以下函数只为满足常用的若干作图需求。

基本作图：  
plot(x)、plot(x, y)    #散点图，最多两个变量  
    #可使用参数type生成不同的效果图。常用‘l’、‘o’、‘h’，分别为折线图，点线图，垂线图。  
    #'s'和'S'是折线图，前者是先水平后垂直，后者是先垂直后水平；'n'是不显示，用于画空白图  
    #若需对三个以上变量两两作图，可先合并在一个数据框，再对数据框使用plot  
    #例：r=data.frame(x,y,z); plot(r);  
boxplot(x)、boxplot(x,y,z,...)    #箱式图，可使用多个参数将多个箱式图做在一起  
coplot(x~y|z)    #在z的每个值或每个区间上做x与y的散点图  
pairs(x)    #当x为矩阵时，做x各列之间的散点图  
hist(x, freq)    #直方图。参数freq默认为TRUE，根据频数作图；若为FALSE，则据构成比（总和为1）作图。  
barplot(table(x,y), beside=FALSE)    #对定性变量x、y做条图，默认堆积条图，使用beside=T则为并列条图。  
qqnorm(x)    #QQ图（正态分位数-分位数图）  
qqplot(x,y)    #y对x的分位数-分位数图  
pie(table(x))    #对定性变量x做饼图  
arrows    #在两点之间画箭头线，箭头与线段之间的夹角可调  
segments    #在两点之间画线段

加参考线：    #低级绘图参数，直接在原图上修改  
lines(x)、lines(x, y)    #添加折线  
abline(lm(y~x))    #添加y对x的回归直线  
abline(a,b)    #a为截距，b为斜率  
abline(v=)    #添加垂直线  
abline(h=)    #添加水平线

添加点：  
points(x,y)    #低级绘图参数，直接在原图上修改

画曲线：   
1、curve(expr, from=0, to=1, n=101, add=FALSE)      
    例：curve(qnorm); curve(log); curve(x^2)    #第一个参数可以是函数，也可以是含x的表达式。   
    例：curve(x^2, 0, 100);    #from和to规定表达式或函数的自变量范围，默认0～1。   
    n：为自变量范围内取多少个点进行描图，n越大，曲线越平滑   
    add：当取值为TURE时，该曲线添加于前一图上。当两图的自变量取值范围不重合时，无法叠加。     
2、plot(fun, from=0, to=1)   
    用法和curve相似，但第一个参数不能使用含x的表达式。 

    注意在plot中尽量避免使用add参数。

#######################################################################

使用参数pch设定点的样式  
1、数字0～20，表示21种不同的符号  
    curve(x/20,0,21,lty=0); for (i in 0:20) {points(i,0.5,pch=i);}  
2、8个字符，表示8种点的样式  
    curve(x/7,0,7,lty=0); chr=c('\*','.','o','O','0','+','-','|');  
    for (i in 0:7) {points(i,0.5,pch=chr[i+1])}  
3、数字21～25，表示可以填充背景色的五种符号，需和参数bg联用  
    curve(x/4,0,4,lty=0); for (i in 0:4) {points(i,0.5,pch=21+i,bg=sample(colors(),1))}

使用参数col设定颜色  
1、plot(dnorm, col='red')  
    使用sort(colors())可以查看所有已命名的颜色  
2、使用gray(x)生成灰度值，x为0~1之间的值，0表示黑色。另，gray也可写作grey。  
    curve(x\*1,0,1,lty=0);  
    points(seq(0,1,0.01),rep(0.5,101),pch=20,col=gray(seq(0,1,0.01)));  
3、使用hsv(h,s,v)生成所需的颜色，h是色调，s是饱和度，v是亮度  
    curve(x\*1,0,1,lty=0);  
    points(seq(0,1,0.01),rep(0.5,101),pch=20,col=hsv(h=seq(0,1,0.01),s=1,v=1));  
4、利用hsv生成双色渐变图，用h设定所用颜色，s设定变化程度  
    curve(x\*1,1,101,lty=0);  
    x=seq(0,1,0.01);   
    x1=which(x < 0.5); x2=which(x >= 0.5);    #双色渐变要对数值二分类才能产生预期效果  
    points(x1,rep(50,length(x1)),pch=20,col=hsv(h=0.65,s=(0.5-x[x1])/0.5,v=1));  
    points(x2,rep(50,length(x2)),pch=20,col=hsv(h=0,s=(x[x2]-0.5)/0.5,v=1));

使用参数lty设定线型  
1、简单取值可以使用0～6  
    curve(x\*8,0,1,lty=0); for (i in 1:16) {abline(h=i/2,lty=i);}  
    #0表示空白，随取值增大，6种线型循环出现  
2、使用十六进制数字组成的字符串  
    长度只能是偶数位，最长8位；奇数位为表示实线长度，偶数位表示空白长度  
    注意：需要引号；不能有0。  
    curve(x^2,0,1,lty='32';)    #3单位实线和2单位空白循环  
    curve(x^2,0,1,lty='32AA';)    #3单位实线、2单位空白、10单位实线、10单位空白循环

使用参数bty设定边框  
    bty='o'    #默认显示四条边框  
    bty='l'    #不显示右上两条边框，例：curve(x^2,0,1,bty='l')  
    bty='n'    #不显示边框

使用参数xlab和ylab修改坐标轴的意义  
    curve(x\*5,0,1,xlab='',ylab='',lty=0)    #不显示坐标轴的符号

使用xlim和ylim限定坐标轴的范围  
    plot(rnorm,xlim=c(0,100),ylim=c(0,1000),lty=0)

使用title加图的标题  
    title(main = NULL, sub = NULL, xlab = NULL, ylab = NULL)  
    #main是写在图上方的标题，sub是写在图下方的标题

更多参数请查看帮助：?par

添加图例  
1、图例在作图区域内  
     legend(x,y,legend,pch,col,lty,ncol=1,bty='o')  
          #x,y为图例左上角的坐标，也可以使用以下字符来标记特殊位置："bottomright", "bottom", "bottomleft", "left", "topleft", "top", "topright", "right" and "center"  
          #legend是图例的文字，一般是一个字符向量  
          #pch,col,lty是图中所用的样式  
          #ncol是图例的列数，bty指明图例的边框显示  
    例：  
    curve(x\*8,0,1,lty=0); for (i in 1:6) {abline(h=i/2,lty=i);}  
    legend(0,8,legend=letters[c(1:6)],lty=1:6,bty='n');  
2、图例在作图区域外  
    需使用par修改参数mar和pty。同时还需要修改参数xpd。  
    默认mar=c(5,4,4,2)+0.1，即图形下左上右四个边界的宽度分别为5.1,4.1,4.1,2.1厘米。  
    默认pty='m'，即最大化作图区域，取值为's'则限制作图区域为方形。  
    默认xpd=FALSE，即不允许在作图区域外作图，改为TRUE即可。  
    例：  
    op=par();     #保存par的原值  
    par(mar=c(5,4,4,5),pty='s');    #准备在图形右边添加图例  
    curve(x\*8,0,1,lty=0); for (i in 1:16) {abline(h=i/2,lty=i);}  
    legend(1.02,4,legend=letters[c(1:6)],lty=1:6,bty='n',xpd=T);  
    par(op);    #恢复par的原值

在任意位置添加文字  
    curve(x\*1,0,1,lty=0)    #做一张空白图  
    text(0.5,0.5,'中心')    #在(0.5,0.5)标上‘中心’  
    text(locator(1),'任意')    #在鼠标点击的位置标上‘任意’  
        locator(n)    #返回鼠标点击n次的坐标  
    text(locator(1),'任意',cex=2)    #利用cex参数将字号放大一倍  
    text(locator(1),expression(sqrt(x)));    #标示公式，不能使用font和vfont修改字体  
    text(locator(1),'x',vfont=c('serif','italic'))    #设定字体  
        #字体设定参看：?Hershey。vfont的值为(typeface, fontindex)。  
    注：text只能在作图区域内添加，而利用mtext可以在外面添加

做一个空白图（什么都没有）  
    plot(rnorm,lty=0,bty='n',xaxt='n',yaxt='n',xlab='',ylab='',xlim=c(-1,1),ylim=c(-1,1));

################################################################################

曲线拟合：（线性回归方法：lm）  
1、x排序  
2、求线性回归方程并赋予一个新变量  
    z=lm(y~x+I(x^2)+...)  
3、plot(x,y)    #做y对x的散点图  
4、lines(x,fitted(z))    #添加拟合值对x的散点图并连线

曲线拟合：（曲线回归，nls）  
lm是将曲线直线化再做回归，nls是直接拟合曲线。  
需要三个条件：曲线方程、数据位置、系数的估计值。  
如果曲线方程比较复杂，可以先命名一个自定义函数。  
例：  
    f=function(x1, x2, a, b) {a+x1+x2^b};  
    result=nls(x$y~f(x$x1, x$x2, a, b), data=x, start=list(a=1, b=2));  
        #x可以是数据框或列表，但不能是矩阵  
        #对系数的估计要尽量接近真实值，如果相差太远会报错：“奇异梯度”  
    summary(result);    #结果包含对系数的估计和p值  
根据估计的系数直接在散点图上使用lines加曲线即可。

曲线拟合：（局部回归）  
lowess(x, y=NULL, f = 2/3, iter = 3)  
    #可以只包含x，也可使用x、y两个变量  
    #f为窗宽参数，越大越平滑  
    #iter为迭代次数，越大计算越慢  
loess(y~x, data, span=0.75, degree=2)  
    #data为包含x、y的数据集；span为窗宽参数  
    #degree默认为二次回归  
    #该方法计算1000个数据点约占10M内存  
举例：  
x=seq(0, 10, 0.1); y=sin(x)+rnorm(101)    #x的值必须排序  
plot(x,y);    #做散点图  
lines(lowess(x,y));    #利用lowess做回归曲线  
lines(x,predict(loess(y~x)));    #利用loess做回归曲线，predict是取回归预测值  
z=loess(y~x); lines(x, z$fit);    #利用loess做回归曲线的另一种做法

核密度估计曲线  
1、hist(x, freq=FALSE)    #根据构成比做直方图  
2、核密度估计  
density(x,  
    bw='nrd0',    #设置窗宽，默认为‘nrd0’（只是为了兼容，并不是推荐数值），可尝试不同数字选择最合适的。  
    kernel = c("gaussian", "epanechnikov", "rectangular", "triangular", "biweight", "cosine", "optcosine"),   
    #选择列表中的一种计算方法，默认为第一种。方法名称可使用首字母代替。  
    weights)    #给不同的x值赋予权重，长度和x相同。默认权重相同。  
例：a=density(x,bw=0.5, kernel='c');  
3、lines(a)    #添加核密度曲线

###################################################################

切割作图区域  
1、layout  
    #将作图区域分为四份（2\*2），上面两个格子用于图1，右下角用于图2，左下角不作图。  
    layout(matrix(c(1,1,0,2), 2, 2, byrow = TRUE));  
    curve(1\*x);  
    curve(x^2);

2、par(mfcol, mfrow)  
    #将作图区域分为2\*3，按列逐个画图  
    par(mfcol=c(2,3))  
    for (i in 1:6) curve(x^i);  
    #将作图区域分为2\*3，按行逐个画图  
    par(mfrow=c(2,3))  
    for (i in 1:6) curve(x^i);

#############################################################

绘制多边形  
    curve(x\*1,0,1,lty=0);  
    polygon(c(0.2,0.8,0.7,0.1),c(0.3,0.5,0.8,0.7));  
    polygon(c(0.2,0.8,0.7,0.1),c(0.3,0.5,0.8,0.7),col='blue');    #颜色填充  
    polygon(c(0.2,0.8,0.7,0.1),c(0.3,0.5,0.8,0.7),density=10);    #斜线填充

制作一个太极图  
    par(pty='s');  
    ##########################################  
    #中间线  
    zjx=function() {  
        pz=NULL;  
        temp=-1\*sqrt(1-seq(0,-1,length=100)^2)-1;  
        pz=rbind(pz,cbind(seq(0,-1,length=100),temp));  
        temp=sqrt(1-seq(-1,0,length=100)^2)-1;  
        pz=rbind(pz,cbind(seq(-1,0,length=100),temp));  
        temp=-1\*sqrt(1-seq(0,1,length=100)^2)+1;  
        pz=rbind(pz,cbind(seq(0,1,length=100),temp));  
        temp=sqrt(1-seq(1,0,length=100)^2)+1;  
        pz=rbind(pz,cbind(seq(1,0,length=100),temp));  
        return(pz);  
    }  
    #######################################  
    #画布  
    plot(rnorm,lty=0,bty='n',xaxt='n',yaxt='n',  
         xlab='',ylab='',xlim=c(-2,2),ylim=c(-2,2));   
      
    #左图  
    pz=NULL;  
    temp=sqrt(4-seq(0,-2,length=100)^2);  
    pz=rbind(pz,cbind(seq(0,-2,length=100),temp));  
    temp=-1\*sqrt(4-seq(-2,0,length=100)^2);  
    pz=rbind(pz,cbind(seq(-2,0,length=100),temp));  
    temp=-1\*sqrt(1-seq(0,-1,length=100)^2)-1;  
    pz=rbind(pz,zjx());  
      
    polygon(pz,col='black');  
    points(0,1,pch=21,bg='white',cex=10);  
      
    #右图  
    pz=NULL;  
    temp=sqrt(4-seq(0,2,length=100)^2);  
    pz=rbind(pz,cbind(seq(0,2,length=100),temp));  
    temp=-1\*sqrt(4-seq(2,0,length=100)^2);  
    pz=rbind(pz,cbind(seq(2,0,length=100),temp));  
    pz=rbind(pz,zjx());  
      
    polygon(pz,col='white');  
    points(0,-1,pch=21,bg='black',cex=10);