Ludwig-Maximilians-Universität München
Institut für Informatik
Dr. Matthias Schubert
Erich Schubert

Knowledge Discovery in Databases

WS 2009/10

Übungsblatt 9

Aufgabe 9-1 EM-Algorithmus

Übungsaufgabe

Gegeben sei eine Datenmenge D mit 100 Punkten, die drei Gausscluster A, B und C und den Punkt p enthält.

Der Cluster A ist repräsentiert durch den Mittelwert aller seiner Punkte (2,2) und die Kovarianzmatrix $\begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix}$.

30 Prozent aller Punkte gehören zu diesem Cluster.

Der Cluster B ist repräsentiert durch den Mittelwert aller seiner Punkte (5,3) und die Kovarianzmatrix $\begin{pmatrix} 2 & 1 \\ 1 & 4 \end{pmatrix}$.

20 Prozent aller Punkte gehören zu diesem Cluster.

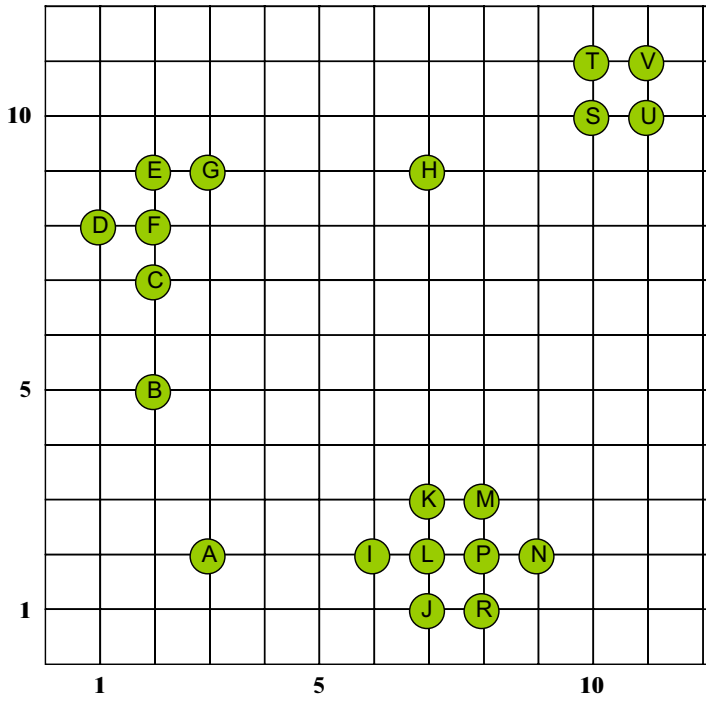
Der Cluster C ist repräsentiert durch den Mittelwert aller seiner Punkte (1,4) und die Kovarianzmatrix $\begin{pmatrix} 16 & 0 \\ 0 & 4 \end{pmatrix}$.

50 Prozent aller Punkte gehören zu diesem Cluster.

Der Punkt p ist durch die Koordinaten (2.5,3.0) gegeben. Geben Sie die drei Wahrscheinlichkeiten an, mit der p zum Cluster A, B bzw. C gehört.

Aufgabe 9-2 OPTICS

Gegeben sei der folgende 2-dimensionale Datensatz:



Verwenden Sie als Distanzfunktion zwischen den Punkten wieder die Manhattan-Distanz (L_1-Norm)

Erzeugen Sie mit OPTICS (Pseudocode am Ende des Übungsblattes) jeweils ein Erreichbarkeitsdiagramm für die folgenden Parameter:

(a) $\epsilon = 5$ und $MinPts = 2$

(b) $\epsilon = 5$ und $MinPts = 4$

(c) $\epsilon = 2$ und $MinPts = 4$

(d) $\epsilon = \infty$ und $MinPts = 4$

(e) Diskutieren Sie, welche Auswirkungen die Parameter $MinPts$ und ϵ haben.

Aufgabe 9-3 Zusammenhang Multivariate Dichte und Mahalanobis-Distanz

Die Dichte einer Multivariaten Normalverteilung (mit Σ, μ) berechnen wir mit der Formel

$$prob(p) := \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T \Sigma^{-1} (p-\mu)\right)}$$

Finden sie einen einfachen Zusammenhang dieser Formel zur Mahalanobis-Distanz (mit Σ) von p zu μ.

Aufgabe 9-4 SNN-Clustering als GDBSCAN

Wie in der Vorlesung angedeutet kann man SNN-Clustering als eine Spezialisierung des GDBSCAN Algorithmus (Generalized DBSCAN) auffassen.

(a) Formulieren sie dazu je ein geeignetes Nachbarschaftsprädikat $NPred$ und Dichteprädikat $MinWeight$.

(b) Naiv implementiert ist die Laufzeitkomplexität problematisch. Welche Maßnahme wurde in der Vorlesung (implizit) verwendet um hier Abhilfe zu schaffen, und zu welchen Kosten geschieht das?

Pseudocode OPTICS

```
seedlist = ∅   // implemented as a heap
for i = 0 to n-1 do
    if(seedlist = ∅ )  then seedlist = {(random_not_handled_point, ∞)}
    (x, x.reach) = get_and_remove_point_with_min_reach(seedlist)
    x.pos = i
    x.handled = TRUE
    neighbors = rangeQuery(x, ε)
    x.core = nnDist(x, neighbors, MinPts)
    if(x.core < ∞)
        for each y ∈ neighbors with not(y.handled)
            if( y ∉ seedlist)  seedlist = seedlist ∪ {(y, reach-dist(y,x))}
            else
                curr_reach = lookup(seedlist, y)
                update(y, min(curr_reach, reach-dist(y,x)))
        endfor
endfor
```