R语言基础

数据结构

构建子集

重要函数的使用

数据结构

- 对象的5种基本类型(classes of objects)
  - 字符 (character)
  - 数值 (numeric: real numbers)
  - 整数 (integer)
  - 复数 (complex): 1+2i
  - 逻辑 (logical: True / False)

数据结构

- 属性 (attribute)
  - 名称 (name)
  - 维度 (dimensions: matrix, array)
  - 类型 (class)
  - 长度 (length)

数据结构

- 向量 (vector)
  - 只能包含同一类型的对象
  - 创建向量
    - vector()
    - c()
    - as.logical() / as.numeric() / as.character()

数据结构

- 矩阵 (matrix)
  - 向量 + 维度属性 (整数向量: nrow, ncol)
  - 创建矩阵
    - matrix(): 先列后行
    - vector() + dim()
    - cbind(), rbind()
    - attributes()

数据结构

- 数组 (array)
  - 与矩阵类似，但是维度可以大于2
  - 创建数组
    - array()

数据结构

- 列表 (list)
  - 可以包含不同类型的对象
  - 创建列表
    - list()

数据结构

- 因子 (factor)
  - 分类数据 / 有序 vs. 无序
  - 整数向量 + 标签(label)（优于整数向量）
    - Male / Female vs. 1 / 2
    - 常用于lm(), glm()
  - 创建因子
    - factor()
    - table() / unclass()

数据结构

- 缺失值 (missing value)
  - NA / NaN: NaN属于NA, NA不属于NaN
  - NA有类型属性 : integer NA, character NA等
  - is.na() / is.nan()

数据结构

- 数据框 (data frame)
  - 存储**表格数据** (tabular data)
  - 视为各元素长度相同的**列表**
    - 每个元素代表**一列数据**
    - 每个元素的长度代表**行数**
    - 元素类型可以不同

数据结构

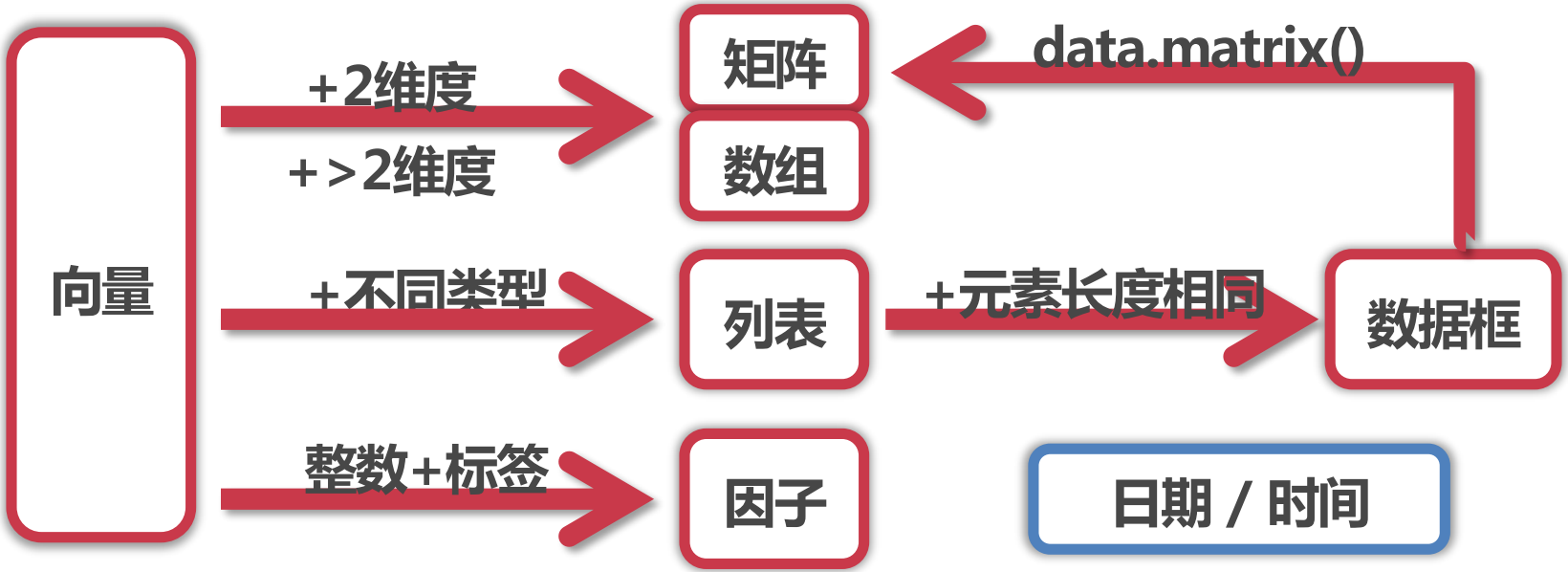
- 日期与时间 (date, time)
  - 日期: Date
    - 距离1970-01-01的天数 / date() / Sys.Date()
    - weekdays() / months() / quarters()
  - 时间: POSIXct / POSIXlt
    - 距离1970-01-01的秒数 / Sys.time()
    - POSIXct: 整数, 常用于存入数据框
    - POSIXlt: 列表, 还包含星期、年、月、日等信息

数据结构

- 日期与时间 (date, time)
  - 字符 → 日期/时间
    - as.Date()
    - as.POSIXct() / as.POSIXlt() / strptime()

小结

- 5种对象类型
  - character, numeric, integer, complex, logical



构建子集 (subsetting)

- 原始数据 (raw dataset) → 预处理后的数据 (clean dataset)

- 基本方法
  - []: 提取一个或多个类型相同的元素
  - [[]]: 从列表或数据框中提取元素
  - $: 按名字从列表或数据框中提取元素

构建子集

- 矩阵的子集
  - [i,j] / [i,] / [,j] / [i, c(j1, j2)]
  - 提取元素，返回的是向量（而不是矩阵）
    - drop = FALSE

构建子集

- 数据框的子集

构建子集

- 列表的子集
  - [[]] / $ / [[]][] / [[]][[]]
  - 嵌套列表 / 不完全匹配 (partial matching)

构建子集

- 处理缺失值

构建子集

- 向量化操作 (vectorized operation)
  - 可以作用于向量、矩阵等结构，使得代码简洁、易于阅读、效率高

小结

构建子集: [] / [[]] / $ / [[]][] / [[]][[]]

处理缺失值: is.na() / complete.cases()

向量化操作

重要函数的使用

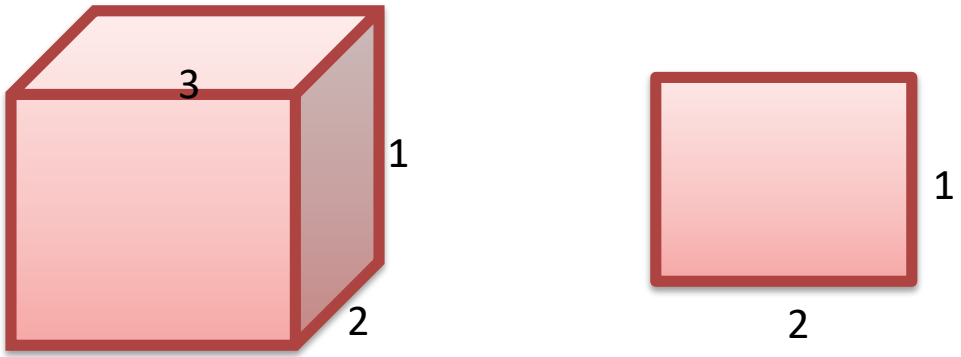
- 处理循环
  - R不仅有for / while循环语句，还有更强大的实现循环的“一句话”函数
- 排序
- 总结数据信息

重要函数的使用

- lapply
  - 可以循环处理列表中的每一个元素
  - lapply(参数): lapply(列表, 函数/函数名, 其他参数)
  - 总是返回一个列表
  - sapply: 简化结果
    - 结果列表元素长度均为1，返回向量
    - 结果列表元素长度相同且大于1，返回矩阵

重要函数的使用

- apply
  - 沿着数组的某一维度处理数据
    - 例如：将函数用于矩阵的行或列
    - 虽然与for/while循环的效率相似，但是只用一句话就可以完成
  - apply(参数): apply(数组, 维度, 函数/函数名)



重要函数的使用

- mapply
  - lapply的多元版本
  - mapply(参数): mapply(函数/函数名, 数据, 函数相关的参数)

重要函数的使用

- tapply
  - 对向量的子集进行操作
  - tapply(参数): tapply(向量, 因子/因子列表, 函数/函数名)

重要函数的使用

- split
  - 根据因子或因子列表将向量或其他对象分组
  - 通常与lapply一起使用
  - split(参数): split(向量/列表/数据框, 因子/因子列表)

重要函数的使用

- 排序
  - sort: 对向量进行排序; 返回排好序的内容
  - order: 返回排好序的内容的下标 / 多个排序标准

重要函数的使用

- 总结数据信息

小结

“一句话”循环: lapply (sapply, split) / apply / mapply / tapply

排序: sort / order

总结数据信息: head / tail / summary / str / table / xtabs / ftable / object.size