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

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

单一维度数据集合

属性类型不同、长度相同

本质上是列表

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

方以类聚、物以群分——聚类

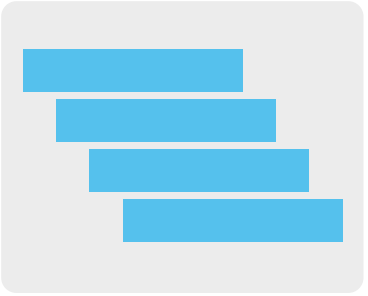
2 因子

4 数组

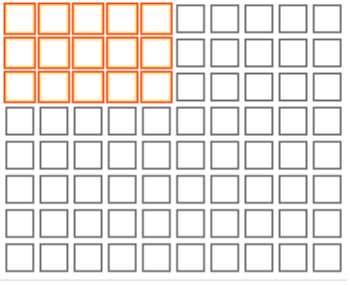
数据整理最后都是为了得到数据框

R语言数据分析人员的大部分时间，都是在把既有数据转换成**数据框**。进而以数据框为基础，开展数据探索和数据建模

各种渠道的数据



主题相关的数据



关系表数据
（数据框/矩阵）

数据框的组成

学习data.frame()先从官方帮助文档学起

data.frame {base} R Documentation

Data Frames

Description

The function `data.frame()` creates data frames, tightly coupled collections of variables which share many of the properties of matrices and of lists, used as the fundamental data structure by most of R's modeling software.

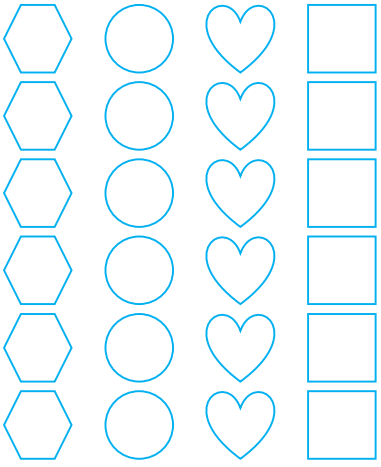
Usage

```
data.frame(..., row.names = NULL, check.rows = FALSE,
           check.names = TRUE, fix.empty.names = TRUE,
           stringsAsFactors = default.stringsAsFactors())
```

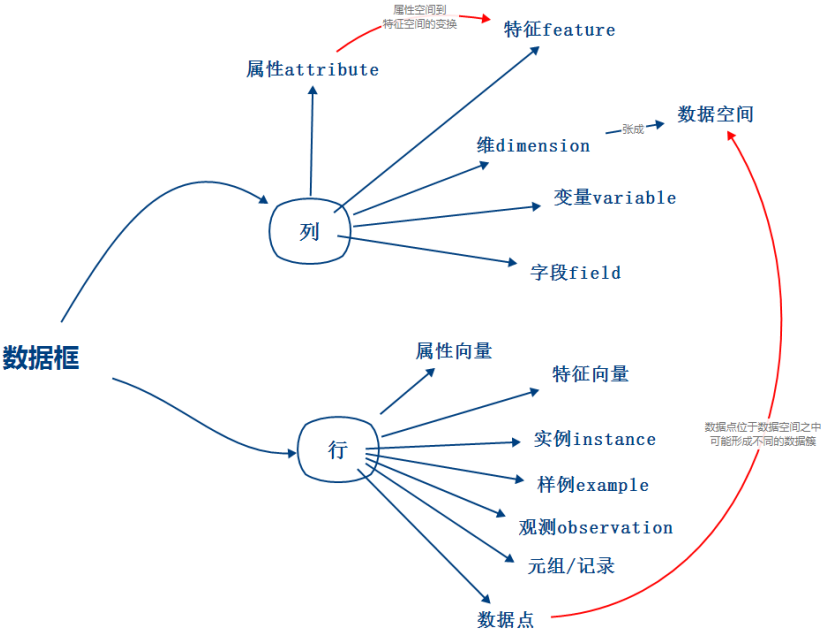
- The function data.frame() creates data frames, tightly coupled collections of variables which share many of the properties of matrices and of lists, used as the fundamental data structure by most of R's modeling software.
- A data frame is a list of variables of the same number of rows with unique row names, given class "data.frame". If no variables are included, the row names determine the number of rows.

数据框的组成

- 数据框（data.frame）是最美好的数据对象
- 形式上是矩阵、本质上是列表
- 与数据库关系表、Excel中的sheet相近
- 个体-变量（cases by variables）矩阵
- 列：变量、属性、特征、维度
- 行：记录，观测值，n维数据空间的一个点



数据框的行与列



自由度，其实也类似于维度、列

数据框的行与列

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	xm	bj	xb	yw	sx	wy	zz	ls	dl	wl	hx	sw	wlfk
2	周黎	1101	女	94	82	96	97	97	98	95	94	88	文科
3	汤海明	1101	男	87	94	89	95	94	94	90	90	89	文科
4	舒江辉	1101	男	92	79	86	98	95	96	89	94	87	文科
5	翁柯	1101	女	91	84	96	93	97	94	82	90	83	文科
6	祁强	1101	男	85	92	82	93	87	88	95	94	93	文科
7	湛容	1101	女	92	82	85	91	90	92	82	98	90	文科
8	穆伶俐	1101	女	88	72	86	94	87	88	89	98	94	文科
9	韦永杰	1101	男	81	89	87	97	94	96	81	88	83	文科
10	龚兰秀	1101	女	88	77	95	94	84	94	87	94	82	文科
11	舒亚	1101	女	94	81	88	91	85	98	81	88	88	文科
12	宰玲玲	1101	女	87	83	92	91	86	94	84	90	87	文科
13	邰友生	1101	男	88	82	91	89	81	98	89	98	75	文科
14	历阳	1101	男	79	84	91	87	91	87	85	96	90	文科
15	卜杰	1101	男	78	81	83	86	88	98	85	90	99	文科
16	桑锡雨	1101	男	83	85	82	91	88	88	88	86	93	文科
17	屠珍锦	1101	女	80	84	91	91	85	100	73	90	90	文科
18	骆英	1101	女	89	71	90	95	91	96	84	88	80	文科
19	韶丽君	1101	女	89	84	85	97	84	86	72	96	89	文科

数据框的创建

```
xm <- c("周黎", "汤海明", "舒江辉", "翁柯", "祁强", "湛容")
xb <- factor(c("女", "男", "男", "女", "男", "女"))
yw <- c(94, 87, 92, 91, 85, 92)
sx <- c(82, 94, 79, 84, 92, 82)
wy <- c(96, 89, 86, 96, 82, 85)
cjb <- data.frame(xm = xm,
                  xb = xb,
                  yw = yw,
                  sx = sx,
                  wy = wy)
```

数据框的基本操作

```
#由于数据框本质上是列表，可以通过以下三种方式访问其中的列
cjb$xm
cjb[[1]]
cjb[["xm"]]
#> [1] "周黎"   "汤海明" "舒江辉" "翁柯"   "祁强"   "湛容"
```

一般来讲[[]]的用法较少，要么采用$，要么采用以下的矩阵式操作

```
cjb[, 1]
cjb[, "xm"]
```

数据框的基本操作

```
cjb[1, ]
#> xm xb yw sx wy
#> 1 周黎 女 94 82 96
cjb[c(1, 3), c("xm", "sx")]
#> xm sx
#> 1   周黎 82
#> 3 舒江辉 79
cjb[1:3, -1]
#>   xb yw sx wy
#> 1 女 94 82 96
#> 2 男 87 94 89
#> 3 男 92 79 86
```

xm	xb	yw	sx	wy
周黎	女	94	82	96
汤海明	男	87	94	89
舒江辉	男	92	79	86
翁柯	女	91	84	96
祁强	男	85	92	82
湛容	女	92	82	85

数据框的基本操作

```
#作为列表，通过美元符号增加一列政治zz
cjb$zz <- c(97, 95, 98, 93, 93, 91)
#像矩阵，cbind也可以
cjb <- cbind(cjb,
             ls = c(97, 94, 95, 97, 87, 90))
```

xm	xb	yw	sx	wy	zz
周黎	女	94	82	96	97
汤海明	男	87	94	89	95
舒江辉	男	92	79	86	98
翁柯	女	91	84	96	93
祁强	男	85	92	82	93
湛容	女	92	82	85	91

xm	xb	yw	sx	wy	zz	ls
周黎	女	94	82	96	97	97
汤海明	男	87	94	89	95	94
舒江辉	男	92	79	86	98	95
翁柯	女	91	84	96	93	97
祁强	男	85	92	82	93	87
湛容	女	92	82	85	91	90

读取数据

```
#数据不会在代码里逐字敲入，也不会通过控制台输入
#而是直接读取已经采集好的数据
cjb_url <-
"https://github.com/byaxb/RDataAnalytics/raw/master/data/cjb.csv"


cjb <- read.csv(cjb_url,
                head = TRUE,
                stringsAsFactors = FALSE)
View(cjb)
```

读取数据

xm	bj	xb	yw	sx	wy	zz	ls	dl	wl	hx	sw	wlfk
周黎	1101	女	94	82	96	97	97	98	95	94	88	文科
汤海明	1101	男	87	94	89	95	94	94	90	90	89	文科
舒江辉	1101	男	92	79	86	98	95	96	89	94	87	文科
翁柯	1101	女	91	84	96	93	97	94	82	90	83	文科
祁强	1101	男	85	92	82	93	87	88	95	94	93	文科
湛容	1101	女	92	82	85	91	90	92	82	98	90	文科
穆伶俐	1101	女	88	72	86	94	87	88	89	98	94	文科
韦永杰	1101	男	81	89	87	97	94	96	81	88	83	文科
龚兰秀	1101	女	88	77	95	94	84	94	87	94	82	文科
舒亚	1101	女	94	81	88	91	85	98	81	88	88	文科
宰玲玲	1101	女	87	83	92	91	86	94	84	90	87	文科
邰友生	1101	男	88	82	91	89	81	98	89	98	75	文科
历阳	1101	男	79	84	91	87	91	87	85	96	90	文科
卜杰	1101	男	78	81	83	86	88	98	85	90	99	文科
桑锡雨	1101	男	83	85	82	91	88	88	88	86	93	文科
屠珍锦	1101	女	80	84	91	91	85	100	73	90	90	文科

查看数据

```
head(cjb)
#>      xm   bj xb yw sx wy zz ls dl wl hx sw wlfk
#> 1   周黎 1101 女 94 82 96 97 97 98 95 94 88 文科
#> 2 汤海明 1101 男 87 94 89 95 94 94 90 90 89 文科
#> 3 舒江辉 1101 男 92 79 86 98 95 96 89 94 87 文科
#> 4   翁柯 1101 女 91 84 96 93 97 94 82 90 83 文科
#> 5   祁强 1101 男 85 92 82 93 87 88 95 94 93 文科
#> 6   湛容 1101 女 92 82 85 91 90 92 82 98 90 文科
tail(cjb, n = 3)
#>       xm   bj xb yw sx wy zz ls dl wl hx sw wlfk
#> 773 徐宏平 1115 男 85 59 89 80 85 82 61 64 75 理科
#> 774 昌肖峰 1115 男 81 62 76 89 76 91 49 68 74 理科
#> 775 郑慕海 1115 男 72 59 82 92 85 82 59 58 55 理科
```

查看数据

```
#Compactly Display the Structure
str(cjb) #查看数据的结构
#> 'data.frame':    775 obs. of  13 variables:
#> $ xm  : chr  "周黎" "汤海明" "舒江辉" "翁柯" ...
#> $ bj  : int  1101 1101 1101 1101 1101 1101 1101 1101 ...
#> $ xb  : chr  "女" "男" "男" "女" ...
#> $ yw  : int  94 87 92 91 85 92 88 81 88 94 ...
#> $ sx  : int  82 94 79 84 92 82 72 89 77 81 ...
#> $ sw  : int  88 89 87 83 93 90 94 83 82 88 ...
#> $ wlfk: chr  "文科" "文科" "文科" "文科" ...
```

查看数据

```
summary(cjb) #对数据进行统计描述
#>  xm                  bj              xb
#>  Length:775         Min.   :1101   Length:775
#>  Class :character   1st Qu.:1104   Class :character
#>  Mode  :character   Median :1107   Mode  :character
#>                     Mean   :1108
#>                     3rd Qu.:1111
#>                     Max.   :1115
#>  yw               sx               wy
#>  Min.   : 0.00   Min.   :  0.00   Min.   : 0.0
#>  1st Qu.:85.00   1st Qu.: 81.00   1st Qu.:84.0
#>  Median :88.00   Median : 89.00   Median :88.0
#>  Mean   :87.27   Mean   : 86.08   Mean   :87.4
#>  3rd Qu.:91.00   3rd Qu.: 95.00   3rd Qu.:92.0
#>  Max.   :96.00   Max.   :100.00   Max.   :99.0
```

查看数据

```
names(cjb)
#> [1] "xm"   "bj"   "xb"   "yw"   "sx"   "wy"   "zz"
#> [8] "ls"   "dl"   "wl"   "hx"   "sw"   "wlfk"
colnames(cjb) #结果同上
nrow(cjb)
#> [1] 775
ncol(cjb)
#> [1] 13
length(cjb) #结果同上
#> [1] 13
```

查看数据

```
#作必要的类型转换
cjb$bj <- factor(cjb$bj)
cjb$xb <- factor(cjb$xb)
cjb$wlfk <- factor(cjb$wlfk)
str(cjb)
#> 'data.frame':    775 obs. of  13 variables:
#> $ xm  : chr  "周黎" "汤海明" "舒江辉" "翁柯" ...
#> $ bj  : Factor w/ 15 levels "1101","1102",..: 1 1 1 1 ...
#> $ xb  : Factor w/ 2 levels "男","女": 2 1 1 2 1 2 2 ...
```

查看数据

```
summary(cjb) #再次summary，数据分析总是反复迭代的
#> xm                  bj            xb
#> Length:775          Min.   :1101   Length:775
#> Class :character    1st Qu.:1104   Class :character
#> Mode  :character    Median :1107   Mode  :character
#>                     Mean   :1108
#>                     3rd Qu.:1111
#>                     Max.   :1115
#> xm                   bj           xb
#> Length:775          1102   : 58   男:369
#> Class :character    1103   : 57   女:406
#> Mode  :character    1105   : 57
#>                     1104   : 56
#>                     1106   : 56
#>                     1107   : 55
#>                     (Other):436
```

数据框记录排序

```
cjb$zcj <- apply(cjb[, 4:12], 1, sum)
order(cjb$zcj, decreasing = TRUE)[1:5]
#> [1] 488 392 438 393 489 337
cjb_sorted <- cjb[order(cjb$zcj, decreasing = TRUE), ]
View(cjb_sorted)
```

	xm	bj	xb	yw	sx	wy	zz	ls	dl	wl	hx	sw	wlfk	zcj
488	宁琦	1110	男	94	97	97	97	100	100	100	100	100	理科	885
392	焦金音	1108	女	91	98	95	99	99	100	100	100	97	理科	879
438	鲁孟秋	1109	女	93	98	98	97	96	98	100	100	98	理科	878
393	伊礼贤	1108	男	92	97	99	95	94	100	100	100	99	理科	876
489	傅世鸿	1110	男	88	98	99	95	100	100	97	100	95	理科	872
337	嵇婉玉	1107	女	94	100	96	98	100	98	88	100	97	文科	871

训练集和测试集

```
#数据集分为训练集和测试集
set.seed(2012)
n_record <- nrow(cjb)
train_idx <- sample(1:n_record, floor(n_record * 0.7))
train_idx <- sample(n_record, n_record * 0.7)
length(train_idx)
#> [1] 542
test_idx <- (1:n_record)[-train_idx]
test_idx <- setdiff(1:n_record, train_idx)
length(test_idx)
#> [1] 233
```

训练集和测试集

```
#得到测试集和训练集
train_set <- cjb[train_idx, ]
test_set <- cjb[-train_idx, ]
test_set <- cjb[test_idx, ]
#显然，下面这种方式是错的
train_set <- cjb[sample(n_record, n_record * 0.7), ]
test_set <- cjb[sample(n_record, n_record * 0.3), ]
```

在工业级/商业级应用中，建议不要重复造轮子，直接采用caret等包中训练集和测试集划分的相关函数如createDataPartition()

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