Übungsserie 11

Abgabe KW 21

Scannen Sie ihre manuellen Lösungen für die Aufgaben 2 in die Datei *Name_ Vorname_ Klasse_ S11_ Aufg2.pdf* und fassen Sie diese mit Ihren MATLAB-Skripts *Name_ Vorname_ Klasse_ S11_ Aufg1.m* und *Name_ Vorname_ Klasse_ S11_ Aufg3*.m für die Aufgaben 1 und 3 in eine ZIP-Datei *Name_ Vorname_ Klasse_ S11.zip* zusammen. Laden Sie dieses File vor der Übungsstunde nächste Woche auf OLAT hoch.

Aufgabe 1 (ca. 20 Minuten):

In Kapitel 1.5.3 (p. 34) im Anhang 'Eine kurze Einführung in MATLAB' ist ein Beispiel gegeben, wie eine Funktion mit zwei unabhängigen Variablen und deren Höhenlinien grafisch mit dem Befehl `mesh()` bzw. `meshc()`in einem dreidimensionalen Koordinatensystem dargestellt werden kann. Höhenlinien selbst lassen sich mit dem Befehl `contour()` in zwei Dimensionen abbilden.

Schreiben Sie ein Skript *Name_ Vorname_ Klasse_ S11_ Aufg1*.m, welches Ihnen jede der folgenden Funktionen in a) und b)

- einmal dreidimensional inkl. Höhenlinien darstellt mit `meshc()`
- einmal in zwei Dimensionen mit nur den Höhenlinien darstellt mit `contour()`
- zweidimensional mit fliessenden farblichen Übergängen darstellt mit `surface()` und dem zusätzlichen Befehl `colorbar`

Versehen Sie jede Abbildung mit passenden Achsenbeschriftungen und einem Titel.

a) Die Funktion $W = W(v_0, \alpha) = \frac{v_0^2 \sin(2\alpha)}{g}$ beschreibt die Wurfweite W eines Körpers, der mit der Anfangsgeschwindigkeit $v_0\,[\frac{\mathrm{m}}{\mathrm{s}}]$ unter einem Winkel α gegen die Horziontal abgeworfen wird. Nehme Sie für die Erdbeschleunigung $g = 9.81\,[\frac{\mathrm{m}}{\mathrm{s}^2}]$ an. Für die Anfangsgeschwindigkeit soll gelten $v_0 \in [0, 100]$. Wählen Sie selbst einen vernünftigen Definitionsbereich für α. Bei welchem Winkel α erreicht W für gegebens v_0 sein Maximum? Schreiben Sie dies als Kommentar in Ihr Skript.

Aufgabe 2 (ca. 20 Minuten):

Die Jacobi-Matrix $Df(x_1, ..., x_n)$ einer Funktion $f: \mathbb{R}^n \longrightarrow \mathbb{R}^n$ für einen gegebenen Vektor ist gemäss Skript (Def. 5.3)

$$Df(x_1,...,x_n) := \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(x_1,...,x_n) & \frac{\partial f_1}{\partial x_2}(x_1,...,x_n) & \cdots & \frac{\partial f_1}{\partial x_n}(x_1,...,x_n) \\ \frac{\partial f_2}{\partial x_1}(x_1,...,x_n) & \frac{\partial f_2}{\partial x_2}(x_1,...,x_n) & \cdots & \frac{\partial f_2}{\partial x_n}(x_1,...,x_n) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1}(x_1,...,x_n) & \frac{\partial f_n}{\partial x_2}(x_1,...,x_n) & \cdots & \frac{\partial f_n}{\partial x_n}(x_1,...,x_n) \end{pmatrix}$$

Berechnen Sie manuell die Jacobi-Matrizen für die folgenden Funktionen und Vektoren:

a) $f(x_1, x_2) = \begin{pmatrix} 5x_1x_2 \\ x_1^2x_2^2 + x_1 + 2x_2 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$

b) $f(x_1, x_2, x_3) = \begin{pmatrix} \ln(x_1^2 + x_2^2) + x_3^2 \\ \exp(x_2^2 + x_3^2) + x_1^2 \\ \frac{1}{(x_3^2 + x_1^2)} + x_2^2 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$

Aufgabe 3 (ca. 80 Minuten):

LORAN steht für LOng RAnge Navigation und ist ein erdgebundenes Funknavigationssystem, welches in der Luft- und Seefahrt zur Positionsbestimmung zum Einsatz kommt (siehe z.B. http://de.wikipedia.org/wiki/LORAN), als unterdessen etwas überholte Alternative zum satellitengestützten GPS. Die Sender emittieren nach einem festen Schema Impulsgruppen. Aus den Laufzeitdifferenzen der bei einem Empfänger eintreffenden Signale ist es möglich, hyperbelförmige Ortskurven und aus deren Schnittpunkten die Position des Empfängers zu berechnen. Die folgenden zwei impliziten Funktionsgleichungen sind ein solches Beispiel zur Bestimmung der $(x, y)-$Koordinaten eines Empfängers:

$$\begin{aligned} f_1(x,y) = \frac{x^2}{186^2} - \frac{y^2}{300^2 - 186^2} - 1 &= 0 \\ f_2(x,y) = \frac{(y-500)^2}{279^2} - \frac{(x-300)^2}{500^2 - 279^2} - 1 &= 0 \end{aligned}$$

Schreiben Sie ein Skript *Name_ Vorname_ Klasse_ S12_ Aufg2.m* welches Ihnen die beiden folgenden Aufgaben löst:

a) Lösen Sie das Gleichungssystem zuerst grafisch, indem Sie mit MATLAB und dem Befehl `ezplot()` (siehe Online-Dokumentation) die beiden durch f_1 und f_2 definierten Hyperbeln in ein Grafikfenster plotten und anschliessend von Auge Näherungswerte für die Koordinaten der vier Schnittpunkte zwischen den beiden Hyperbeln bestimmen.

b) Benutzen Sie Ihre unter a) bestimmten Näherungsvektoren und bestimmen Sie mit dem Newton-Verfahren die vier Lösungen mit einer Genauigkeit $\| f(x^{(k)} \|_2 < 10^{-5}$.

Bemerkung: falls Sie bei b) symbolisch rechnen, helfen Ihnen die Befehle `syms`, `jacobian()`, `matlabFunction()`.