

Züchtungslehre - Lösung 2

Peter von Rohr

2016-10-07

Aufgabe 1: Matrixdefinitionen in R

In R werden Matrizen mit der Funktion `matrix` erstellt. In der Vorlesung hatten wir gesehen, dass die Funktion `matrix()` verschiedene Optionen akzeptiert. Wir wollen uns hier anschauen, wie sich die Parameter auswirken.

Ihre Aufgabe wird es sein die Matrizen mit den verschiedenen Optionen zu erstellen und so besser zu verstehen, was die Optionen für eine Bedeutung haben.

Parameter data

- **data:** Angabe der Matrix-Elemente

```
(matA <- matrix(data = c(1:9), nrow = 3, ncol = 3))
```

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

- **data:** Ohne Angabe der Matrix-Elemente

```
(matB <- matrix(nrow = 3, ncol = 3))
```

```
##      [,1] [,2] [,3]
## [1,]   NA   NA   NA
## [2,]   NA   NA   NA
## [3,]   NA   NA   NA
```

- **data:** Spezifikation nicht aller Elemente

```
(matC <- matrix(data = c(1,2,3), nrow = 3, ncol = 3))
```

```
##      [,1] [,2] [,3]
## [1,]    1    1    1
## [2,]    2    2    2
## [3,]    3    3    3
```

```
(matC2 <- matrix(data = c(1,2,3,4), nrow = 3, ncol = 3))
```

```
## Warning in matrix(data = c(1, 2, 3, 4), nrow = 3, ncol = 3): data length
## [4] is not a sub-multiple or multiple of the number of rows [3]
```

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

Parameter nrow und ncol

- Weglassen einer der beiden Parameter

```
(matD <- matrix(data = c(1:9), nrow = 3))
```

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

```
(matE <- matrix(data = c(1:9), ncol = 3))
```

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

Parameter byrow

```
(matF <- matrix(data = c(1:9), nrow = 3, ncol = 3, byrow = TRUE))
```

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

```
(matG <- matrix(data = c(1:9), nrow = 3, ncol = 3, byrow = FALSE))
```

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

Aufgabe 2: Matrixmultiplikation in R

Matrixmultiplikationen können in R mit dem Operator `%*%` oder mit den Funktionen `crossprod()` oder `tcrossprod()` ausgeführt werden. Gegeben sind die folgenden Matrizen

```
matA <- matrix(data = c(1:9), ncol = 3)
```

Aufgabe 3: Gleichungssysteme

Gegeben ist das folgende Gleichungssystem.

$$\begin{aligned} 2x_2 + 2x_3 &= 1 \\ 2x_1 + 4x_2 + 5x_3 &= 9 \\ x_1 - x_2 + 2x_3 &= 3 \end{aligned} \tag{1}$$

- a) Bestimmen Sie die Lösungsmenge des Gleichungssystems (1) mit dem Gaussverfahren

Lösung

- Vertauschen der ersten und der zweiten Gleichung

$$2x_1 + 4x_2 + 5x_3 = 9$$

$$2x_2 + 2x_3 = 1$$

$$x_1 - x_2 + 2x_3 = 3$$

- 1/2-fache der ersten Gleichung von dritter abziehen

$$2x_1 + 4x_2 + 5x_3 = 9$$

$$2x_2 + 2x_3 = 1$$

$$-3x_2 - \frac{1}{2}x_3 = -\frac{3}{2}$$

- Addition des 3/2-fache der zweiten zur dritten Gleichung

$$2x_1 + 4x_2 + 5x_3 = 9$$

$$2x_2 + 2x_3 = 1$$

$$\frac{5}{2}x_3 = 0$$

Somit ist $x_3 = 0$.

- Rückwärtseinsetzen in der zweiten Gleichung führt zu $x_2 = 1/2$. Aufgrund der ersten Gleichung folgt $x_1 = 7/2$.

b) Verwandeln Sie das Gleichungssystem (1) in Matrix-Vektor-Schreibweise

Lösung

$$A \cdot x = b$$

wobei die sogenannte Koeffizientenmatrix A , der Vektor x und die rechte Handseite b wie folgt definiert sind

$$A = \begin{bmatrix} 0 & 2 & 2 \\ 2 & 4 & 5 \\ 1 & -1 & 2 \end{bmatrix}$$

,

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

und

$$b = \begin{bmatrix} 1 \\ 9 \\ 3 \end{bmatrix}$$

c) Wie lautet die Lösung des Gleichungssystem (1) in Matrix-Vektor-Schreibweise

$$x = A^{-1} \cdot b$$

d) Berechnen Sie die Lösung aus c) mit R

Hinweis * Für die Multiplikation der Matrix A^{-1} mit dem Vektor b muss der Vektor b in eine Matrix verwandelt werden. Dies kann mit der Funktion `as.matrix()` gemacht werden.

Lösung

```
matA <- matrix(data = c(0,2,2,2,4,5,1,-1,2),nrow = 3,ncol = 3, byrow = TRUE)
matAInv <- solve(matA)
vecB <- c(1,9,3)
sol <- matAInv %*% as.matrix(vecB)
print(sol)
```

```
##           [,1]
## [1,] 3.500000e+00
## [2,] 5.000000e-01
## [3,] 2.220446e-16
```