

Ausgleichsrechnung

wr@isg.cs.uni-magdeburg.de

SoSe 2019

1 Lineare Regression

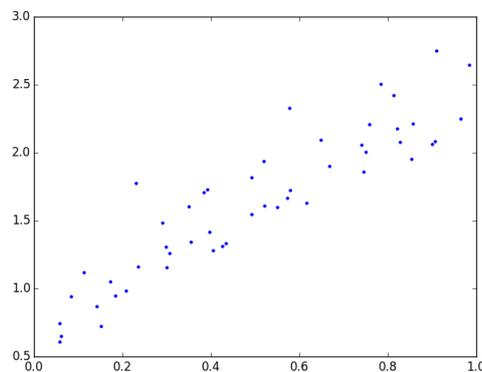


Abbildung 1: Messergebnisse, für welche ein linearer Zusammenhang zwischen x und y Werten vermutet wird.

Wir werden die Ausgleichsrechnung, oder lineare Regression, zunächst anhand eines einfachen Modellexperiments kennenlernen. Nehmen wir an, wir haben experimentell Messdatenpaare $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ bestimmt, siehe Abb. 1, und wir vermuten, dass es einen linearen Zusammenhang zwischen den x_i und y_i gibt, d.h. gilt

$$y_i \approx f(x_i) = m x_i + b. \quad (1)$$

Offensichtlich wird der lineare Zusammenhang nur approximativ halten, da die Messdaten mit zufälligem Rauschen, welches zum Beispiel auf Messfehler oder vorherige Rechnungen in Gleitkommazahlen zurückzuführen ist, kontaminiert sind. Anhand der gemessenen Daten soll der Anstieg m und die Verschiebung b , welche den linearen Zusammenhang beschreiben, bestimmt werden. Da die Datenpaare (x_i, y_i) linear von den unbekanntenen Parametern m und b abhängen, können wir für diese mit einem linearen Gleichungssystem lösen:

$$m x_i + b = y_i \quad (2a)$$

$$m x_j + b = y_j \quad (2b)$$

oder in kompakter Matrix-Vektor-Notation,

$$\begin{pmatrix} x_i & 1 \\ x_j & 1 \end{pmatrix} \begin{pmatrix} m \\ b \end{pmatrix} = \begin{pmatrix} y_i \\ y_j \end{pmatrix} \quad (3)$$

wobei i und j Indizes für zwei beliebige (unterschiedliche) Punktpaare sind. Die Lösung, welche wir für einige zufällig gewähltes Tuple (i, j) erhalten, ist in Abb. 2 dargestellt.

Wir sehen, dass m und b , und damit der lineare Zusammenhang, sehr stark von der Wahl von i und j abhängen und das Ergebnis in vielen Fällen nicht der anhand der Punktwolke intuitiv zu erwartenden Gerade entspricht.

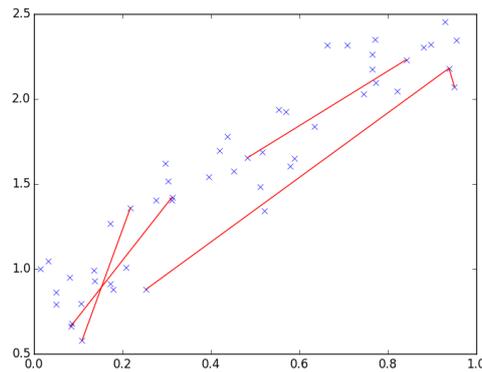


Abbildung 2: Linearer Zusammenhang mit jeweils zwei zufällig gewählten Datenpunkten bestimmt. Das Ergebnis hängt signifikant von den gewählten Punkten ab.

Die unterschiedlichen Ergebnisse für verschiedene Tuple (i, j) resultieren vom Rauschen, welches in den Daten enthalten ist. Unser Ziel ist es im Folgenden, den durch das Rauschen verursachten zufälligen Fehler in den Datenpaaren (x_i, y_i) auszugleichen, um ein Parameterpaar m und b zu bestimmen, welches den Daten ohne Rauschen entspricht. Da der Fehler in den Datenpaaren (x_i, y_i) unbekannt ist, werden wir uns im Allgemeinen mit einer Schätzung für die "wahren" Werte für m und b zufrieden geben müssen. Um die bestmögliche Schätzungen zu erhalten, sollen alle Datenpaare berücksichtigt werden. Dies ergibt folgendes lineares Gleichungssystem:

$$\underbrace{\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}}_A \underbrace{\begin{pmatrix} m \\ b \end{pmatrix}}_{\gamma} = \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_y. \quad (4)$$

Dieses Gleichungssystem ist überbestimmt und hat für mit Rauschen kontaminierte Daten wie in Abb. 1 keine Lösung, d.h. es existieren keine Parameter m und b , so dass das Gleichheitszeichen in der obigen Gleichung für jedes Datenpaar (x_i, y_i) erfüllt ist.

Um zu verstehen, wie wir eine "gute" Lösung finden, betrachten wir den Fall $n = 3$, d.h.

$$\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{pmatrix} \begin{pmatrix} m \\ b \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}. \quad (5)$$

Wie im Zusammenhang mit exakt lösbar linearen Gleichungssystemen besprochen, kann das Gleichungssystem spaltenweise interpretiert werden:

$$m \underbrace{\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}}_{\vec{u}} + b \underbrace{\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}}_{\vec{v}} = \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}}_{\vec{y}} \quad (6)$$

Die Vektoren \vec{u} , \vec{v} , \vec{y} sind in \mathbb{R}^3 definiert und m und b Gewichte, für welche die Linearkombination von \vec{u} und \vec{v} die rechte Seite \vec{y} ergeben soll, siehe Abb. 3. Die Vektoren \vec{u} und \vec{v} spannen dabei eine zwei-dimensionale Ebene in \mathbb{R}^3 . Ist das Gleichungssystem erfüllt, dann liegt \vec{y} in dieser Ebene. Für verrauschte Daten, wie die in Abb. 1, ist dies nicht der Fall.

Anstatt zu versuchen, eine exakte Lösung zu finden, die nicht existiert, können wir zumindest eine bestmögliche Lösung bestimmen, d.h. eine, welche so gut wie möglich ist in Bezug auf die gegebenen Daten. Je nach Anwendung kann "bestmöglich" unterschiedlich definiert werden. Die einfachste und am meisten verwendete Definition ist, dass der Punkt \vec{p}_y in der durch \vec{u} und \vec{v} aufgespannten Ebene gesucht wird, welcher am nächsten an \vec{y} liegt. Der verbleibende Fehler ist dann durch

$$\|\vec{r}\| = \|\vec{y} - \vec{p}_y\| \quad (7)$$

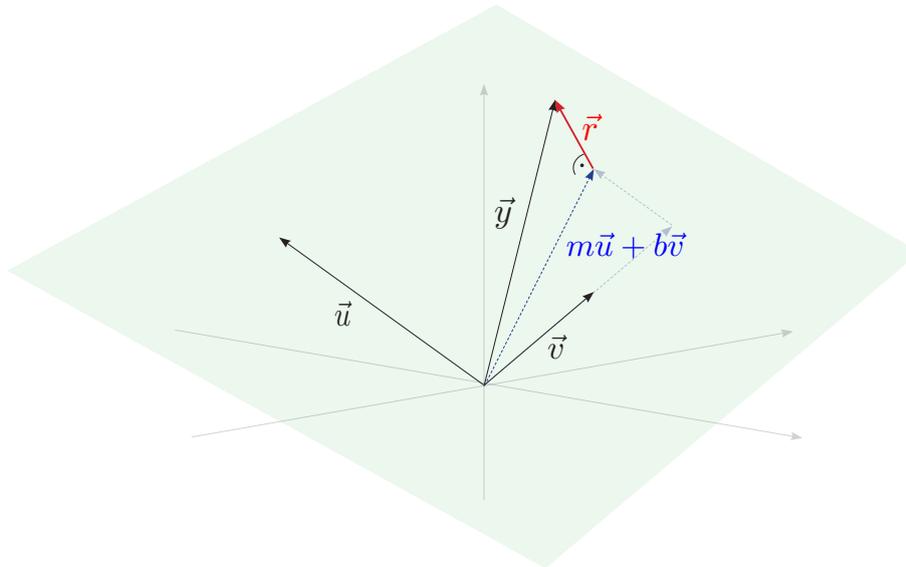


Abbildung 3: Geometrische Interpretation des linearen Regressionsproblems für drei Datenpunkte. Die Vektoren \vec{u} und \vec{v} , welche der linearen Gleichung entsprechen, spannen eine Ebene (grün) und gesucht wird der Punkt $A\gamma = m\vec{u} + b\vec{v}$ in der Ebene, welcher den Abstand zum Datenvektor y minimiert

gegeben, wobei \vec{r} der Differenzvektor zwischen \vec{y} und dem nächsten Punkt \vec{p}_y in der Ebene ist. Der Punkt \vec{p}_y bestimmt auch die gesuchten optimalen Parameter m' und b' , siehe Abb. 3. Da \vec{u} und \vec{v} genau die Spalten von A sind, kann \vec{p}_y in Matrix-Vektor-Notation wie folgt geschrieben werden:

$$\vec{p}_y = m'\vec{u} + b'\vec{v} = m' \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + b' \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = A\gamma. \quad (8)$$

wobei $\gamma = (m', b')$ die optimalen Parameter sind. Der Differenzvektor \vec{r} ist damit durch

$$\vec{r} = y - A\gamma \quad (9)$$

gegeben. Formal können die bestmöglichen Parameter m' und b' damit wie folgt definiert werden:

$$(m', b') = \operatorname{argmin}_{\gamma} \|y - A\gamma\|. \quad (10)$$

Geometrisch kann aus Abb. 3 abgelesen werden, dass die Länge $\|\vec{r}\|$ des Differenzvektors \vec{r} minimiert wird, wenn \vec{r} orthogonal auf der Ebene steht.¹

Ein beliebiger Punkt \vec{p} in der durch \vec{u} und \vec{v} aufgespannten Ebene ist:

$$\vec{p} = z_1\vec{u} + z_2\vec{v} = Az, \quad (11)$$

wobei $z = (z_1, z_2)^T \in \mathbb{R}^2$ ein beliebiger Parametervektor ist. Mit Gl. 11 kann die Orthogonalitätsbedingung für \vec{r} mit Hilfe des Skalarproduktes formuliert werden:

$$0 = \langle \vec{r}, Az \rangle. \quad (12a)$$

wobei die Gleichung für alle $z \in \mathbb{R}^2$ gelten muss. Einsetzen aus Gl. 9 ergibt:

$$0 = \langle y - A\gamma, Az \rangle. \quad (12b)$$

Unter Verwendung der Linearität des Skalarproduktes erhält man

$$0 = \langle y, Az \rangle - \langle A\gamma, Az \rangle. \quad (12c)$$

¹Wenn wir annehmen, dass ein Vektor, welcher nicht orthogonal auf der Ebene steht, die Distanz minimiert, so können wir diesen als eine Linearkombination von \vec{r} und eines Vektors in der Ebene schreiben. Dieser hat dann bei der Dreiecksungleichung eine größere Länge als \vec{r} .

Da $\langle x, Ay \rangle = \langle A^T x, y \rangle$ haben wir außerdem

$$0 = \langle A^T y, z \rangle - \langle A^T A \gamma, z \rangle \quad (12d)$$

und wenn wir die Linearität des Skalarproduktes erneut verwenden erhalten wir schließlich

$$0 = \langle A^T y - A^T A \gamma, z \rangle. \quad (12e)$$

Da $z \in \mathbb{R}^2$ aber jeder beliebiger Vektor sein kann, d.h. \vec{r} orthogonal zu jedem beliebigen Vektor in der Ebene sein soll, gilt Gl. 12e nur wenn

$$0 = A^T y - A^T A \gamma. \quad (13)$$

Die letzte Gleichung wird als **Normalengleichung** bezeichnet. Betrachten wir die Größen in dieser Gleichung, so haben wir

$$M = A^T A \in \mathbb{R}^{2 \times 2} \quad (14a)$$

$$b = A^T y \in \mathbb{R}^2 \quad (14b)$$

und

$$M \gamma = b \quad (15)$$

ist also ein quadratisches 2×2 Gleichungssystem. Wir können dieses System mit Gauß-Elimination lösen. Die Matrix M ist jedoch symmetrisch und positiv semi-definit,² so dass die Cholesky-Zerlegung, welche in Abschnitt 2 besprochen wird, das geeignetere Verfahren ist.

Nebenbemerkung: In der obigen Herleitung haben wir benutzt, dass gilt $\langle x, Ay \rangle = \langle A^T x, y \rangle$. Dies kann man zeigen, wenn man das Skalarprodukt und die Matrix-Multiplikation in Koordinaten schreibt,

$$\langle x, Ay \rangle = \sum_{i=1}^n x_i \left(\sum_{j=1}^n A_{ij} y_j \right) \quad (16)$$

und durch eine Umordnung der Summe erhält man

$$\langle x, Ay \rangle = \sum_{j=1}^n \left(\sum_{i=1}^n x_i A_{ij} \right) y_j. \quad (17)$$

Da die innere Summe über die Zeile von A ist, entspricht dies $x^T A$, für was gilt $x^T A = A^T x$. Damit haben wir

$$\langle x, Ay \rangle = \langle A^T x, y \rangle. \quad (18)$$

Die transponierte ist damit eine Art inverse Matrix, da sie in die gleiche "Richtung" wie A^{-1} geht, aber "nur" das innere Product erhält. Dies ist auch die geometrische Eigenschaft, welche die transponierte definiert. In der Tat haben wir in unserem Fall $A : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ und damit $A^T : \mathbb{R}^3 \rightarrow \mathbb{R}^2$, d.h. A^T ist eine Projektionsmatrix auf die von \vec{u} und \vec{v} aufgespannte Ebene. Wir sollten also eigentlich auch genauer schreiben

$$\langle x, Ay \rangle_{\mathbb{R}^3} = \langle A^T x, y \rangle_{\mathbb{R}^2}. \quad (19)$$

In der Herleitung der Normalengleichung haben wir drei Punkte betrachtet, um eine graphische Darstellung des Problems zu ermöglichen. Verwenden wir eine beliebige Anzahl von n Punkten, dann spannen die zwei Spalten von A eine 2-dimensionale Ebene im \mathbb{R}^n auf und man versucht in diesem Raum, den Abstand zu $y \in \mathbb{R}^n$ zu minimieren. Der minimale Abstand ist weiterhin durch

$$0 = \langle \vec{r}, Az \rangle = \langle y - A \gamma, Az \rangle \quad (20)$$

beschrieben. Es kann kontrolliert werden, dass die Rechnung in Gl. 12 unabhängig von der Dimension ist und damit auch die Normalengleichung in Gl. 15 für eine beliebige Anzahl von Punkten n gilt. Darüberhinaus erhält man auch im n -dimensionalen Fall ein 2×2 Gleichungssystem, d.h. die Bestimmung von m und b erfordern unabhängig von der Anzahl der Datenpunkte nur die Lösung eines 2×2 Gleichungssystems. Die Lösung des linearen Regressionsproblems für 50 Punkte ist in Abb. 4 dargestellt. Der folgende Quellcode zur Lösung des linearen Regressionsproblems befindet sich in Abb. 5.

Eine wichtige Eigenschaft der Ausgleichsrechnung ist, dass der Fehler bei der Schätzung der Parameter geringer wird, wenn mehr Datenpunkte verwendet werden. Für die lineare Regression ist dies in Abb. 6 dargestellt (gemittelt über mehrere Instanzen des Rauschens).

²Eine Matrix $M \in \mathbb{R}^{n \times n}$ ist positiv semi-definit, wenn $z^T M z \geq 0$.

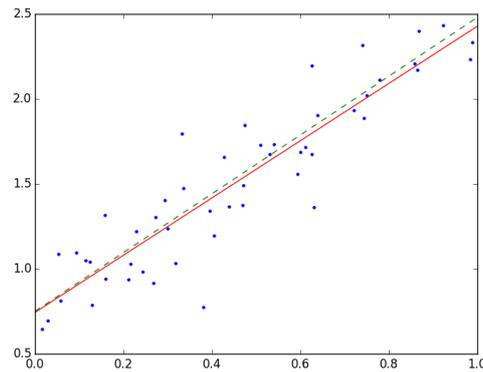


Abbildung 4: Ergebnis für lineare Regression mit 50 Datenpunkten. Der geschätzte lineare Zusammenhang ist in rot dargestellt, der zur Erzeugung der Datenpaare verwendete in grün.

```
# -*- coding: utf-8 -*-  
"""  
Linear regression  
"""  
  
from triangularSolve import forwardSub, backwardSub  
  
import matplotlib.pyplot as plt  
import numpy as np  
  
# "unknown" linear relation  
def f(x):  
    return 1.73 * x + 0.75  
  
# input data  
N = 1000  
x = np.random.random((N,))  
y = f(x) + 0.2 * np.random.randn(N)  
  
# setup linear system  
A = np.ones((N, 2))  
A[:,0] = x  
M = np.dot( np.transpose(A), A);  
b = np.dot( np.transpose(A), y)  
  
params = np.linalg.solve( M, b)  
  
## Solve linear system  
#chol = np.linalg.cholesky(M)  
#params_t = forwardSub(chol, b)  
#params = backwardSub(np.transpose(chol), params_t)  
  
# visualize result  
# plt.plot( x, y, '.b')  
x_end = np.array([0.0,1.0])  
y_fit = x_end * params[0] + params[1]  
y_true = f(x_end)  
plt.plot( x_end, y_fit, '-r')  
plt.plot( x_end, y_true, '--g')  
plt.show()
```

Abbildung 5: Quellcode für lineares Regressionsproblem.

2 Cholesky Zerlegung

Angenommen wir möchten eine Menge von Regressionsproblemen lösen, bei denen die x Koordinaten der Punkte für alle Probleme gleich sind und nur die y Koordinaten sich unterscheiden. Wir können für jeden Satz der y Werte die Normalgleichung mithilfe der Gauß-Elimination und anschließendem Rück-

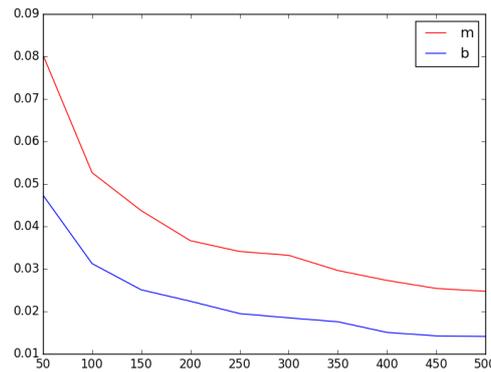


Abbildung 6: Fehler in m und b in Abhängigkeit von der Anzahl der Datenpunkte.

wärtseinsetzen lösen. Da die Matrix A , und damit auch $M = A^T A$, lediglich von den x Koordinaten der Punkte abhängt, werden wir dabei jedoch immer die gleiche Matrix auf Dreiecksgestalt bringen und nur die rechte Seite $b = A^T y$ des Gleichungssystems ändert sich. Dies ist offensichtlich eine ineffiziente Vorgehensweise und es wäre günstiger, einmal die Matrix $M = A^T A$ auf Dreiecksgestalt zu bringen und sich die notwendigen Transformationen, welche auf die rechte Seite angewendet werden müssen, zu merken. Eine Möglichkeit genau dies zu erreichen, bietet die sogenannte Cholesky Zerlegung, welche eine Matrix $M \in \mathbb{R}^{n \times n}$ als Produkt einer unteren Dreiecksmatrix $L \in \mathbb{R}^{n \times n}$ und deren Transponierter $L^T \in \mathbb{R}^{n \times n}$ darstellt:

$$M = LL^T = \begin{pmatrix} l_{1,1} & 0 & 0 & \cdots & 0 \\ l_{2,1} & l_{2,2} & 0 & \cdots & 0 \\ l_{3,1} & l_{3,2} & l_{3,3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n,1} & l_{n,2} & l_{n,3} & \cdots & l_{n,n} \end{pmatrix} \begin{pmatrix} l_{1,1} & l_{2,1} & l_{3,1} & \cdots & l_{n,1} \\ 0 & l_{2,2} & l_{3,2} & \cdots & l_{n,2} \\ 0 & 0 & l_{3,3} & \cdots & l_{n,3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & l_{n,n} \end{pmatrix}. \quad (21)$$

Diese Zerlegung existiert, wenn $M \in \mathbb{R}^{n \times n}$

- i) symmetrisch ist, d.h. $B = B^T$,
- ii) positiv semi-definit ist, d.h. $x^T B x \geq 0$ für alle $x \in \mathbb{R}^n$.

Diese beiden Eigenschaften sind für Matrizen der Form $M = A^T A$ immer erfüllt. Das Verfahren ist also auf die Normalgleichung anwendbar.

Ist die Matrix L bekannt, dann lässt sich das Gleichungssystem

$$Mx = LL^T x = b \quad (22)$$

effizient lösen, indem man sich die Dreiecksstruktur der Matrix zunutze macht. Dafür substituiert man zunächst

$$y = L^T x \quad (23)$$

und erhält das Gleichungssystem

$$Ly = b \quad (24)$$

welches sich effizient durch Vorwärtseinsetzen lösen lässt. Kennt man y , so kann man Gl. 23 durch Rückwärtseinsetzen lösen und erhält dadurch die Lösung des Gleichungssystems.

Die Zerlegung $M = LL^T$ ermöglicht damit, dass keine wiederholte Transformation der Matrix $M = A^T A$ in obere Dreiecksform für verschiedene rechte Seiten notwendig ist. In der Tat ist die obere Dreiecksmatrix L^T identisch zur oberen Dreiecksmatrix, welche man mithilfe der Gauß-Elimination erhält und die Matrix L repräsentiert die Zeilenumformungen, welche bei der Durchführung der Gauß-Elimination vorgenommen wurden. Das Lösen von Gl. 24 bedeutet damit nichts anderes, als das Anwenden der Umformungen auf den Vektor b . Das Lösen von Gl. 23 entspricht dem Rückwärtseinsetzen, welches nach der Gauß-Elimination benutzt wird, um die Lösung zu erhalten.

Berechnen der Cholesky Zerlegung

Aber wie erhält man die Cholesky Zerlegung? Für die Elemente der unteren Dreiecksmatrix L gelten folgende Formeln:

$$l_{i,i} = \sqrt{b_{i,i} - \sum_{k=1}^{i-1} l_{i,k}^2} \quad (25a)$$

$$l_{i,j} = \frac{1}{l_{j,j}} \left(b_{i,j} - \sum_{k=1}^{j-1} l_{i,k} l_{j,k} \right), \quad \text{falls } i > j \quad (25b)$$

Betrachtet man diese Formeln, so stellt man fest, dass Element $l_{i,j}$ der unteren Dreiecksmatrix L nur von Elementen, welche „darüber“ oder „links“ sind, abhängt. Mit anderen Worten, um $l_{i,j}$ zu berechnen, reicht es aus die Elemente $l_{m,n}$ mit $m \leq i$ und $n \leq j$ zu kennen. Wir können also von links oben beginnend, d.h. angefangen von $l_{1,1}$, welches nur von $a_{1,1}$ abhängt, Zeile für Zeile von links nach rechts die Elemente $l_{i,j}$ berechnen. Die Formeln in Gl. 25 können auch verwendet werden, um festzustellen, ob die Eingabematrix positiv semi-definit ist. Wenn dies nicht der Fall ist, dann erhält man bei der Berechnung der Faktorisierung eine negative Zahl unter der Wurzel in Gl. 25a.

Folgender Pythoncode implementiert die Cholesky Zerlegung:

```
import numpy as np
from math import sqrt

def cholesky(m):
    N = m.shape[0]
    assert(N == m.shape[1])
    L = np.zeros((N,N))

    for i in range(N):
        for j in range(0, i):
            L[i, j] = (m[i, j] - np.dot(L[i, :], L[j, :])) / L[j, j]
        L[i, i] = sqrt(m[i, i] - np.sum(np.square(L[i, :i])))

    return L
```

Die Cholesky-Faktorisierung bietet den Vorteil, dass sie nur ca. halb so viele Berechnungen benötigt, wie bei einer Lösung des Gleichungssystems mit der Gauß-Elimination. Darüber hinaus ist sie stabil, auch ohne Pivoting.

LR Zerlegung

Eine Analogon zur Cholesky Zerlegung existiert auch für quadratischen, invertierbare Matrizen $A \in \mathbb{R}^{n \times n}$, welche nicht symmetrisch und positiv semi-definit sind. Dies ist die sogenannte LR Zerlegung:

$$A = LR \quad (26)$$

wobei $L \in \mathbb{R}^{n \times n}$ eine untere Dreiecksmatrix ist und $R \in \mathbb{R}^{n \times n}$ eine obere Dreiecksmatrix. Die Cholesky Zerlegung ist damit ein Spezialfall der LR Zerlegung für symmetrische, positiv semi-definite Matrizen, bei welcher $R = L^T$. Große Bedeutung erhält die LR Zerlegung in der Praxis wieder durch die Möglichkeit, ein Gleichungssystem für verschiedene rechte Seiten effizient zu lösen. Dies erreicht man durch ein Vorgehen analog zu Gl. 23 und Gl. 24. Die LR Zerlegung wird im Anhang ausführlich diskutiert.

A LR Zerlegung³.

In der Praxis will man häufig ein Gleichungssystem $Ax = b$ für verschiedene Vektoren b lösen. Das Gauß-Verfahren, wie wir es bis jetzt betrachtet haben, erfordert jedoch für jeden Vektor b eine vollständig neue Lösung des Gleichungssystems. Die LR Zerlegung bietet hierfür eine Lösung und ermöglicht es, ein Gleichungssystem effizient für verschiedene "Eingabedaten" b zu lösen. Einen Teil der LR Zerlegung haben wir bereits kennengelernt: R steht für die obere Dreiecksmatrix, welche das Ergebnis der Gauß-Elimination ist. Was für eine bereits berechnete Matrix R fehlt, um diese für eine beliebige rechte Seite b verwenden zu können, sind die Transformationen, Zeilenaddition und Zeilentausch, welche von der Eingabematrix A zu R geführt haben. Der zweite Teil der LR Zerlegung, die untere Dreiecksmatrix L , enthält diese Informationen. Damit haben wir $A = LR$ und L und R repräsentieren zusammen wieder das ursprüngliche lineare Gleichungssystem. Da L und R beide Dreiecksmatrizen sind, ist eine sehr effiziente Lösung des Gleichungssystems mit Vorwärts- und Rückwärtseinsetzen möglich. Betrachten wir nun näher, wie L konstruiert werden kann und wie damit eine effiziente Lösung eines Gleichungssystems möglich ist.

Betrachten wir ein 2×2 Gleichungssystem, welche in Matrixform als

$$A \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ -a & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} e \\ f \end{pmatrix} \quad (27)$$

gegeben ist. Für die Gauß-Elimination müssen wir $-a$ in der zweiten Zeile durch die Addition der beiden Zeile eliminieren. Wenn wir das Produkt von A mit einer speziellen Matrix

$$L_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} = \text{Id} + \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \quad (28)$$

betrachten, so haben wir

$$L_1 A = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} a & b \\ -a & c \end{pmatrix} = \begin{pmatrix} a & b \\ a - a & b + c \end{pmatrix} \quad (29)$$

Wir sehen also, dass $L_1 A$ genau die gewünschten Operationen implementiert: die erste Zeile des Produkts ist wieder die erste Zeile von A und die zweite Zeile ist die Summe der Zeilen in A .

Wenn wir mehrere Zeilen-Transformationen haben, dann kann können diese als Produkt von mehreren L_a Matrizen ausgedrückt werden. Zum Beispiel, für ein 3×3 Gleichungssystem drücken

$$L_1 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad L_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \quad (30)$$

die Addition der ersten mit der zweiten und der ersten mit der dritten Zeile aus. Das Produkt der Matrizen

$$L_2 L_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \quad (31)$$

entspricht, entsprechend der Definition der Matrixmultiplikation, der aufeinanderfolgenden Umsetzung der beiden Transformationen. An den vorhergehenden Beispielen sehen wir auch, dass L_a die Summe aus der Einheitsmatrix und einer Matrix \bar{L}_a ist, deren (i, j) -te Element 1 ist, wenn wir die j -te Zeile der Matrix zur i -ten addieren. Im Allgemeinen, wenn wir Vielfache von Zeilen addieren, gilt für das (i, j) -te Element der Matrix \bar{L}_a das

$$\bar{L}_{ij}^a = \begin{cases} c & \text{wenn } c \text{ multipliziert mit } j\text{-ten Zeile zur } i\text{-ten addiert wird} \\ 0 & \text{ansonsten} \end{cases} \quad (32)$$

und $L_a = \text{Id} + \bar{L}_a$.

³LR steht für „links-rechts“. Im englischen spricht man von der *LU-Decomposition* für „lower-upper“

Betrachten wir nun, wie wir mit Hilfe der Matrizen L_a die LR Zerlegung von A erhalten können. Wenn wir Gauß-Elimination (zunächst ohne Pivoting) durch die Matrizen L_a ausdrücken, dann haben wir

$$L_k \cdots L_1 A = R. \quad (33a)$$

wobei R die obere Dreiecksmatrix ist, mit welcher das Gleichungssystem effizient mit Rückwertseinsetzen gelöst werden kann. Bezeichnen wir nun die akkumulierten Transformationen mit $L^{-1} = L_k \cdots L_1$ (die Bezeichnung L^{-1} ist unintuitiv aber Konvention), dann können wir die Matrix A , welche das ursprüngliche Gleichungssystem repräsentiert, als

$$L^{-1} A = R \implies A = L R \quad (33b)$$

schreiben. Damit haben wir die LR Zerlegung von A erhalten. Um die Matrix L zu bestimmen, muss L^{-1} invertiert werden. Durch die spezielle Struktur von $L^{-1} = L_k \cdots L_1$ ist die Invertierung sehr einfach möglich. Betrachten wir dafür zunächst eine einzelne Matrix L_1 , z.B.

$$L_1 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (34)$$

welche, wie bereits besprochen, die Addition der ersten und zweiten Zeile der Matrix A darstellt. Die umgekehrte Operation ist die Subtraktion der ersten Zeile von der zweiten. Die entsprechende Matrix ist

$$L'_1 = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (35)$$

Für die Anwendung von L_1 gefolgt von L'_1 haben wir also

$$L'_1 L_1 A = A \quad (36)$$

da L_1 und L'_1 inverse Operationen darstellen. Aber dies bedeutet auch, dass

$$L'_1 L_1 = \text{Id} \rightarrow L'_1 = L_1^{-1}, \quad (37)$$

d.h. die Matrix L'_1 ist die Inverse von L_1 . Man kann natürlich auch per Hand leicht nachrechnen, dass $L'_1 L_1 = I$ gilt. Mit anderen Worten: es gilt

$$L_a^{-1} = \text{Id} - \overline{L_a}. \quad (38)$$

Mit den Inversen der L_a Matrizen können wir die gesuchte Matrix L bestimmen:

$$L = (L^{-1})^{-1} = (L_k \cdots L_1)^{-1} \quad (39a)$$

und mit $(AB)^{-1} = B^{-1}A^{-1}$ erhalten wir

$$L = L_1^{-1} \cdots L_k^{-1}. \quad (39b)$$

Es muss beachtet werden, dass die Matrizen L_a^{-1} in der umgekehrten Reihenfolge multipliziert werden, wie die L_a Matrizen, d.h.

$$L_k \cdots L_1 A = R \implies A = L_1^{-1} \cdots L_k^{-1} R. \quad (40)$$

In der Praxis wird für eine Computerimplementierung folgende Formel zur Bestimmung von L verwendet:

$$L = \text{Id} - \sum_{a=1}^k \overline{L_a}. \quad (41)$$

Diese zeigt auch explizit, dass sowohl das Produkt als auch die Inverse von unteren Dreiecksmatrizen wieder eine untere Dreiecksmatrix ist.

Beispiel Schauen wir uns noch einmal das einführende Beispiel von Aufgabe 1 an. Um die Matrix umzuformen, haben wir zunächst die erste Zeile auf die zweite und dritte addiert, und dann das Doppelte der zweiten Zeile zur dritten addiert. Wir erhalten damit in Matrixform

$$L_3 L_2 L_1 A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} A. \quad (42a)$$

Als nächstes müssen die Matrizen auf die rechte Seite gebracht werden. Damit erhalten wir

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix}^{-1} R. \quad (42b)$$

Wie im Vorausgehenden beschrieben sind die Matrizen L_a einfach zu invertieren, in dem wir das Vorzeichen des nicht-trivialen Matrixelementes umdrehen, was dem Wechsel zwischen Addition und Subtraktion einer Zeile entspricht. Damit erhalten wir

$$A = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{pmatrix} R. \quad (42c)$$

Durch formelles Multiplizieren der L_a Matrizen, welches wie in Eq. 41 durch Addieren der nicht-diagonalen Einträge möglich ist, erhält man

$$A = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & -2 & 1 \end{pmatrix} R \quad (42d)$$

welches die gesuchte Faktorisierung $A = LR$ ist. Explizit haben wir also für die Zerlegung von A :

$$\begin{pmatrix} 1 & 0 & -1 \\ -1 & -1 & 2 \\ -1 & 2 & -3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & -2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -1 \\ 0 & -1 & 1 \\ 0 & 0 & -2 \end{pmatrix} \quad (43)$$

wobei die Matrix R in der Praxis durch das Ausführen von Gauss-Elimination bestimmt wird und L durch das Akkumulieren der Transformationen.

Berechnung der Lösung eines Gleichungssystems mit Hilfe der LR Zerlegung Mithilfe der LR Zerlegung lässt sich ein Gleichungssystem leicht und effizient für beliebige rechte Seiten lösen. Durch Einsetzen der Faktorisierung erhalten wir zunächst:

$$Ax = (LR)x = LRx = b. \quad (44)$$

Auf der linken Seite des Systems stehen jetzt zwei Matrizen, L und R , so dass keine direkte Lösung möglich ist. Zunächst lösen wir deshalb für einen temporären Vektor

$$y = Rx \quad (45)$$

mit welchem Eq. 44 ein reguläres Gleichungssystem

$$Ly = b \quad (46)$$

ist. Darüberhinaus ist L eine untere Dreiecksmatrix, so dass das System effizient durch Vorwärtseinsetzen beginnend von der ersten Zeile gelöst werden kann. Ist y bestimmt so kann das Gleichungssystem in Gleichung 45 gelöst werden, was auch wieder effizient möglich ist, da R eine obere Dreiecksmatrix ist, so dass Rückwärtseinsetzen verwendet werden kann. Die vollständige Lösung des Gleichungssystems erhalten wir als mit einmaligen Vorwärtseinsetzen zur Bestimmung des Zwischenergebnisses y in Gleichung 46 und dann der Lösung nach x in Gleichung 45 mit Rückwärtseinsetzen.

A.1 Übungsaufgabe

Wie muss das Vorgehen modifiziert werden, damit Pivoting bei der Berechnung der LR-Zerlegung verwendet werden kann.

Bei der Spaltenpivotsuche werden die Zeilen von A neu angeordnet. Diese Vertauschung der Zeilen lässt sich formal durch die Multiplikation von links mit einer Permutationsmatrix beschreiben. Eine solche Matrix hat die Eigenschaft, dass in jeder Zeile und Spalte nur eine 1 auftritt und alle anderen Element in jeder Spalte und Zeile null sind. Zum Beispiel

$$P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (47)$$

ist eine Permutationsmatrix. Wenn wir dies auf eine Matrix A anwenden dann erhalten wir

$$PA = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = \begin{pmatrix} A_{21} & A_{22} \\ A_{11} & A_{12} \end{pmatrix} \quad (48)$$

und wir sehen, dass P die zwei Zeilen von A vertauscht. Formell lässt sich eine Permutationsmatrix als eine Permutation der Einheitsvektoren $e_i = \delta_{ii}$ beschreiben, wenn diese als Spalten in einer Matrix angeordnet werden. Ist $\pi: \{1, \dots, n\} \rightarrow \{1, \dots, n\}$ eine Permutation von n Elementen, so ist die zugehörige Permutationsmatrix gegeben durch:

$$P_\pi = (e_{\pi(1)}, \dots, e_{\pi(n)}) \quad (49)$$

Es sein nun P die Permutationsmatrix, welche die Vertauschungen bei der Durchführung des Gaußschen Eliminationsverfahrens beschreibt. Dann gilt: $PA = LR$. Um eine Lösung für $Ax = b$ können wir also $PAx = LRx = Pb$ mittels Vorwärts-/Rückwärtseinsetzen lösen:

$$Ly = Pb \quad (\text{Vorwärtseinsetzen}) \quad (50a)$$

$$Rx = y \quad (\text{Rückwärtseinsetzen}) \quad (50b)$$

Beispiel Gegeben sei die Matrix

$$A = \begin{pmatrix} -2 & 2 & -2 \\ -1 & -1 & 2 \\ 1 & 2 & -1 \end{pmatrix}. \quad (51)$$

Um die LR-Zerlegung mit Pivoting durchzuführen, starten wir mit den drei Matrizen $P = \text{Id}$, $L = 0$ und $R = A$. Es ist zu beachten, dass die Matrix L beim Pivoting auch permutiert werden muss, da sich die hier gespeicherten Operationen ja auf die unpermutierte Matrix beziehen.

R	L	P	Beschreibung
$\begin{pmatrix} -2 & 2 & -2 \\ -1 & -1 & 1 \\ 1 & 3 & -1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	Initialisierung.
$\begin{pmatrix} -2 & 2 & -2 \\ 0 & -2 & 2 \\ 0 & 4 & -2 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0.5 & 0 & 0 \\ -0.5 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	Erste Spalte eliminieren.
$\begin{pmatrix} -2 & 2 & -2 \\ 0 & 4 & -2 \\ 0 & -2 & 2 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ -0.5 & 0 & 0 \\ 0.5 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$	Pivoting: Zeile 2 und 3 tauschen.
$\begin{pmatrix} -2 & 2 & -2 \\ 0 & 4 & -2 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ -0.5 & 1 & 0 \\ 0.5 & -0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$	Zweite Spalte eliminieren.

Nachdem wir nun das Gaußsche Eliminationsverfahren mit Pivoting durchgeführt haben, erhalten wir die gewünschte Zerlegung $PA = LR$.