NUMERIK

Inhaltsverzeichnis

**1 MATLAB**

1.1 Basics

1.2 Funktionen

1.3 Vektoren

1.4 Abbildungen

1.5 Balkendiagramm

1.6 Flächendiagramm

**2 Rechenarithmetik**

2.1 Umrechen zwischen Basen

2.2 Maschinengenauigkeit

2.3 Absoluter / Relativer Fehler & Konditionszahl

**3 Numerische Lösungen von Nullstellenproblemen**

3.1 Bisektionverfahren

3.2 Fixpunktiteration

3.3 Banachscher Fixpunktsatz

3.4 Newton Verfahren

3.5 Konvergenzgeschwindigkeit

3.6 Fehlerabschätzung

**4 Numerische Lösung linearen Gleichungssysteme**

4.1 Gauss Algorithmus

4.2 LR-Zerlegung

4.3 LR-Zerlegung druch Zeilenvertauschung (P Multiplikation)

4.4 Cholesky-Zerlegung (Spezielle LR-Zerlegung)

**5 Fehlerrechnung & Aufwandabschätzung (linearen Gleichungssysteme)**

5.1 Vektornormen

5.2 Fehlerbehafteter x-Vektor

5.3 Fehlerbehaftete A-Matrix

5.4 Aufwandabschätzung

**6 Iterativ-Verfahren**

6.1 Jacobi-Verfahren (eig. nur Gleichung wichtig)

6.2 Gauss-Siedel-Verfahren (eig. nur Gleichung wichtig)

6.3 Konvergenz

**7 Numerische Lösungen nicht linearen Gleichungssysteme**

7.1 Funktionen mit mehreren Variablen

7.2 Partielle Ableitung 1. Ordnung

7.3 Problemstellung zur Nullstellenbestimmung für nichtlineare Systeme

7.4 Das Newton-Verfahren für Systeme

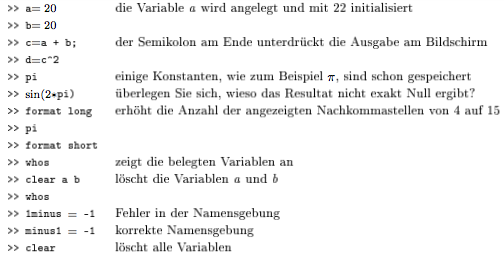
7.5 Quadratisch konvergentes Newton-Verfahren

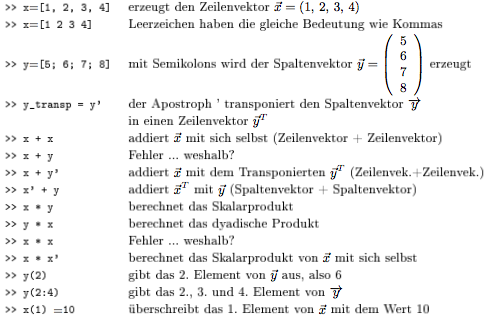
7.6 Vereinfachtes Newton-Verfahren

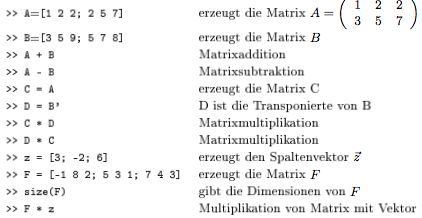
7.7 Gedämpftes Newton-Verfahren

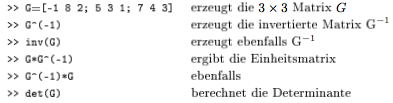
**1 MATLAB**

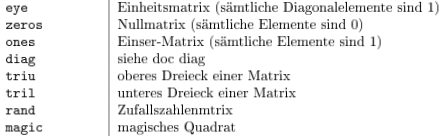
1.1 Basics

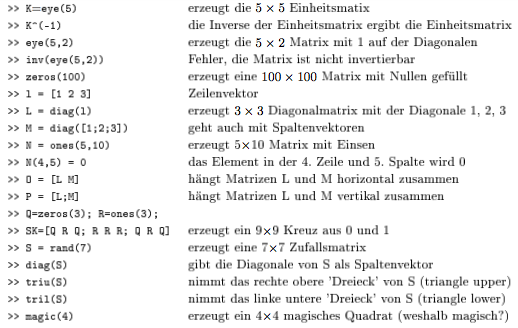
****

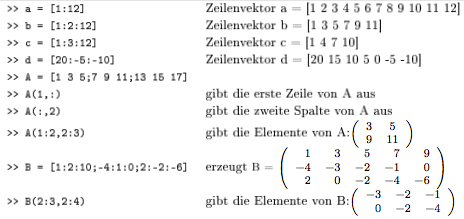
****

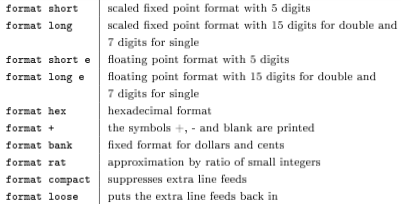
****

****

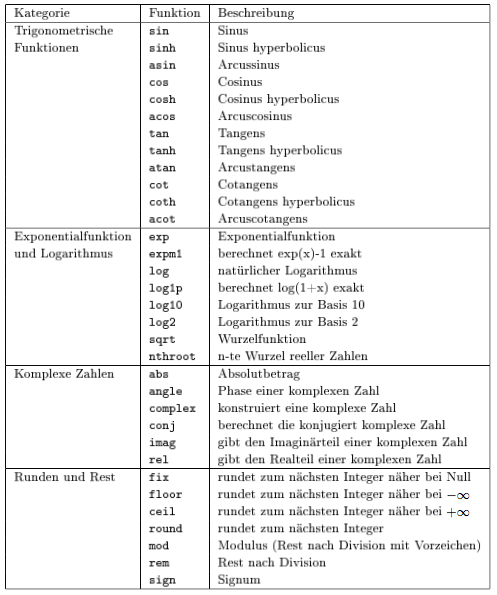
****

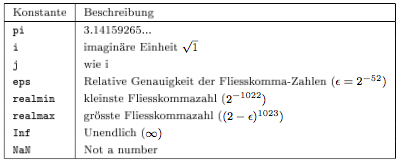
****

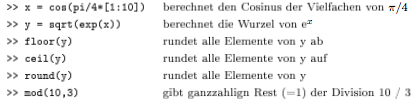
****

****

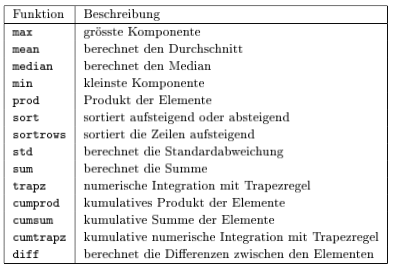
1.2 Funktionen

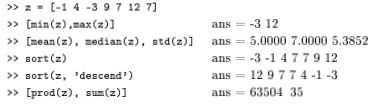
****

****

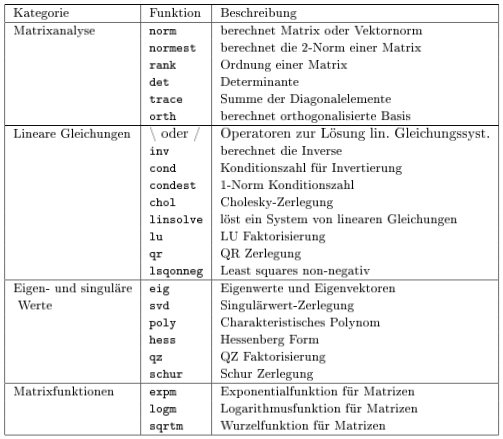
****

1.3 Vektoren

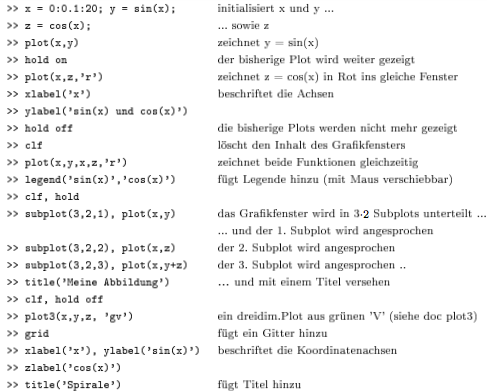
****

****

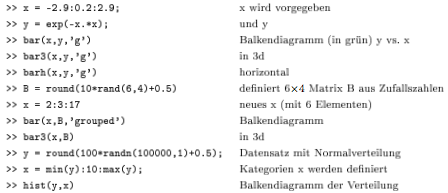
1.4 Matrizen

****

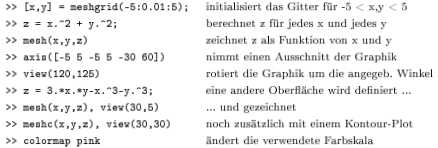
1.5 Abbildungen

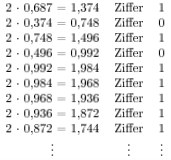
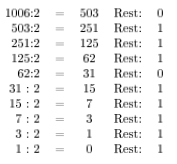
****

1.6 Balkendiagramm

****

1.7 Flächendiagramm



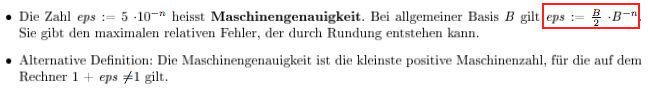
**2 Rechenarithmetik**

2.1 Umrechnen zwischen Basen

../../Desktop/Bildschirmfoto%202017-10-06%20um%2016.59.38.png

2.2 Maschinengenauigkeit unten -> oben oben -> unten

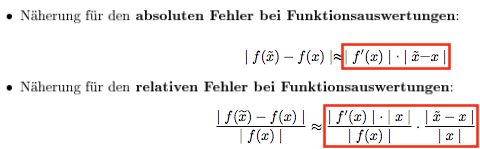
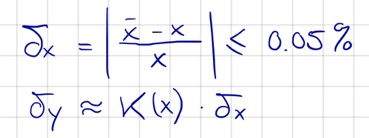
**Definition**

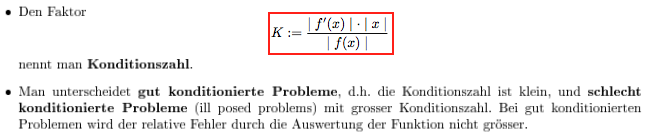


in Prozent

2.3 Absoluter/Relativer Fehler & Konditionszahl

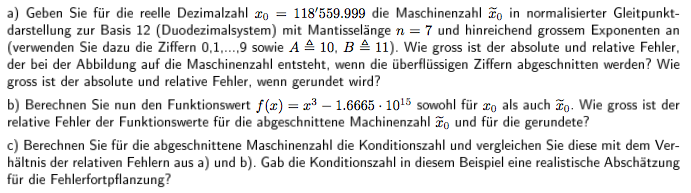
**Definitionen**

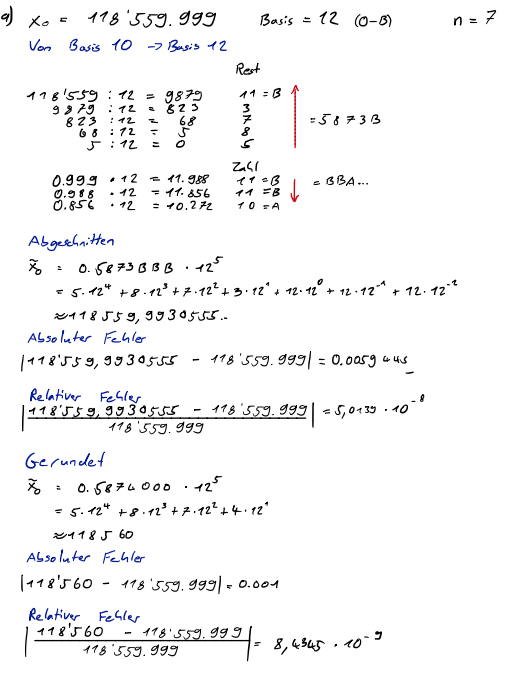
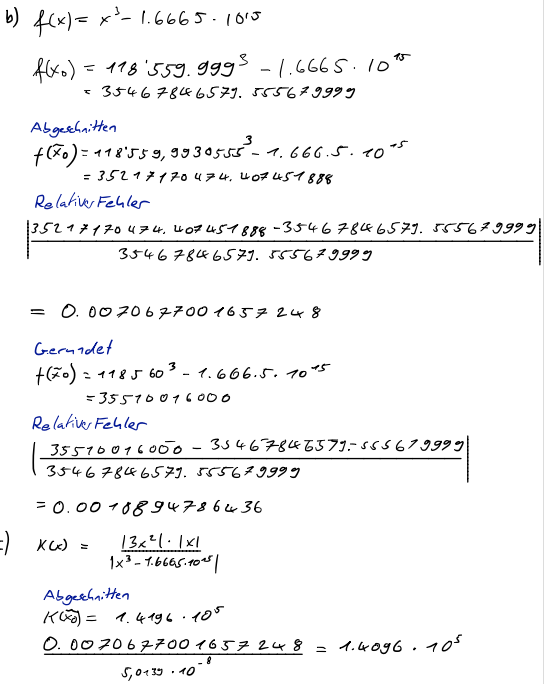
 in Prozent

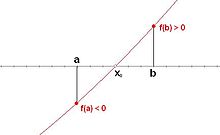


Beispiel

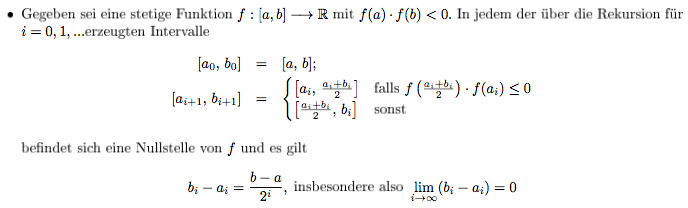
1) Relativer/Absoluter Fehler

****

****

**3 Numerische Lösungen von Nullstellenproblemen**

3.1 Bisektionsverfahren

****

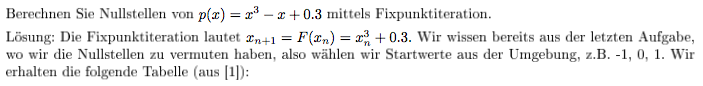
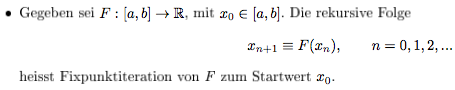
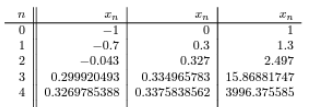
Funktioniert für:

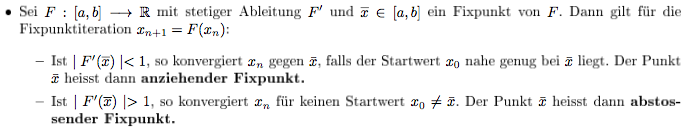
-stetige allg. Funktionen

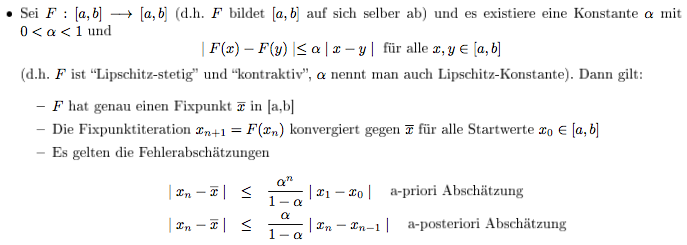
-a & b **richtig** wählen !!!

kommt fkt. von oben oder unten

3.2 Fixpunktiteration, anziehender/abstossender Fixpunkt

****

****

****3.3 Banachscher Fixpunktsatz

**-> Fixpunkt muss konvergieren**

**-> grösste Steigung wählen beim**

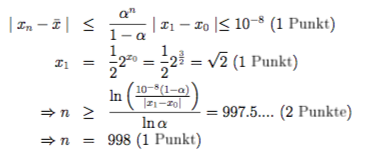
**einsetzen**

**-> Prüfen ob konvegiert:  
 Intervall einsetzen**

**a < 1 sein**

../../Desktop/Bildschirmfoto%202017-10-28%20um%2002.16.08.png

../../Desktop/Bildschirmfoto%202017-10-28%20um%2002.22.01.png



Beispiel - spezifisch

Gesucht: • Intervall [a, b] • Konstante a (a<1 ein muss)

Gegeben: • f(x) = x3 - x + 0.3

--> x = x3 + 0.3 = F(xn) = xn3 + 0.3

Lösung:

1. Fixpunkt finden (y konvergiert gegen Konstante) ≈ **0.3389**

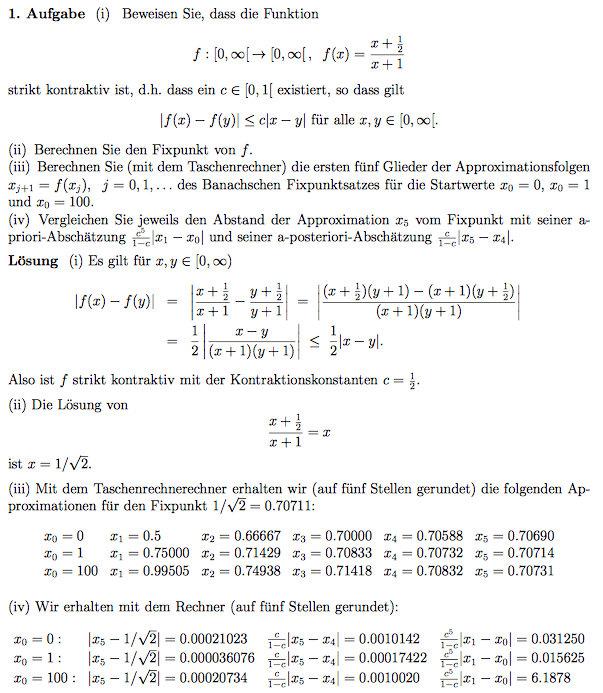
2. Intervall bestimmen zB = **[0, 0.5]** für das gilt: F(xa) = xa3 + 0.3 ≥ 0.3 und F(xb) = xb3 + 0.3 ≤ 0.5

Intervall prüfen: F(0) = 03 + 0.3 = 0.3 F(0.5) = 0.53 + 0.3 = 0.425

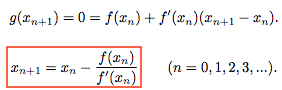
3. a < 1 für das gilt : |F(x) - F(y)| ≤ a|x - y| also max. Ableitung berechnen amax = |F'(xa oder b)| (je nach Fkt. a oder b nehmen)

F'(x) = 3x2 F'(0.5) = 3\*0.52 = 0.75 < 1 und F'(0) = 0 a = 0.75

Beispiel - allgemein Beweisen

Wenn keine Fixpunktfunkt. möglich ist wie zB f(x) = dann = x setzen (siehe heavyFixpunkt paper)  


3.4 Newton Verfahren

****

**Vereinfachtes Newton Verfahren Sekanten Verfahren**

**../../Desktop/Bildschirmfoto%202017-10-28%20um%2002.34.32.png**../../Desktop/Bildschirmfoto%202017-10-28%20um%2002.37.48.png

**Konvergiert**:**../../Desktop/Bildschirmfoto%202017-10-28%20um%2002.32.22.png**

Beispiel

Gegeben: • ex^2 + x-3 = 10

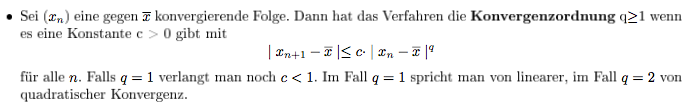
• [-3, 3] 3 Lösungen/Nullstellen (2, 0.5, -1 & -1.2)

Gesucht: Nullstellen (Schnittpunkte)

**Lösung** (Serie 6\_Aufg 1):

1. Fkt. umformen = f(x) = ex^2 g(x) = 10 - x-3

2. Verfahren anwenden

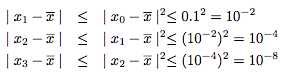
3.5 Konvergenzgeschwindigkeit

Beispiel

Gegeben: c = 1

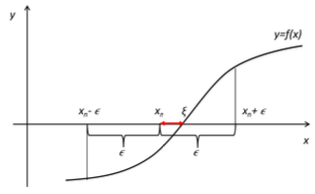
|x0 - |≤ 0.1

Gesucht: Wie schnell konverigert?

Lösung:

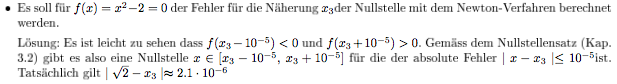
Für **einfache** Nullstellen konvergiert **Newton-Verf. quadratisch**, **vereinf. Newton-Verf. linear** und **Sekanten-Verf.** (1 + 50.5)/2 = **1.618**

3.6 Fehlerabschätzung

../../Desktop/Bildschirmfoto%202017-10-28%20um%2003.11.40.png../../Desktop/Bildschirmfoto%202017-10-28%20um%2003.11.33.png

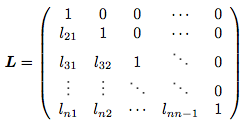
e finden, sodass es die gesuchte Nullstelle eindeckt

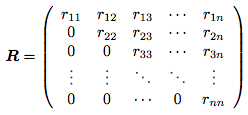
Beispiel

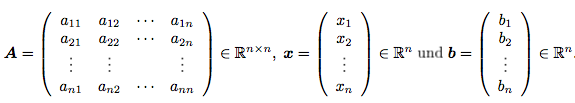


**4 Numerische Lösung linearen Gleichungssysteme**

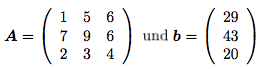
4.1 Gauss Algorithmus

**Matrizen Definition Normierte Dreiecks-Matrix Obere Dreiecks-Matrix**

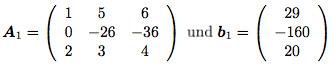
Ax = b



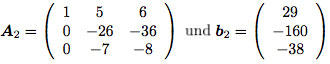
**Gauss Alg. (Reihenfolge der Schritte einhalten) --> l-Zeile ≠ 0 Determinante**

**** --> Diagonale der Dreiecks-Matrix multiplizieren

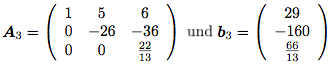
ll - (7/1)\*l (= ll - λl)



lll - (2/1)\*l



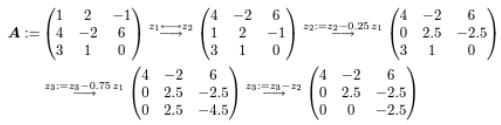
lll - (7/26)\*ll



../../Desktop/Bildschirmfoto%202017-11-26%20um%2016.42.29.png

**Spaltenpivotisierung**

Ziel: Fehlerminimierung von λ --> In Spalte immer höchstmöglichste Zahl pro Step

****

4.2 LR-Zerlegung

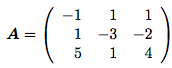
Ziel: Ax = b --> LRx = b

1. Ly = b

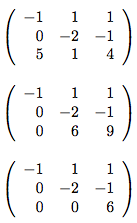
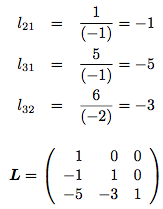
2. Rx = y

--> Ax = LRx = Ly = b

Bsp.



1. Ax = b lösen = **R** 2. **L** erzeugen mit λ von R & untere Dreiecks-Matrix & Diagonale 1

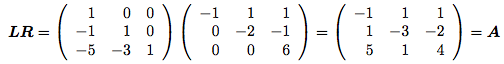


ll - (1/-1)l ( = ll - λl )

lll - (5/-1)l

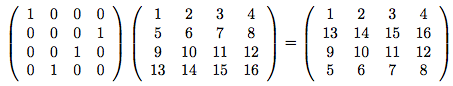
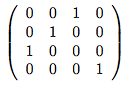
lll - (6/-2)ll

= **R**



4.3 LR-Zerlegung durch Zeilenvertauschung (Multiplitaktion P)

ll mit lV -Zeile vertauschen l mit lll -Zeile vertauschen = Multip. mit 1er Matrix



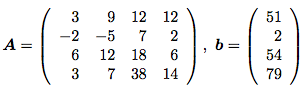
Somit nichts anders als PA = LR, zu lösen

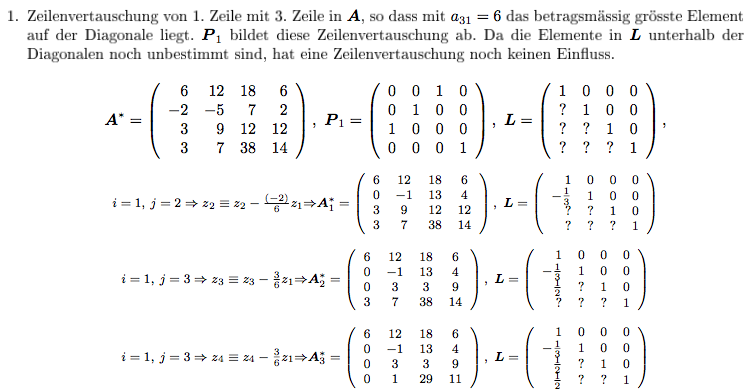
1. Ly = Pb

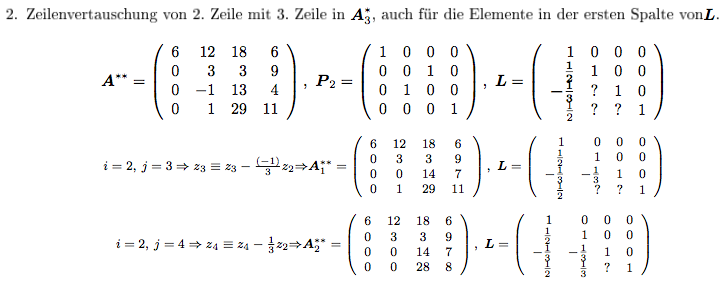
2. Rx = y --> Zeilenvertauschen hat Auswirkung auf L, ausser bei der ersten Vertauschung. **Danach nur λ vertauschen**

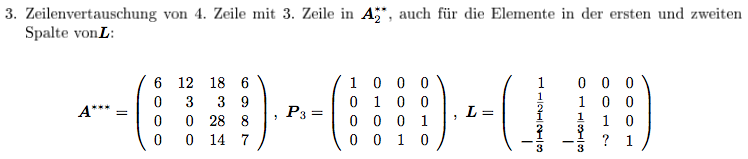
Bsp.

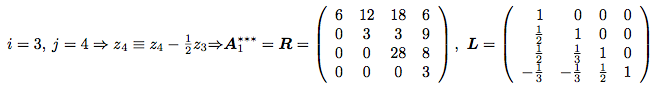
../../Desktop/Bildschirmfoto%202017-11-26%20um%2018.01.15.png

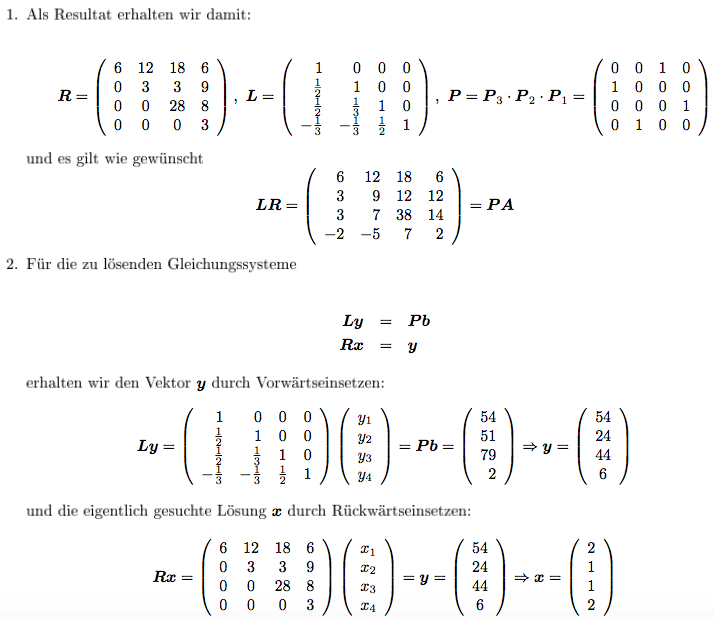








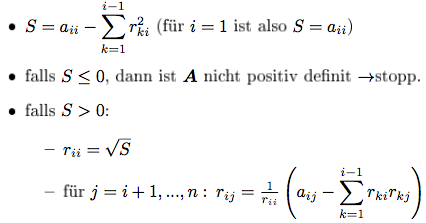


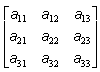
****

4.5 Cholesky-Zerlegung (Spezielle LR-Zerlegung)

**Definition**

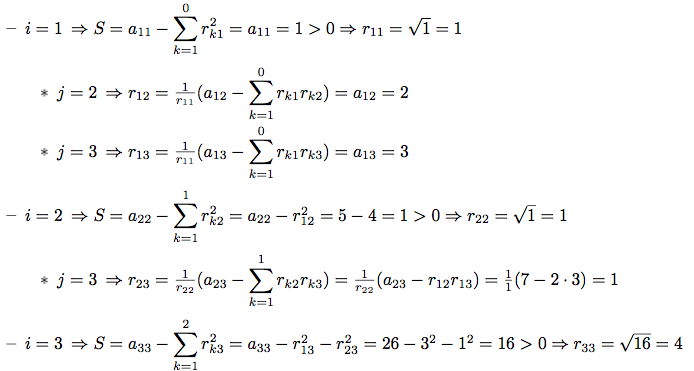
**../../Desktop/Bildschirmfoto%202017-11-26%20um%2018.22.09.png**

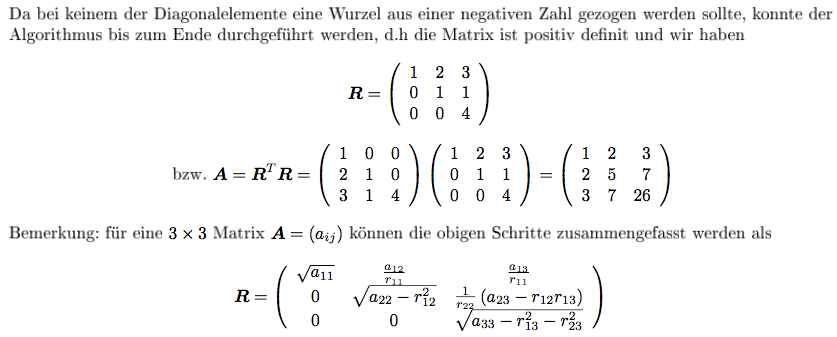


****Bsp.

**../../Desktop/Bildschirmfoto%202017-11-26%20um%2018.23.38.png**

1. AT = A, sprich A ist symmetrisch, also können wir den Cholesky Alg. anweden

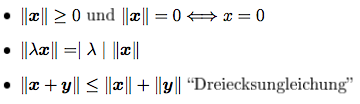
****2.

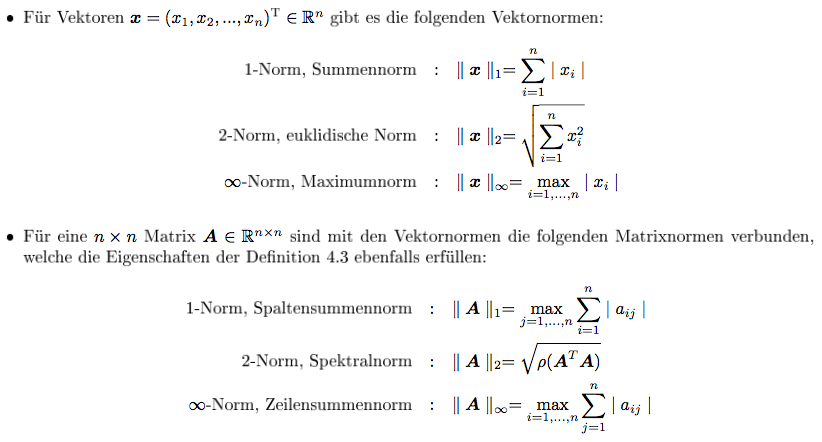
****

**5 Fehlerrechnung & Aufwandabschätzung**

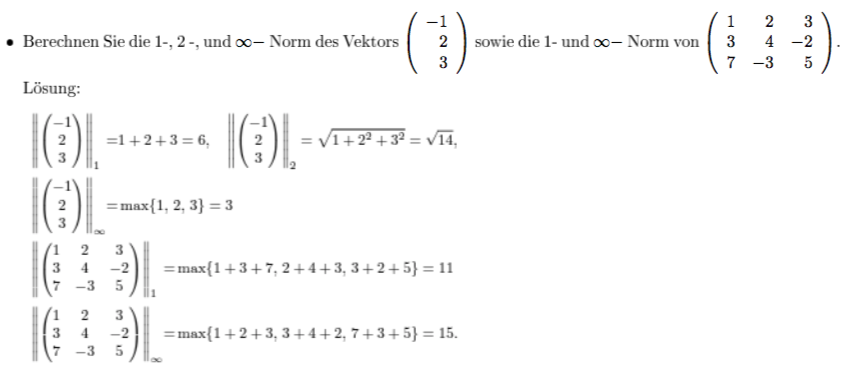
5.1 Vektornormen

**Definition**



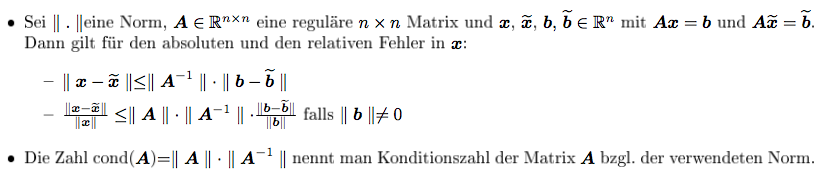


Bsp.



5.2 Fehlerbehafteter x-Vektor

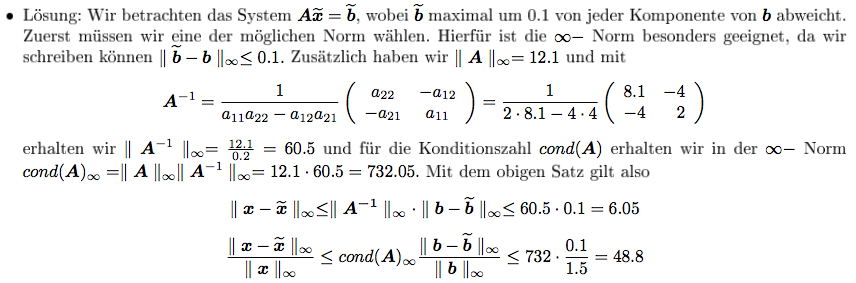
**../../Desktop/Bildschirmfoto%202017-11-26%20um%2020.16.21.png**

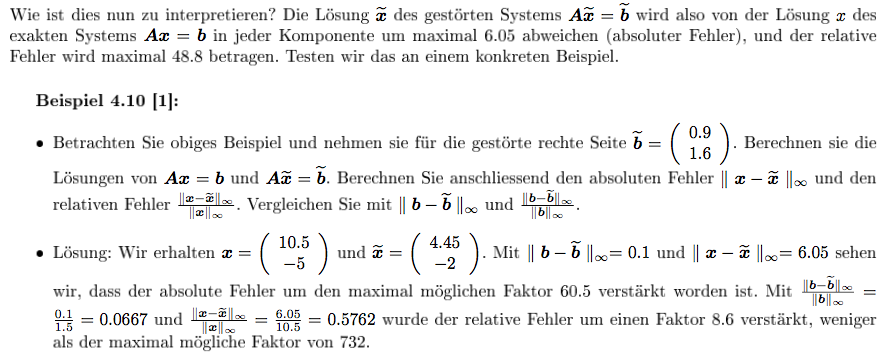


Bsp

maximal 0.1 von b abweichen.

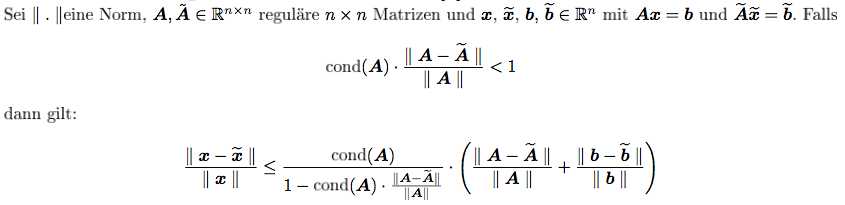
../../Desktop/Bildschirmfoto%202017-11-26%20um%2020.25.43.png

**1. Norm wählen: ∝-Norm 2. A-1 berechnen inv(A) 3. cond(A) berechnen: ||A||∝||A-1||∝**



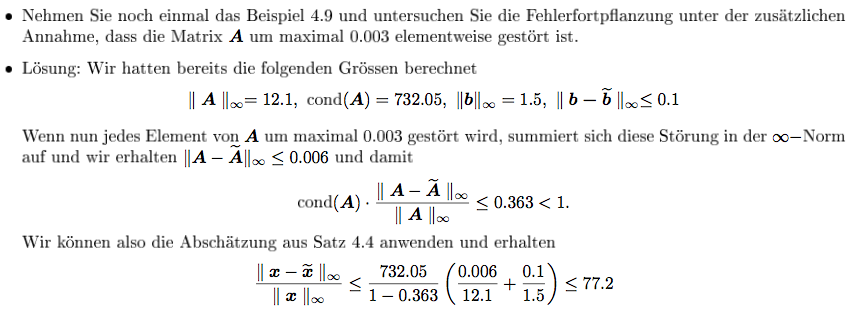
5.3 Fehlerbehaftete A-Matrix

**Definition**

../../Desktop/Bildschirmfoto%202018-01-10%20um%2016.00.43.png****

../../Desktop/Bildschirmfoto%202017-11-26%20um%2020.37.37.png

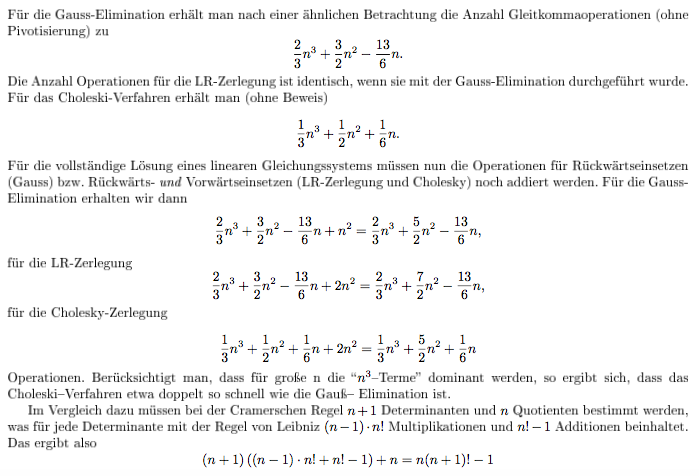
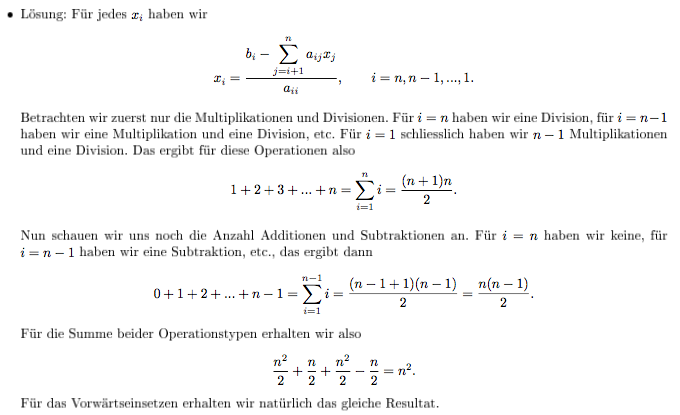
Bsp



5.4 Aufwandabschätzung

**Definition**

**../../Desktop/Bildschirmfoto%202017-12-03%20um%2017.57.33.png**

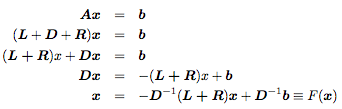
****

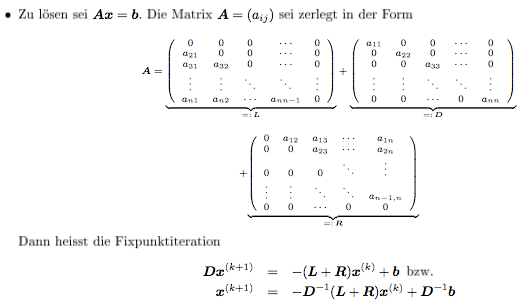
**6. Iterativ-Verfahren**

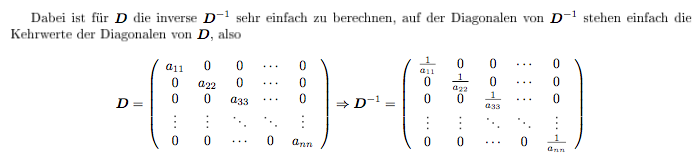
**../../Desktop/Bildschirmfoto%202017-12-03%20um%2018.27.51.png**

6.1 Jacobi-Verfahren (eig. nur Gleichung wichtig)

**Definition**

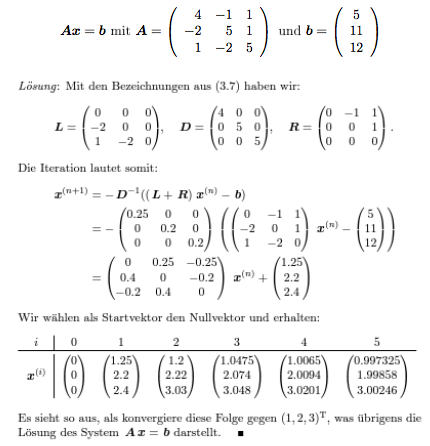


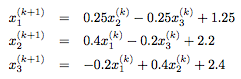


****

4.13 Bsp. 0 = Start-Vektor

**Programmier-Implementation**

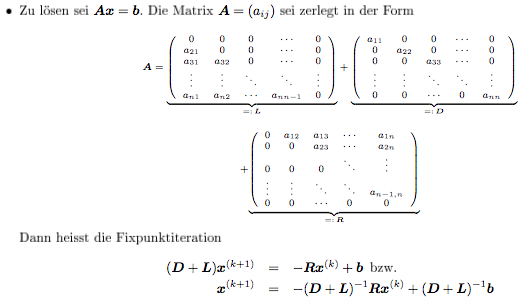
../../Desktop/Bildschirmfoto%202017-12-03%20um%2018.36.40.png

****

6.2 Gauss-Siedel-Verfahren (eig. nur Gleichung wichtig) Iterationsverfahren

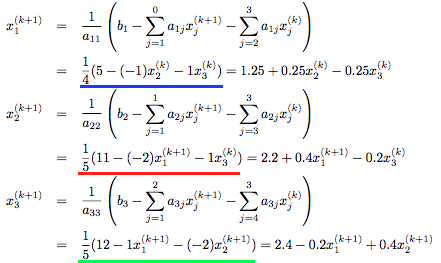
**../../Desktop/Bildschirmfoto%202017-12-03%20um%2018.46.36.pngDefinition**

**../../Desktop/Bildschirmfoto%202017-12-03%20um%2018.45.06.png =**

****

Bsp. 0 = Start-Vektor

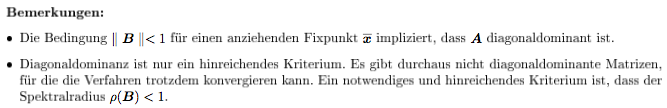
../../Desktop/Bildschirmfoto%202017-12-03%20um%2018.57.49.png



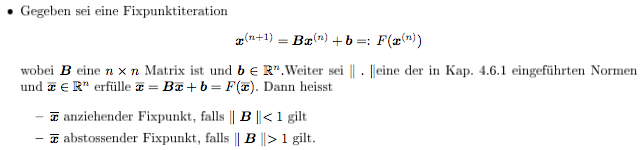


6.3 Konvergenz

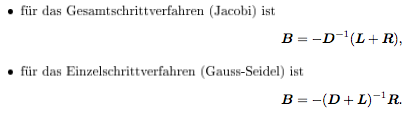
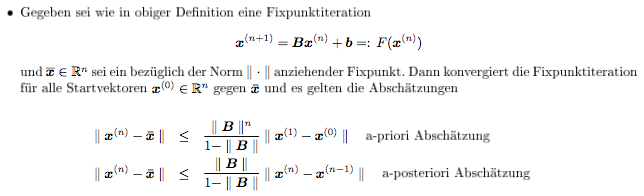
**Konvergenz**

**../../Desktop/Bildschirmfoto%202017-12-03%20um%2020.49.27.png**

**Fixpunkt**

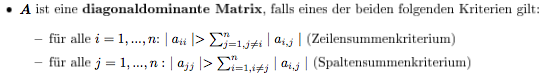


**Abschätzung**

****

**Iterationsmatrix B berechnen**

**Diagonaldominanz - OB MATRIX KONVEGIERT**

****

Z.sum: Z.|addition| ohne diagonale < |diagonale|

S.sum: dito für S.

Bsp

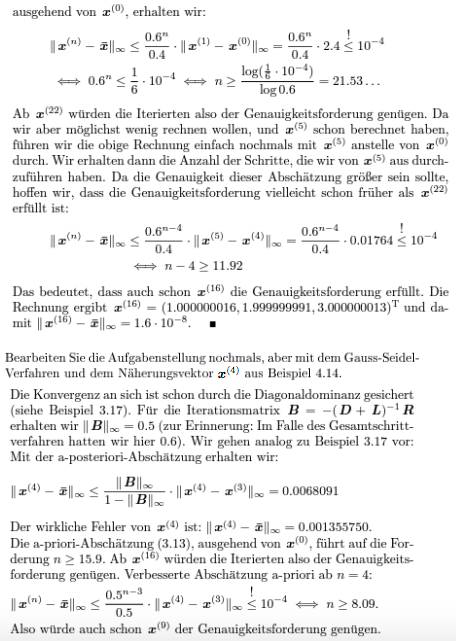
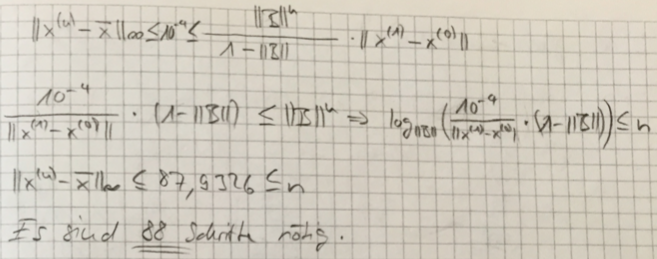
-Prüfen ob Matrix A konvegiert.

-Fehlerabschätzen des Vekors x(5).  
-Wieviel Schritte rechnen, damit berechneter **Näherungsvektor**  in jeder Komponente um max. 10-4 von exakten Lösung

**x = (1 2 3)T** abweicht? Vergleichen.

**Fehlerabschätzung der ∝-Norm mit Iterationsmatrix B**

**--> flasche Werte von x(4) für x(5) genommen**

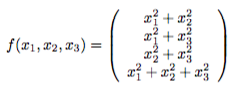
****

**7 Numerische Lösung von Nullstellenproblem**

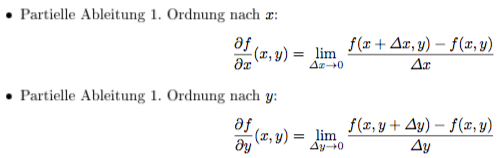
7.1 Funktionen mit mehreren Variablen  
**Definition**Jedem Zahlentupel x1 x2 ... xn aus D ⊂ Rn wird genau ein y aus W ⊂ R zugeordnet.

**../../Desktop/Bildschirmfoto%202017-12-18%20um%2015.39.37.png**

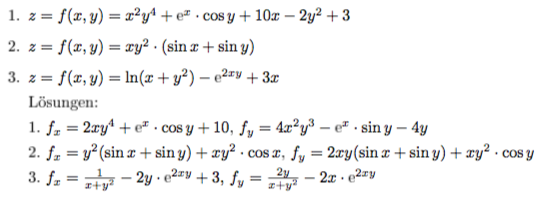
**Addition & Multiplikation Ohmsches Gesetz Reihenschaltung Vektorwertige Fkt**

**../../Desktop/Bildschirmfoto%202017-12-18%20um%2015.49.44.png../../Desktop/Bildschirmfoto%202017-12-18%20um%2015.49.19.png../../Desktop/Bildschirmfoto%202017-12-18%20um%2015.49.38.png**R2 --> R R2 --> R RGesamt R3 --> R4

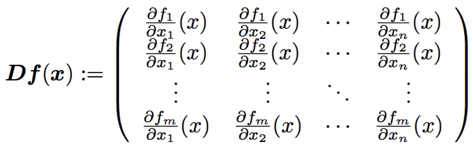
7.2 Partielle Ableitung 1. Ordnung



Bsp.



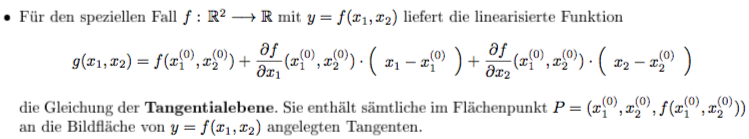
**Jacobi Matrix: pratielle Ableitung 1. Ordnung**



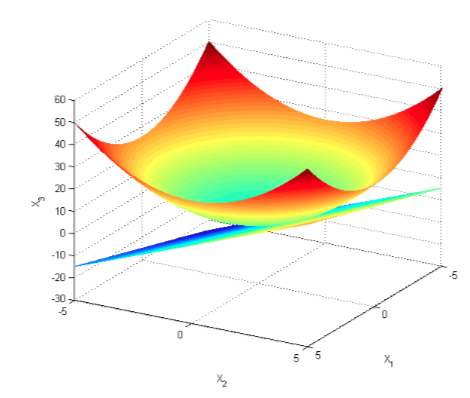
**Tangentengleichung**

../../Desktop/Bildschirmfoto%202017-12-18%20um%2019.29.15.png

**Liniearisierte Funktion**

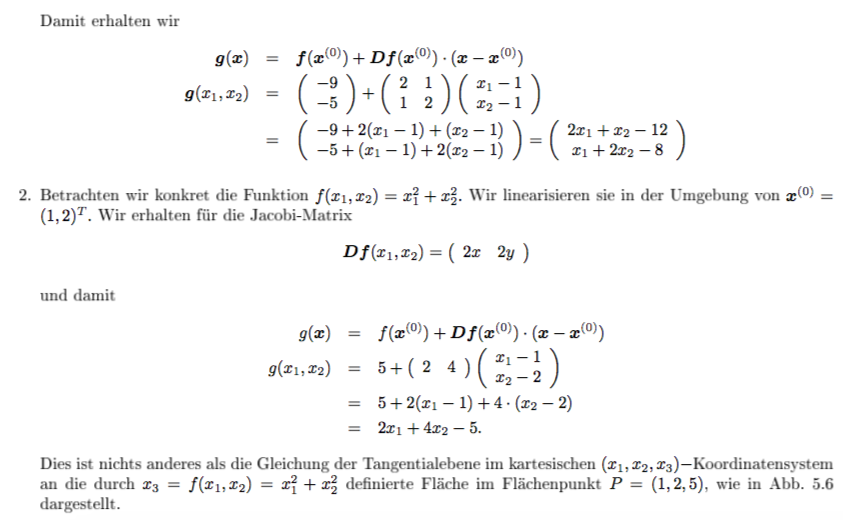


Bsp.

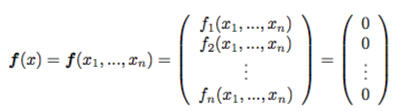
../../Desktop/Bildschirmfoto%202017-12-18%20um%2019.37.28.png../../Desktop/Bildschirmfoto%202017-12-18%20um%2019.36.45.png

Lösung

../../Desktop/Bildschirmfoto%202017-12-18%20um%2019.36.59.png../../Desktop/Bildschirmfoto%202017-12-18%20um%2019.40.30.png



7.3 Problemstellung zur Nullstellenbestimmung für nichtlineare Systeme



Bsp.

../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.01.48.png

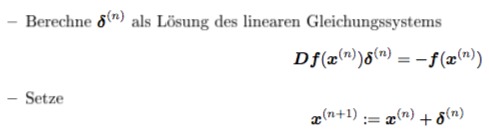
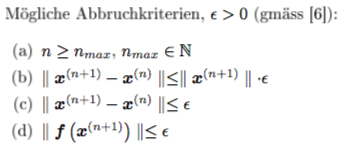
../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.01.41.png

7.4 Das Newton-Verfahren für Systeme

../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.17.52.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.17.45.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.21.13.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.37.52.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.21.06.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.19.11.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.19.18.png **Herleitung**

../../Desktop/Bildschirmfoto%202017-12-19%20um%2018.31.11.png

7.5 Quadratisch konvergentes Newton-Verfahren

../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.43.24.png

Wenn sich **keine 0-Stelle** sondern ein **lokales Minimum** ergibt, dann **Df(xmin) = nicht regulär**

Bsp. **1 von 3 Nullstellen (-2,1)T (2,-1)T & 0**

Geg.:

../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.46.41.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.46.53.png**../../Desktop/Bildschirmfoto%202017-12-19%20um%2017.48.37.png**

Lsg.:

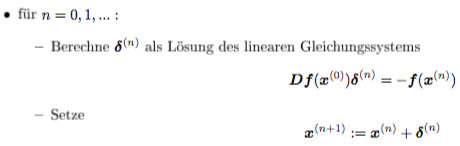
**../../Desktop/Bildschirmfoto%202017-12-19%20um%2018.18.27.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2018.18.20.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2018.10.05.png1. Ableiten 2. Einsetzen 3. Iteration**

../../Desktop/Bildschirmfoto%202017-12-19%20um%2018.33.35.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2018.33.02.png../../Desktop/Bildschirmfoto%202017-12-19%20um%2018.31.11.pngBsp. **Keine Nullstellen**

7.6 Vereinfachtes Newton-Verfahren

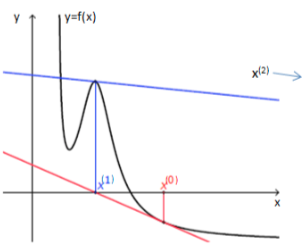
**Ziel**

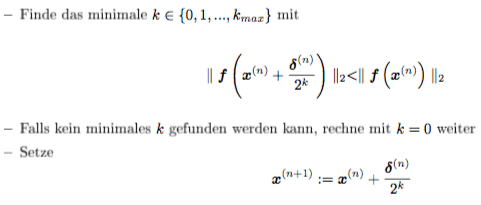
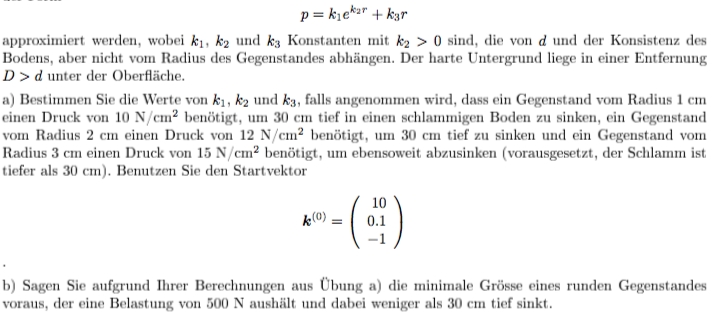
Df(x(n)) nicht jedes mal neu ausrechnen, sondern immer Df(x(0)) verwenden. Konvergiert linear.



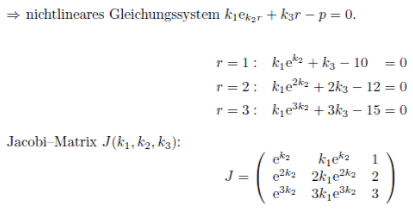
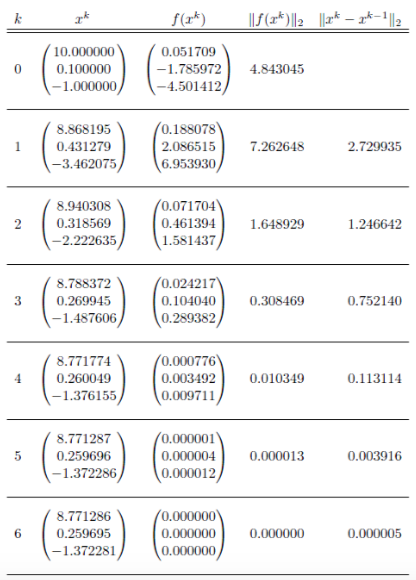
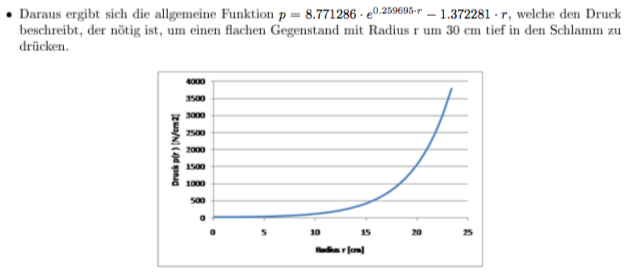
Bsp.

Dito wie Newton-Verfahren einfach nur erstes δ berechnen.

****7.7 Gedämpftes Newton-Verfahren   
**Falls schlecht oder gar nicht konvergiert**



Bsp.

a)

b)