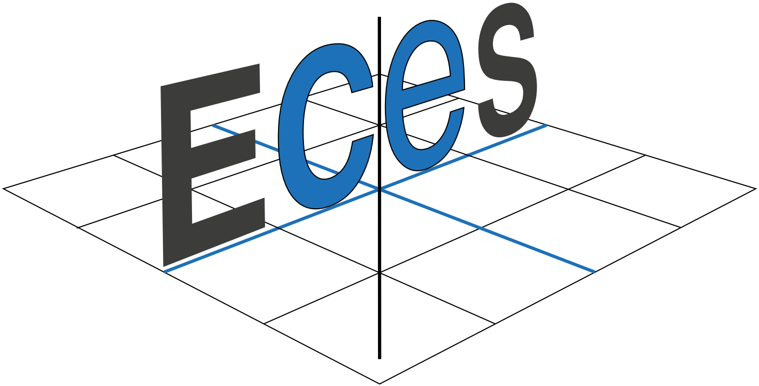
Wie wirtschaftlich ist mein Betrieb?

Denis Andrić, Bastian Koch, Dirk Schweickard

TECHNISCHE UNIVERSITÄT DARMSTADT



Inhaltsverzeichnis

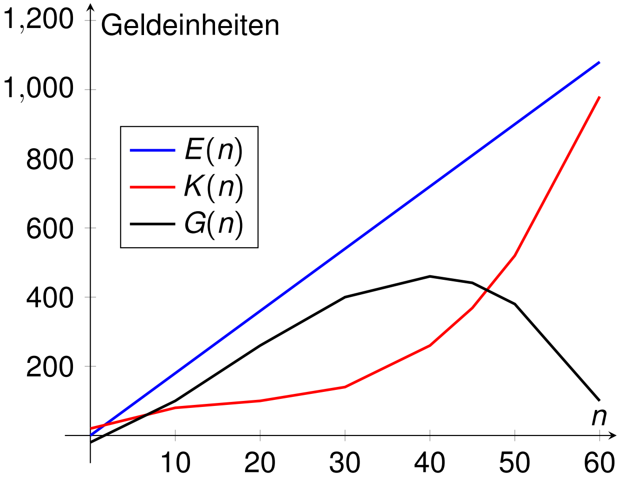
Inhaltsverzeichnis

Gewinnfunktion

$$G(n) = E(n) - K(n)$$

- $n \rightarrow$ Stückzahl
- $E \rightarrow$ Erlös
- $K \rightarrow$ Kosten
- $G \rightarrow$ Gewinn

Graphen der Beispieldaten von [Son01]



Ziele der mathematischen Methoden

- Berechnung einer reellen Gewinnfunktion aus den gegebenen Daten

Ziele der mathematischen Methoden

- Berechnung einer reellen Gewinnfunktion aus den gegebenen Daten
- Gewinnmaximierung durch Berechnung der Maxima

Ziele der mathematischen Methoden

- Berechnung einer reellen Gewinnfunktion aus den gegebenen Daten
- Gewinnmaximierung durch Berechnung der Maxima
- Nullstellenberechnung zur Ermittlung der Gewinnzone

Inhaltsverzeichnis

Ausgangssituation

gegeben

- Argumentmenge $A \subset \mathbb{R}$

Ausgangssituation

gegeben

- Argumentmenge $A \subset \mathbb{R}$
- Bildmenge $B \subset \mathbb{R}$

Ausgangssituation

TECHNISCHE UNIVERSITÄT DARMSTADT

gegeben

- Argumentmenge $A \subset \mathbb{R}$
- Bildmenge $B \subset \mathbb{R}$
- Funktion $F\colon A \to B$

Ausgangssituation

TECHNISCHE UNIVERSITÄT DARMSTADT

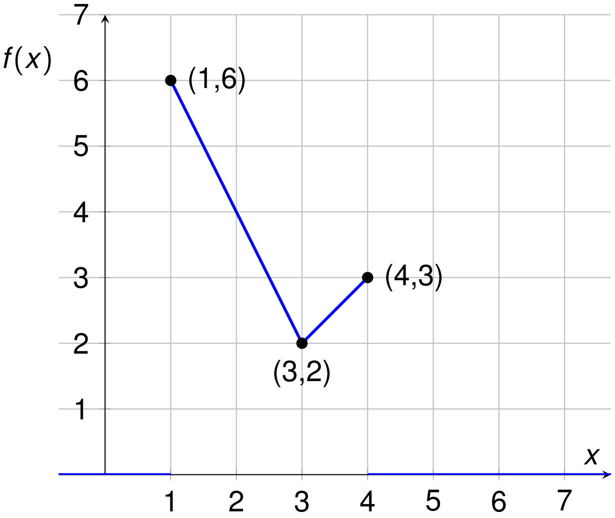
gegeben

- Argumentmenge $A \subset \mathbb{R}$
- Bildmenge $B \subset \mathbb{R}$
- Funktion $F\colon A \to B$

gesucht

- Reelle Funktion $f : \mathbb{R} \to \mathbb{R}$, sodass $\forall a \in A\colon f(a) = F(a)$ gilt

Interpolierender Spline

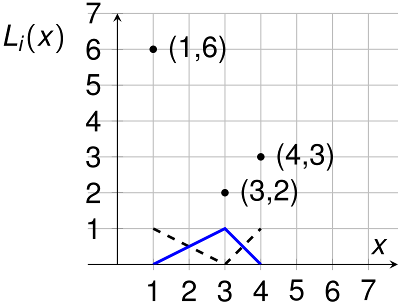


Sortierung der Daten nach Größe des Arguments

$$\forall i \in \{i \in \mathbb{N} \colon 0 \leq i \leq n-1\} \colon a_i < a_{i+1} \text{ mit } a_i, a_{i+1} \in A$$
$$\forall i \in \{i \in \mathbb{N} \colon 0 \leq i \leq n\} \colon F(a_i) = b_i$$

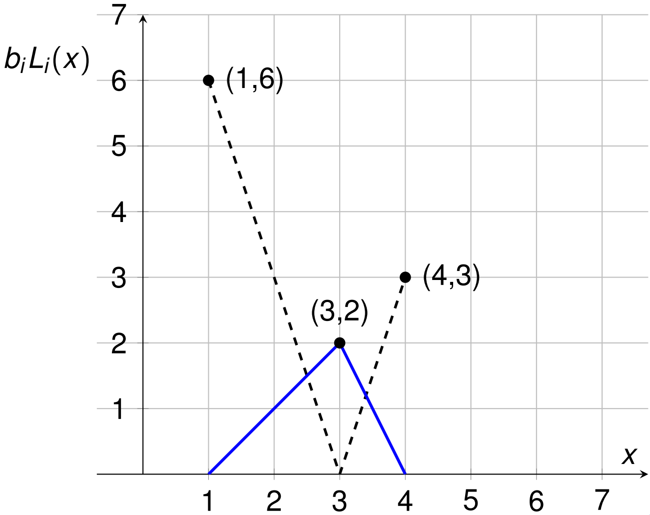
- a_i und a_{i+1} werden im Folgenden als benachbart bezeichnet.

Linearer Spline

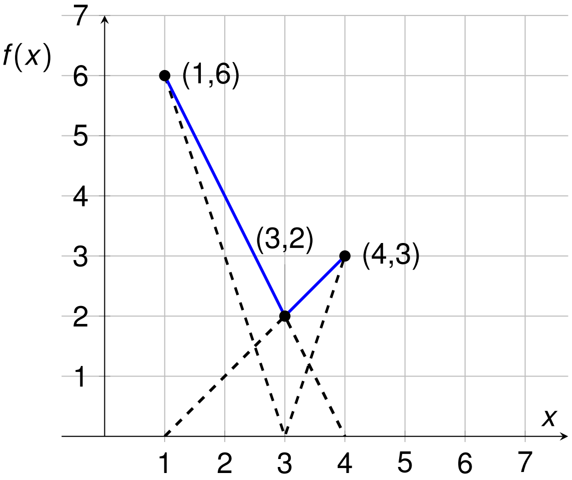


$$L_i(x) := \begin{cases} \frac{x-a_{i-1}}{a_i-a_{i-1}} & \text{für } x \in [a_{i-1}, a_i], \\ \frac{a_{i+1}-x}{a_{i+1}-a_i} & \text{für } x \in (a_i, a_{i+1}], \\ 0 & \text{sonst} \end{cases}$$

Gewichteter linearer Spline $b_i L_i$



Interpolierender Spline



$$f : [a_0, a_n] \to \mathbb{R}$$

$$f(x) = \sum_{i=0}^{n} b_i L_i(x)$$

Extremstellenbestimmung bei linearen Splines

TECHNISCHE UNIVERSITÄT DARMSTADT

Menge der Stellen, an denen das globale Maximum angenommen wird

$A_M := \{a_i \in A \colon \forall a \in A : f(a_i) \geq f(a)\}$

Extremstellenbestimmung bei linearen Splines

Menge der Stellen, an denen das globale Maximum angenommen wird

$$A_M := \{a_i \in A \colon \forall a \in A : f(a_i) \geq f(a)\}$$

Menge der Stellen, an denen lokale Maxima angenommen werden

$$A_m := \{a_i \in A \colon f(a_{i-1}) < f(a_i) > f(a_{i+1})\}$$

Interpolationspolynom

gegeben

- Argumentmenge $A \subset \mathbb{R}$
- Bildmenge $B \subset \mathbb{R}$
- Funktion $F\colon A \to B$

gesucht

- Polynom $f(x) = \sum_{i=0}^{n} k_i x^i \in \Pi_n$ mit $n = |A| - 1$, sodass $\forall a \in A\colon f(a) = F(a)$ gilt.

Eindeutigkeit des Polynoms

- Annahme der Existenz eines zweiten Polynoms $g \in \Pi_n$, das die Daten korrekt modelliert

Eindeutigkeit des Polynoms

- Annahme der Existenz eines zweiten Polynoms $g \in \Pi_n$, das die Daten korrekt modelliert
- Bildung der Differenz $f - g$

Eindeutigkeit des Polynoms

- Annahme der Existenz eines zweiten Polynoms $g \in \Pi_n$, das die Daten korrekt modelliert
- Bildung der Differenz $f - g$
- $f - g$ hat an allen $|A| = n + 1$ gegebenen Stellen eine Nullstelle.

Eindeutigkeit des Polynoms

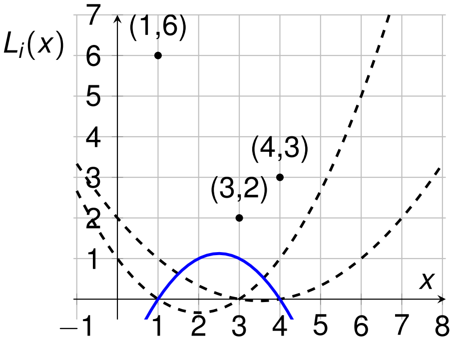
- Annahme der Existenz eines zweiten Polynoms $g \in \Pi_n$, das die Daten korrekt modelliert
- Bildung der Differenz $f - g$
- $f - g$ hat an allen $|A| = n + 1$ gegebenen Stellen eine Nullstelle.
- Polynome in Π_n können jedoch höchstens n Nullstellen haben; es ergibt sich ein Widerspruch.

Eindeutigkeit des Polynoms

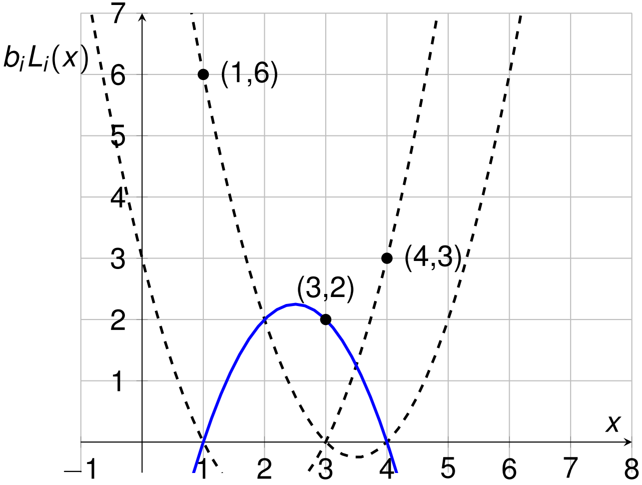
- Annahme der Existenz eines zweiten Polynoms $g \in \Pi_n$, das die Daten korrekt modelliert
- Bildung der Differenz $f - g$
- $f - g$ hat an allen $|A| = n + 1$ gegebenen Stellen eine Nullstelle.
- Polynome in Π_n können jedoch höchstens n Nullstellen haben; es ergibt sich ein Widerspruch.
- Durch Widerlegung des Gegenteils ist die Eindeutigkeit des Polynoms gegeben.

Lagrangesches Basispolynom

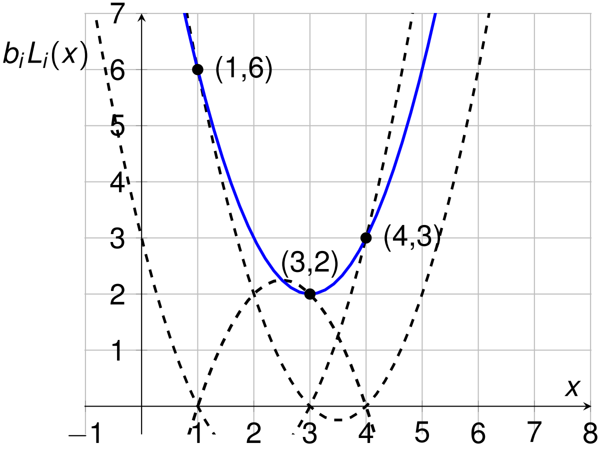
$$L_i(x) := \frac{\prod_{j=0}^{i-1}(x-a_j)\cdot\prod_{j=i+1}^{n}(x-a_j)}{\prod_{j=0}^{i-1}(a_i-a_j)\cdot\prod_{j=i+1}^{n}(a_i-a_j)}$$



Gewichtetes Basispolynomes $b_i L_i$



Interpolationspolynom



$$f : \mathbb{R} \to \mathbb{R}$$

$$f(x) = \sum_{i=0}^{n} b_i L_i(x)$$

Vandermondsche Matrizen

- Alternativer Lösungsansatz: Einsetzen aller Datenpunkte in die allgemeine Gleichung des Polynoms $f \in \Pi_n$

Vandermondsche Matrizen

- Alternativer Lösungsansatz: Einsetzen aller Datenpunkte in die allgemeine Gleichung des Polynoms $f \in \Pi_n$
- Es entsteht ein lineares Gleichungssystem:

$$\underbrace{\begin{bmatrix} 1 & a_0 & a_0^2 & \dots & a_0^n \\ 1 & a_1 & a_1^2 & \dots & a_1^n \\ 1 & a_2 & a_2^2 & \dots & a_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_n & a_n^2 & \dots & a_n^n \end{bmatrix}}_{M} \cdot \underbrace{\begin{bmatrix} k_0 \\ k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}}_{\vec{k}} = \underbrace{\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}}_{\vec{b}}$$

Vandermondsche Matrizen

- Alternativer Lösungsansatz: Einsetzen aller Datenpunkte in die allgemeine Gleichung des Polynoms $f \in \Pi_n$
- Es entsteht ein lineares Gleichungssystem:

$$\underbrace{\begin{bmatrix} 1 & a_0 & a_0^2 & \dots & a_0^n \\ 1 & a_1 & a_1^2 & \dots & a_1^n \\ 1 & a_2 & a_2^2 & \dots & a_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_n & a_n^2 & \dots & a_n^n \end{bmatrix}}_{M} \cdot \underbrace{\begin{bmatrix} k_0 \\ k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}}_{\vec{k}} = \underbrace{\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}}_{\vec{b}}$$

- Die Matrix M nennt man Vandermondsche Matrix

Vandermondsche Matrizen

- Alternativer Lösungsansatz: Einsetzen aller Datenpunkte in die allgemeine Gleichung des Polynoms $f \in \Pi_n$
- Es entsteht ein lineares Gleichungssystem:

$$\underbrace{\begin{bmatrix} 1 & a_0 & a_0^2 & \dots & a_0^n \\ 1 & a_1 & a_1^2 & \dots & a_1^n \\ 1 & a_2 & a_2^2 & \dots & a_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_n & a_n^2 & \dots & a_n^n \end{bmatrix}}_{M} \cdot \underbrace{\begin{bmatrix} k_0 \\ k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}}_{\vec{k}} = \underbrace{\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}}_{\vec{b}}$$

- Die Matrix M nennt man Vandermondsche Matrix
- Auflösen nach dem Gaußschen Eliminationsverfahren

Vandermondsche Matrizen

- Alternativer Lösungsansatz: Einsetzen aller Datenpunkte in die allgemeine Gleichung des Polynoms $f \in \Pi_n$
- Es entsteht ein lineares Gleichungssystem:

$$\underbrace{\begin{bmatrix} 1 & a_0 & a_0^2 & \dots & a_0^n \\ 1 & a_1 & a_1^2 & \dots & a_1^n \\ 1 & a_2 & a_2^2 & \dots & a_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_n & a_n^2 & \dots & a_n^n \end{bmatrix}}_{M} \cdot \underbrace{\begin{bmatrix} k_0 \\ k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}}_{\vec{k}} = \underbrace{\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}}_{\vec{b}}$$

- Die Matrix M nennt man Vandermondsche Matrix
- Auflösen nach dem Gaußschen Eliminationsverfahren
- Wegen Langwierigkeit nicht empfehlenswert

Inhaltsverzeichnis

TECHNISCHE UNIVERSITÄT DARMSTADT

Nullstellensuche

TECHNISCHE UNIVERSITÄT DARMSTADT

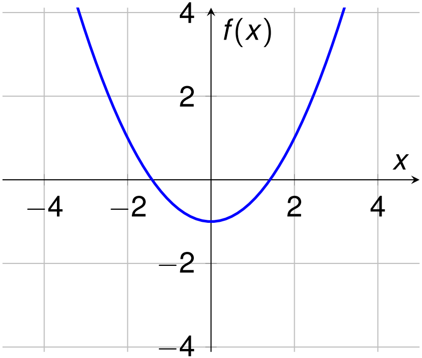


Abbildung: $f(x) = \frac{1}{2}x^2 - 1$

Bisektionsalgorithmus

- Einer der berühmtesten Algorithmen zur Bestimmung von Nullstellen von Funktionen
- Löwenfangalgorithmus, Intervallschachtelung

Vorbedingungen

TECHNISCHE UNIVERSITÄT DARMSTADT

Theorem

von Bolzano nach [Bog17] - sei eine Funktion $f(x)$ im Intervall $[a, b]$ stetig, sodass:

$$f(a) \cdot f(b) < 0$$

dann existiert ein $c \in (a, b)$:

$$f(c) = 0$$

Hauptschritte

TECHNISCHE UNIVERSITÄT DARMSTADT

Mittelpunkt

$$x = \frac{a+b}{2}$$

Neues Intervall

- wenn $f(x) \cdot f(b) < 0 \rightarrow [x, b]$
- wenn $f(x) \cdot f(a) < 0 \rightarrow [a, x]$

Abbruchbedingung

$$\frac{b-a}{2} < \varepsilon; \varepsilon \geq 0$$

Visualisierung

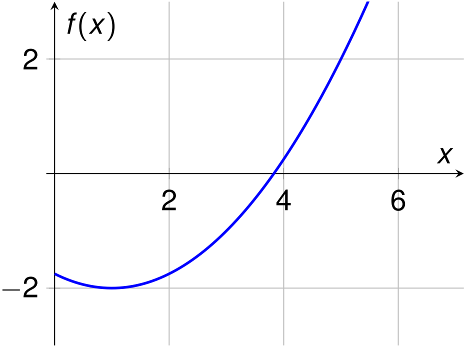


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_n	a_n	b_n	L_n
1				
2				
3				

Tabelle: Werte der Abbildung

Visualisierung

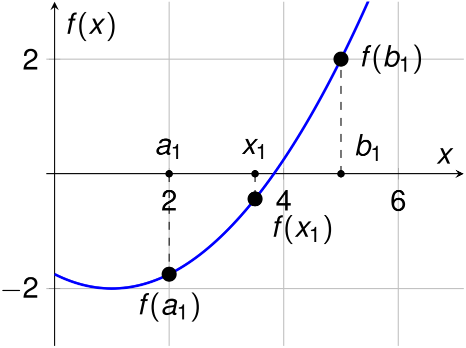


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_n	a_n	b_n	L_n
1	3.5	2	5	3
2				
3				

Tabelle: Werte der Abbildung

Visualisierung

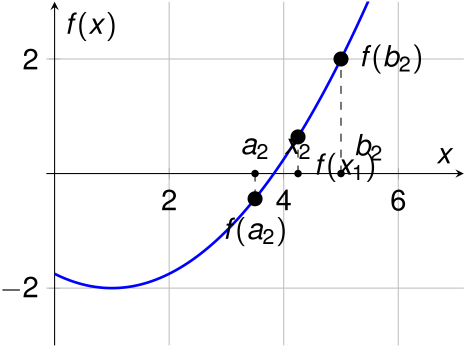


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_n	a_n	b_n	L_n
1	3.5	2	5	3
2	4.25	3.5	5	1.5
3				

Tabelle: Werte der Abbildung

Visualisierung

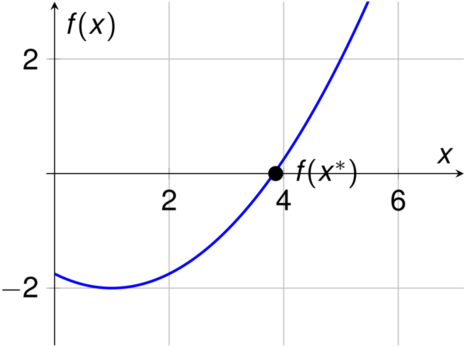


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_n	a_n	b_n	L_n
1	3.5	2	5	3
2	4.25	3.5	5	1.5
3	3.86	3.5	4.25	0.75
	3.83	3.83	3.83	0

Tabelle: Werte der Abbildung

Konvergenz

Aus Tabelle:

Konvergenz

- Nullstelle: $\lim_{n\to\infty} x_n = x^*$
- Intervalllänge: $\lim_{n\to\infty} L_n = \lim_{n\to\infty} \frac{b-a}{2^n} = 0$

Javaimplementierung


```
do {
        x = (a + b) / 2;
        if (f(a) * f(x) < 0)
                b = x;
        else
                a = x;
} while (Math.abs(b-a)/2 > e);
return x;
```

TECHNISCHE UNIVERSITÄT DARMSTADT

Regula Falsorum

- verbesserte Variante des Bisektionsalgorithmus
- konstruiert eine Näherung an die Nullstelle aus zwei falschen Werten
- Regel der Falschen oder Regula Falsi

Hauptschritte

Näherung an die Nullstelle durch Zweipunktgerade

$$x = a - f(b)\frac{b-a}{f(b)-f(a)} = \frac{f(b)\cdot a - f(a)\cdot b}{f(b)-f(a)}$$

Neues Intervall

- wenn $f(x) \cdot f(b) < 0 \rightarrow [x, b]$
- wenn $f(x) \cdot f(a) < 0 \rightarrow [a, x]$

Abbruchbedingung

$$|x_{n-1} - x_n| < \varepsilon; \varepsilon \geq 0$$

Visualisierung

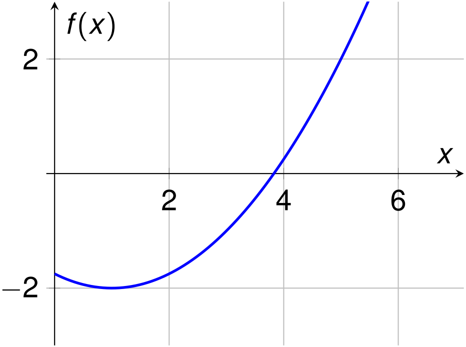


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_{n-1}	x_n	a_n	b_n
1				
2				
3				

Tabelle: Werte der Abbildung

Visualisierung

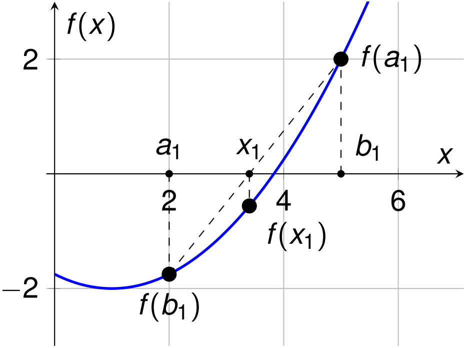


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_{n-1}	x_n	a_n	b_n
1	2	3.4	2	5
2				
3				

Tabelle: Werte der Abbildung

Visualisierung

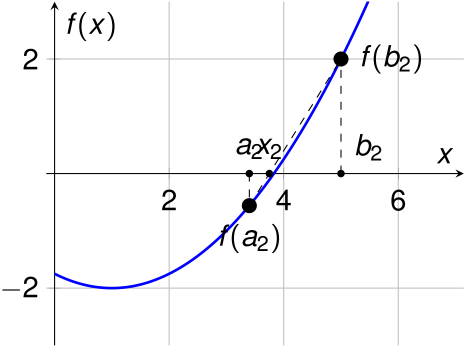


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_{n-1}	x_n	a_n	b_n
1	2	3.4	2	5
2	3.4	3.75	3.4	5
3				

Tabelle: Werte der Abbildung

Visualisierung

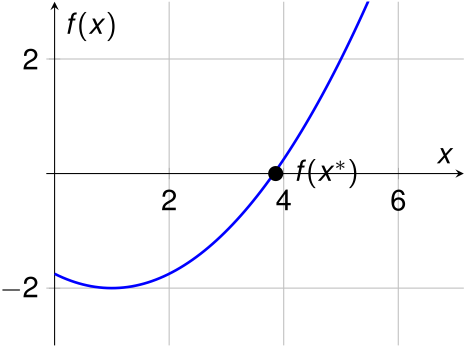


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_{n-1}	x_n	a_n	b_n
1	2	3.4	2	5
2	3.4	3.75	3.4	5
3	3.75	3.82	3.75	5
	3.83	3.83	?	?

Tabelle: Werte der Abbildung

Konvergenz

aus Tabelle:

Konvergenz

- Nullstelle: $\lim_{n\to\infty} x_{n-1} = \lim_{n\to\infty} x_n = x^*$
- $|x_{n-1} - x_n| \geq 0$

Java-Implementierung

```java
double x = a;
double x1;
do {
        x1 = x;
        x = (f(b) * a - f(a) * b) / (f(b) - f(a));

        if (f(a) * f(b) * f(x) <= 0)
                b = x;
        else
                a = x;
} while (Math.abs(x - x1) >= e);
return x;
```

Modifizierte Regula Falsorum

- Nullstellensuche mit der Regula Falsorum im Intervall $[-1, 2]$ der Funktion $f(x) = x^3$ mit $\varepsilon = 10^{-9}$ benötigt 792858 Iterationsschritte.
- Modifizierte Methoden wie z.B. der Illinois Algorithmus oder der Algorithmus von Anderson und Björk's
- Illinois Algorithmus benötigt 58 Iterationen

Hauptschritte

- Die erste Iteration ist ähnlich wie die der normalen Regula Falsorum

Neues Intervall

- $F = f(a), G = f(b)$
- $x_2 = \frac{G \cdot a - F \cdot b}{G - F}$

Neues Intervall

- wenn $f(x_2) \cdot f(b) < 0 \rightarrow [x_2, b]$
  wenn $f(x_1) \cdot f(x_2) > 0 \; G = \frac{G}{2}$
- wenn $f(x") \cdot f(a) < 0 \rightarrow [a, x_2]$
  wenn $f(x_1) \cdot f(x_2) > 0 \; F = \frac{F}{2}$

Visualisierung

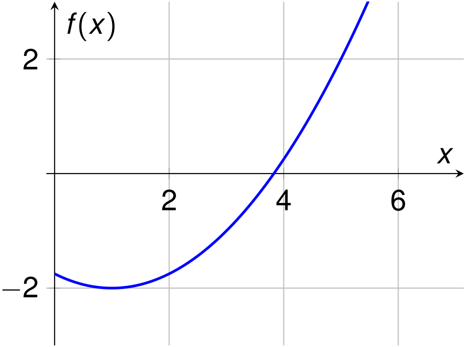


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_{n-1}	x_n	a_n	b_n
1				
2				
3				

Tabelle: Werte der Abbildung

Visualisierung

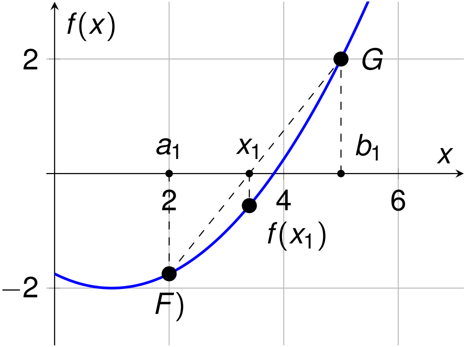


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_{n-1}	x_n	a_n	b_n
1	2	3.4	2	5
2				
3				

Tabelle: Werte der Abbildung

Visualisierung

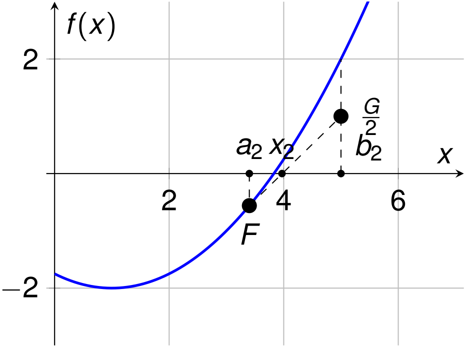


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_{n-1}	x_n	a_n	b_n
1	2	3.4	2	5
2	3.4	3.97	3.4	5
3				

Tabelle: Werte der Abbildung

Visualisierung

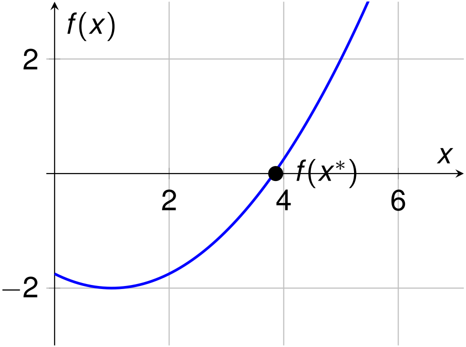


Abbildung: $f(x) = \frac{1}{4}(x-1)^2 - 2$

n	x_{n-1}	x_n	a_n	b_n
1	2	3.4	2	5
2	3.4	3.97	3.4	5
3	3.97	3.81	3.4	3.97
	3.83	3.83	?	?

Tabelle: Werte der Abbildung

Java-Implementierung

```
double x = a;
do {
        x1 = x;
        x = (G * a - F * b) / (G - F);

        if (f(a) * f(b) * f(x) <= 0) {
                b = x;
                G = f(x);
                if (f(x) * f(x1) > 0)
                        F = F / 2;
```

Java-Implementierung

```
        } else {
                a = x;
                F = f(x);
                if (f(x) * f(x1) > 0)
                        G = G / 2;
        }
} while (Math.abs(x - x1) > e);
```

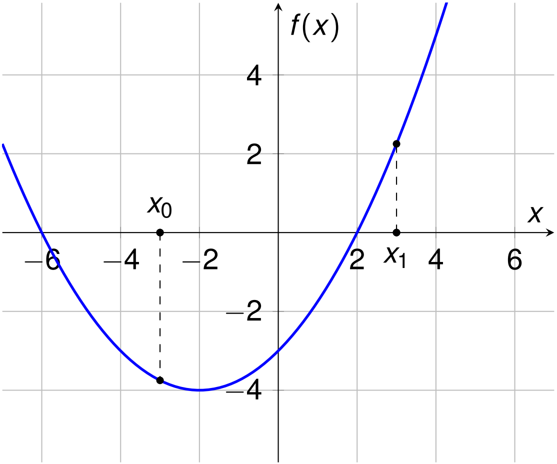
Allgemeines

- Numerisches Näherungsverfahren
- Benötigt zwei Startwerte x_0 und x_1

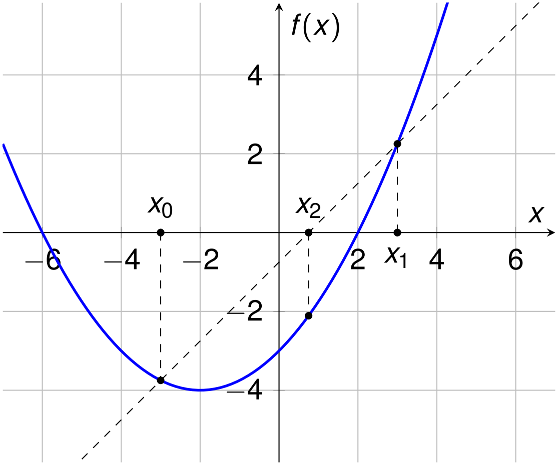
Vorbedingungen

- Funktion f muss stetig sein
- einfache Nullstelle x^*

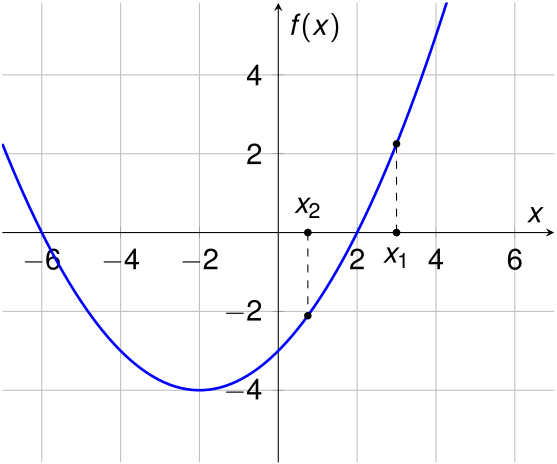
Beispiel



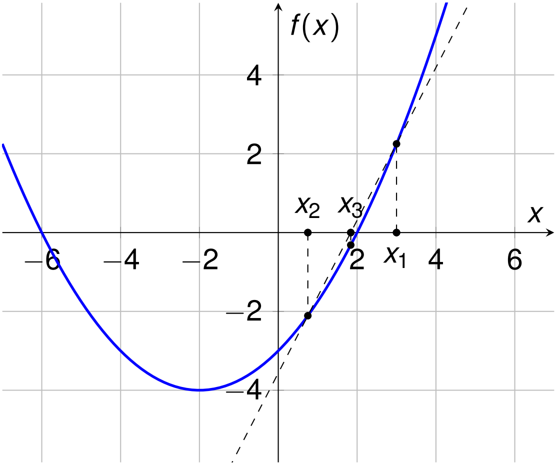
Beispiel



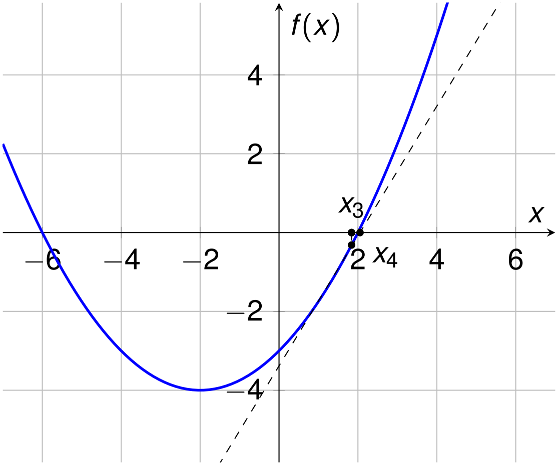
Beispiel



Beispiel



Beispiel



TECHNISCHE UNIVERSITÄT DARMSTADT

Sekantengleichung

$$y = \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_1) + f(x_1)$$

TECHNISCHE UNIVERSITÄT DARMSTADT

Sekantengleichung

$$y = \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_1) + f(x_1)$$

Nullstelle der Sekante

$$0 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_1) + f(x_1)$$

Mathematik

Nullstelle der Sekante

$$0 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_1) + f(x_1)$$

Umstellen nach x

$$x = \frac{f(x_1)x_0 - f(x_1)x_1}{f(x_1) - f(x_0)}$$

Mathematik

Umstellen nach x

$$x = \frac{f(x_1)x_0 - f(x_1)x_1}{f(x_1) - f(x_0)}$$

Iterationsvorschrift

$$x_{n+1} = \frac{f(x_n)x_{n-1} - f(x_{n-1})x_n}{f(x_n) - f(x_{n-1})}$$

Konvergenz

- keine garantierte Konvergenz
- Verfahren bricht zusammen falls gilt $f(x_n) = f(x_{n+1})$
- konvergiert linear mit Konvergenzordnung $\Phi = \frac{1+\sqrt{5}}{2} \approx 1,618$

Java-Implementierung

```
public static double Sekante(double eps, double x1,
 double x2){
        double dummy;
        int n=0;
        while (Math.abs(f2(x2)) > eps && n<1000){
                dummy = x2;
                x2 = (f2(x2)*x1-f2(x1)*x2)/(f2(x2)-
                x1 = dummy;
                n++;
        }
        return x2;
}
```

TECHNISCHE UNIVERSITÄT DARMSTADT

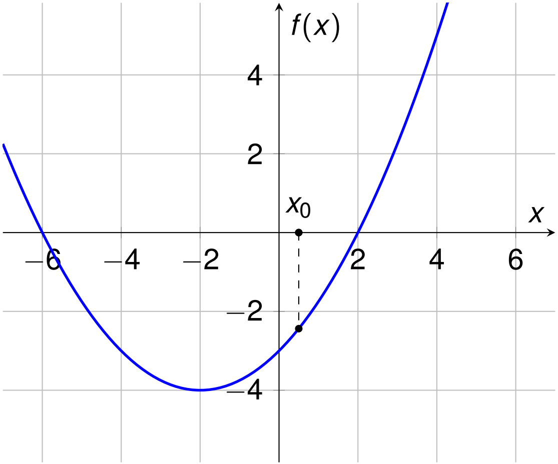
Allgemein

- numerisches Verfahren
- benötigt nur noch einen Startwert x_0

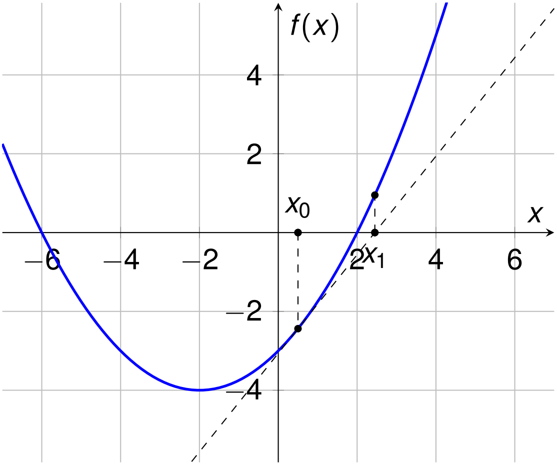
Vorbedingungen

- Funktion muss zweimal stetig partiell differenzierbar sein
- einfache Nullstelle x^*
- Startwert x_0 muss in der Umgebung von x^* sein

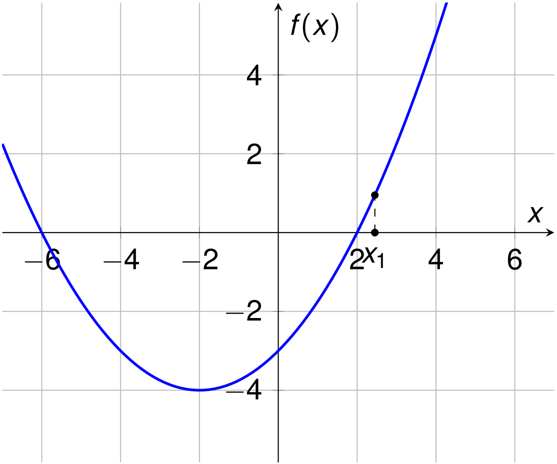
Beispiel



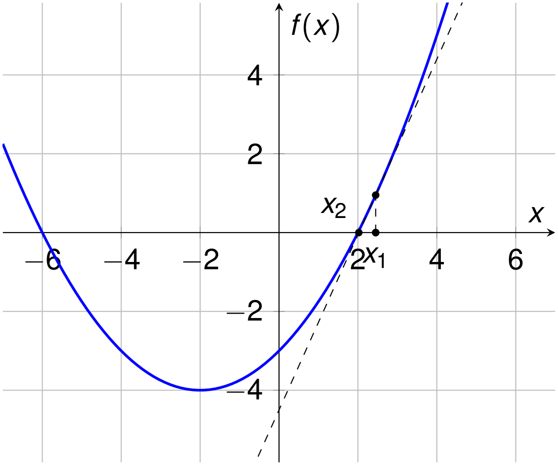
Beispiel



Beispiel



Beispiel



Tangentengleichung

$$y = f'(x_0)x + f(x_0) - f'(x_0)x_0$$

Tangentengleichung

$$y = f'(x_0)x + f(x_0) - f'(x_0)x_0$$

Nullstelle der Tangente

$$0 = f'(x_0)(x_0 - x) + f(x_0)$$

Nullstelle der Tangente

$$0 = f'(x_0)(x_0 - x) + f(x_0)$$

Umstellen nach x

$$x = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Umstellen nach x

$$x = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Iterationsvorschrift

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Konvergenz

- nur noch garantierte Konvergenz unter Einhaltung der Vorbedingungen
- quadratische Konvergenzordnung
- falls Vorbedingungen nicht erfüllt sind, kann man keine sichere Aussage über Konvergenz machen

Java-Implementierung

```
public static double Newton(double eps, double x) {
        int n = 0;
        while (Math.abs(f1(x)) > eps && n <= 1000) {
                x = x-f1(x)/df1(x);
                n++;
        }
        return x;
}
```

Quellenverzeichnis I

BOGOMOLNY, Alexander:
Intermediate Value Theorem - Bolzano Theorem from Interactive Mathematics Miscellany and Puzzles.
http://www.cut-the-knot.org/Generalization/ivt.shtml.
Version: 2017

PAPULA, Lothar:
Mathematik für Ingenieure und Naturwissenschaftler.
11., überarb. Aufl.
Wiesbaden : Springer Vieweg, 2007. –
ISBN 3834803049

Quellenverzeichnis II

TECHNISCHE UNIVERSITÄT DARMSTADT

SONAR, Thomas:
Angewandte Mathematik, Modellbildung und Informatik: Eine Einführung für Lehramtsstudenten, Lehrer und Schüler.
1. Aufl.
Braunschweig u.a., 2001. –
ISBN 3528031794

Abschluss

Vielen Dank für Ihre Aufmerksamkeit!
Haben Sie noch Fragen?