

2. Praktikum zur Höhere Mathematik 2 für (Wirtschafts-)Informatik

Ziel dieses Praktikums ist eine Implementierung des Newtonverfahrens.

Dabei wird die Klasse `CMyVektor` des ersten Praktikums weiter verwendet.

1. Aufgabe

Um bequem mit Matrizen $A \in \mathbb{R}^{m \times n}$ arbeiten zu können, soll eine Klasse `CMyMatrix` implementiert werden:

- Implementieren Sie die Informationen über die Dimensionen und die Einträge als private Attribute.
- Implementieren Sie (public-)Methoden, um
 - eine Matrix einer bestimmten Dimension anzulegen,
 - eine bestimmte Komponente der Matrix zu setzen,
 - eine bestimmte Komponente der Matrix auszugeben.
- Implementieren Sie eine (public-)Methode `CMyMatrix invers()`, die
 - bei einer 2×2 -Matrix A mit $\det A \neq 0$ die Inverse A^{-1} mittels der Formel

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

liefert,

- ansonsten eine Fehlermeldung liefert und zum Programmabbruch führt.

Implementieren Sie ferner eine überladene Operator-Funktion

```
CMyVektor operator*(CMyMatrix A, CMyVektor x)
```

die eine Matrix-Vektor-Multiplikation realisiert.

2. Aufgabe

Zu einer Funktion $f : \mathbb{R}^m \rightarrow \mathbb{R}^n$ soll die Jacobi-Matrix an einer Stelle $\vec{x} \in \mathbb{R}^m$ berechnet werden:

- Implementieren Sie eine Funktionen

`CMyMatrix jacobi(CMyVektor x, CMyVektor (*funktion)(CMyVektor x)),`

der man im ersten Parameter die Stelle \vec{x} und im zweiten Parameter die Funktion f als Funktionspointer übergibt, und die die Jacobi-Matrix $J_f = f'(\vec{x})$ numerisch durch entsprechende Differenzenquotienten zu festem $h = 10^{-4}$ berechnet.

- Testen Sie die Berechnung an

$$f : \mathbb{R}^4 \rightarrow \mathbb{R}^3, \quad f(x_1, x_2, x_3, x_4) = \begin{pmatrix} x_1 x_2 e^{x_3} \\ x_2 x_3 x_4 \\ x_4 \end{pmatrix}, \quad \vec{x} = \begin{pmatrix} \frac{1}{2} \\ \frac{2}{0} \\ 3 \end{pmatrix}.$$

3. Aufgabe

Zu einer Funktion $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ soll ausgehend von einer Stelle $\vec{x} \in \mathbb{R}^n$ das Newtonverfahren zur Bestimmung einer Nullstelle von f durchgeführt werden.

- Implementieren Sie das entsprechende Verfahren unter Benutzung der Klassen `CMyVektor` und `CMyMatrix`.
- Entsprechend der Implementierung der `invers`-Methode braucht das Verfahren nur für den Fall $n = 2$ zu funktionieren.
- Nutzen Sie wieder einen Funktions-Pointer zur Angabe der Funktion f .
- Führen Sie die Newton-Iteration durch, bis $\|f(\vec{x})\| < 10^{-5}$ ist, oder bis 50 Schritte gemacht wurden.
- Testen Sie das Verfahren an

$$f : \mathbb{R}^2 \rightarrow \mathbb{R}^2, \quad f(x, y) = \begin{pmatrix} x^3 y^3 - 2y \\ x - 2 \end{pmatrix}$$

mit Startwert $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$.

Für Interessierte:

Die Suche nach einer lokalen Extremstelle einer Funktion $f : \mathbb{R}^n \rightarrow \mathbb{R}$ kann man auffassen als Suche nach einer Nullstelle von $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $g(\vec{x}) = \text{grad } f(\vec{x})$.

Schreiben Sie eine Funktion, die den Gradienten zu der Testfunktion $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ von Praktikum 1 zurückgibt, und wenden Sie darauf das Newtonverfahren an.

Testen Sie verschiedene Startwerte und vergleichen Sie die Ergebnisse (Konvergenzpunkt und dazu nötige Schrittzahl) des Gradientenverfahrens und des Newton-Verfahrens.