



《第三章 R语言绘图》

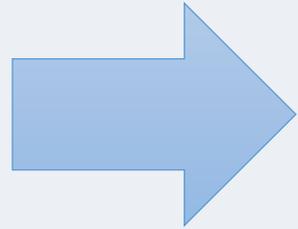
第三章 R语言绘图

实验4 R语言的绘图

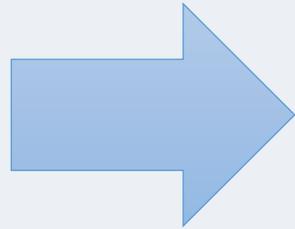


《第三章 R语言绘图》

目录



4.1 实验目的



4.2 实验过程



《第三章 R语言绘图》

4.1 实验目的

- 1. 掌握R高级水平作图的方法；
- 2. 掌握R低级水平作图的方法；
- 3. 掌握R交互作图的方法。



《第三章 R语言绘图》

4.2 实验过程

1. 高级水平绘图

- 基本绘图函数(plot())

在基本的绘图函数中，最常用的函数是plot()函数，它可以绘制数据的散点图、曲线图等，用途非常广泛，其使用格式为

```
plot(x, y, ...)
```

参数x和y分别表示所绘图形横坐标和纵坐标构成的对象。“...”为附加参数。



《第三章 R语言绘图》

函数的默认使用格式为

```
plot(x, y = NULL, type = "p", xlim = NULL, ylim = NULL, log = "",  
main = NULL, sub = NULL, xlab = NULL, ylab = NULL, ann = par("ann"),  
axes = TRUE, frame.plot = axes, panel.first = NULL, panel.last = NULL,  
asp = NA, ...)
```

其中，各参数的取值及意义如下：

(1) 图形类型

参数type为所绘图形的类型，其取值和类型如下：



《第三章 R语言绘图》

- 'p' : 绘点(默认值) ;
- 'l' : 画线 ;
- 'b' : 同时画点和线 , 但点线不相交 ;
- 'c' : 将type = 'b'中的点去掉 , 只剩下相应的线条部分 ;
- 'o' : 同时画点和线 , 且相互重叠 , 这是它与type = 'b'的区别 ;
- 'h' : 画铅垂线 ;
- 's' : 画阶梯线 , 从一点到下一点时 , 先画水平线 , 再画垂直线 ;
- 'S' : 也是画阶梯线 , 但从一点到下一点是先画垂直线 , 再画水平线 ;
- 'n' : 作一幅空图 , 没有任何内容 , 但坐标轴、标题等其它元素都照样显示。



《第三章 R语言绘图》

(2) 图形范围

用参数xlim和ylim来控制图形的范围，log是对数据取对数，其取值及意义如下：

- xlim : 二维向量，表示所绘图形x轴的范围；
- ylim : 二维向量，表示所绘图形y轴的范围；
- log : "x"或"y",表示对x轴或对y轴的数据取对数；"xy"或"yx",表示对x轴与对y轴的数据同时取对数。



《第三章 R语言绘图》

(3) 图题及坐标轴

可通过设置下列参数来补充图形的说明，如图的标题、副标题、x轴和y轴的说明等。

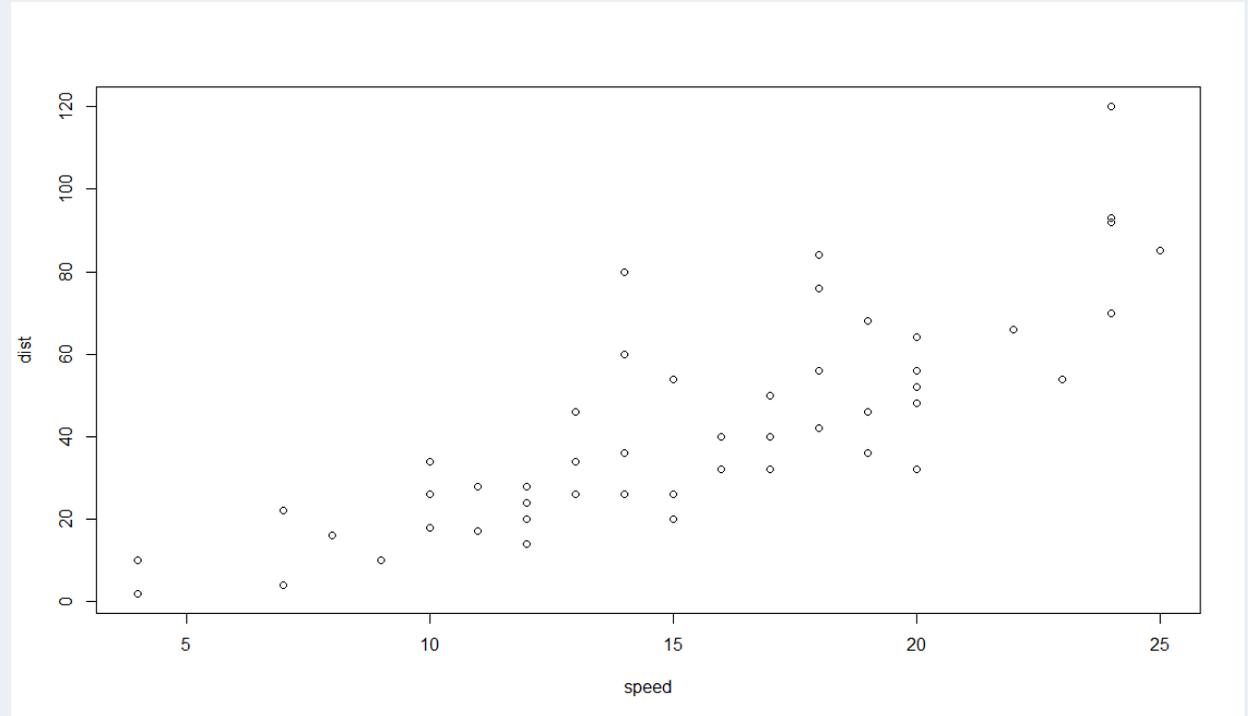
- main : 字符串，描述图形的标题；
- sub : 字符串，描述图形的副标题；
- xlab : 字符串，描述x轴的标签，默认值为对象名；
- ylab : 字符串，描述y轴的标签，默认值为对象名。



《第三章 R语言绘图》

例4.1：cars数据集是R自带的数据库，包括50组数据，两个变量：speed和dist，用plot()画出这两个变量的散点图。

```
data(cars) # 读取数据集到工作空间  
attach(cars) # 将数据集添加到搜索路径  
plot(speed,dist)  
detach() # 用完数据集后，将其从搜索  
路径中移除
```





《第三章 R语言绘图》

● 绘制多组图(pairs())

plot()可以绘制出两个变量的散点图或是曲线图，但是当统计数据中包含有多个变量时，想要查看变量之间的关系时，需要把任意两个变量的散点图在同一窗口中展示，这时可以使用pair()函数，其作用是绘制多个变量的散点图，以阵列的方式排列。

pair()的默认调用格式为：

```
pairs(x, labels, panel = points, ...,horInd = 1:nc, verInd = 1:nc,
```

```
lower.panel = panel, upper.panel = panel,diag.panel = NULL, text.panel  
= textPanel,label.pos = 0.5 + has.diag/3, line.main = 3,cex.labels = NULL,  
font.labels = 1,row1atop = TRUE, gap = 1, log = "")
```



《第三章 R语言绘图》

- `x` : 向量、矩阵或数据框，对应数据的坐标；
- `labels` : 字符串，变量名称，默认值为对象的名称；
- `panel` : 函数，绘制面板数据的方法，默认值为`points`；
- `horInd, verInd` : 在横轴与纵轴上变量的下标；
- `lower.panel` : 函数，表示阵列下三角部分图形的绘制方法，默认值为`points`；
- `upper.panel` : 函数，表示阵列上三角部分图形的绘制方法，默认值为`points`；
- `diag.panel` : 函数，表示阵列对角部分图形的绘制方法，默认值为`NULL`；
- `label.pos` : 函数，标签的位置；
- `cex.labels` : 数值，标签的大小，默认值为`NULL`；
- `font.labels` : 数值，标签字体的类型，默认值为1。



《第三章 R语言绘图》

pairs()也可以以公式的方式调用：

```
pairs(formula, data = NULL, ..., subset, na.action = stats::na.pass)
```

- formula : 形如 “~ x + y+ z” 的公式；
- data : 数据框；
- subset : 数值或字符串向量，表示数据的子集。



《第三章 R语言绘图》

例4.2：iris数据集是一个著名的数据集，许多的科研著作都在iris数据集上作分类的操作。该数据集是由3种不同类的鸢尾花共150条记录构成，包括五个变量：

Sepal.Length：花萼长度，单位：cm

Sepal.Width：花萼宽度，单位：cm

Petal.Length：花瓣长度，单位：cm

Petal.Width：花瓣宽度，单位：cm

R语言的基础包中自带了iris数据，用pairs()函数画出各变量的散点图。

《第三章 R语言绘图》

```
data(iris)
```

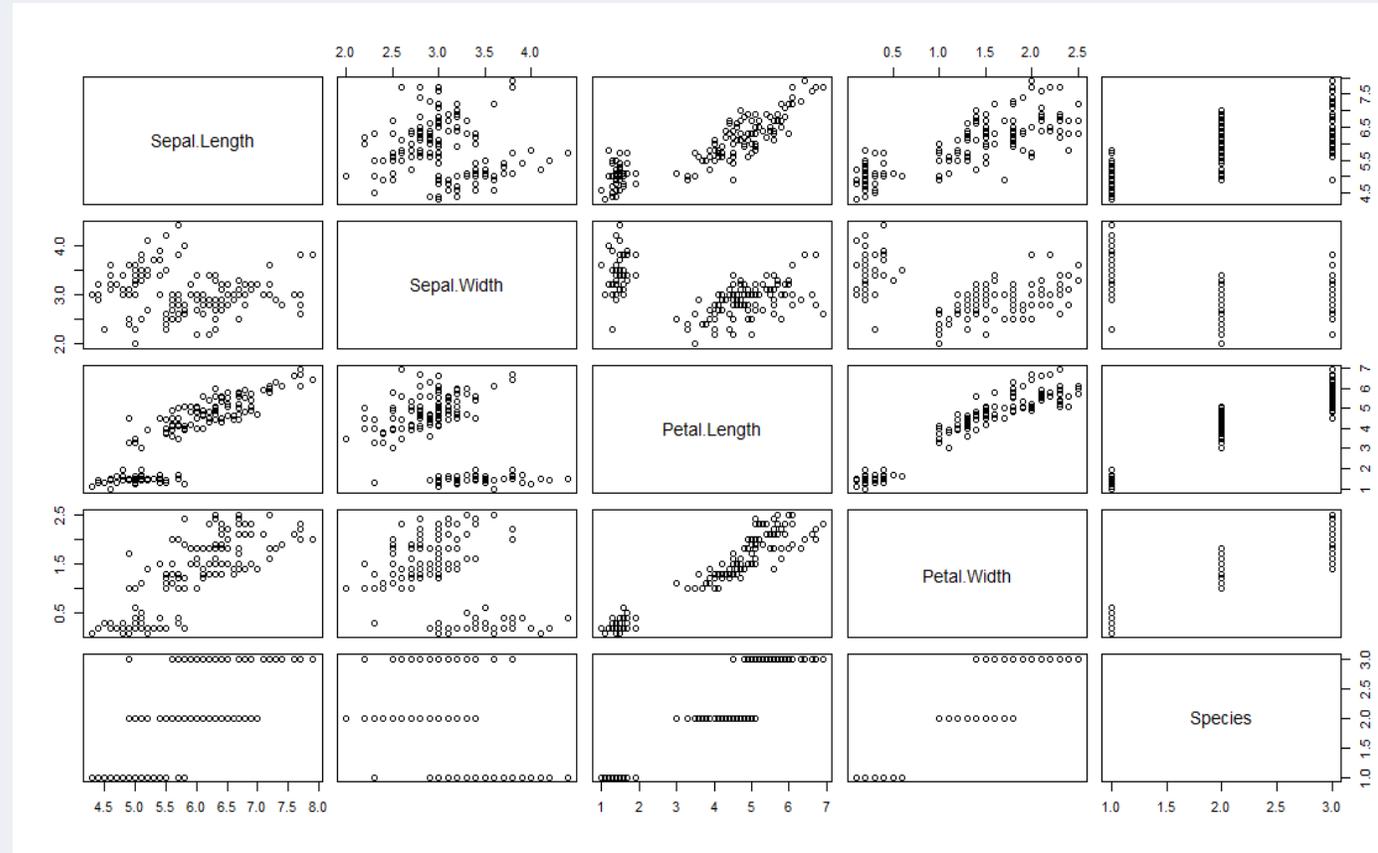
```
pairs(iris)
```

```
#第二种方法：使用公式形式的调用方法
```

```
pairs(~Sepal.Length + Sepal.Width +
```

```
Petal.Length + Petal.Width + Species,
```

```
data = iris)
```





《第三章 R语言绘图》

● 绘制点图(dotchart())

点图也叫克里夫兰点图，常用于检测数据的离群值。在R中可以使用dotchart()函数绘制克里夫兰点图，其调用格式为：

```
dotchart(x, labels = NULL, groups = NULL, gdata = NULL,  
         cex = par("cex"), pch = 21, gpch = 21, bg = par("bg"),  
         color = par("fg"), gcolor = par("fg"), lcolor = "gray",  
         xlim = range(x[is.finite(x)]),  
         main = NULL, xlab = NULL, ylab = NULL, ...)
```



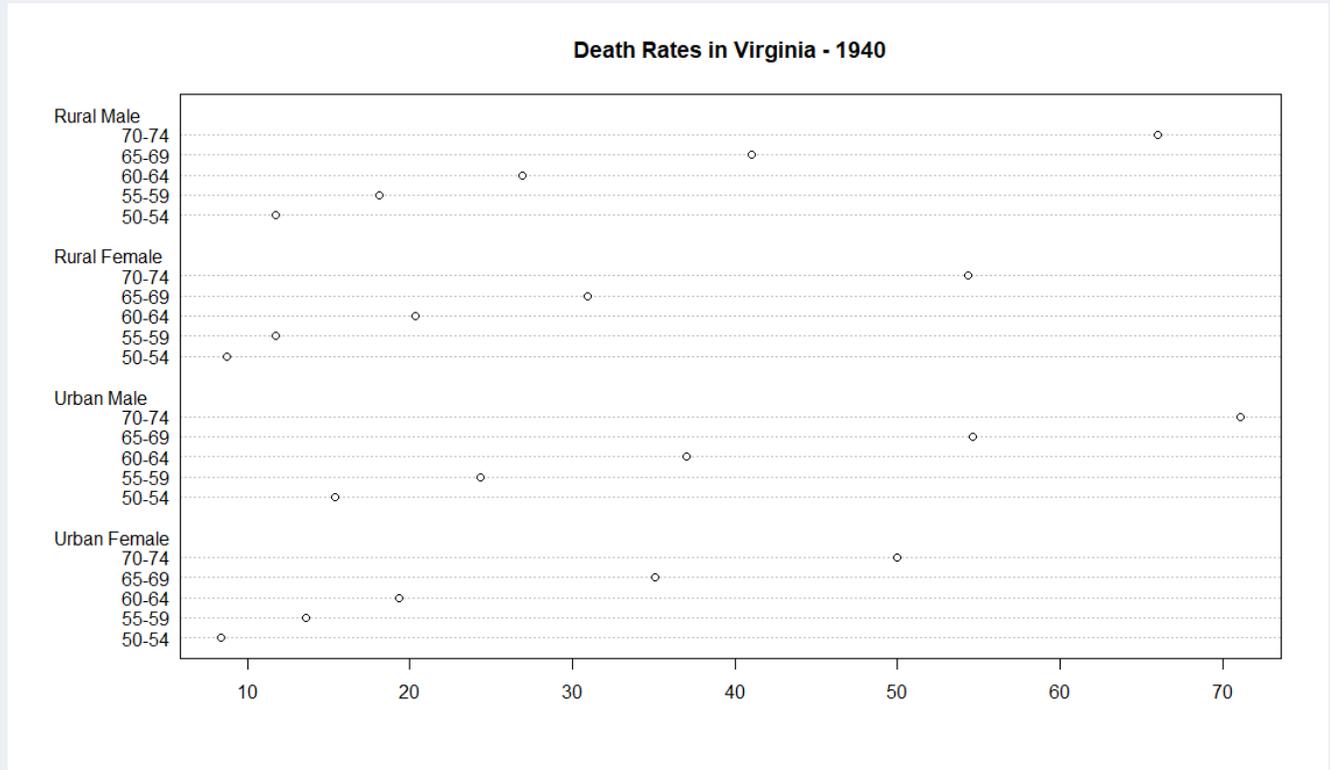
《第三章 R语言绘图》

- `x` : 向量或矩阵, 表示绘制克里夫兰点图的数据;
- `labels` : 向量, 表示每个点的标签, 当`x`为向量时, 默认值是 `'names(x)'`, 当`x`为矩阵时, 默认值是 `'dimnames(x)[1]'`
- `groups` : 因子向量, 表示`x`的分组情况, 当`x`为矩阵时, 默认值按`x`的列分组;
- `gdata` : 数值向量, `group`的数据值, 最具代表性的就是每个组的中位数或平均值。

《第三章 R语言绘图》

例4.4：R自带的数据集VADeaths保存的是美国弗吉尼亚州1940年的人口死亡率（1/1000），绘制出该数据集的克里夫兰点图。

```
data(VADeaths)
dotchart(VADeaths, main =
'Death Rates in Virginia - 1940')
```





《第三章 R语言绘图》

● 绘制饼图(pie())

饼图是一种常用的统计图形，用整个圆表示总数，圆内各个扇形的大小表示各部分数量占总数的百分数，可以用于描述数量、频率及百分比之间的关系。在R中，可以使用pie()函数绘制饼图，其调用格式为：

```
pie(x, labels = names(x), edges = 200, radius = 0.8,  
    clockwise = FALSE, init.angle = if(clockwise) 90 else 0,  
    density = NULL, angle = 45, col = NULL, border = NULL,  
    lty = NULL, main = NULL, ...)
```



《第三章 R语言绘图》

- `x` : 非负值构成的向量, `x`的分量表示扇形所占饼图的大小;
- `labels` : 表达式或字符串, 表示饼图中各扇形的名称, 默认值是 "`namex(x)`";
- `edged` : 正号数, 表示近似圆的多边形的边数;
- `radius` : 数值, 饼图的半径, 默认值为0.8;
- `clockwies` : 逻辑变量, 当取FALSE (默认) 时为逆时针, 取TRUE时为顺时针;
- `init.angle` : 数值, 表示饼图开始的角度。逆时针的默认值是0度 (3点钟的位置), 顺时针的默认值为90度 (12点钟的位置);
- `density` : 正整数, 表示阴影线条的密度, 即每英寸的线条的条数;
- `angle` : 数值或向量, 表示阴影线条倾斜的角度。

《第三章 R语言绘图》

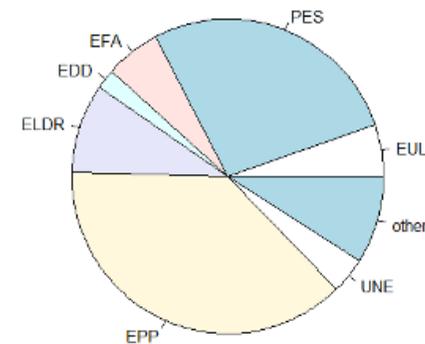
例4.5 : 2004年欧洲议会选举的初步结果数据如下 :

EUL	PES	EFA	EDD	ELDR	EPP	UNE	other
39	200	42	15	67	276	27	66

根据这组数据绘制出饼图。

```
pie.data <- c(39, 200, 42, 15, 67, 276, 27, 66)
names(pie.data) <- c("EUL", "PES", "EFA",
"EDD", "ELDR", "EPP", "UNE", "other")
pie(pie.data, radius = 0.9, main = "2004 年欧
洲议会选举结果")
```

2004 年欧洲议会选举结果





《第三章 R语言绘图》

● 绘制条形图(barplot())

条形图是一种以长方形的长度表示变量的统计表，它在统计图形中得到广泛应用。在R中，可以使用barplot()函数来绘制条形图。



《第三章 R语言绘图》

barplot()函数调用格式如下：

```
barplot(height, width = 1, space = NULL,  
        names.arg = NULL, legend.text = NULL, beside = FALSE,  
        horiz = FALSE, density = NULL, angle = 45,  
        col = NULL, border = par("fg"),  
        main = NULL, sub = NULL, xlab = NULL, ylab = NULL,  
        xlim = NULL, ylim = NULL, xpd = TRUE, log = "",  
        axes = TRUE, axisnames = TRUE,  
        cex.axis = par("cex.axis"), cex.names = par("cex.axis"),  
        inside = TRUE, plot = TRUE, axis.lty = 0, offset = 0,  
        add = FALSE, args.legend = NULL, ...)
```



《第三章 R语言绘图》

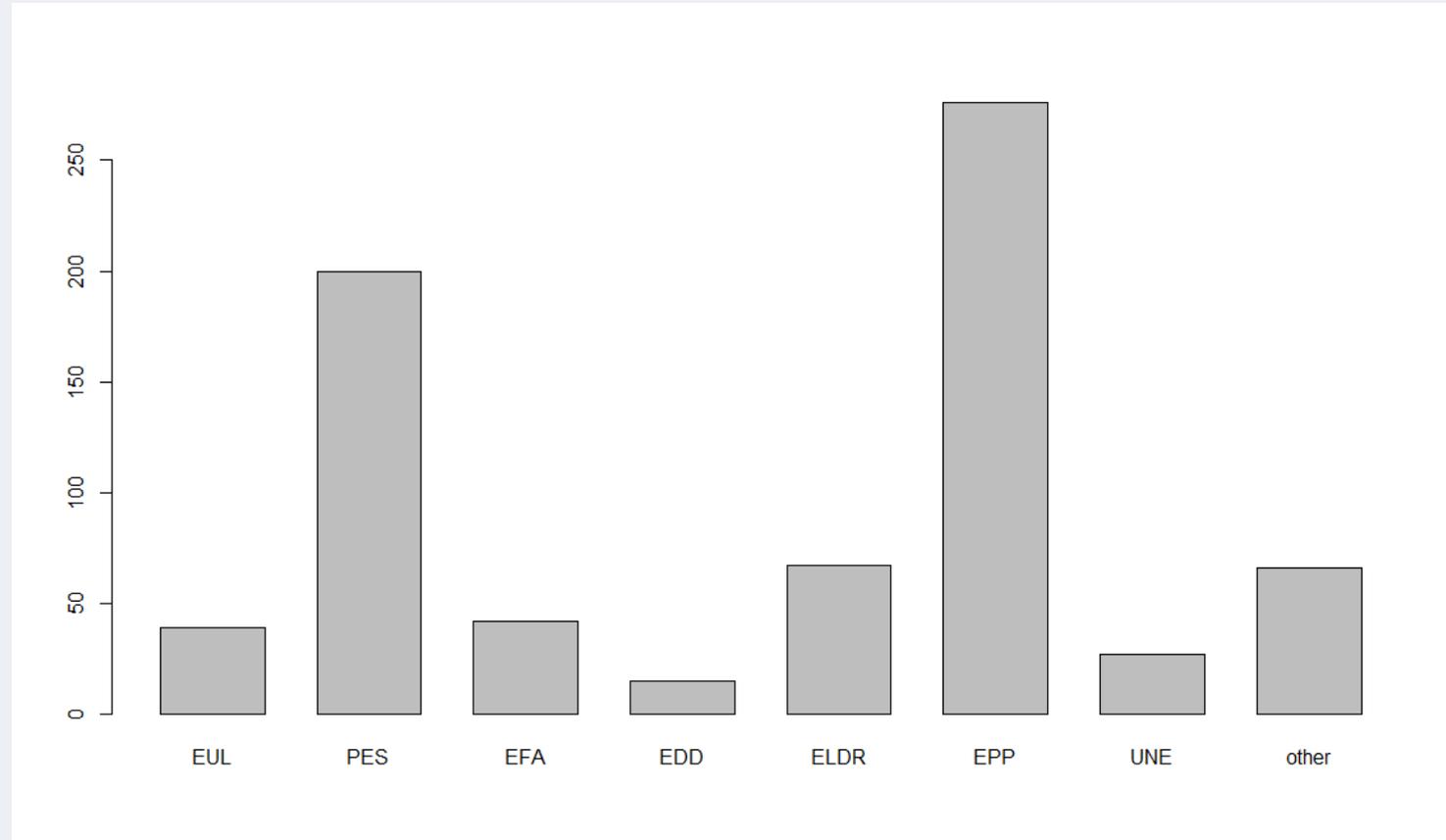
- height : 向量或矩阵, 表示条形的长度;
- width : 数值或向量, 表示条形的宽度, 默认值为1;
- space : 数值, 表示条形之间的空白的宽度, 默认值为NULL ;
- legend.text : 字符串, 表示图例说明;
- beside : 逻辑变量, FALSE (默认值) 表示重叠, TRUE表示平行排列;
- horiz : 逻辑变量, FALSE (默认值) 表示绘制竖条, TRUE表示绘制横条。



《第三章 R语言绘图》

例4.6：使用例4.5中的数据，绘制出条形图。

```
barplot(pie.data, space = 0.5)
```





《第三章 R语言绘图》

● 绘制直方图(hist())

直方图又称为质量分布图，由一系列高度不等的纵向条纹或线段表示数据分布的情况，是一种常用的统计报告图。直方图常被用于展示连续数据的分布，也常用于估计数据的概率分布。在R中，可以使用hist()函数来绘制直方图。

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/896225025110010211>