Livestock Breeding and Genomics - Solution 2

Peter von Rohr

2022-09-29

Problem 1: Vectors in R

Vector Definition

Although there exists a function called `vector()` in R, vectors are always defined in R using the function `c()` which stands for "concatenation".

Vector Assignment

Let us assume we want to assign the following vector a

$$a = \left[\begin{array}{c} 10 \\ 7 \\ 43 \end{array}\right]$$

to the variable named `a` in R, then this can be done with the following statement

```
a <- c(10,7,43)
```

Access of single Vector Element

A single vector element can be accessed using the variable name followed by the element index in brackets. Hence, if we want to know the first element of vector `a`, we have to write

```
a[1]
```

```
## [1] 10
```

Computations with Vector Elements

Vector elements can be used in arithmetic operations such as summation, subtraction and multiplication as shown below

```
a[1] + a[3]
```

```
## [1] 53
```

```r
a[2] * a[3]
```

```
## [1] 301
```

```r
a[3] - a[1]
```

```
## [1] 33
```

The function `sum()` can be used to compute the sum of all vector elements. The function `mean()` computes the mean of all vector elements.

```r
sum(a)
```

```
## [1] 60
```

```r
mean(a)
```

```
## [1] 20
```

Vector Computations

Arithmetic operations can also be performed not only on elements of vectors but also on complete vectors. Hence, we can add the vector `a` to itself or we can multiply it by a factor of 3.5 which is shown in the following code-chunk

```r
a + a
```

```
## [1] 20 14 86
```

```r
3.5 * a
```

```
## [1]  35.0  24.5 150.5
```

More Computations on Vectors

Given are the following two vectors v and w.

$$v = \begin{bmatrix} 3 \\ -5 \\ 1 \\ 9 \end{bmatrix}$$

$$w = \begin{bmatrix} 1 \\ 9 \\ -12 \\ 27 \end{bmatrix}$$

Compute

- the sum $v + w$,
- the difference $v - w$ and
- the dot product $v \cdot w$.

Solution

```
v <- c(3, -5, 1, 9)
w <- c(1, 9, -12, 27)
```

Now we do the computations.

The sum $v + w$ is

```
v+w
```

```
## [1]   4   4 -11  36
```

The difference $v - w$ is

```
v-w
```

```
## [1]   2 -14  13 -18
```

and the dot-product is

```
crossprod(v,w)
```

```
##      [,1]
## [1,]  189
```

or

```
v %*% w
```

```
##      [,1]
## [1,]  189
```

Please note: Although the R-function is called `crossprod()` what is computed is the dot product between the two vectors. The function name `crossprod()` is used because in Statistics the product (X^TX) of a transposed matrix (X^T) and itself (X) is called a matrix crossproduct. This has nothing to do with the crossproduct $v \times w$ between two vectors v and w.

Problem 2: Matrices in R

Matrices in R are defined using the function `matrix()`. The function `matrix()` takes as first arguments all the elements of the matrix as a vector and as further arguments the number of rows and the number of columns. The following statment generates a matrix with 4 rows and 3 columns containing all integer numbers from 1 to 12.

```
mat_by_col <- matrix(1:12, nrow = 4, ncol = 3)
mat_by_col
```

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

As can be seen, the matrix elements are ordered by columns. Often, we want to define a matrix where elements are filled by rows. This can by done using the option `byrow=TRUE`

```
mat_by_row <- matrix(1:12, nrow = 4, ncol = 3, byrow = TRUE)
mat_by_row
```

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

Access of Matrix Elements

Matrix elements can be accessed similarly to what was shown for vectors. But to access a single element, we need two indices, one for rows and one for columns. Hence the matrix element in the second row and third column can be accessed by

```
mat_by_row[2,3]
```

```
## [1] 6
```

Arithmetic Computations with Matrices

Arithmetic computations with matrices can be done with the well-known operators as long as the matrices are compatible. For summation and subtraction matrices must have the same number of rows and columns. For matrix-multiplication, the number of columns of the first matrix must be equal to the number of rows of the second matrix.

In R the arithmetic operators `+`, `-` and `*` all perform element-wise operations. The matrix multiplication can either be done using the operator `%*%` or the function `crossprod()`. It has to be noted that the statement

```
crossprod(A, B)
```

computes the matrix-product $A^T \cdot B$ where A^T stands for the transpose of matrix A. Hence the matrix product $A \cdot B$ would have to be computed as

```
crossprod(t(A), B)
```

More Examples

Given the matrices X and Y

```r
X <- matrix(1:15, nrow = 5, ncol = 3)
Y <- matrix(16:30, nrow = 5, ncol = 3)
```

Compute

- $X + Y$
- $Y - X$
- multiplication of elements between X and Y
- matrix-product $X^T \cdot Y$
- matrix-product $X^T \cdot X$
- matrix-product $Y^T \cdot Y$

Solution

```r
X + Y
```

```
##      [,1] [,2] [,3]
## [1,]   17   27   37
## [2,]   19   29   39
## [3,]   21   31   41
## [4,]   23   33   43
## [5,]   25   35   45
```

```r
Y - X
```

```
##      [,1] [,2] [,3]
## [1,]   15   15   15
## [2,]   15   15   15
## [3,]   15   15   15
## [4,]   15   15   15
## [5,]   15   15   15
```

```r
X * Y
```

```
##      [,1] [,2] [,3]
## [1,]   16  126  286
## [2,]   34  154  324
## [3,]   54  184  364
## [4,]   76  216  406
## [5,]  100  250  450
```

```r
crossprod(X, Y)
```

```
##      [,1] [,2] [,3]
## [1,]  280  355  430
## [2,]  730  930 1130
## [3,] 1180 1505 1830
```

```
crossprod(X)
```

```
##      [,1] [,2] [,3]
## [1,]   55  130  205
## [2,]  130  330  530
## [3,]  205  530  855
```

```
crossprod(Y)
```

```
##      [,1] [,2] [,3]
## [1,] 1630 2080 2530
## [2,] 2080 2655 3230
## [3,] 2530 3230 3930
```