

Übungsblatt 1: Matlab

Abgabeschluss: Montag, der 28.04.2008 um 10:00 Uhr.

Die genauen Abgabemodalitäten werden noch bekannt gegeben.

Aufgaben

Dieses erste Übungsblatt dient dazu, Euch mit Matlab vertraut zu machen. Für die Bearbeitung der folgenden Übungsblätter sind gute Matlab-Kenntnisse sehr hilfreich. In der ersten Woche findet am Mittwoch ein Matlab Crash-Kurs statt.

Aufgabe 1 [10 Punkte]

Schreibe eine Funktion `Aufgabe1` in einer Datei namens `Aufgabe1.m` mit der Signatur

$$[D1, D2, td] = \text{Aufgabe1}(X)$$

die für die Spaltenvektoren in X die Distanzmatrizen (nach L_2 -Norm) $D1$ und $D2$ nach zwei unterschiedlichen Verfahren berechnet und deren Laufzeitdifferenz td ermittelt: Es sei

$$X = [x_1, x_2, \dots, x_n]$$

die $(d \times n)$ -Matrix von Spaltenvektoren. Dann ist

$$(D_1)_{ij} = (D_2)_{ij} = \|x_i - x_j\|$$

wobei D_1 mit Hilfe von `for`-Schleifen berechnet und für die Berechnung von D_2 die Gleichung

$$\|x_i - x_j\|^2 = (x_i - x_j)^\top (x_i - x_j) = x_i^\top x_i - 2x_i^\top x_j + x_j^\top x_j$$

benutzt wird, um `for`-Schleifen zu vermeiden. Berechne die Differenz der Laufzeiten td mittels `tic` und `toc` wobei td positiv ist, wenn die Methode D_2 schneller war.

Anmerkungen: Da `tic` und `toc` kein sehr hochauflösender Timer ist, sollte die Matrix X gross genug gewählt werden, damit die Differenz messbar wird. Desweiteren könnten die Befehle `repmat` und `sum` für die Lösung der Aufgabe hilfreich sein. Versuche am besten die drei Terme der letzten Gleichung erstmal als drei getrennte Matrizen zu berechnen um sie dann zu einer Matrix zu addieren. Merke auch, dass die Gleichung die quadrierte Norm berechnet.

Aufgabe 2 [10 Punkte]

Schreibe eine Funktion `Aufgabe2` mit der Signatur

$$d = \text{Aufgabe2}(A, k)$$

welche die Determinante von A durch Entwicklung nach der k -ten Zeile durch Rekursion berechnet.

Zur Erinnerung: Die Determinante einer $(n \times n)$ -Matrix A kann durch Entwicklung nach der k -ten Zeile berechnet werden als

$$\det(A) = \sum_{j=1}^n (-1)^{k+j} A_{kj} \det(\tilde{A}_{k,j})$$

wobei $\tilde{A}_{k,j}$ aus A durch Löschen der k -ten Zeile und j -ten Spalte entsteht. Für (1×1) -Matrizen gilt $\det(A) = A_{1,1}$. Teste die Funktion anhand einiger Matrizen und vergleiche das Ergebnis mit `det`.

Aufgabe 3 [10 Punkte]

Bei einer Gameshow hat der Kandidat die Auswahl zwischen drei Türen (links, mitte, rechts). Hinter zwei der Türen befinden sich Ziegen, hinter einer Tür winkt der Hauptgewinn. Das Spiel läuft so ab: Der Kandidat nennt eine der Türen, doch bevor sie geöffnet wird, erhöht der Spielleiter die Spannung, indem er eine andere Tür, hinter der sich eine Ziege befindet, öffnet. Der Showmaster bietet dem Kandidaten an, sich nochmals zwischen beiden noch verschlossenen Türen zu entscheiden. Die vom Kandidaten gewählte Tür wird dann im Showdown geöffnet und es zeigt sich, ob der Kandidat den Hauptgewinn oder eine Ziege mit nach Hause nehmen darf.

Schreibe eine Funktion `Aufgabe3` (ohne Parameter), in der Du durch wiederholte Simulation des Spiels ermittelst, ob es günstiger ist, die Tür zu wechseln oder nicht. Visualisiere das Ergebnis durch einen geeigneten Plot.

Aufgabe 4 [10 Punkte]

Ein Graph $G = (V, E)$ mit $|V| = n$ Knoten kann als Adjazenzmatrix A dargestellt werden. Für deren Elemente gilt

$$A_{ij} = \begin{cases} 1 & : \text{ Es existiert eine Kante zwischen den Knoten } i \text{ und } j \\ 0 & : \text{ sonst.} \end{cases}$$

Der Grad eines Knotens $v \in V$ ist die Anzahl der Knoten, mit denen er verbunden ist.

Für eine Menge von Vektoren $X = \{x_1, x_2, \dots, x_n\}$ definieren wir den k -nearest-neighbour als ungerichteten Graphen mit Knotenmenge $V = X$ wobei wir zwei Knoten $x_i, x_j \in V$ verbinden, wenn x_i einer der k nächsten Nachbarn (bezüglich der L_2 -Norm) von x_j ist oder umgekehrt. Die zugehörige Adjazenzmatrix ist also symmetrisch.

Schreibe eine Funktion

$$A = \text{Aufgabe4}(X, k)$$

welche die Adjazenzmatrix A für den k -nearest-neighbour Graphen auf den Spaltenvektoren in X berechnet. Dazu könnte die Funktion `sort` hilfreich sein. Die Matrix X hat das Format $(2 \times n)$. Ausserdem soll die Funktion die folgenden drei Plots erzeugen.

- Plot (als Punkte) der zweidimensionalen Spaltenvektoren in X . Ausserdem soll der k -nearest-neighbour Graph durch gestrichelte Linien zwischen den Datenpunkten dargestellt werden. Hinweis: Verwende `line`.
- Histogramm der Verteilung der Längen der Kanten des k -nearest-neighbour Graphen.
- Histogramm der Verteilung der Grade der Knoten des k -nearest-neighbour Graphen.

Aufgabe 5 [10 Punkte]

In dieser Aufgabe sollen Eigenschaften des berühmten USPS Datensatzes untersucht werden. Der USPS Datensatz enthält Bilder (16×16 Pixel) handgeschriebener Ziffern von 0–9 und die dazugehörigen Labels.

Schreibe eine Funktion `Aufgabe5` (ohne Parameter), welche die Daten aus der Datei `usps.mat` (siehe website) lädt und die folgenden Plots erstellt.

- Die Mittelwerte (dargestellt als 16×16 Pixel Bild in Graustufen) der zehn Klassen als Unterplots (subplots) in einer figure.
- Die Standardabweichung an jedem Pixel für die Klassen '1', '2' und '7' als Unterplots (subplots) in einer figure als 3D Balkendiagramm.
- Für die Klassen '1', '2' und '7', zeige die drei Datenpunkte (als 16×16 Pixel Bild in Graustufen), die am weitesten entfernt vom Mittelwert der Klasse liegen.

Die Funktionen `load`, `bar3`, `mean`, `std`, `reshape`, `imagesc` und `colormap` könnten hilfreich sein.