

Name: _____

Aufwand (h): _____

Punkte: _____

Aufgabe 1 (8 Pkt): Polynome

Polynome, eine wesentliche Grundlage bei der Beschreibung des dynamischen Verhaltens von beliebigen Systemen, können als Zeilenvektoren angegeben werden; Polynome von Grad n können allgemein durch Vektoren der Länge $n+1$ dargestellt werden.

Beispiel: Das Polynom

$$p(x) = 5x^5 - 3x^4 + x^2 + 2x$$

lautet in seiner vollständigen Darstellung

$$p(x) = 5x^5 - 3x^4 + 0x^3 + x^2 + 2x + 0x^0$$

und kann durch den Vektor

$$p = [5 \ -3 \ 0 \ 1 \ 2 \ 0]$$

definiert werden.

Schreiben Sie eine MATLAB-Funktion, welcher ein Polynom beliebigen Grades als Zeilenvektor übergeben wird plus Angabe des Bereiches, für den dieses Polynom ausgewertet und gezeichnet werden soll; die Schrittweite der Auswertung bzw. graphischen Darstellung soll optional übergeben werden können, wenn nicht definiert soll sie 0.01 betragen.

Achten Sie bei der graphischen Ausgabe auf die Generierung von sinnvollen Diagramm-Überschriften. Und rufen Sie für diese Übung nicht einfach eine Polynom-Funktion von MATLAB auf!

Implementieren Sie diese Aufgabe auf 2 verschiedene Arten:

- „herkömmlich“, also wie aus C++ / Java / C# gewohnt mit geschachtelten for-Schleifen
- „schlau“, also unter Ausnutzung der Vorteile von MATLAB (Vektor-Initialisierungen und Multiplikationen etc.)

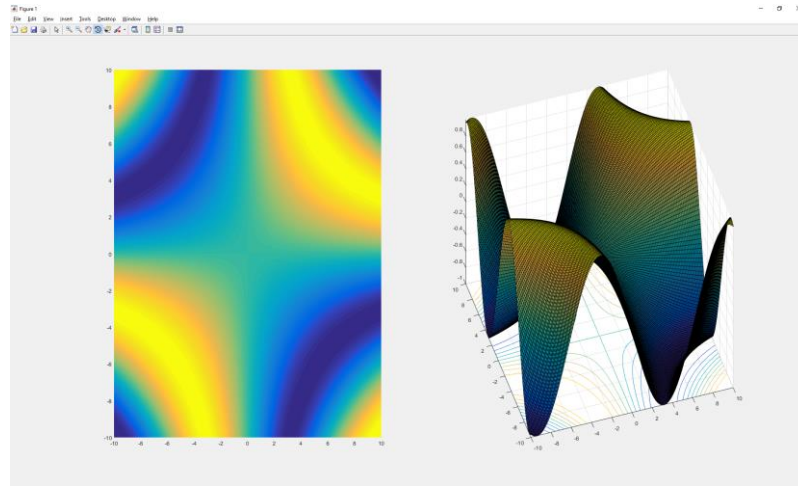
Messen Sie den Unterschied im Laufzeitverbrauch und zeigen Sie, um wie viel sich (bei unterschiedlichen Auswertebereichen) die Laufzeiten der beiden Varianten unterscheiden.

Aufgabe 2 (8 Pkt): Malen nach Zahlen (Pt 1)

Implementieren Sie die graphische Darstellung von zweidimensionalen Funktionen. Die Visualisierung der Funktion $f(x, y) = \sin(x * y / 10)$ könnte aussehen wie in Abbildung 1 gezeigt.

Implementieren Sie nun in MATLAB eine Funktion, die eine gegebene Funktion $f(x, y)$ in einem gegebenen Bereich zeichnet, und zwar sowohl 2- als auch 3-dimensional.

Die Funktion, die zu visualisieren ist, wird als String übergeben, zB `'sin(x_*y_ / 20)'`. Verwenden Sie dafür die MATLAB-Funktion `eval(s)`, die einen gegebenen String auswertet; der eigentlich auszuwertende Befehl sollte die übergebene Funktion beinhalten.



Aufgabe 3 (8 Pkt): Malen nach Zahlen (Pt 2)

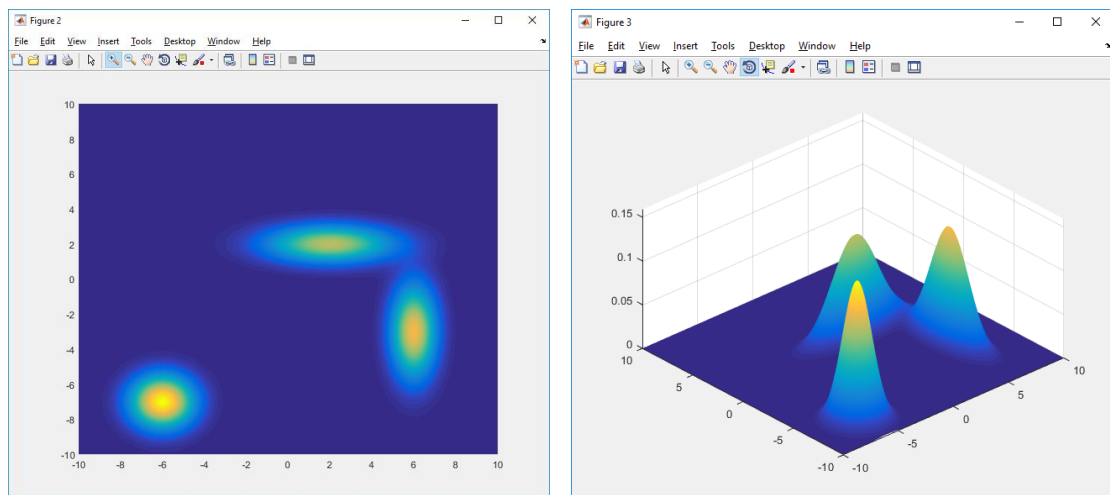
Nun geht es um das Zeichnen von 2-dimensionalen Normalverteilungen. Eine Normalverteilung wird ja definiert durch ihren Mittelpunkt und ihre Kovarianzmatrix; die Wahrscheinlichkeitsdichte einer Normalverteilung mit Mittelwert μ und Varianz σ^2 ist ja bekanntlich gegeben als

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

bzw. im mehrdimensionalen (d = Anzahl der Dimensionen) als

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu})\right]$$

Abbildung 2 zeigt drei solche Normalverteilungen zweidimensional gezeichnet, Abbildung 3 zeigt diese Überlagerung mehrerer Verteilungen dreidimensional.



Implementieren Sie nun eine Funktion, die alle Informationen über die (beliebig vielen) zu zeichnenden Normalverteilungen und auch Information über den zu zeichnenden Bereich aus einer Text-Datei einliest und eine Datenstruktur zurückgibt, die alle bereits berechneten und zu zeichnenden Werte beinhaltet; definieren Sie selber die Klasse, die diese Datenstruktur definiert sowie die Struktur der zu lesenden Datei. Diese Datenstruktur soll dann einer weiteren Funktion übergeben werden können, die das Zeichnen übernimmt.

Hinweise: Geben Sie elektronisch ab und testen Sie ausführlich!