(1) Matrizen

Ziel dieses Seminar ist es, die Matrizenrechnung in **R** anhand der in der Vorlesung diskutierten Beispiele nachzuvollziehen und optional weitere Übungsaufgaben mithilfe von **R** zu lösen.

Matrizenrechnung in R

Wir betrachten zunächst die spaltenweise und zeilenweise Definition von Matrizen in **R** anhand zweier in der Vorlesung gegebener Beispiele zur Matrixaddition und Matrixsubtraktion. Man beachte, dass die (non-default) zeilenweise Definition einer Matrix mithilfe von `byrow = TRUE` eine höhere Korrespondenz zwischen **R** Codebild und **R** Repräsentation ermöglicht.

```r
# Spaltenweise Definition von A (R default)
A = matrix(c(2,1,-3,6,0,5), nrow = 2)
print(A)
```

```
     [,1] [,2] [,3]
[1,]    2   -3    0
[2,]    1    6    5
```

```r
# Zeilenweise Definition von B
B = matrix(c(4,1,0,
            -4,2,0), nrow = 2, byrow = TRUE)
print(B)
```

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

Die Addition und Subtraktion von Matrizen werden in **R** dann mit den Operatoren `+` und `-` implementiert. Entsprechend der analytischen Betrachtung in der Vorlesung ergeben sich

```r
# Addition
C = A + B
print(C)
```

```
     [,1] [,2] [,3]
[1,]    6   -2    0
[2,]   -3    8    5
```

und

```
# Subtraktion
D = A - B
print(D)
```

```
     [,1] [,2] [,3]
[1,]   -2   -4    0
[2,]    5    4    5
```

Das in der Vorlesung betrachtete Beispiel zur Skalarmultiplikation ergibt sich mit dem Skalarmultiplikationsoperator `*` wie folgt.

```
# Definitionen
A = matrix(c(3,1,1,
             5,2,5,
             2,7,1,
             3,4,2),
           nrow = 4,
           byrow = TRUE)
c = -3

# Skalarmultiplikation
B = c*A
print(B)
```

```
     [,1] [,2] [,3]
[1,]   -9   -3   -3
[2,]  -15   -6  -15
[3,]   -6  -21   -3
[4,]   -9  -12   -6
```

Das in der Vorlesung betrachtete Beispiel zur Matrixtransposition schließlich ergibt sich mit dem Transpositionsoperator `t()` wie folgt.

```
# Definition
A = matrix(c(2,3,0,
             1,6,5),
           nrow = 2,
           byrow = TRUE)

# Transposition
AT = t(A)
print(A)
```

```
     [,1] [,2] [,3]
[1,]    2    3    0
[2,]    1    6    5
```

```
print(AT)
```

```
     [,1] [,2]
[1,]    2    1
[2,]    3    6
[3,]    0    5
```

Das erste in der Vorlesung betrachtete Beispiel zur Matrixmultiplikation implementiert man in **R** mithilfe des Matrixmultiplikationsoperators `%*%` wie folgt.

```
# Definitionen
A = matrix(c(2,-3,0,
             1, 6,5),
           nrow  = 2,
           byrow = TRUE)
B = matrix(c( 4,2,
             -1,0,
              1,3),
           nrow  = 3,
           byrow = TRUE)

# Matrixmultiplikation
C = A %*% B
print(C)
```

```
     [,1] [,2]
[1,]   11    4
[2,]    3   17
```

Das zweite Beispiel zur Matrixmultiplikation ergibt sich als

```
# Definitionen
A = matrix(c(2,-3,0,
             1, 6,5),
           nrow  = 2,
           byrow = TRUE)
B = matrix(c( 4,2,
             -1,0,
              1,3),
           nrow  = 3,
           byrow = TRUE)

# Matrixmultiplikation
D = B %*% A
print(D)
```

```
     [,1] [,2] [,3]
[1,]   10    0   10
[2,]   -2    3    0
[3,]    5   15   15
```

Sind in **R** definierte Matrizen nicht von für die Definition der Matrixmultiplikation erforderlichen Größen, so ergibt sich ein Fehler.

```
# Beispiel für eine undefinierte Matrixmultipliation
E = t(A) %*% B      # (3 x 2)(3 x 2)
```

```
Error in t(A) %*% B: nicht passende Argumente
```

Inverse Matrizen berechnet man in **R** für gewöhnlich mit dem Befehl `solve()`. Für das in der Vorlesung betrachtete Beispiel einer invertierbaren 2×2 Matrix ergibt sich entsprechend folgender **R** Code.

```
# Definition
A = matrix(c(2,1,
             3,4),
           nrow  = 2,
           byrow = TRUE)

# Berechnen von A^{-1}
print(solve(A))
```

```
     [,1] [,2]
[1,]  0.8 -0.2
[2,] -0.6  0.4
```

```
# Reproduktion der definierenden Eigenschaft von A^{-1}
print(solve(A) %*% A)
```

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

```
print(A %*% solve(A))
```

```
     [,1]          [,2]
[1,]    1 -5.551115e-17
[2,]    0  1.000000e+00
```

Man beachte, dass sich bei der rechtsseitigen Multiplikation von A mit ihrer inversen minimale Rundungsfehler eingeschlichen haben. Die in der Vorlesung beispielhaft betrachtete nicht-invertierbare Matrix ist auch numerisch nicht invertierbar, wie folgender Beispielcode demonstriert.

```
# Nicht-invertierbare Matrizen sind auch numerisch nicht-invertierbar (singulär)
B = matrix(c(1,0,
             0,0),
           nrow  = 2,
           byrow = 2)
solve(B)
```

```
Error in solve.default(B): Lapackroutine dgesv: System ist genau singulär: U[2,2] = 0
```

Determinanten berechnet man in **R** mithilfe des Befehls `det()`. Für die in der Vorlesung betrachteten Beispiele ergibt sich folgender **R** Code.

```
# Beispiel 1
A = matrix(c(2,1,                   # Matrixdefinition
             3,4),
           nrow = 2,
           byrow = TRUE)
det(A)                              # Determinantenberechnung
```

```
[1] 5
```

```
B = matrix(c(1,0,                   # Matrixdefinition
             0,0),
           nrow = 2,
           byrow = TRUE)
det(B)                              # Determinantenberechnung
```

```
[1] 0
```

```
# Beispiel 2
C = matrix(c(2,0,0,                 # Matrixdefinition
             0,1,0,
             0,0,3),
           nrow = 3,
           byrow = TRUE)
det(C)                              # Determinantenberechnung
```

```
[1] 6
```

Optionale Aufgaben zur Übung

(1) Es seien

$$A := \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}, B := \begin{pmatrix} 3 & 0 \\ 1 & 2 \end{pmatrix}, \text{ und } c := 2 \tag{1}$$

Berechnen Sie

$$D := c\left(A - B^T\right) \text{ und } E := (cA)^T + B. \tag{2}$$

mit **R**.

(2) Es seien

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 3 & 2 & 0 \end{pmatrix} B := \begin{pmatrix} 1 & 2 & 2 \\ 1 & 3 & 1 \\ 2 & 0 & 0 \end{pmatrix} \text{ und } C := \begin{pmatrix} 1 \\ 3 \\ 2 \end{pmatrix}. \tag{3}$$

Berechnen Sie die Matrixprodukte

$$AB, \qquad B^T A^T, \qquad \left(B^T A^T\right)^T \text{ und } AC \tag{4}$$

mit **R**.

(3) Invertieren Sie die Matrizen A und B aus der vorherigen Aufgabe mithilfe von `solve()` und überprüfen Sie die Inverseeigenschaft der inversen Matrizen mithilfe von **R**.

(4) Berechnen Sie die Determinanten von

$$A := \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, B := \begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 2 \\ 1 & 2 & 3 \end{pmatrix} \text{ und } C := \operatorname{diag}(1, 2, 3) \tag{5}$$

mit **R**.