LAK-ComputerMathematik

Projekt **SR-3**

Übungsprogramm Newton-Algorithmus

Stand: 19. Oktober 2026, 13:36

Konsultationen zum Projekt: nach Terminvereinbarung (E-mail)

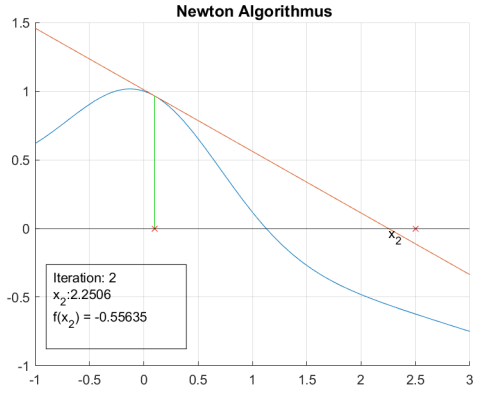
Übungsprogramm um den Newton-Algorithmus darzustellen:

Der Newton-Algorithmus war vor der Zentralmatura[1] ein Musterbeispiel bei der Matura. Der Algorithmus ist eine effiziente Methode um die Nullstellen von Funktionen zu finden.
Sei $f \in C^1(\mathbb{R})$ eine reelle differenzierbare Funktion. Die Folge

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

definiert den Newton-Algorithmus.
Die Idee des Newton-Algorithmus ist es, dass an jeder Stelle x_n die Tangente an $f(x_n)$ gelegt wird, und die Nullstelle der Tangente der nächste Iterations-Punkt ist.
Vorgehen:

- Initialisieren Sie eine symbolische Variable x und eine Funktion (z.B.: $f(x) = e^{-x^2} - \frac{x}{4}$).
- Bestimmen Sie die Ableitung von f und definieren Sie einen Startwert x_0 (z.B.: $x_0 = 2,5$).
- Berechnen Sie nun die Folgenglieder iterativ. Wählen Sie hierfür ein passendes Abbruchkriterium (z.B.: $|x_{n+1} - x_n| < \varepsilon$ mit geeignetem ε).[2]
- Stellen Sie nun die Folge iterativ dar. Zeichnen Sie dazu die Tangente an $f(x_n)$.
- Markieren Sie nun die Nullstelle der Tangente als nächsten Iterationspunkt.
- Eine mögliche Darstellung kann folgende Form haben.



- Wählen Sie einen Startpunkt, mit dem der Newton-Algorithmus ein Problem hat! Erklären Sie das Problem!

---

[1] Er wird teilweise in HTL's noch gelehrt, und inbesondere ist dieser ein Kandidat, welcher wieder in das *Repertoire* der Matura mitaufgenommen werden kann.

[2] Hinweis: Wenn Sie in Ihrem Algorithmus die neu berechneten Werte x_{n+1} mit `eval` approximieren lassen wird Matlab **markant** schneller! Warum ist das so? Versuchen Sie eine Erklärung zu formulieren, welche auch für einen Schüler nachvollziehbar ist!

Bonuspunkte:

- Verallgemeinerns Sie Ihr Programm so, dass der Algorithmus in eine Matlab-Funktion ausgelagert ist, und Sie das Programm wiederholt für verschiedene Funktionen f ausführen können.

Hinweise: `syms`, `diff`, `pause`, `delete`