



# Inverse der Verwandtschaftsmatrix

Peter von Rohr

# Inverse einer Matrix

## Definition

- Gegeben eine quadratische Matrix **A**
- Finde eine quadratische Matrix **B** so, dass gilt

$$\mathbf{B} * \mathbf{A} = \mathbf{A} * \mathbf{B} = \mathbf{I}$$

wobei  $\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 & \dots \\ 0 & 1 & 0 & \dots \\ 0 & 0 & 1 & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix}$  die Einheitsmatrix

# Berechnung der Inversen

## Gauss-Jordan

- Schreibe die gesuchte Matrix

$$\mathbf{B} = [ \mathbf{b}_1 \quad \mathbf{b}_2 \quad \dots \quad \mathbf{b}_n ]$$

und die Einheitsmatrix

$$\mathbf{I} = [ \mathbf{e}_1 \quad \mathbf{e}_2 \quad \dots \quad \mathbf{e}_n ]$$

je als Sequenz von Spaltenvektoren

- Löse die Gleichungssysteme für  $j = 1, \dots, n$

$$\mathbf{A} * \mathbf{b}_j = \mathbf{e}_j$$

## Beispiel für Gauss-Jordan

- Gegeben ist die Matrix  $\mathbf{A} = \begin{bmatrix} 8 & 4 \\ 4 & 3 \end{bmatrix}$
- Folgende Schreibweise für das Lösen der Gleichungen

$$[\mathbf{A}|\mathbf{I}] = \left[ \begin{array}{cc|cc} 8 & 4 & 1 & 0 \\ 4 & 3 & 0 & 1 \end{array} \right]$$

- Elimination von Koeffizienten auf der linken Seite bis links  $\mathbf{I}$  und rechts  $\mathbf{B}$  steht

# Rechenschritte für Gauss-Jordan

- Schritt 1 - Element  $a_{11}$  muss eine 1 sein  $\rightarrow$  erste Zeile durch 8 teilen

$$\left[ \begin{array}{cc|cc} 8 & 4 & 1 & 0 \\ 4 & 3 & 0 & 1 \end{array} \right] \rightarrow \left[ \begin{array}{cc|cc} 1 & 0.5 & 0.125 & 0 \\ 4 & 3 & 0 & 1 \end{array} \right]$$

- Schritt 2 - Element  $a_{21}$  muss eine 0 sein  $\rightarrow$  vier mal erste Zeile von zweiten abziehen

$$\left[ \begin{array}{cc|cc} 1 & 0.5 & 0.125 & 0 \\ 4 & 3 & 0 & 1 \end{array} \right] \rightarrow \left[ \begin{array}{cc|cc} 1 & 0.5 & 0.125 & 0 \\ 0 & 1 & -0.5 & 1 \end{array} \right]$$

- Schritt 3 - Element  $a_{12}$  muss eine 0 sein  $\rightarrow$  erste Zeile minus die Hälfte der zweiten Zeile

$$\left[ \begin{array}{cc|cc} 1 & 0.5 & 0.125 & 0 \\ 0 & 1 & -0.5 & 1 \end{array} \right] \rightarrow \left[ \begin{array}{cc|cc} 1 & 0 & 0.375 & -0.5 \\ 0 & 1 & -0.5 & 1 \end{array} \right] = [\mathbf{I}|\mathbf{B}]$$

# Cramersche Regel

- Lösung für Gleichungssystem

$$\mathbf{A} * \mathbf{b}_j = \mathbf{e}_j$$

- Das  $i$ -te Element des Lösungsvektors  $\mathbf{b}_j$  entspricht

$$(\mathbf{b}_j)_i = \frac{\det(\mathbf{A}_i)}{\det(\mathbf{A})}$$

- Matrix  $\mathbf{A}_i$  entsteht durch Ersetzen der  $i$ -ten Spalte von  $\mathbf{A}$  mit dem Einheitsvektor
- Determinante

$$\det(\mathbf{A}) = \sum_{i=1}^n (-1)^{i+j} * a_{ij} * \det(\mathbf{A}_{ij})$$

wobei  $\mathbf{A}_{ij}$  die Untermatrix von  $\mathbf{A}$  ist, welche durch Streichen der  $i$ -ten Zeile und der  $j$ -ten Kolonne entsteht

## Cramersche Regel II

- In

$$(\mathbf{b}_j)_i = \frac{\det(\mathbf{A}_i)}{\det(\mathbf{A})}$$

wird  $\det(\mathbf{A}_i)$  nach der Zeile entwickelt, d.h. es wird über die Kolonne summiert, welche durch den Einheitsvektor ersetzt wurde

- Somit ist

$$\det(\mathbf{A}_i) = \sum_{j=1}^n (-1)^{i+j} * a_{ij} * \det(\mathbf{A}_{ij})$$

wobei alle  $a_{ij} = 0$  sind ausser eines ist gleich 1

- Es folgt

$$\det(\mathbf{A}_i) = (-1)^{i+j} \det(\mathbf{A}_{ij})$$

# Beispiel für Cramersche Regel

- Element  $b_{ij}$  der Inversen kann berechnet werden als

$$b_{ij} = \frac{1}{\det(\mathbf{A})} \left( (-1)^{i+j} \det(\mathbf{A}_{ij}) \right)$$

- Für  $\mathbf{A} = \begin{bmatrix} 8 & 4 \\ 4 & 3 \end{bmatrix}$
- $b_{11} = \frac{1}{8} * 3 = 0.375$
- $b_{12} = \frac{1}{8} * (-1) * 4 = -0.5$
- $b_{21} = \frac{1}{8} * (-1) * 4 = -0.5$
- $b_{22} = \frac{1}{8} * 8 = 1$

## Anwendung für Verwandtschaftsmatrix

- Gauss-Jordan und vor allem Cramer sind sehr aufwändig und ungenau für grosse Matrizen
- Verwandtschaftsmatrix hat spezielle Eigenschaften, welche wir ausnützen wollen, z.Bsp Symmetrie, d.h.,  $\mathbf{A} = \mathbf{A}^T$  und positiv-definit, d.h., Eigenwerte  $\lambda > 0$ , welche erfüllen  $\mathbf{A} * \mathbf{x} = \lambda \mathbf{x}$
- Regeln für Inverse eines Produktes

$$\mathbf{A} = \mathbf{X} * \mathbf{Y} * \mathbf{Z}$$

die Inverse von  $\mathbf{A}$  ist dann

$$\mathbf{A}^{-1} = \mathbf{Z}^{-1} * \mathbf{Y}^{-1} * \mathbf{X}^{-1}$$

da

$$\mathbf{A} * \mathbf{A}^{-1} = \mathbf{X} * \mathbf{Y} * \mathbf{Z} * \mathbf{Z}^{-1} * \mathbf{Y}^{-1} * \mathbf{X}^{-1} = \mathbf{I}$$

# Zerlegung der Verwandtschaftsmatrix

- Symmetrische, positive-definite Matrizen  $\mathbf{A}$  können in folgendes Produkt zerlegt werden

$$\mathbf{A} = \mathbf{U} * \mathbf{U}^T$$

wobei  $\mathbf{L}$  eine untere Dreiecksmatrix ist

- Diese Zerlegung heisst **Cholesky**-Zerlegung
- In R wird diese Zerlegung mit der Funktion `chol()` berechnet
- Variante der Cholesky-Zerlegung

$$\mathbf{A} = \mathbf{L} * \mathbf{D} * \mathbf{L}^T$$

wobei  $\mathbf{L}$  eine untere Dreiecksmatrix mit 1 auf der Diagonalen und  $\mathbf{D}$  eine Diagonalmatrix ist

## Berechnung der Matrizen $\mathbf{L}$ und $\mathbf{D}$

- Definiert man  $\mathbf{U} = \mathbf{L} * \mathbf{S}$ , wobei  $\mathbf{D} = \mathbf{S} * \mathbf{S}$ , dann ist

$$\mathbf{A} = \mathbf{U} * \mathbf{U}^T = \mathbf{L} * \mathbf{S} * (\mathbf{L} * \mathbf{S})^T = \mathbf{L} * \mathbf{S} * \mathbf{S}^T * \mathbf{L}^T = \mathbf{L} * \mathbf{D} * \mathbf{L}^T$$

- $\mathbf{S}$  ist eine Diagonalmatrix wobei Elemente von  $\mathbf{S}$  der Wurzel der Elemente von  $\mathbf{D}$  entsprechen
- Somit ist  $\mathbf{L} = \mathbf{U} * \mathbf{S}^{-1}$
- $\mathbf{U}$  und  $\mathbf{D}$  können mit R bestimmt werden
- **Wichtig** die Funktion `chol()` in R macht die Zerlegung  $\mathbf{A} = \mathbf{U}^T * \mathbf{U}$ , welche dank Symmetrie völlig äquivalent ist, aber bei der Kontrolle muss man aufpassen.

## Bestimmung von **U** und **D** mit **R**

- Diagonalmatrix **D** (und somit **S**) kann mit der Funktion `Dmat()` aus Package "pedigreemm" bestimmt werden
- Matrix **U** kann mit Funktion `chol()` bestimmt werden
- Für ein Beispielpedigree ohne Inzucht

```
> library(pedigreemm)
> pedNoIb <- pedigree(sire = as.integer(c(NA,NA,1, 1,4,4)),
+                   dam  = as.integer(c(NA,NA,2,NA,2,2)),
+                   label = as.character(1:6))
> spmatANoIb <- getA(pedNoIb)
```

# Beispiel Pedigree

```
> print(pedNoIb)
```

	sire	dam
1	<NA>	<NA>
2	<NA>	<NA>
3	1	2
4	1	<NA>
5	4	2
6	4	2

## Verwandtschaftsmatrix zum Beispiel

```
> print(spmatANoIb)
```

```
6 x 6 sparse Matrix of class "dsCMatrix"
```

```
      1  2    3  4    5  6
1  1.00 .   0.500 0.50 0.250 0.250
2  .    1.0 0.500 .   0.500 0.500
3  0.50 0.5 1.000 0.25 0.375 0.375
4  0.50 .   0.250 1.00 0.500 0.500
5  0.25 0.5 0.375 0.50 1.000 0.500
6  0.25 0.5 0.375 0.50 0.500 1.000
```

“Sparse Matrix” heisst, es werden nur Elemente  $\neq 0$  gespeichert, überall wo ein Punkt steht, da ist das Element = 0

# Verwandtschaftsmatrix als normale Matrix

```
> matANoIb <- as.matrix(spmatANoIb)
> print(matANoIb)
```

	1	2	3	4	5	6
1	1.00	0.0	0.500	0.50	0.250	0.250
2	0.00	1.0	0.500	0.00	0.500	0.500
3	0.50	0.5	1.000	0.25	0.375	0.375
4	0.50	0.0	0.250	1.00	0.500	0.500
5	0.25	0.5	0.375	0.50	1.000	0.500
6	0.25	0.5	0.375	0.50	0.500	1.000

Tiere sind nicht ingezüchtet, alle Diagonalelemente = 1

# Zerlegung der Verwandtschaftsmatrix $U^T$

## ■ Matrix $U^T$

```
> (matCholUt <- chol(matANoIb))
```

	1	2	3	4	5	6
1	1	0	0.5000000	0.5000000	0.2500000	0.2500000
2	0	1	0.5000000	0.0000000	0.5000000	0.5000000
3	0	0	0.7071068	0.0000000	0.0000000	0.0000000
4	0	0	0.0000000	0.8660254	0.4330127	0.4330127
5	0	0	0.0000000	0.0000000	0.7071068	0.0000000
6	0	0	0.0000000	0.0000000	0.0000000	0.7071068

# Zerlegung der Verwandtschaftsmatrix $S$

## ■ Matrizen $D$ , $S$ und $S^{-1}$

```
> vecD <- Dmat(pedNoIb)
> matD <- diag(vecD)
> (matSinv <- diag(1/sqrt(vecD)))
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
[1,]	1	0	0.000000	0.000000	0.000000	0.000000
[2,]	0	1	0.000000	0.000000	0.000000	0.000000
[3,]	0	0	1.414214	0.000000	0.000000	0.000000
[4,]	0	0	0.000000	1.154701	0.000000	0.000000
[5,]	0	0	0.000000	0.000000	1.414214	0.000000
[6,]	0	0	0.000000	0.000000	0.000000	1.414214

# Zerlegung der Verwandtschaftsmatrix $L^T$

## ■ Matrix $L^T$

```
> (matLt <- matSinv %*% matCholUt)
```

```
      1 2   3   4   5   6
[1,] 1 0 0.5 0.5 0.25 0.25
[2,] 0 1 0.5 0.0 0.50 0.50
[3,] 0 0 1.0 0.0 0.00 0.00
[4,] 0 0 0.0 1.0 0.50 0.50
[5,] 0 0 0.0 0.0 1.00 0.00
[6,] 0 0 0.0 0.0 0.00 1.00
```

# Kontrolle der Zerlegung

- Matrixmultiplikation  $\mathbf{A} = \mathbf{L} * \mathbf{D} * \mathbf{L}^T$

```
> (t(matLt) %*% matD %*% matLt - matANoIb)
```

	1	2	3	4	5	6
1	0	0	0	0.000000e+00	0.000000e+00	0.000000e+00
2	0	0	0	0.000000e+00	0.000000e+00	0.000000e+00
3	0	0	0	0.000000e+00	0.000000e+00	0.000000e+00
4	0	0	0	1.110223e-16	5.551115e-17	5.551115e-17
5	0	0	0	5.551115e-17	0.000000e+00	0.000000e+00
6	0	0	0	5.551115e-17	0.000000e+00	0.000000e+00

# Allgemeine Berechnung der Matrizen $\mathbf{D}$ und $\mathbf{L}$

$$\mathbf{A} = \mathbf{L} * \mathbf{D} * \mathbf{L}^T = \begin{bmatrix} 1 & 0 & 0 \\ L_{21} & 1 & 0 \\ L_{31} & L_{32} & 1 \end{bmatrix} * \begin{bmatrix} D_1 & 0 & 0 \\ 0 & D_2 & 0 \\ 0 & 0 & D_3 \end{bmatrix} * \begin{bmatrix} 1 & L_{21} & L_{31} \\ 0 & 1 & L_{32} \\ 0 & 0 & 1 \end{bmatrix}$$

Für die Elemente in  $\mathbf{D}$  und  $\mathbf{L}$  gelten folgende rekursive Beziehungen

- $D_j = A_{jj} - \sum_{k=1}^{j-1} L_{jk}^2 D_k$
- $L_{ij} = \frac{1}{D_j} \left( A_{ij} - \sum_{k=1}^{j-1} L_{ik} L_{jk} D_k \right)$ , für  $i > j$

# Matrix L für Verwandtschaftsmatrix

- Diagonalelement für Tier  $i$ :  $L_{ij} = 1$
- Für Tiere  $i$  mit bekannten Eltern  $m$  und  $v$ :  $L_{ij} = 0.5(L_{mj} + L_{vj})$
- Falls nur ein Elternteil  $m$  bekannt ist:  $L_{ij} = 0.5L_{mj}$
- Beide Eltern unbekannt:  $L_{ij} = 0$

## Matrix **D** für Verwandtschaftsmatrix

- Mendelian sampling für Tier  $i$  mit Eltern  $m$  und  $v$  und den entsprechenden Zuchtwerten  $u_i$ ,  $u_m$  und  $u_v$ :

$$m_i = u_i - 0.5(u_m + u_v)$$

- Die Varianz der mendelian Sampling Effekte ist definiert als  $\mathbf{D} * \sigma_u^2$  wobei  $\sigma_u^2$  der genetisch additiven Varianz entspricht.

$$\begin{aligned} \text{var}(m_i) &= \text{var}(u_i) - \text{var}(0.5u_m + 0.5u_v) \\ &= \text{var}(u_i) - \text{var}(0.5u_m) - \text{var}(0.5u_v) - 2\text{cov}(0.5u_m, 0.5u_v) \\ &= (1 + F_i)\sigma_u^2 - 0.25a_{mm}\sigma_u^2 - 0.25a_{vv}\sigma_u^2 - 0.5a_{mv}\sigma_u^2 \end{aligned}$$

- Somit ist das Element  $D_{ii}$  für Tier  $i$

$$D_{ii} = \frac{\text{var}(m_i)}{\sigma_u^2} = (1 + F_i) - 0.25a_{mm} - 0.25a_{vv} - 0.5a_{mv}$$

## Verwendung der Zerlegung zur Inversion

- Aufgrund Regel zur Inversen eines Produktes (siehe Folie 9) gilt

$$\mathbf{A}^{-1} = (\mathbf{L} * \mathbf{D} * \mathbf{L}^T)^{-1} = (\mathbf{L}^T)^{-1} * \mathbf{D}^{-1} * \mathbf{L}^{-1}$$

- Matrizen  $\mathbf{L}$  und  $\mathbf{D}$  sind viel einfacher zu invertieren als  $\mathbf{A}$
- Matrix  $\mathbf{D}^{-1}$  auch eine Diagonalmatrix mit inversen Elementen der Ursprungsmatrix - Überprüfung mit

> solve(matD)

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
[1,]	1	0	0	0.000000	0	0
[2,]	0	1	0	0.000000	0	0
[3,]	0	0	2	0.000000	0	0
[4,]	0	0	0	1.333333	0	0
[5,]	0	0	0	0.000000	2	0
[6,]	0	0	0	0.000000	0	2

# Inverse von Matrix L

- Wie sieht  $L^{-1}$  aus
- Diagonalelemente sind alle 1
- Verbindet Eltern und Nachkommen mit Elementen von  $-0.5$

```
> solve(matLt)
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
1	1	0	-0.5	-0.5	0.0	0.0
2	0	1	-0.5	0.0	-0.5	-0.5
3	0	0	1.0	0.0	0.0	0.0
4	0	0	0.0	1.0	-0.5	-0.5
5	0	0	0.0	0.0	1.0	0.0
6	0	0	0.0	0.0	0.0	1.0

# Direktes Aufstellen von $\mathbf{A}^{-1}$ ohne Inzucht

- Setze  $\alpha_i$  auf den Wert von  $\mathbf{D}^{-1}$  für Tier  $i$
- Hat Tier  $i$  bekannte Eltern  $m$  und  $v$ , dann wird
- $\alpha_i$  zum Element  $(i, i)$  addiert
- $-\frac{\alpha_i}{2}$  zu den Elementen  $(m, i)$ ,  $(i, m)$ ,  $(v, i)$  und  $(i, v)$
- $\frac{\alpha_i}{4}$  zu den Elementen  $(m, m)$ ,  $(m, v)$ ,  $(v, m)$  und  $(v, v)$
- Falls Elternteile fehlen, dann werden entsprechende Teile weggelassen

# Direktes Aufstellen von $\mathbf{A}^{-1}$ mit Inzucht

- Zurück zur Zerlegung  $\mathbf{A} = \mathbf{U} * \mathbf{U}^T$
- Dann gilt  $a_{ii} = \sum_{k=1}^i u_{ik}^2$
- Diagonalelement von
 
$$u_{ij} = \sqrt{d_i} = \sqrt{[0.5 - 0.25(F_m + F_v)]} = \sqrt{[1 - 0.25(a_{mm} + a_{vv})]}$$
- Einsetzen:  $u_{ij} = \sqrt{[1 - 0.25 (\sum_{k=1}^m u_{mk}^2 + \sum_{k=1}^v u_{vk}^2)]}$
- Diagonalelement von  $\mathbf{D}^{-1}$  ist berechnet als  $\alpha_i = \frac{1}{u_{ii}^2}$
- Off-Diagonalelemente sind:  $u_{ij} = 0.5(u_{mj} + u_{vj})$