北京郵電大學
BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

Data Analytics with R
语言数据分析

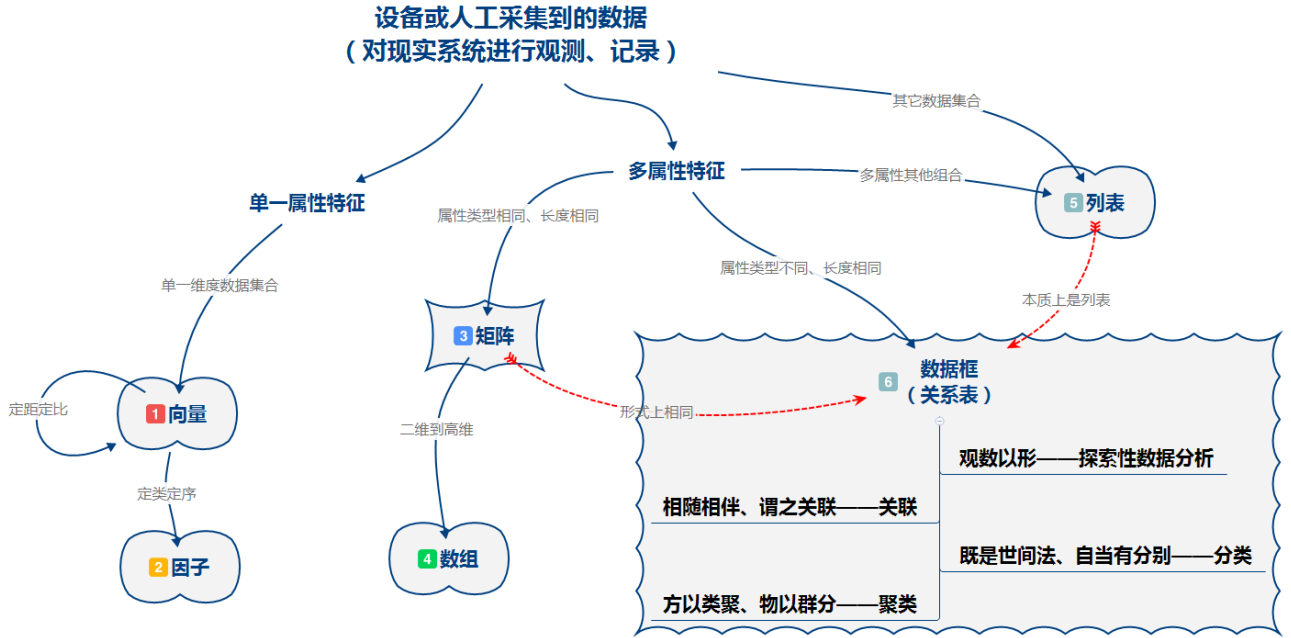
数据对象

艾新波 / 2018 • 北京

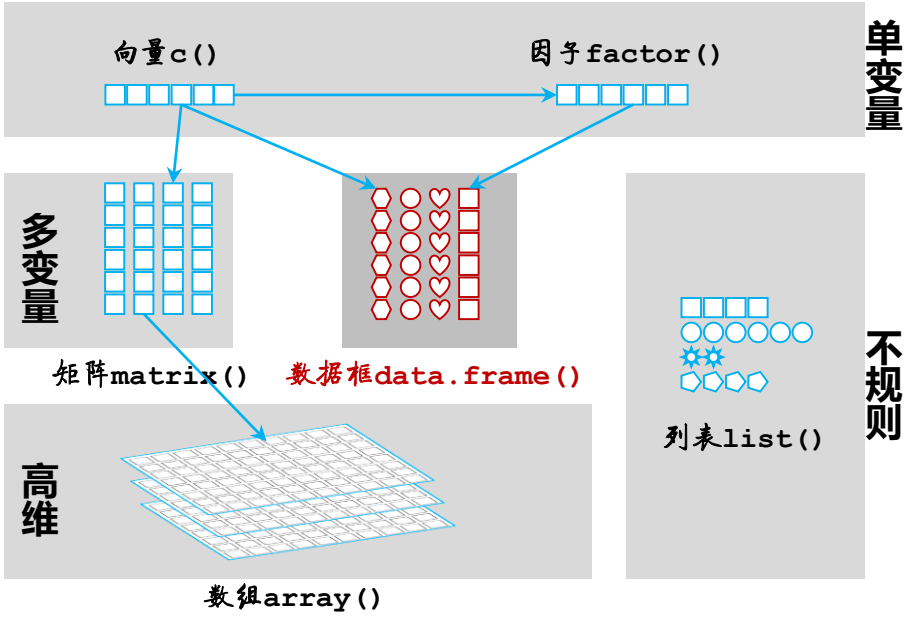
课程体系

- R语言数据分析
  - 上部：论道 
    - 第1章 气象万千、数以等观
    - 第2章 所谓学习、归类而已
    - 第3章 格言联璧话学习
    - 第4章 源于数学、归于工程
  - 中部：执具 
    - 第5章 工欲善其事必先利其器
    - 第6章 基础编程
    - 第7章 数据对象
  -
    - 第8章 人人都爱tidyverse
    - 第9章 最美不过数据框
  - 下部 博术 
    - 第10章 观数以形
    - 第11章 相随相伴、谓之关联
    - 第12章 既是世间法、自当有分别
    - 第13章 方以类聚、物以群分
    - 第14章 庐山烟雨浙江潮

数据对象


设备或人工采集到的数据
（对现实系统进行观测、记录）
单一属性特征
多属性特征
其它数据集合
多属性其他组合
5 列表
属性类型相同、长度相同
属性类型不同、长度相同
本质上是列表
单一维度数据集合
3 矩阵
1 向量
定距定比
定类定序
2 因子
二维到高维
4 数组
形式上相同
6 数据框
（关系表）
观数以形——探索性数据分析
相随相伴、谓之关联——关联
既是世间法、自当有分别——分类
方以类聚、物以群分——聚类

数据对象


向量c()
因子factor()
单变量
多变量
矩阵matrix()
数据框data.frame()
不规则
列表list()
高维
数组array()

数据的材质

序号	类型	名称	示例	备注
1	logical	逻辑型	TRUE, FALSE, T, F	※
2	integer	整型	1L, 300L	※
3	numeric	双精度型	1, 3.14	※
4	complex	复数型	1+2i	
5	character	字符型	‘Hello, world’	※
6	raw	原始字节数据	b0 ac d0 c2 b2 a8	

相关内容可参考帮助文档：`?atomic`

数据对象

当前位置

设备或人工采集到的数据
（对现实系统进行观测、记录）

其它数据集合

单一属性特征

多属性特征

多属性其他组合

5 列表

属性类型相同、长度相同

属性类型不同、长度相同

本质上是列表

单一维度数据集合

3 矩阵

定距定比

1 向量

形式上相同

6 数据框
（关系表）

二维到高维

定类定序

观数以形——探索性数据分析

相随相伴、谓之关联——关联

既是世间法、自当有分别——分类

2 因子

4 数组

方以类聚、物以群分——聚类

创建向量

```
#c()创建向量最常见的方式
#Combine Values into a Vector
#字符型向量
xm <- c("周黎", "汤海明", "舒江辉", "翁柯", "祁强", "湛容")
xb <- c("女", "男", "男", "女", "男","女")
#数值型向量
yw <- c(94, 87, 92, 91, 85, 92)
#逻辑型向量
xb2 <- c(F, T, TRUE, FALSE, T, F)
```

创建向量

```
my_pi <- c(3, ".", 1, 4, 1, 5, 9, 2, 6) #不能有混合类型
my_pi
#> [1] "3" "." "1" "4" "1" "5" "9" "2" "6"
my_pi <- c(3, TRUE, 4, TRUE, 5, 9, 2, 6) #强制类型转换
my_pi
#[1] 3 1 4 1 5 9 2 6
c(1, 2, c(4, 3), c(1, 0)) #不存在包含向量的向量，一律拆包
#> [1] 1 2 4 3 1 0
c(1, 2, 4, 3, 1, 0)
#> [1] 1 2 4 3 1 0
```

创建向量

```
(x1 <- vector("numeric", 8)) #事先知道长度和类型
#> [1] 0 0 0 0 0 0 0 0
(x2 <- numeric(8))
#> [1] 0 0 0 0 0 0 0 0
(x3 <- character(8))
#>  [1] "" "" "" "" "" "" "" ""
(x4 <- vector(len = 8))
#> [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
(x5 <- logical(8))
#> [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

创建向量：等差数列

```
#规则序列
#等差数列
seq(from = 1, to = 20, by = 2)
#> [1]  1  3  5  7  9 11 13 15 17 19
seq(from = 20, to = 1, by = -2)
#>  [1] 20 18 16 14 12 10  8  6  4  2
seq(from = 1, to = 20, len = 10)
#> [1]  1.0  3.1  5.2  7.3  9.4 11.6 13.7 15.8 17.9 20.0
```

显然，此时的by = (to - from) / (len - 1)

创建向量：等差数列

```
1:10 #from:to,步长为1的等差数列
#> [1]  1  2  3  4  5  6  7  8  9 10
pi:1
#> [1] 3.14 2.14 1.14
#注意运算符的优先级
1:10 - 1 #长度为10
# [1] 0 1 2 3 4 5 6 7 8 9
1:(10 - 1) #长度为9
#> [1] 1 2 3 4 5 6 7 8 9
#不要有记忆的负担，在R里边，不要吝啬{}和()的使用
```

创建向量：随机数列

```
#产生随机数
sample(10) #随机抽样
#> [1]  6  5  8  7  2  3  4  1 10  9
sample(c("b", "u", "p", "t", "a", "x", "b")) #随机抽样
#> [1] "u" "x" "t" "b" "a" "p" "b"
set.seed(2012) #设定随机数种子
sample(10) #结果应该是一致的，Reproducible Research
#[1]  3  7 10  9  5  6  8  4  2  1
(train_idx <- sample(1:10, 7))
#> [1]  3  6 10  5  4  1  8
```

创建向量：随机数列

```
#有放回的抽样
re_sample <- sample(1:100,
                    100,
                    replace = TRUE)
unique_re_sample <- unique(re_sample)
length(unique_re_sample) #有放回的抽样，有约36.8%的数不被抽到
#> [1] 62
```

向量的下标

向量的子集通过[]来指定

第1种方法：采用1~n的正整数来指定，n为向量的长度

```
yw <- c(94, 87, 92, 91, 85, 92)
yw[c(2, 5)]
#> [1] 87 85
yw[c(2, 5)] - 90
#> [1] -3 -5
yw[c(2, 5)] <- yw[c(2, 5)] + 6
yw
#> [1] 94 93 92 91 91 92
```

向量的下标

```
yw[] <- mean(yw)
yw
#> [1] 92.17 92.17 92.17 92.17 92.17 92.17
yw <- mean(yw)
yw
#> [1] 92.17

xm <- c("周黎", "汤海明", "舒江辉")
xm[c(1, 3, 2, 3)]
#> [1] "周黎"   "舒江辉" "汤海明" "舒江辉"
```

子集不子：下标可重复，顺序可变

向量的下标

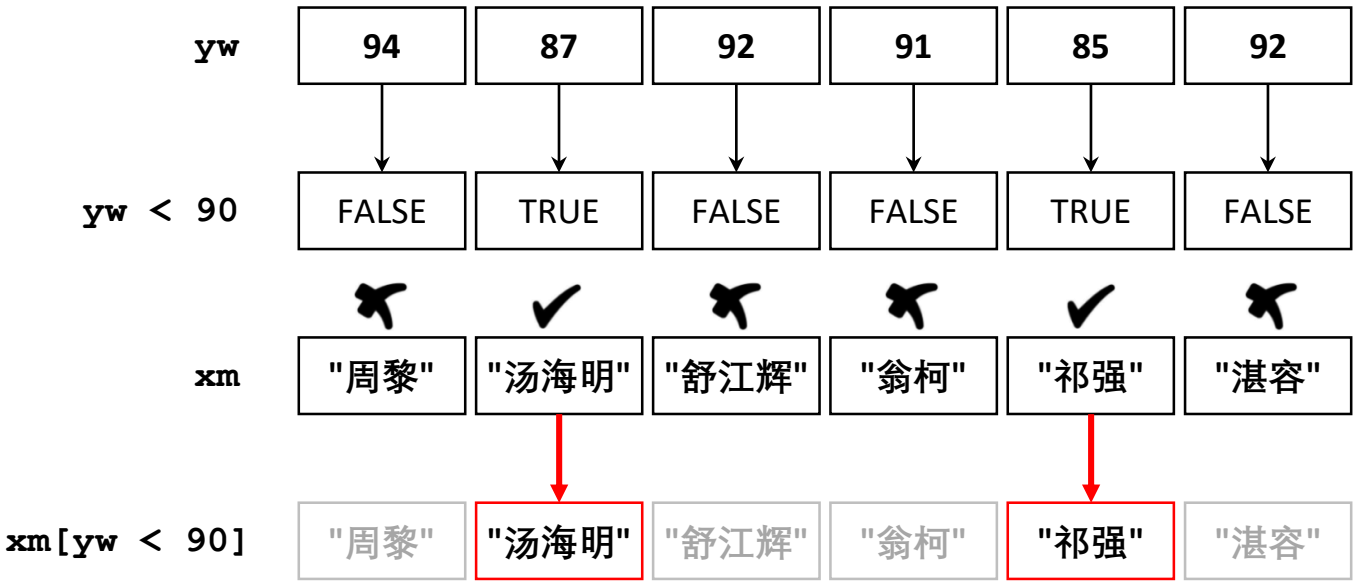
```
#方法二：采用负整数，反向选出某些元素
yw <- c(94, 87, 92, 91, 85, 92)
yw[-c(2, 5)]
#> [1] 94 92 91 92
which(yw < 90)
#> [1] 2 5
idx <- which(yw < 90)
yw[-idx] #避免了硬代码，增强了代码的可维护性
#> [1] 94 92 91 92
```

向量的下标

```
#方法三：逻辑下标
xm <- c("周黎", "汤海明", "舒江辉", "翁柯", "祁强", "湛容")
yw <- c(94, 87, 92, 91, 85, 92)
yw < 90
#> [1] FALSE  TRUE FALSE FALSE  TRUE FALSE
yw[yw < 90]
#> [1] 87 85
xm[yw < 90]
#> [1] "汤海明" "祁强"
```

R为何如此智能识别出了语文成绩小于90分（yw < 90）的同学?

向量的下标


yw
94
87
92
91
85
92
yw < 90
FALSE
TRUE
FALSE
FALSE
TRUE
FALSE
xm
"周黎"
"汤海明"
"舒江辉"
"翁柯"
"祁强"
"湛容"
xm[yw < 90]
"周黎"
"汤海明"
"舒江辉"
"翁柯"
"祁强"
"湛容"

向量的下标

```
#方法四：通过元素名访问相应的子集
xm <- c("周黎", "汤海明", "舒江辉", "翁柯", "祁强", "湛容")
yw <- c(94, 87, 92, 91, 85, 92)
names(yw) <- xm
yw
#> 周黎  汤海明  舒江辉    翁柯    祁强    湛容
#> 94      87      92      91      85      92
yw[c("汤海明", "祁强")]
#> 汤海明    祁强
#> 87      85
```

向量排序

```
fen_shu_xian2016 <- c(中国科学院大学 = 671, 中央民族大学 = 625,
  北京大学 = 678, 中国人民大学 = 670, 清华大学 = 680,
  北京交通大学 = 640, 北京科技大学 = 635, 北京化工大学 = 620,
  北京邮电大学 = 646, 中国农业大学 = 634, 北京林业大学 = 621)

sort(fen_shu_xian2016)
#> 北京化工大学 北京林业大学 中央民族大学 中国农业大学
#> 620          621          625          634
#> 北京科技大学 北京交通大学 北京邮电大学 中国人民大学
#> 635          640          646          670
#> 中国科学院大学       北京大学       清华大学
#> 671          678          680
```

向量排序

```
fen_shu_xian2016 <- c(中国科学院大学 = 671, 中央民族大学 = 625,
  北京大学 = 678, 中国人民大学 = 670, 清华大学 = 680,
  北京交通大学 = 640, 北京科技大学 = 635, 北京化工大学 = 620,
  北京邮电大学 = 646, 中国农业大学 = 634, 北京林业大学 = 621)
order(fen_shu_xian2016, decreasing = TRUE)
#> [1]  5  3  1  4  9  6  7 10  2 11  8
fen_shu_xian2016[order(fen_shu_xian2016, decreasing = TRUE)]
#> 清华大学       北京大学 中国科学院大学   中国人民大学
#> 680            678            671            670
#> 北京邮电大学   北京交通大学   北京科技大学   中国农业大学
#> 646            640            635            634
#> 中央民族大学   北京林业大学   北京化工大学
#> 625            621            620
```

向量逆序排列

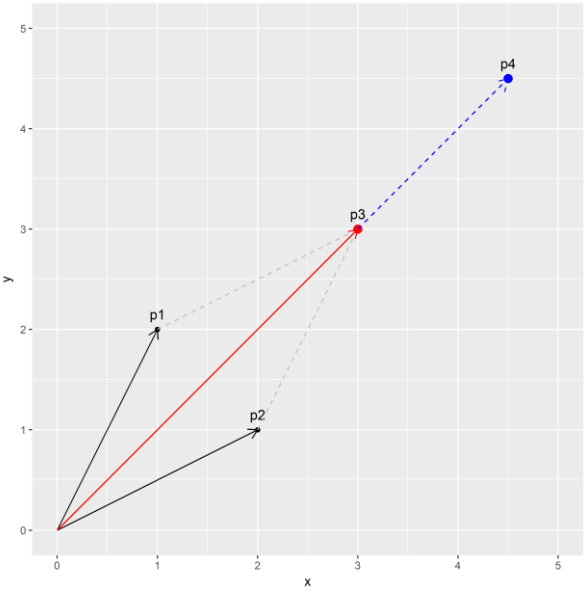
```
yw <- c(94, 87, 92, 91, 85, 92)
rev(yw)
#> [1] 92 85 91 92 87 94
yw[6] #可以用来取最后一个元素，但是这种硬代码很难维护
#> [1] 92
yw[length(yw)] #基本可行的办法
#> [1] 92
tail(yw, n = 1) #更好的选择
#> [1] 92
rev(tail(yw, n = 3)) #等价于head(rev(yw), n = 3)
#> [1] 92 85 91
```

数值向量运算

```
#原点
p0 <- c(x = 0, y = 0)
#向量1
p1 <- c(x = 1, y = 2)
#向量2
p2 <- c(x = 2, y = 1)

#求和
p3 <- p1 + p2

#数乘
p4 <- 1.5 * p3
```

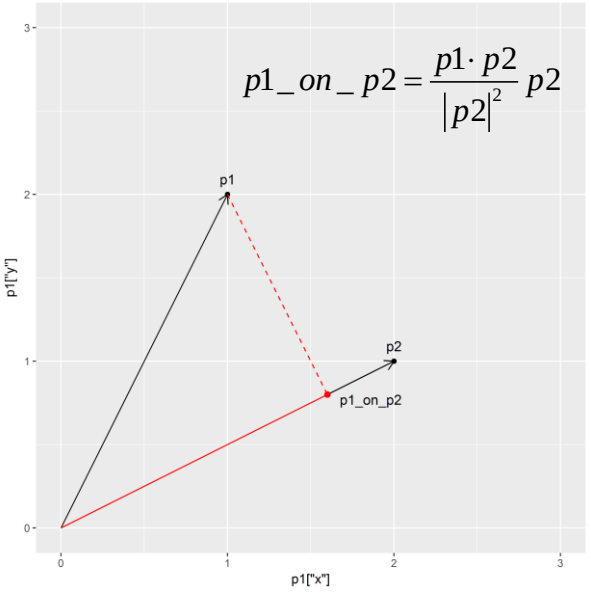


数值向量运算

```
#原点
p0 <- c(x = 0, y = 0)
#向量1
p1 <- c(x = 1, y = 2)
#向量2
p2 <- c(x = 2, y = 1)

#投影向量
p1_on_p2 <-
  sum(p1 * p2) /
  sum(p2 * p2) * p2
```

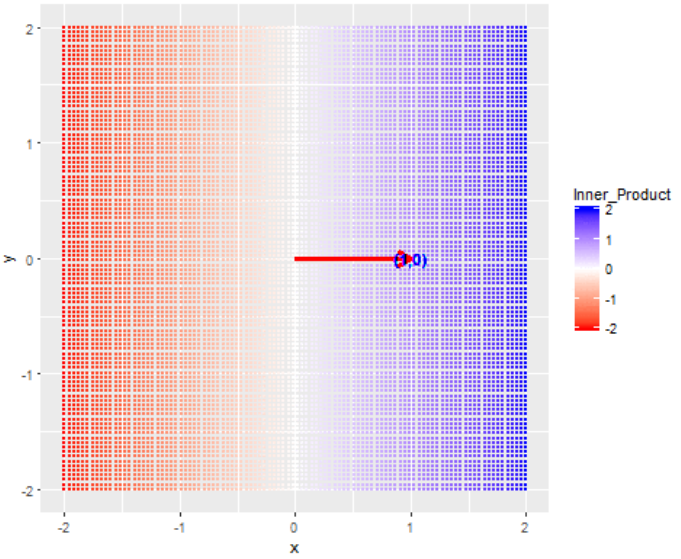
$$p1\_on\_p2 = \frac{p1 \cdot p2}{|p2|^2} p2$$



数值向量运算

```r
#原点
p0 <- c(x = 0, y = 0)
#向量1
p1 <- c(x = 1, y = 2)
#向量2
p2 <- c(x = 2, y = 1)

#内积
sum(p1 * p2)
```



数值向量运算

```
#向量的内积
set.seed(2012)
x <- rnorm(100)
y <- rnorm(100)
#求向量的内积
sum(x * y)
#> [1] -11.1336

sum(sort(x) * sort(y))
#> [1] 128.3501
sum(sort(x) * sort(y, decreasing = TRUE))
#[1] -127.108
```

谢谢聆听

Thank you

教师个人联系方式

艾新波

手机：13641159546

QQ： 23127789

微信：13641159546

E-mail: 13641159546@126.com

axb@bupt.edu.cn

地址：北京邮电大学科研楼917室

课程网址：https://github.com/byaxb/RDataAnalytics