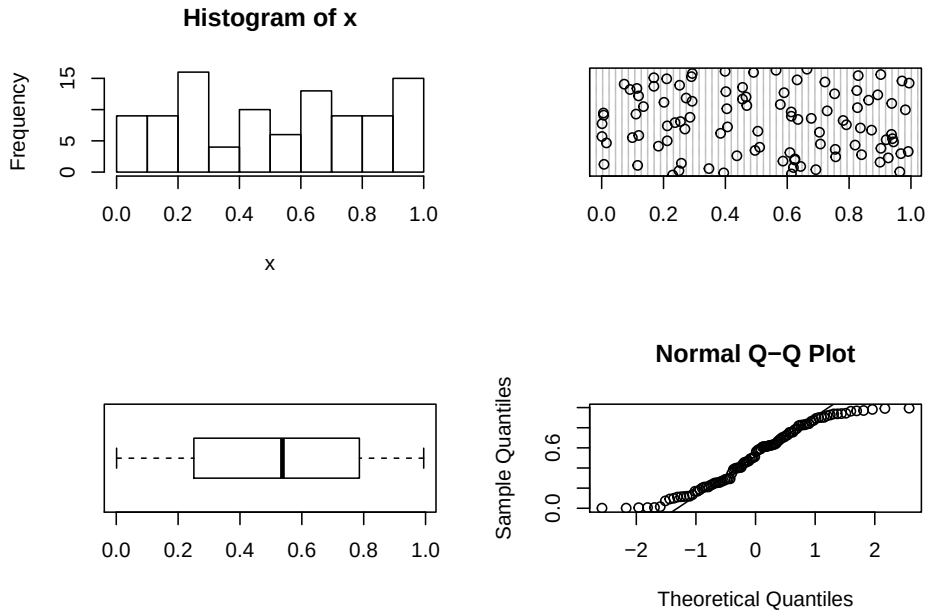
一个完整的统计图，包含以下几个要素或者注意事项：

- 要有图标题，一般在图的下方，标题要简洁明了。同时，报告中的统计图要有标号。
- 横轴和纵轴要标注清楚（横轴：职称；纵轴：频数）。如果有单位的话，需要注明。
- 图的标题、横轴、纵轴等，出现的文字要统一和准确，不要一会儿中文，一会儿英文。写中文报告，就都标注中文。
- 图的比例要协调，别太胖别太瘦，别太高也别太矮。
- 图的内容要正确、简明，避免出现不必要的标签、背景等。
- 注意图的配色。不要精挑细选一组非常难看的配色！画完了自己先看看，是不是人类能够接受的颜色。
- 画完图要有适当的评述，尤其是在报告里，这点非常重要。

3.2 图形布局

合理图形布局有利于读者阅读。通过 *par( )* 函数将不同的作图同时做出进行比较、每种图形各有优劣。

```
X =  runif(100)
EDA <- function(x)
{par(mfrow = c(2, 2))
  hist(x);
  dotchart(x)
  boxplot(x,horizontal = T)
  qqnorm(x);qqline(x)
  par(mfrow = c(1, 1))
  }
EDA(X)
```



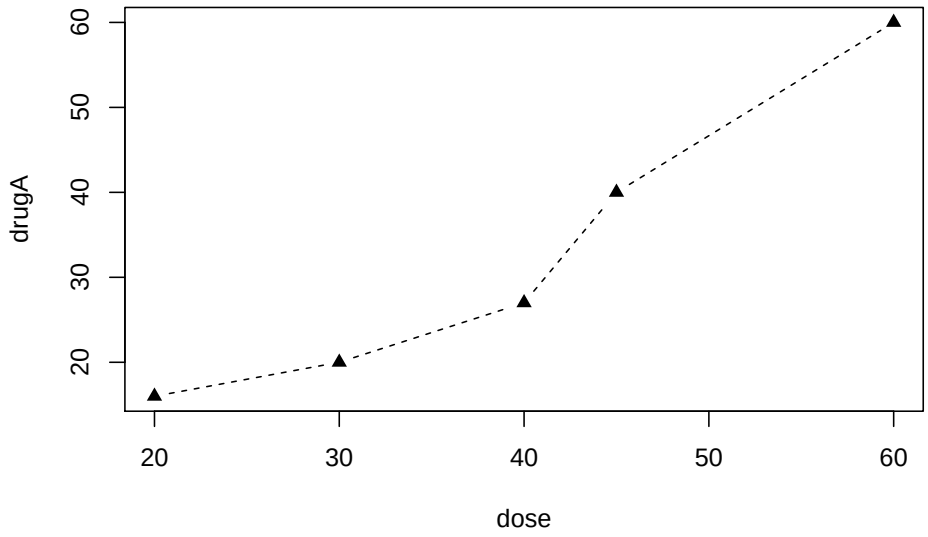
3.3 图形参数

我们可以通过修改称为图形参数的选项来自定义一幅图形的多个特征。大致参数如下：

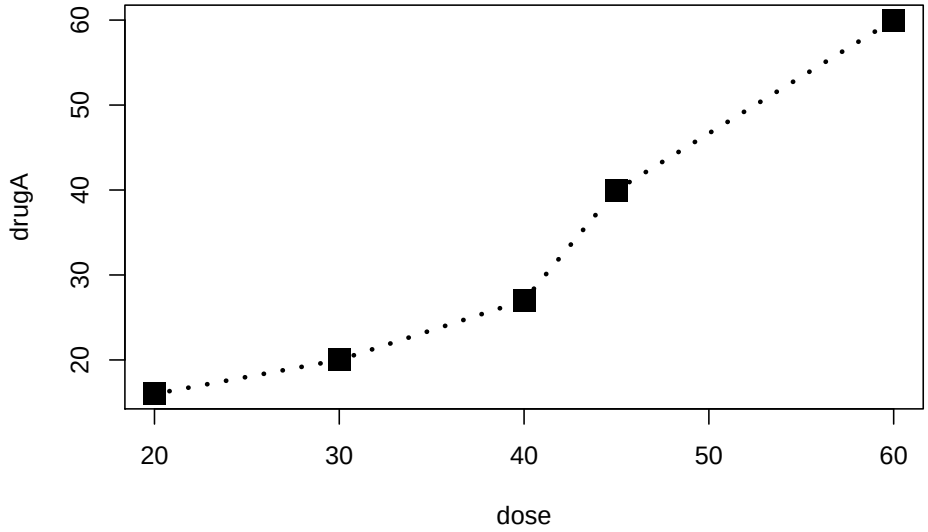
- *pch*：指定绘制点时使用的符号。
- *cex*：指定符号的大小。cex 是一个数值，表示绘图符号相对于默认大小的缩放倍数。
- *lty*：指定线条类型。
- *lwd*：指定线条宽度。lwd 是以默认值的相对大小来表示的（默认值为 1）。例如，lwd=2 将生成一条两倍于默认宽度的线条。
- *font*：整数。用于指定绘图使用的字体样式。1= 常规，2= 粗体，3= 斜体，4= 粗斜体，5= 符号字体。

```
dose <- c(20, 30, 40, 45, 60)
drugA<- c(16, 20, 27, 40, 60)
drugB<- c(15, 18, 25, 31, 40)
```

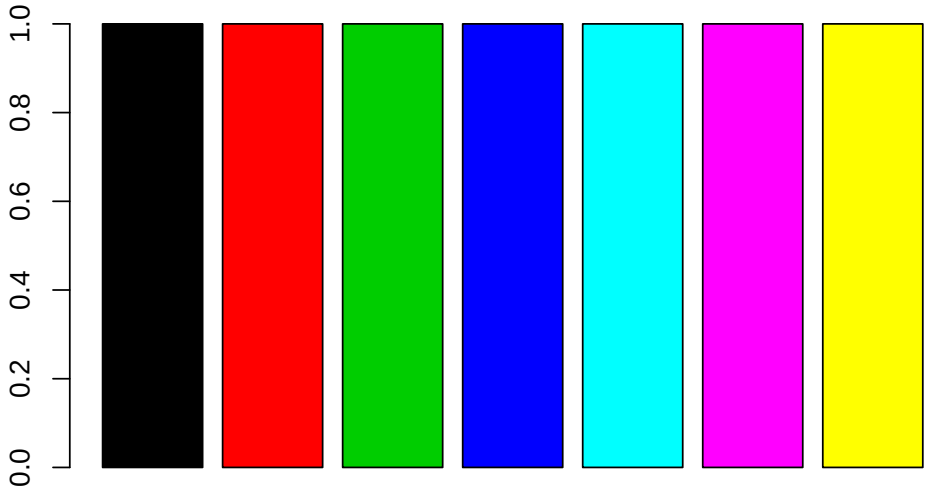
```r
opar <- par(no.readonly=TRUE)# 对原始参数进行处理
# 添加参数 no.readonly=TRUE 可以生成一个可以修改的当前图形参数列表

# 符号和线条
par(lty = 2,pch = 17)
plot(dose, drugA, type = "b")
```



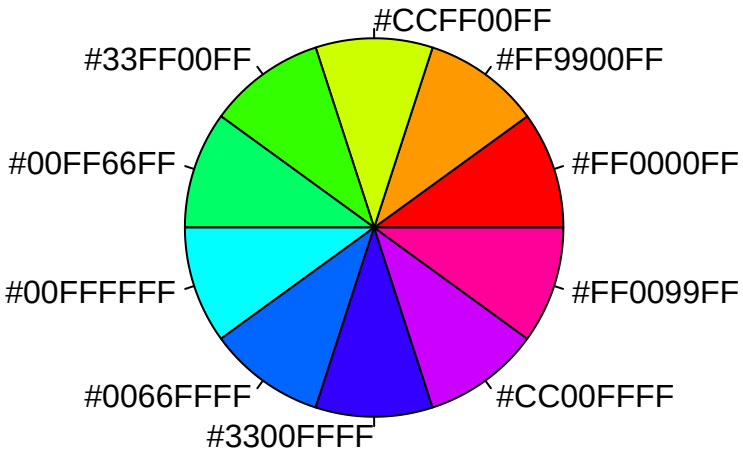

```r
par(opar)
# 可以随心所欲多次使用 par() 函数, par(lty=2,pch=17) 也可以写成
#par(lty=2)
#par(pch=17)
# 工科试验经常会用到三角等图形 操作点
plot(dose, drugA, type = "b",lty = 3, lwd = 3, pch = 15, cex = 2)
```



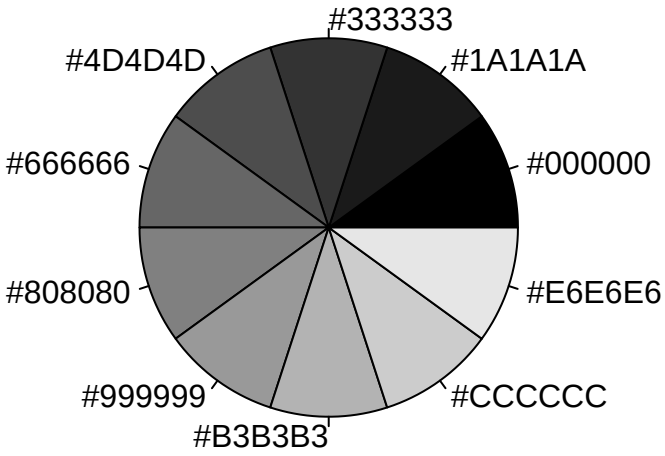

```
# 颜色 饼状图
#col=1 col="white" col="#FFFFFF" col=rgb(1,1,1) col=hsv(0,0,1)
#rgb()基于红绿蓝三色值生成颜色，hsv() 基于色相—饱和度—亮度值生成颜色

n <- 7
barplot(rep(1, n), col = 1:n)
```



```
## 饼状图
n <- 10
mycolors <- rainbow(n)
#10 种连续的彩虹色
pie(rep(1, n), labels = mycolors, col = mycolors)
```



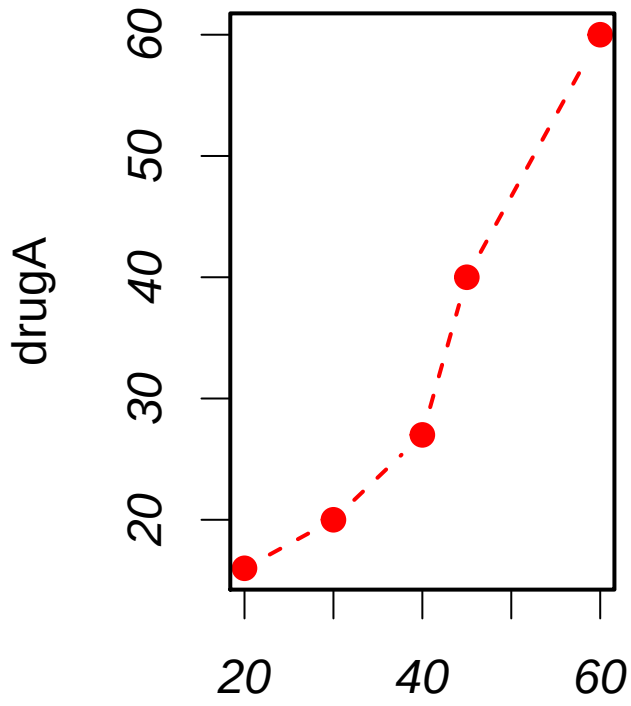

```
mygrays <- gray(0:n/n)
#10 阶灰度色
pie(rep(1, n), labels = mygrays, col = mygrays)
```



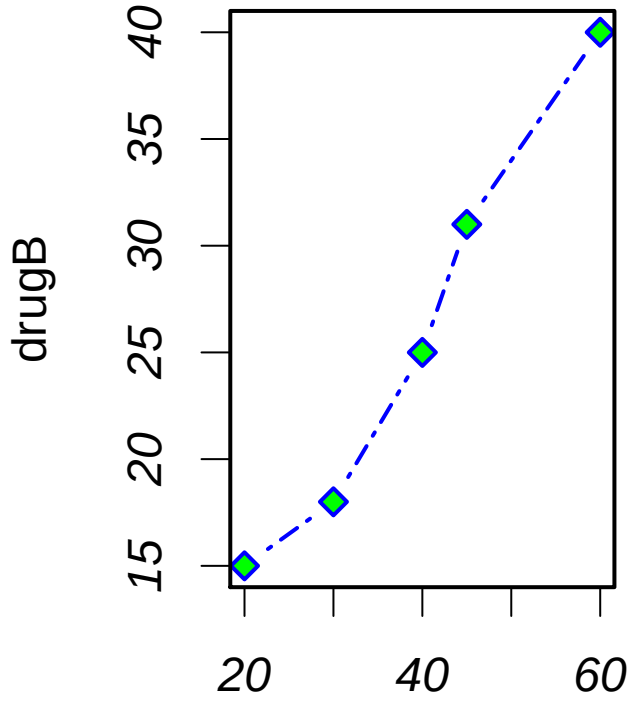
```
# 图形尺寸与边界尺寸
opar <- par(no.readonly = TRUE)
par(pin = c(2, 3))
par(lwd = 2, cex = 1.5)
par(cex = 1.5, axis = 0.75, font.axis = 3)
```

```
## Warning in par(cex = 1.5, axis = 0.75, font.axis = 3): "axis" is not a
## graphical parameter
```

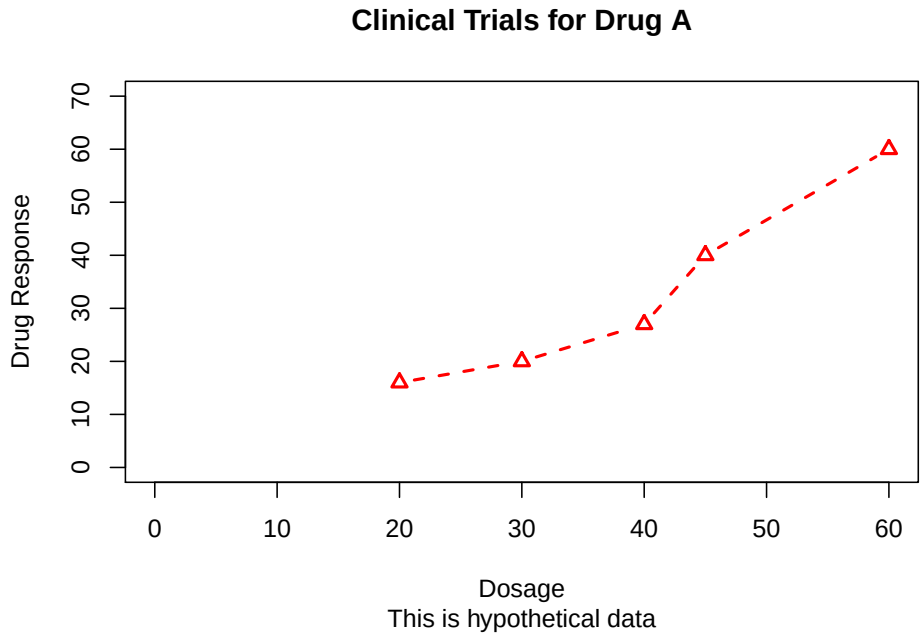
```
plot(dose, drugA, type = "b", pch = 19, lty = 2, col = "red")
```




```
plot(dose, drugB, type = "b", pch = 23, lty = 6, col = "blue", bg = "green")
```




```
par(opar)

# 添加标题(main), 副标题(sub),
# 坐标轴标签(xlab,ylab) 指定坐标轴范围(xlim,ylim)
plot(dose, drugA, type = "b", col = "red", lty = 2, pch = 2, lwd = 2,
     main= "Clinical Trials for Drug A",
     sub = "This is hypothetical data",xlab="Dosage",ylab="Drug Response",
     xlim= c(0, 60), ylim = c(0, 70))
```



3.4 自定义坐标轴

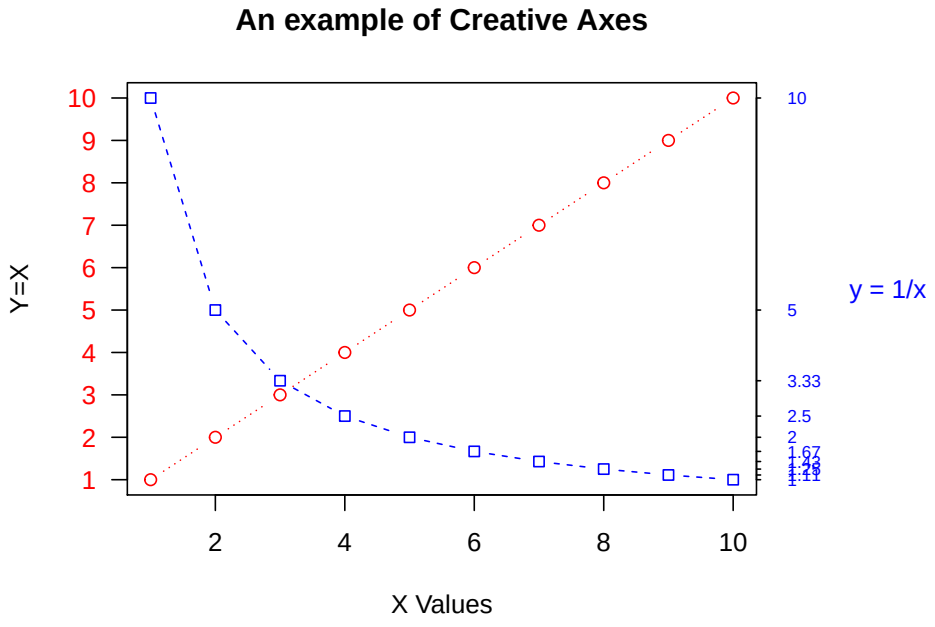
除了函数自带的关于坐标轴的设定，我们有时还需要自定义坐标轴，此时，我们需要用到 *axis( )* 函数。

其中主要参数如下：

- *side*：一个整数，表示在图形的哪边绘制坐标轴（1= 下，2= 左，3 = 上，4= 右）。
- *at*：一个数值型向量，表示需要绘制刻度线的位置。
- *labels*：一个字符型向量，表示置于刻度线旁边的文字标签（如果为 null, 则将直接使用 at 中的值）。
- *pos*：坐标轴线绘制位置的坐标（即与另一条坐标轴相交位置的值）。
- *lty*：线条类型。
- *col*：线条和刻度线颜色。
- *las*：标签是否平行于（=0）或垂直于（=2）坐标轴。

```
x <- c(1:10)
y <- x
z <- 10/x
```

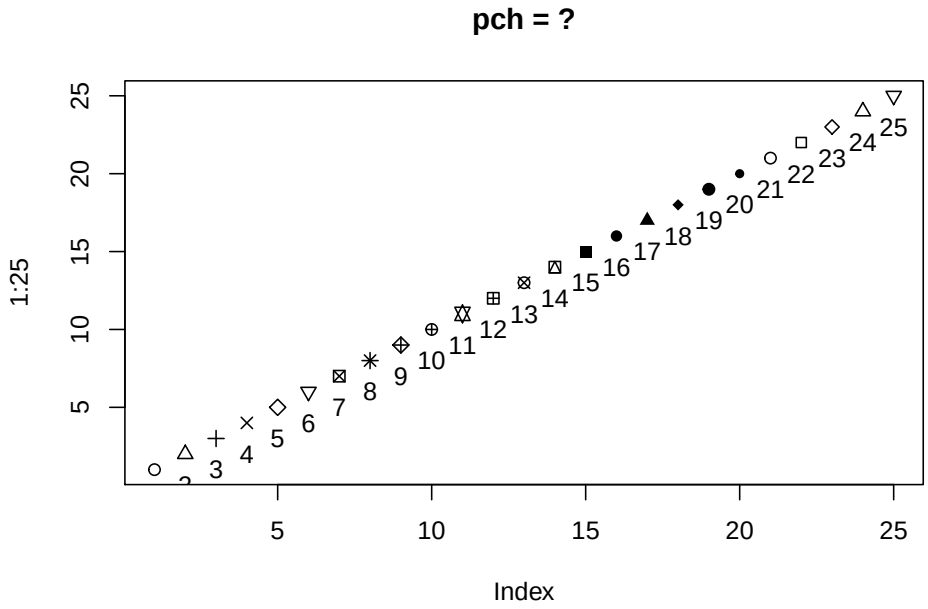
```
opar <- par(no.readonly = TRUE)
par(mar = c(5, 4, 4, 8)+0.1) # 增加边界大小
plot(x, y, type = "b", pch = 21,col = "red", yaxt = "n", lty = 3,
     ann =  FALSE)# 绘制 x 对 Y 的图形
lines(x, z, type = "b", pch = 22, col = "blue", lty = 2)# 添加 x 对 1/x 的直线
#plot 函数优先级比较高, lines 次之
axis(2, at = x, labels = x, col.axis = "red", las = 2) # 绘制自己的坐标轴
#"2" 图形的左边绘制坐标轴,"at=x" 需要绘制刻度线的位置在 x 轴,
#"labels" 置于刻度线旁边的文字标签,"las=2" 标签垂直于坐标轴
axis(4, at = z, labels = round(z,digits = 2),
     col.axis = "blue", las = 2, cex.axis = 0.7, tck = -.01)
# 添加标题和文本
mtext("y = 1/x", side = 4, line = 3,cex.lab = 1,las = 2,col = "blue")
title("An example of Creative Axes", xlab = "X Values", ylab = "Y=X")
```




```
par(opar)
```

3.5 学以致用' 图形参数' 这部分通过 base 包展示，展示所有的 pch，lty，col 参数。因为老是查 cheatsheet 太麻烦了。

```
plot(1:25, pch = 1:25, main = "pch = ?")
text(x = 1:25, y = 1:25 - 2, labels = 1:25)
```

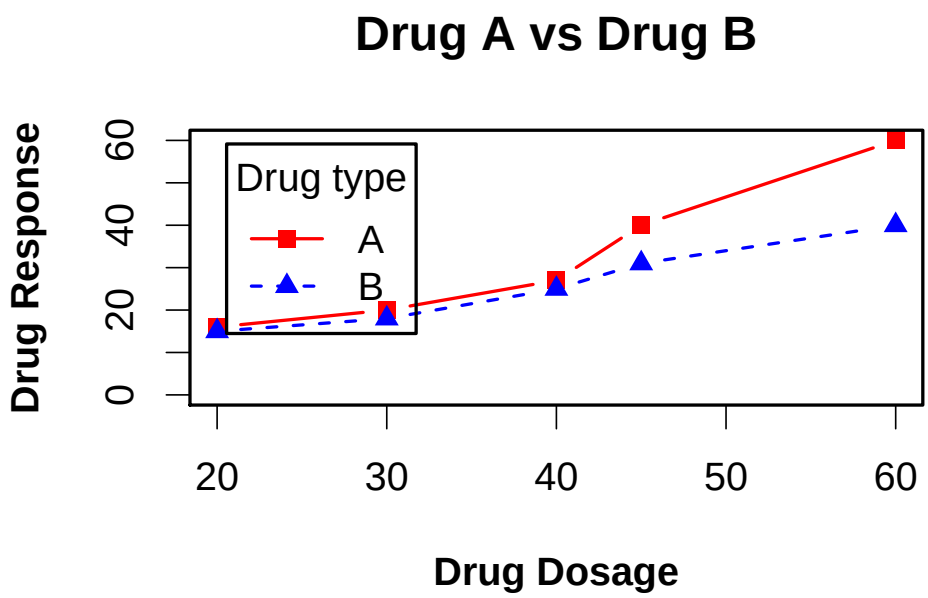



```
plot(1:6, type = "n", axes = FALSE)
abline(h = 1:6, lty = 1:6)
#?text
text(x = 3, y = 1:6, labels = paste0("lty=", 1:6))
```

3.6 添加图例

除了上述参数外，我们有时还需要使用 *legend( )* 函数添加图例，对图形进行说明。请看如下实例：

```
opar <- par(no.readonly = TRUE)
par(lwd = 2, cex = 1.5, font.lab = 2)# 线条宽度为 2, 放大 1.5 倍, 标签字体为粗体
# 绘制图形
plot(dose, drugA, type = "b",
pch = 15, lty = 1, col = "red", ylim = c(0, 60),
main= "Drug A vs Drug B",
xlab= "Drug Dosage", ylab = "Drug Response")

lines(dose, drugB, type = "b",
      pch = 17, lty = 2, col = "blue")

# 添加图例
legend("topleft", inset = .05, title = "Drug type",
       c("A", "B"), lty = c(1,2), pch = c(15,17), col = c("red","blue"))
```

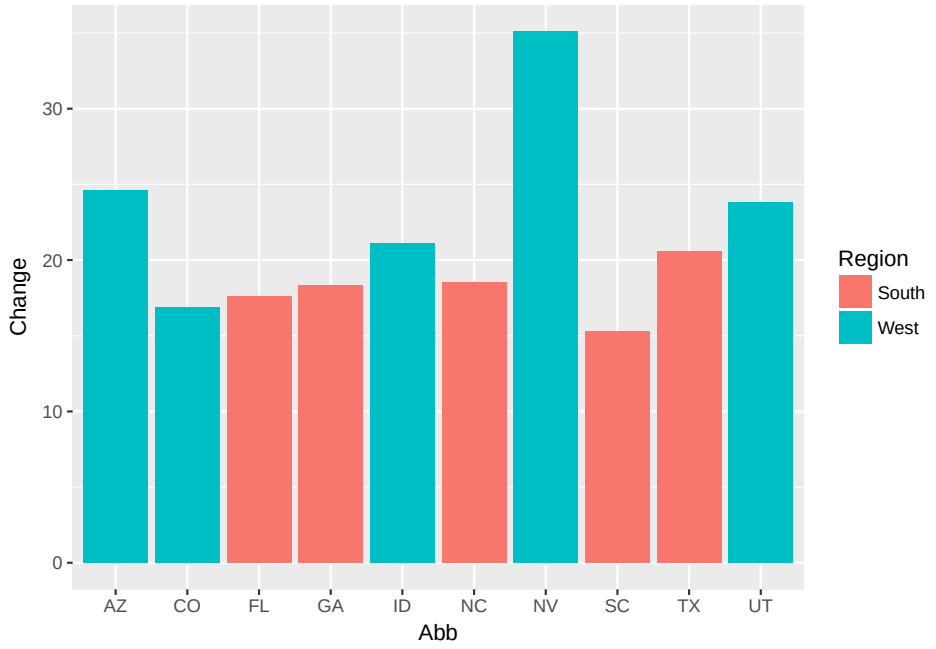


3.7 图形参数调整实例

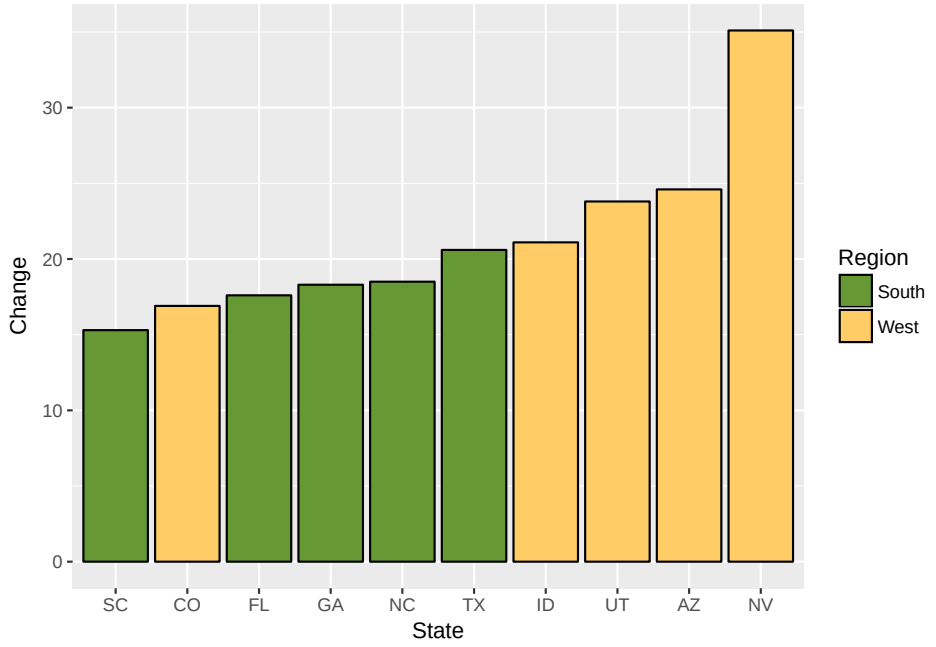
```
##install.packages("gcookbook")
library("gcookbook")
upc <- subset(uspopchange, rank(Change)>40)
upc
```

```
##             State Abb Region Change
## 3         Arizona  AZ   West   24.6
## 6        Colorado  CO   West   16.9
## 10        Florida  FL  South   17.6
## 11        Georgia  GA  South   18.3
## 13          Idaho  ID   West   21.1
## 29         Nevada  NV   West   35.1
## 34 North Carolina  NC  South   18.5
## 41 South Carolina  SC  South   15.3
## 44          Texas  TX  South   20.6
## 45           Utah  UT   West   23.8
```

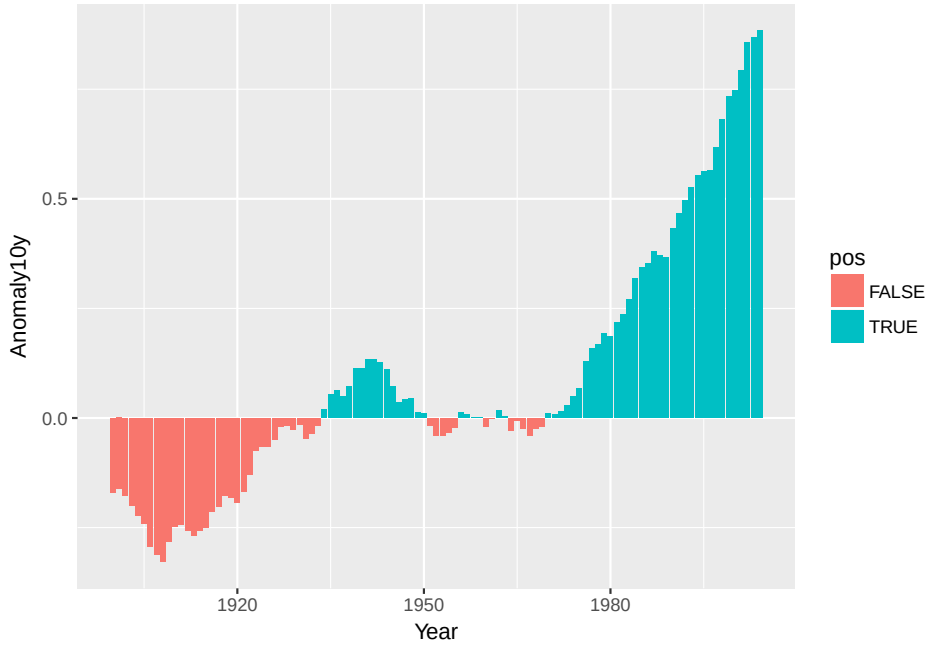
```
ggplot(upc, aes(x=Abb, y=Change, fill = Region))+
        geom_bar(stat = "identity")
```



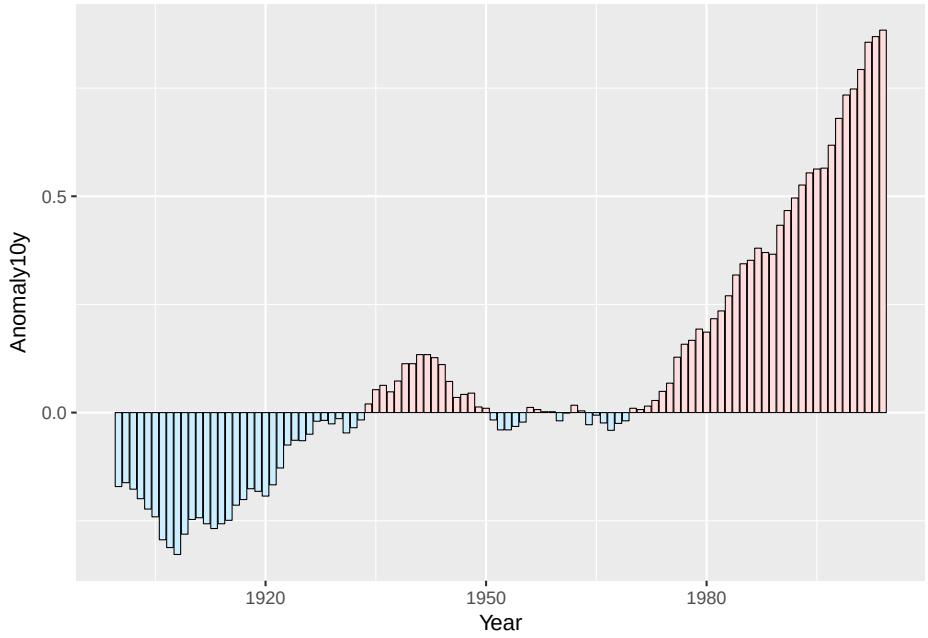

```
# 将分类变量映射给 fill 参数
ggplot(upc, aes(x = reorder(Abb, Change), y = Change, fill = Region)) +
        geom_bar(stat = "identity", colour = "black") +
        scale_fill_manual(values = c("#669933", "#FFCC66")) +
        xlab("State")
```




```
# 颜色的映射设定是在 aes() 内部完成的，而颜色的重新设定是在 aes() 外部完成的
# 根据条形图的高进行排序比按照字母排序对分类变量排序更有意义
# 对正负条形图分别着色

csub <- subset(climate, Source=="Berkeley"&Year>=1900)
csub$pos <- csub$Anomaly10y>=0
#csub
ggplot(csub, aes(x = Year, y = Anomaly10y, fill = pos))+
        geom_bar(stat = "identity", position = "identity")
```




```
# 我们可以通过 scale_fill_manual() 参数对图形颜色进行调整,
# 设定参数 guide=FALSE 可以删除图例。同时，我们通过设定边框颜色（colour）
# 和边框线宽度（size）为图形添加一个细黑色边框。其中,
# 边框线宽度（size）是用来控制边框线宽度的参数，单位是毫米
ggplot(csub, aes(x = Year, y = Anomaly10y, fill = pos)) +
geom_bar(stat = "identity", position = "identity", colour = "black",size = 0.25) +
        scale_fill_manual(values = c("#CCEEFF", "#FFDDDD"), guide = FALSE)
```



4 拓展

在课程结束前，我们来看一个很多人都关注的图：K 线图。

```
library("quantmod") #install.packages("quantmod")
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: TTR
```

```
## Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
getSymbols("BABA", src = "yahoo", from = "2015-1-1", to = Sys.Date())
```

```
##     As of 0.4-0, 'getSymbols' uses env=parent.frame() and
##  auto.assign=TRUE by default.
##
## This behavior will be phased out in 0.5-0 when the call will
## default to use auto.assign=FALSE. getOption("getSymbols.env") and
## getOptions("getSymbols.auto.assign") are now checked for alternate defaults
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for more details.
```

```
## [1] "BABA"
```

```
chartSeries(BABA, name = 'baba', TA = NULL, up.col = 'red', dn.col = 'green')
```


```
# 走势上看  在 2015 年 11 月和 2016 年 2 月出现二次探底，估值得到支撑，走出反转走势，
# 根据波浪理论，目前股价处于第五浪
#View(BABA)
```

```
# 显示成交量的 K 线图
chartSeries(BABA, up.col = 'red', dn.col = 'green')
```


```
# 第一根巨量在第一次探底回升后的高点，多空分歧增发，使得成交量巨大
# 第二根巨量出现在横盘调整期间，股价在调整器没有回调，看出多方继续做多的决心
# 第三根巨量出现在向上跳空缺口的位置，那是多方发起攻势的新起点
# 周线图
BABAW <- to.weekly(BABA)
chartSeries(BABAW, name = 'baba', TA = NULL, up.col = 'red', dn.col = 'green')
```


baba
[2015−01−02/2017−03−16]
110
100
90
80
70
60
Jan 02
2015
May 01
2015
Sep 04
2015
Jan 08
2016
May 06
2016
Sep 02
2016
Jan 06
2017