

# 数据分析与处理技术

## 变量类型

南京审计大学商学院物流管理系

# 变量基础

---

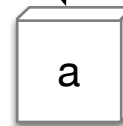
- 变量对应了内存中的一块记录空间
- **赋值** 将数据写入一个变量中，改变变量存储
- **调用(或称为访问)** 调取变量中存储的数据，但不改变变量存储

赋值	<-	->	=
----	----	----	---

赋值命令

a<-3

数字3



调用命令

a

赋值符号有：<- 或 = 左赋值 -> 右赋值

# 变量命名规则

---

- 命名规则
  - 不能以数字或非字母符号开头
  - 大小写敏感
  - 不能占用已有命令名
- 常用方式
  - 可以用下划线、小数点表示
  - 大小写混合方式
  - 避免用中文

## 变量命名的风格

1. 变量名通常是名词
2. 全部使用小写字母，避免驼峰式命名
3. 避免使用非英文符号的变量名，包括中文作变量名

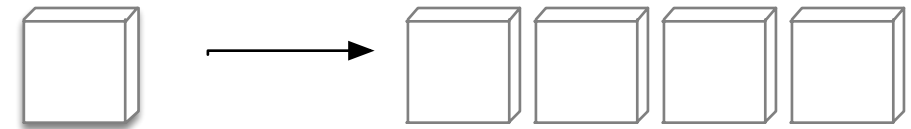
例如：  
day\_one  
day\_1

# 原子向量

向量赋值时必须使用c()将数据组合成一个整体写入变量中

变量单元

向量



赋值命令

`a<-3`

`a<-c(3,5,6,7)`

访问整个向量时仍然直接用变量名a

```
> a=c(3,5,6,7)
> a
[1] 3 5 6 7
```

注意，向量只能保存单一的数据类型，例如

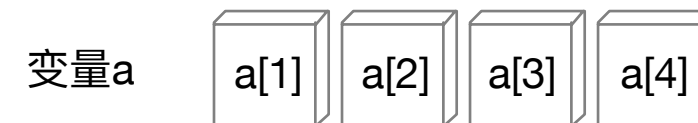
```
> y<-c(1,2,'a')
> y
[1] "1" "2" "a"
```

原子向量是与计算机存储结构最接近的变量类型，也是所有变量类型的基础

# 向量索引

---

访问向量内的元素时需索引号[]指示元素位置



索引的基本功能是指示变量元素的位置

```
a[1]=3  
a[2]=5  
a[3]=6  
a[4]=7
```

变量：调取变量a所有元素

```
> a  
[1] 3 5 6 7  
> a[2]
```

变量+索引：调取变量a第二个元素

加负号：调取变量a的所有元素除了第二个元素

```
> a[-2]  
[1] 3 6 7
```

# 索引的逻辑性

---

索引的基本功能是指示变量元素的位置

例如a向量的操作，a代表了整体的名称，索引[]则指示了a中第几个元素

```
> a=c(1.2,3.7,5,12.1,-19,0.75)
> a
[1] 1.20 3.70 5.00 12.10 -19.00 0.75
> a[2]
[1] 3.7
> a[2:4]
[1] 3.7 5.0 12.1
> a[c(1,3,5)]
[1] 1.2 5.0 -19.0
```

索引同样可以根据筛选条件返回对应元素，而不需具体指示元素位置

例如选择性调取a中大于4的元素

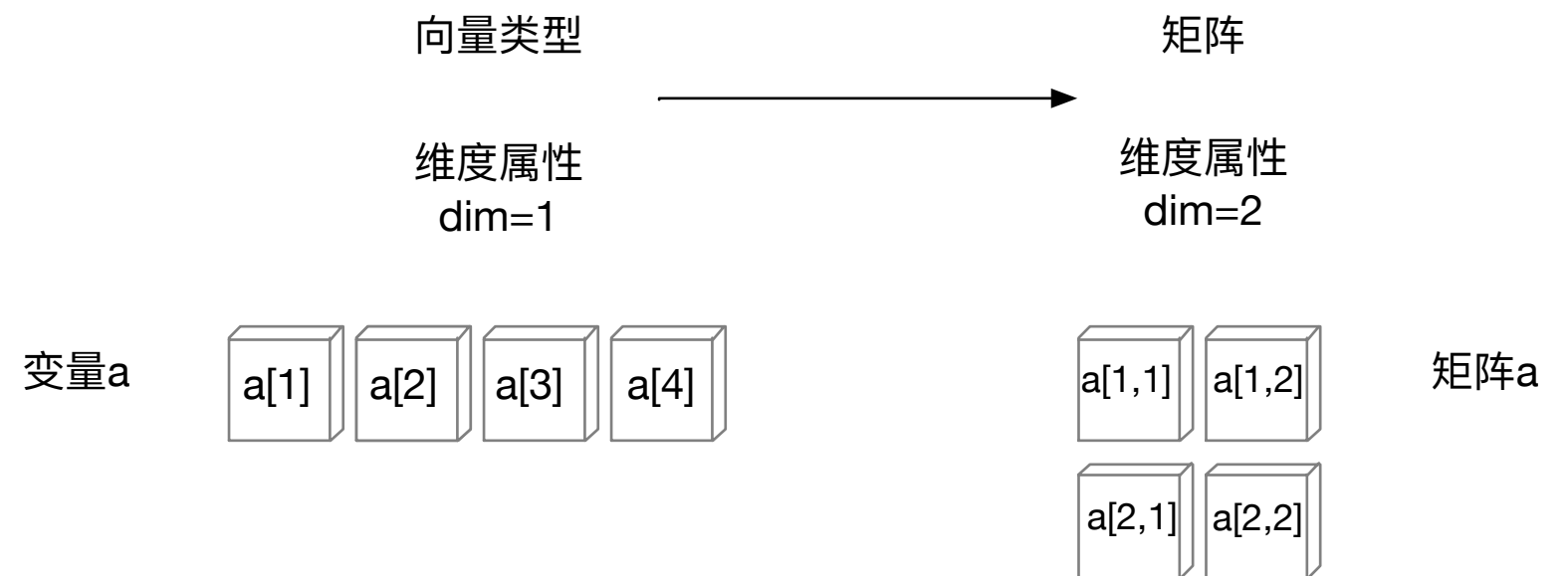
```
> a[a>4]
[1] 5.0 12.1
```

其原因在于比较过程的向量化运算产生的逻辑向量

```
> a>4
[1] FALSE FALSE TRUE TRUE FALSE FALSE
```

# 矩阵

在原子向量基础上改变维度属性，dim=2时向量变成了矩阵类型



赋值矩阵变量时需先创建原子向量，再用matrix()创建出矩阵

```
a=matrix(data=c(3,5,6,7), ncol=2, nrow=2, byrow=TRUE)
```

当然，许多参数有默认值，因此可以简化

```
a=matrix(c(3,5,6,7),ncol=2)
```

```
> y<-matrix(c(1,2,3,'r'),ncol = 2)
> y
      [,1] [,2]
[1,] "1"  "3"
[2,] "2"  "r"
```

tips:直接用dim(a)可以取出维度

# 矩阵索引

相对原子向量，矩阵变量有了行列属性，索引也就自然得拓展到二维

例如：右图创造了一个矩阵m，取出m第一行第二列的元素

取第二行所有元素，空缺的列属性表示全选

同时，索引的逻辑特性也能发挥作用

```
> m<-matrix(1:12,ncol=3)
```

```
> m
```

	[,1]	[,2]	[,3]
[1,]	1	5	9
[2,]	2	6	10
[3,]	3	7	11
[4,]	4	8	12

```
>
```

```
> m[1,2]
```

```
[1] 5
```

```
> m[2,]
```

```
[1] 2 6 10
```

```
> m[-2,]
```

	[,1]	[,2]	[,3]
[1,]	1	5	9
[2,]	3	7	11
[3,]	4	8	12

```
> m[m>3]
```

```
[1] 4 5 6 7 8 9 10 11 12
```



# 矩阵运算

常见算术运算符在矩阵中并非数学中的左乘、右乘、内积等含义，而是元素间点对点的算术运算，如右侧例子

矩阵乘法符号是`%*`，即

```
> m%*%u
      [,1] [,2]
[1,]    4    4
[2,]    6    6
```

另外一些矩阵常用运算：

转秩 `t()`

求行列式 `det()`

取特征值 `eigen()`

取行列长度 `dim()`

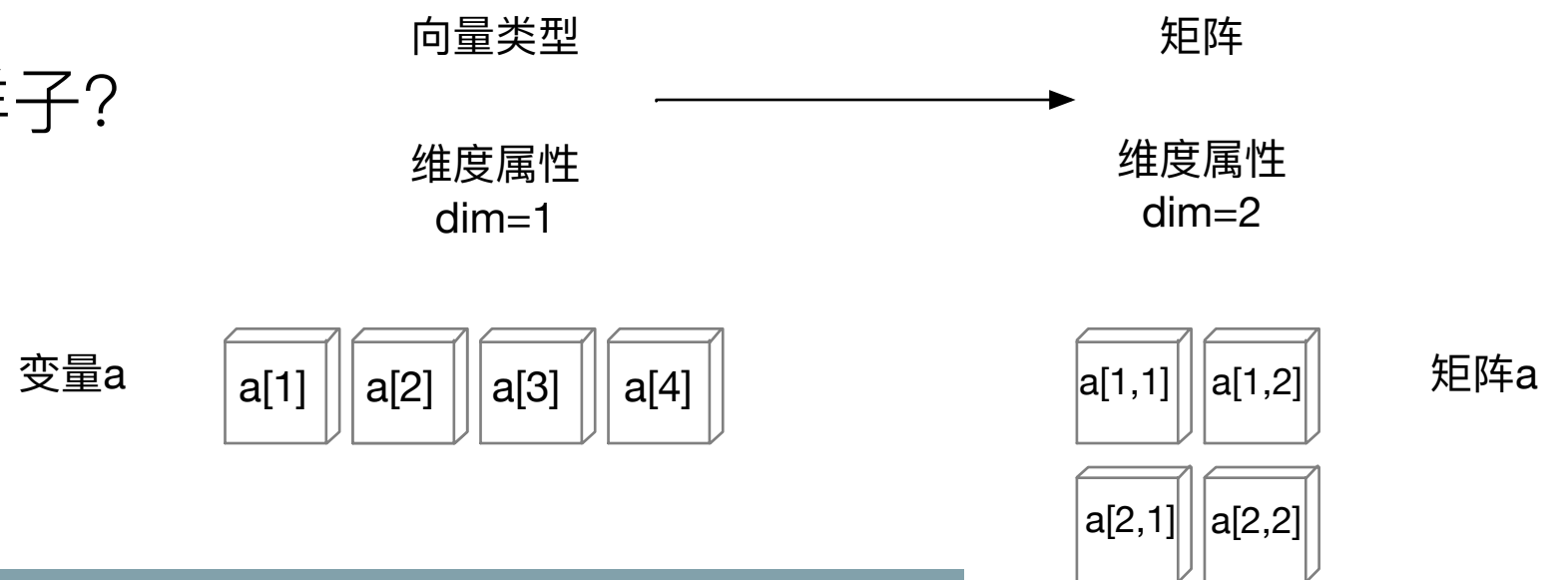
解线性方程 `solve(A,b)`

```
> m<-matrix(c(1,2,3,4),ncol=2)
> m
      [,1] [,2]
[1,]    1    3
[2,]    2    4
> u<-matrix(c(1,1,1,1),ncol=2)
> u
      [,1] [,2]
[1,]    1    1
[2,]    1    1
> m+u
      [,1] [,2]
[1,]    2    4
[2,]    3    5
> m*u
      [,1] [,2]
[1,]    1    3
[2,]    2    4
```

# 数组变量

dim=3时，变量结构变成什么样子？

dim=3时，索引格式如何变化？



当维度大于2时，变量从matrix类型变为array类型，即数组类型

当dim>3时，只能在思维中想象变量间关系的形式

# 列表变量

列表变量，即list类型，与矩阵的构造方法不同。列表将变量作为元素放入其中，如

```
> a
[1] 1 2 3 4 5 6 7 8 9 10
> b
[1] "my"      "name"    "wangqi"
> w<-list(a,b)
> w
[[1]]
[1] 1 2 3 4 5 6 7 8 9 10

[[2]]
[1] "my"      "name"    "wangqi"
```



```
> a
[1] 1 2 3 4 5 6 7 8 9 10
> b
[1] "my"      "name"    "wangqi"
> w<-list(a=a,b=b)
> w
$a
[1] 1 2 3 4 5 6 7 8 9 10

$b
[1] "my"      "name"    "wangqi"
```

注意元素命名的差

变量w将两个向量作为元素放入了自身存储空间，此时w与a,b之间构成了一种类似家族名称与个人名字的关系。那么，可以通过两种方式调用变量，w[[ 1 ]]调用w的第一个元素；w\$a调用w的a元素

```
> w[[1]]
[1] 1 2 3 4 5 6 7 8 9 10
> w$a
[1] 1 2 3 4 5 6 7 8 9 10
```

# 列表变量

---

假设 环境中存在a、b、y三个向量

```
> a
[1] 1.20 3.70 5.00 12.10 -19.00 0.75
> b
[1] "my" "name" "wangqi"
> y
[1] 1 2 3 4 5 6
```

右侧代码的第二个元素  
将正弦函数sin装入其中

```
> lista<-list(a=y,b=sin)
> lista$b(10)
[1] -0.5440211
```

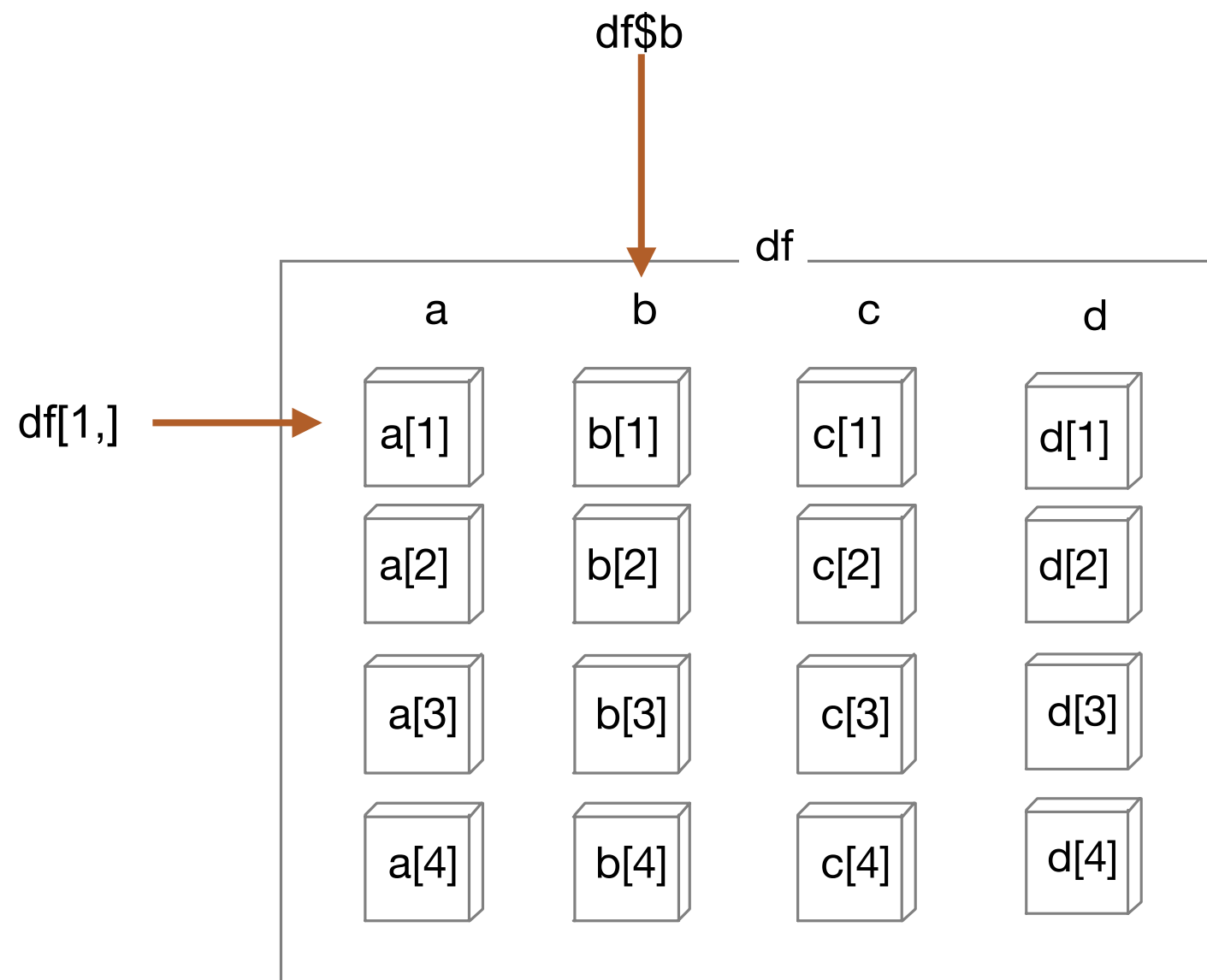
列表变量本质上是原子向量的拓展，不同元素的类型互不干扰。  
同时又称为递归向量，可以将另一个列表装入其中，它甚至可以将自己作为元素装入其中

```
> lista<-list(aa=b,bb=lista)
```

# 数据框/data.frame

数据框的思路：以向量为单位，将多个等长变量装入一个大的变量，装入的变量即为数据框的元素

数据框在结构上等同于矩阵，大部分适用于矩阵的操作都适用于数据框变量



# 操作data.frame类型数据

---

```
> df<-data.frame(first=x,second=letters[1:4])  
> df
```

变量



对象 →

	first	second
1	1	a
2	2	b
3	3	c
4	9	d

data.frame变量是一种同时具备矩阵外形的特殊列表类型，因此同时具备了列表和矩阵大多数操作方法。

调用元素方式：

```
> df$second[3]  
[1] c  
Levels: a b c d
```

矩阵方式调用：

```
> df[3,2]  
[1] c  
Levels: a b c d
```

混合按名称调用：

```
> df[3,'second']  
[1] c  
Levels: a b c d
```

```
> dim(df)  
[1] 4 2  
> colnames(df)  
[1] "first" "second"  
> row.names(df)  
[1] "1" "2" "3" "4"
```

	专业	学号	姓名	性别	第一学期考试成绩	第二学期考试成绩	考级考证数量	获奖证书数量	所学专业兴趣
1	教育技术学	11002001	刘 芮	女	657	492	2	3	2
2	教育技术学	11002002	付 恒	男	600	419	1	2	1
3	教育技术学	11002003	廖瑞斌	男	628	450	1	0	3
4	教育技术学	11002004	李 星	女	627	445	2	1	1
5	教育技术学	11002005	张倩倩	女	675	501	3	3	1
6	教育技术学	11002006	甄晓静	女	670	511	3	4	1
7	教育技术学	11002007	温金涛	男	630	496	2	0	2
8	教育技术学	11002008	岳 亮	男	577	432	1	0	2
9	教育技术学	11002009	后世强	男	602	453	1	1	2
10	教育技术学	11002010	邱 月	女	639	436	2	0	3
11	教育技术学	11002011	王春雷	男	568	428	1	0	3
12	教育技术学	11002012	杨少青	男	630	435	1	1	2
13	教育技术学	11002013	姬永浩	男	611	452	1	0	3
14	教育技术学	11002014	朱 震	男	617	480	2	2	1
15	教育技术学	11002015	黄慧娟	女	674	522	4	4	1



# 变量类型间的关系

- `c()`: 组合函数, 生成向量, 基础数据结构
- 列表: 每个位置可以放任意长度、类型的数据
- 数据框: 变量长度等长, 结构整体的列表
- 数组: 数据可以有多个维度
- 矩阵: 仅有行列两个维度时候的数组

