

# 第2章 多元数据的数学表达及R使用

武慧

[wuh@hit.edu.cn](mailto:wuh@hit.edu.cn)

经济管理学院  
哈尔滨工业大学（威海）

## 本章内容

1. 收集和整理多元数据
2. 数据矩阵和数据框
3. 多元数据的数学表达
4. 多元数据的R语言调用

# 1. 收集和整理多元数据

3

## 收集多元数据

- **公开数据集**：查找政府、学术机构和其他组织提供的公开数据。
- **调查问卷**：设计并分发问卷收集原始数据。
- **访谈**：进行面对面的或在线访谈，收集定性数据。
- **在线抓取**：使用网络爬虫技术从网站上提取数据。

## 数据变量类型：按数据性质划分

### • 定量变量 (Quantitative Variables)

- 离散变量 (Discrete Variables)：只能取特定值，通常为整数。  
例如，学生人数、投票数。
- 连续变量 (Continuous Variables)：可以在某个范围内取任意值。例如，身高、体重、时间。

### • 定性变量 (Qualitative Variables)

- 名义变量 (Nominal Variables)：没有内在顺序的分类变量。  
例如，性别、颜色、城市。
- 顺序变量 (Ordinal Variables)：有内在顺序但没有固定间隔的变量。例如，满意度评分（满意、中立、不满意）。

5

单选题 1分

以下哪个变量属于离散变量？

- ☐ A 身高
- ☒ B 学生人数
- ☐ C 温度
- ☐ D 时间

6

单选题 1分

哪种变量类型没有内在顺序?

- ☐ A 连续变量
- ☐ B 离散变量
- ☐ C 顺序变量
- ☒ D 名义变量

7

单选题 1分

身高属于哪种类型的变量?

- ☐ A 离散变量
- ☐ B 名义变量
- ☒ C 连续变量
- ☐ D 顺序变量

8



单选题 1分

以下哪个例子是顺序变量?

- ☐ A 颜色
- ☒ B 满意度评分
- ☐ C 城市
- ☐ D 投票数

9

单选题 1分

关于名义变量的描述，以下哪项是正确的？

- ☒ A 具有固定的数值范围
- ☐ B 具有内在顺序
- ☐ C 仅用于分类，没有顺序
- ☐ D 可以取任意实数

10

## 整理数据

- 数据清洗:

- 去除重复项: 确保数据的唯一性。
- 处理缺失值: 选择填补缺失值的方法 (如均值、中位数填补等) 或删除缺失数据的记录。

- 数据转换:

- 归一化和标准化: 将数据缩放到同一范围, 便于比较。
- 编码分类变量: 将分类数据转换为数值形式, 以便进行分析 (如独热编码)。

11

## R语言实现

【例】下面我将通过一个简单的示例，介绍如何使用R语言进行多元数据整理，特别是数据清洗和数据转换的操作。

Name	Age	Score	Gender
Alice	23	85	F
Bob	25	NA	M
Alice	23	85	F
Charlie	NA	90	M
Dave	30	95	M
Eva	29	NA	F

12

## 步骤 1: 去除重复项distinct()

```
library(dplyr)
data <- data.frame(
  Name = c("Alice", "Bob", "Alice", "Charlie",
            "Dave", "Eva"),
  Age = c(23, 25, 23, NA, 30, 29),
  Score = c(85, NA, 85, 90, 95, NA),
  Gender = c("F", "M", "F", "M", "M", "F")
)
# 去除重复项
data_unique <- distinct(data)
# 结果
print(data_unique)
```

13

## 步骤 1: 去除重复项distinct()

```
> print(data_unique)
```

	Name	Age	Score	Gender
1	Alice	23	85	F
2	Bob	25	NA	M
3	Charlie	NA	90	M
4	Dave	30	95	M
5	Eva	29	NA	F

14

## 步骤 2: 处理缺失值

```
# 填补缺失值
data_cleaned <- data_unique %>%
  mutate(
    Age = ifelse(is.na(Age), mean(Age, na.rm = TRUE),
Age),
    Score = ifelse(is.na(Score), mean(Score, na.rm =
TRUE), Score)
  )

# 结果
print(data_cleaned)
```

15

## 步骤 2: 处理缺失值

```
> print(data_cleaned)
```

	Name	Age	Score	Gender
1	Alice	23.00	85	F
2	Bob	25.00	90	M
3	Charlie	26.75	90	M
4	Dave	30.00	95	M
5	Eva	29.00	90	F

16



### 步骤3：归一化和标准化

```
# 标准化
data_normalized <- data_cleaned %>%
  mutate(
    Age = scale(Age),
    Score = scale(Score)
  )

# 结果
print(data_normalized)
```

17

### 步骤3：归一化和标准化

```
> print(data_normalized)
```

	Name	Age	Score	Gender
1	Alice	-1.3105561	-1.414214	F
2	Bob	-0.6115928	0.000000	M
3	Charlie	0.0000000	0.000000	M
4	Dave	1.1358153	1.414214	M
5	Eva	0.7863337	0.000000	F

18

## 步骤4: 编码分类变量

```
# 独热编码
data_encoded <- cbind(data_normalized,
  model.matrix(~ Gender - 1, data=data_normalized))
data_encoded$Gender <- NULL

# 结果
print(data_encoded)
```

19

## 步骤4: 编码分类变量

```
> print(data_encoded)
```

	Name	Age	Score	GenderF	GenderM
1	Alice	-1.3105561	-1.414214	1	0
2	Bob	-0.6115928	0.000000	0	1
3	Charlie	0.0000000	0.000000	0	1
4	Dave	1.1358153	1.414214	0	1
5	Eva	0.7863337	0.000000	1	0

20

单选题 1分

在数据清洗过程中，去除重复项的主要目的是为了：

- ☐ A 增加数据的复杂性
- ☒ B 确保数据的唯一性
- ☐ C 随机删除部分数据
- ☐ D 改变数据的类型

21

单选题 1分

数据归一化和标准化的主要目的是：

- ☐ A 增加数据的维度
- ☒ B 将数据缩放到同一范围，便于比较
- ☐ C 删除重复记录
- ☐ D 改变数据的格式

22

单选题 1分

将分类数据转换为数值形式的过程称为：

- ☐ A 数据清洗
- ☒ B 数据转换
- ☐ C 数据归一化
- ☐ D 数据合并

23

单选题 1分

在数据清洗中，什么情况下使用均值填补缺失值是合适的？

- ☐ A 数据分布为偏态分布时
- ☒ B 数据分布为正态分布时
- ☐ C 缺失值占比超过50%时
- ☐ D 数据为分类变量时

24



## 2. 数据矩阵和数据框

25

## 数据矩阵及R语言表示

【例 2.2】测得 12 名学生的生长发育指标：身高(x1)、体重(x2)的数据，  
试用 R 语言表述该数据。

x1	171, 175, 159, 155, 152, 158, 154, 164, 168, 166, 159, 164
x2	57, 64, 41, 38, 35, 44, 41, 51, 57, 49, 47, 46

在R中可以用函数c()来创建向量：

```
x1 <- c(171, 175, 159, 155, 152, 158, 154, 164, 168, 166, 159, 164)
x2 <- c(57, 64, 41, 38, 35, 44, 41, 51, 57, 49, 47, 46)
```

在R中结果输出如下：

```
> x1 <- c(171, 175, 159, 155, 152, 158, 154, 164, 168, 166, 159, 164)
> x2 <- c(57, 64, 41, 38, 35, 44, 41, 51, 57, 49, 47, 46)
> x1
[1] 171 175 159 155 152 158 154 164 168 166 159 164
> x2
[1] 57 64 41 38 35 44 41 51 57 49 47 46
```

26

## 数据矩阵及R语言表示

#将向量按列合并

**cbind(x1,x2)**

> cbind(x1,x2)

	x1	x2
[1,]	171	57
[2,]	175	64
[3,]	159	41
[4,]	155	38
[5,]	152	35
[6,]	158	44
[7,]	154	41
[8,]	164	51
[9,]	168	57
[10,]	166	49
[11,]	159	47
[12,]	164	46

#利用x1数据创建矩阵

**matrix(x1,nrow=3,ncol=4)**

> matrix(x1,nrow=3,ncol=4)

	[,1]	[,2]	[,3]	[,4]
[1,]	171	155	154	166
[2,]	175	152	164	159
[3,]	159	158	168	164

#创建按照行排列的矩阵

**matrix(x1,nrow=3,ncol=4, byrow=T)**

> matrix(x1,nrow=3,ncol=4,byrow=T)

	[,1]	[,2]	[,3]	[,4]
[1,]	171	175	159	155
[2,]	152	158	154	164
[3,]	168	166	159	164

27

## 数据矩阵及R语言表示

#创建两个相同的矩阵

```
A=B=matrix(1:12,nrow=3,ncol=4)
```

```
> A;B
```

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

#矩阵转置

```
t(A)
```

```
> t(A)
```

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

#矩阵加法

```
A+B
```

```
> A+B
```

```
      [,1] [,2] [,3] [,4]  
[1,]     2     8    14    20  
[2,]     4    10    16    22  
[3,]     6    12    18    24
```

#矩阵减法

```
A-B
```

```
> A-B
```

```
      [,1] [,2] [,3] [,4]  
[1,]     0     0     0     0  
[2,]     0     0     0     0  
[3,]     0     0     0     0
```

28

## 数据矩阵及R语言表示

#矩阵相乘

```
A=matrix(1:12,nrow=3,ncol=4)
```

```
B=matrix(1:12,nrow=4,ncol=3)
```

```
A%*%B
```

```
      [,1] [,2] [,3]  
[1,]   70  158  246  
[2,]   80  184  288  
[3,]   90  210  330
```

#获取对角线元素

```
A=matrix(1:16,nrow=4,ncol=4)
```

```
diag(A)
```

```
> diag(A)
```

```
[1]  1  6 11 16
```

#利用对角线元素创建对角矩阵

```
diag(diag(A))
```

```
> diag(diag(A))
```

```
      [,1] [,2] [,3] [,4]  
[1,]    1    0    0    0  
[2,]    0    6    0    0  
[3,]    0    0   11    0  
[4,]    0    0    0   16
```

#创建3阶单位矩阵

```
diag(3)
```

```
> diag(3)
```

```
      [,1] [,2] [,3]  
[1,]    1    0    0  
[2,]    0    1    0  
[3,]    0    0    1
```

29



## 数据矩阵及R语言表示

#求逆矩阵

```
A=matrix(rnorm(16), 4, 4)
```

```
solve(A)
```

```
> solve(A)
```

	[,1]	[,2]	[,3]	[,4]
[1,]	0.2485820	-0.35092701	0.19955797	-0.03945507
[2,]	-1.0308041	-0.18913835	-0.03197376	0.90517759
[3,]	-0.9322897	0.22579897	0.25031260	0.46842094
[4,]	2.3471280	0.08939981	-0.36256217	-0.26584474

#求矩阵特征根与特征向量

```
A=diag(4)+1
```

```
A.e=eigen(A, symmetric=T)
```

```
$values
```

```
[1] 5 1 1 1
```

```
$vectors
```

	[,1]	[,2]	[,3]	[,4]
[1,]	-0.5	0.000000e+00	0.0000000	0.8660254
[2,]	-0.5	-6.408849e-17	0.8164966	-0.2886751
[3,]	-0.5	-7.071068e-01	-0.4082483	-0.2886751
[4,]	-0.5	7.071068e-01	-0.4082483	-0.2886751

#矩阵的Choleskey分解

```
A.c=chol(A)
```

	[,1]	[,2]	[,3]	[,4]
[1,]	1.414214	0.7071068	0.7071068	0.7071068
[2,]	0.000000	1.2247449	0.4082483	0.4082483
[3,]	0.000000	0.0000000	1.1547005	0.2886751
[4,]	0.000000	0.0000000	0.0000000	1.1180340

30

## 数据矩阵及R语言表示

#矩阵奇异值分解

```
A=matrix(1:18,3,6)
```

```
A.s=svd(A)
```

```
$d
```

```
[1] 4.589453e+01 1.640705e+00 3.627301e-16
```

```
$u
```

```
      [,1]      [,2]      [,3]  
[1,] -0.5290354  0.74394551  0.4082483  
[2,] -0.5760715  0.03840487 -0.8164966  
[3,] -0.6231077 -0.66713577  0.4082483
```

```
$v
```

```
      [,1]      [,2]      [,3]  
[1,] -0.07736219 -0.71960032 -0.18918124  
[2,] -0.19033085 -0.50893247  0.42405898  
[3,] -0.30329950 -0.29826463 -0.45330031  
[4,] -0.41626816 -0.08759679 -0.01637004  
[5,] -0.52923682  0.12307105  0.64231130  
[6,] -0.64220548  0.33373889 -0.40751869
```

#矩阵的维数

```
A=matrix(1:12,3,4)
```

```
dim(A)
```

```
[1] 3 4
```

#矩阵的行数

```
nrow(A)
```

```
[1] 3
```

#矩阵的列数

```
ncol(A)
```

```
[1] 4
```

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

31

## 数据矩阵及R语言表示

#矩阵按行求和

`rowSums(A)`

```
[1] 22 26 30
```

#矩阵按行求均值

`rowMeans(A)`

```
[1] 5.5 6.5 7.5
```

#矩阵按列求和

`colSums(A)`

```
[1] 6 15 24 33
```

#矩阵按列求均值

`colMeans(A)`

```
[1] 2 5 8 11
```

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

### apply()函数

`apply(X, MARGIN, FUN, ...)`

#矩阵按行求和

`apply(A, 1, sum)`

```
[1] 22 26 30
```

#矩阵按行求均值

`apply(A, 1, mean)`

```
[1] 5.5 6.5 7.5
```

32



## 数据矩阵及R语言表示

#矩阵按列求和

```
apply(A, 2, sum)
```

```
[1] 6 15 24 33
```

#矩阵按列求均值

```
apply(A, 2, mean)
```

```
[1] 2 5 8 11
```

#矩阵按列求方差

```
A=matrix(rnorm(100), 20, 5)
```

```
apply(A, 2, var)
```

```
[1] 1.2748524 1.8964186 1.2920973 0.6991467 0.5818300
```

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

#矩阵按列求函数结果

```
A=matrix(1:12, 3, 4)
```

```
apply(A, 2, function(x, a) x*a, a=2)
```

```
      [,1] [,2] [,3] [,4]
[1,]    2    8   14   20
[2,]    4   10   16   22
[3,]    6   12   18   24
```

注意:

`apply(A, 2, function(x, a) x*a, a=2)` 与 `A*2` 效果相同, 此处旨在说明如何应用 `apply` 函数。



33

单选题 1分

在R中，如何创建一个矩阵？

- ☒ A `matrix(data, nrow, ncol)`
- ☐ B `mat(data, nrow, ncol)`
- ☐ C `create.matrix(data, nrow, ncol)`
- ☐ D `data.matrix(data)`

34

单选题 1分

在R中，如何访问矩阵A的第2行第3列的元素？

- ☐ A  $A[2, 3]$
- ☒ B  $A[2][3]$
- ☐ C  $A[3, 2]$
- ☐ D  $A(2, 3)$

35

单选题 1分

在R中，如何对矩阵A进行转置？

- ☒ A `t(A)`
- ☐ B `transpose(A)`
- ☐ C `A.t()`
- ☐ D `rev(A)`

多选题 1分

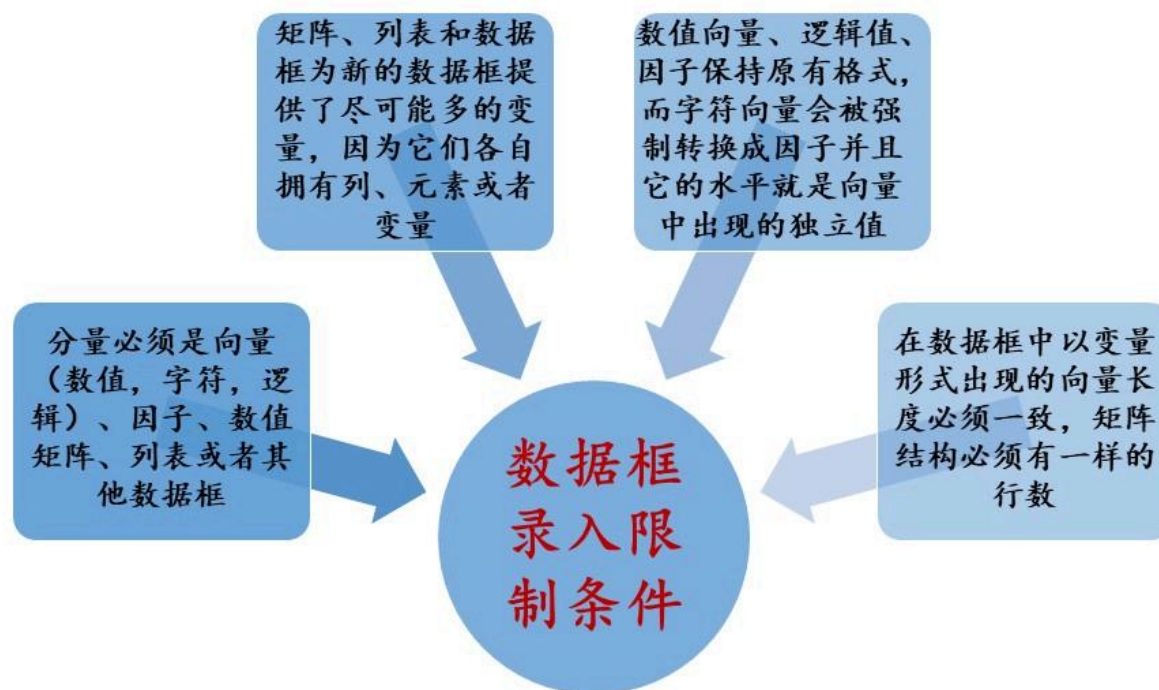
使用哪个函数可以在R中计算矩阵A的行和?

- ☒ A rowSums(A)
- ☐ B sumRows(A)
- ☒ C apply(A, 1, sum)
- ☐ D apply(A, 2, sum)

37

# 数据框

——一种矩阵形式的数据，但其各列可以是不同类型的数据



38

## 数据框

#由x1和x2构建数据框

```
X=data.frame(x1,x2)
```

	x1	x2
1	171	57
2	175	64
3	159	41
4	155	38
5	152	35
6	158	44
7	154	41
8	164	51
9	168	57
10	166	49
11	159	47
12	164	46

#赋予数据框新的列标签

```
X=data.frame('身高'=x1,'体重'=x2)
```

	身高	体重
1	171	57
2	175	64
3	159	41
4	155	38
5	152	35
6	158	44
7	154	41
8	164	51
9	168	57
10	166	49
11	159	47
12	164	46

39

单选题 1分

在R中，如何创建一个包含多个变量的数据框（data frame）？

- ☒ A `data.frame(var1, var2)`
- ☐ B `c(var1, var2)`
- ☐ C `matrix(var1, var2)`
- ☐ D `list(var1, var2)`

40



### 3. 多元数据的数学表达

41

# 期望

## 总体期望（均值）

一元数据	多元数据 ( $p$ 维向量 $\mathbf{X}$ )
标量: $\mu = E[X]$	向量: $\boldsymbol{\mu} = E[\mathbf{X}] = (E[X_1], \dots, E[X_p])^T$
单变量的期望。	多变量各分量的期望组成的向量。

## 样本期望（样本均值）

一元数据	多元数据 ( $n$ 个样本)
标量: $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$	向量: $\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$
所有样本点的算术平均。	各分量的样本均值组成的向量。

42

## 方差 vs 协方差矩阵

一元数据 (方差)	多元数据 (协方差矩阵)
标量: $\sigma^2 = E[(X - \mu)^2]$	矩阵: $\Sigma = E[(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^T]$
描述单变量的离散程度。	描述多变量间的线性关系: <ul style="list-style-type: none"><li>- 对角线元素为各变量方差,</li><li>- 非对角线元素为协方差 <math>\text{Cov}(X_j, X_k)</math>。</li></ul>

## 样本方差 vs 样本协方差矩阵

一元数据 (样本方差)	多元数据 (样本协方差矩阵)
标量: $s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$	矩阵: $\mathbf{S} = \frac{1}{n-1} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$
无偏估计 (分母 $n - 1$ ) 。	无偏估计 (分母 $n - 1$ )

单选题 1分

如果一组一元数据的方差很小，这意味着：

- ☐ A 数据分布广泛
- ☒ B 数据集中在均值附近
- ☐ C 数据均匀分布
- ☐ D 数据存在显著的偏差

45

单选题 1分

样本方差的计算公式中，通常使用哪个值作为分母以获得无偏估计？

- ☐ A  $n$
- ☐ B  $n-1$
- ☒ C  $n+1$
- ☐ D  $1$

单选题 1分

关于多元数据的样本均值，下列说法哪个是正确的？

- ☐ A 它是一个标量值
- ☒ B 它是一个向量，由每个维度的样本均值组成
- ☐ C 它与方差相同
- ☐ D 它不包含任何统计信息

47

单选题 1分

在多元数据中，协方差矩阵的主要作用是：

- ☐ A 计算每个数据点的均值
- ☒ B 衡量各维度间的相关性和变异性
- ☐ C 仅用于一维数据分析
- ☐ D 作为样本均值的替代

48



## 例题

收集了 3 名学生的身高（米）和体重（千克）数据：

$$\mathbf{x}_1 = \begin{pmatrix} 1.6 \\ 50 \end{pmatrix}, \mathbf{x}_2 = \begin{pmatrix} 1.7 \\ 55 \end{pmatrix}, \mathbf{x}_3 = \begin{pmatrix} 1.8 \\ 70 \end{pmatrix}$$

计算样本协方差矩阵，并解释身高与体重的相关性。

解析：

1. 样本均值：

$$\bar{\mathbf{x}} = \begin{pmatrix} \frac{1.6+1.7+1.8}{3} \\ \frac{50+55+70}{3} \end{pmatrix} = \begin{pmatrix} 1.7 \\ 58.33 \end{pmatrix}$$

2. 中心化数据：

$$\mathbf{x}_1 - \bar{\mathbf{x}} = \begin{pmatrix} -0.1 \\ -8.33 \end{pmatrix}, \quad \mathbf{x}_2 - \bar{\mathbf{x}} = \begin{pmatrix} 0 \\ -3.33 \end{pmatrix}, \quad \mathbf{x}_3 - \bar{\mathbf{x}} = \begin{pmatrix} 0.1 \\ 11.67 \end{pmatrix}$$

3. 外积求和：

$$\sum_{i=1}^3 (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T = \begin{bmatrix} 0.01 & 0.833 \\ 0.833 & 69.44 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 11.09 \end{bmatrix} + \begin{bmatrix} 0.01 & 1.167 \\ 1.167 & 136.11 \end{bmatrix} = \begin{bmatrix} 0.02 & 2.0 \\ 2.0 & 216.64 \end{bmatrix}$$

4. 标准化 (分母  $n - 1 = 2$ ) :

$$\mathbf{S} = \frac{1}{2} \begin{bmatrix} 0.02 & 2.0 \\ 2.0 & 216.64 \end{bmatrix} = \begin{bmatrix} 0.01 & 1.0 \\ 1.0 & 108.32 \end{bmatrix}$$

5. 解释：

- 协方差  $1.0 > 0$ ，说明身高与体重呈正相关（身高越高，体重可能越大）。

50

## 例题（样本协方差矩阵的无偏性）

证明：样本协方差矩阵  $\mathbf{S}$  的无偏性，即  $E[\mathbf{S}] = \mathbf{\Sigma}$ 。

解析（关键步骤）：

1. 定义样本协方差矩阵：

$$\mathbf{S} = \frac{1}{n-1} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$$

2. 展开并取期望：

$$E[\mathbf{S}] = \frac{1}{n-1} E \left[ \sum_{i=1}^n (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T - n(\bar{\mathbf{x}} - \boldsymbol{\mu})(\bar{\mathbf{x}} - \boldsymbol{\mu})^T \right]$$

3. 利用线性性质化简：

$$E[(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T] = \mathbf{\Sigma}, \quad E[(\bar{\mathbf{x}} - \boldsymbol{\mu})(\bar{\mathbf{x}} - \boldsymbol{\mu})^T] = \frac{1}{n} \mathbf{\Sigma}$$

4. 合并得：

$$E[\mathbf{S}] = \frac{1}{n-1} \left( n\mathbf{\Sigma} - n \cdot \frac{1}{n} \mathbf{\Sigma} \right) = \mathbf{\Sigma}$$

51

# R语言实现

一元数据示例

数据定义

```
# 一元数据  
x <- c(5, 6, 7, 8, 9)
```

样本均值 (Sample Mean)

```
sample_mean_x <- mean(x) # 计算样本均值  
sample_mean_x
```

结果:

```
[1] 7
```

样本方差 (Sample Variance)

```
var_x <- var(x) # 计算样本方差  
var_x
```

结果:

```
[1] 2.5
```

52

# R语言实现

多元数据示例

数据定义

```
# 多元数据
X <- matrix(c(23, 25, 30, 85, 90, 95), nrow = 3, byrow = TRUE)
```

样本均值 (Sample Mean Vector)

```
sample_mean_X <- colMeans(X) # 计算每列的样本均值
sample_mean_X
```

结果:

```
[1] 26 90
```

样本协方差 (Sample Covariance Matrix)

```
sample_cov_X <- cov(X) # 计算样本协方差矩阵
sample_cov_X
```

结果:

```
      [,1]      [,2]
[1,] 5.000000 5.000000
[2,] 5.000000 25.000000
```

53

## 4. 多元数据的R语言调用

54

## 从剪切板读取

选择需要进行计算的数据块，拷贝之

01

在R中使用 `dat <- read.table("clipboard", header=T)`

02

55

## 从文本文件读取



```
X=read.table('textdata.txt', header=T)
```

第一行作为标题时

#读取名为textdata的txt格式文档

```
X=read.table("textdata.txt")
```

	V1	V2
1	x1	x2
2	171	57
3	175	64
4	159	41
5	155	38
6	152	35
7	158	44
8	154	41
9	164	51
10	168	57
11	166	49
12	159	47
13	164	46

56



## 读取csv格式和excel格式



1. 下载读取excel文件的包 “readxl”
2. 调用包：library(readxl)
3. 读取文件：X=read\_excel(“data.xls”)

也可以使用包 “openxlsx”

1. 下载读取excel文件的包 “openxlsx”
2. 调用包：library(openxlsx)
3. 读取文件：X=read.xlsx(“data.xlsx”)

X=read.csv("textdata.csv")

57

## 本章小结

1. 收集和整理多元数据
2. 数据矩阵和数据框
3. 多元数据的数学表达
4. 多元数据的R语言调用