

## Statistische Software (R-Vertiefung)

Paul Fink, M.Sc.

Institut für Statistik  
Ludwig-Maximilians-Universität München

Lineare Algebra



## Matrixoperationen

- Transponieren einer Matrix  $X$ :  $X'$

```
> x <- matrix( nrow=4, ncol=2, data=1:8, byrow=T )
> x
      [,1] [,2]
[1,]    1    2
[2,]    3    4
[3,]    5    6
[4,]    7    8
> xt <- t(x)
> xt
      [,1] [,2] [,3] [,4]
[1,]    1    3    5    7
[2,]    2    4    6    8
```

## Matrixoperationen

- Vorbelegung aller Matrixelemente mit einer bestimmten Zahl

```
> x <- matrix( nrow=4, ncol=2, data=2 )
> x
      [,1] [,2]
[1,]    2    2
[2,]    2    2
[3,]    2    2
[4,]    2    2
```

- Konstruktion einer Diagonalmatrix, hier der Einheitsmatrix der Dimension 2

```
> d <- diag(1, nrow=2, ncol=2)
> d
      [,1] [,2]
[1,]    1    0
[2,]    0    1
```

## Matrixoperationen

- Multiplikation einer Matrix mit einer Konstanten

```
> x <- matrix( nrow=4, ncol=2, data=1:8, byrow=T )
> x
      [,1] [,2]
[1,]    1    2
[2,]    3    4
[3,]    5    6
[4,]    7    8
> 4*x
      [,1] [,2]
[1,]    4    8
[2,]   12   16
[3,]   20   24
[4,]   28   32
```

## Matrixoperationen

- Matrixmultiplikation: Operator `%*%`, hier am Beispiel der Multiplikation von  $X'$  mit  $X$

```
> xtx <- t(x) %*% x
> xtx
      [,1] [,2]
[1,]  84 100
[2,] 100 120
```

- Kreuzprodukt einer Matrix  $X$ ,  $X'X$ , mittels der Funktion `crossprod`

```
> xtx2 <- crossprod(x)
> xtx2
      [,1] [,2]
[1,]  84 100
[2,] 100 120
```

Bemerkung: dieser Befehl führt die Multiplikation schneller aus als die herkömmliche Methode mit `t(x)%*%x`

## Übersicht

Das Rechnen mit Matrizen ist Bestandteil vieler statistischer Methoden. Hier eine Übersicht hilfreicher Befehle zum Rechnen mit Matrizen in R:

<code>matrix</code>	Erstellen einer Matrix
<code>t</code>	Transponieren einer Matrix
<code>%*%</code>	Operator zu Matrixmultiplikation
<code>%o%</code> , <code>outer</code>	Äußeres Produkt
<code>crossprod</code>	Kreuzprodukt
<code>kronecker</code>	Kroneckerprodukt
<code>solve</code>	Invertieren symmetrischer Matrizen
<code>det</code>	Determinante einer quadratischen Matrix
<code>chol</code>	Cholesky-Zerlegung
<code>backsolve</code> , <code>forwardsolve</code>	Lösen von Gleichungssystemen
<code>qr</code>	QR-Zerlegung
<code>eigen</code>	Eigenwerte und Eigenvektoren
<code>svd</code>	Singulärwertzerlegung

## Matrixoperationen

- Addition und Subtraktion von Matrizen (gleicher Dimension!) kann mittels der üblichen Operatoren `+` und `-` durchgeführt werden

- Zugriff auf komplette Zeilen, Spalten oder Blöcke einer Matrix

```
> x <- matrix( nrow=5, ncol=3, byrow=T, data=1:15)
> x
      [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
[4,]   10   11   12
[5,]   13   14   15
> x[3,]
[1] 7 8 9
> x[,2]
[1] 2 5 8 11 14
> x[4:5, 2:3]
      [,1] [,2]
[1,]   11  12
[2,]   14  15
```

## Aufgaben

- Erstellen Sie die Matrix

$$X = \begin{pmatrix} 17 & 25 & 32 \\ 23 & 18 & 12 \\ 10 & 12 & 16 \\ 28 & 156 & 167 \end{pmatrix}$$

- Berechnen Sie  $X'X$  auf 2 Arten! Wie ist die Dimension der Ergebnismatrix?
- Berechnen Sie  $XX'$ ! Wie ist die Dimension der Ergebnismatrix?

- Erstellen Sie mit dem Matrixobjekt einen  $10 \times 1$  Spaltenvektor  $a$ , der die Zahlen 1,3,5, usw. enthält und einen  $1 \times 10$  Zeilenvektor  $b$ , der die Zahlen 2,4,6, usw. enthält.

## Aufgaben

---

- a) Was liefert `a*a`? (Gemeint ist: unter Verwendung der herkömmlichen Multiplikation)
  - b) Was liefert `b'*b'`?
  - c) Berechnen Sie das Matrixprodukt  $ab'$ ! Wie ist die Dimension der Ergebnismatrix?
  - d) Berechnen Sie das Matrixprodukt  $a'b'$  auf 2 Arten! Welche Dimension hat dieses Ergebnis? Ist das Ergebnis vom Typ Matrix?
3. Mittels des Befehls `rnorm()` (siehe Hilfe) lassen sich eine oder mehrere normalverteilte (Pseudo-)Zufallszahlen erzeugen.
- a) Erzeugen Sie eine Liste `l1` von 2 ( $3 \times 3$ ) Matrizen `A1`, `A2`, die mit standardnormalverteilten Zufallsvariablen vorbelegt sind.

## Aufgaben

---

5. Der Befehl `system.time()` erlaubt die Messung der Zeit, die bei der Ausführung eines Befehls oder einer Funktion benötigt wird. Vergleichen Sie die benötigte Zeit Ihrer implementierten Funktion mit dem `R` Operator zur Matrix-Multiplikation `%*%` für verschiedene Dimensionen der Matrizen.

## Aufgaben

---

- b) Erzeugen Sie eine weitere Liste `l2`, die als Elemente jeweils  $A_i'A_i$ ,  $i = 1, 2$  enthält. Mit welchem Befehl erhalten Sie das Element `A2[2,3]`?
  - c) Eine weitere Liste `l3` soll jetzt die Inversen der Matrizen in `l2` enthalten. Hinweis: `solve()`. Testen Sie ihre Ergebnisliste.
4. Schreiben Sie eine eigene Funktion zur Matrix-Multiplikation mittels Schleifen. Testen Sie die Funktion, indem Sie sie mit der in `R` vorhandenen Operation vergleichen.