

VERİ MADENCİLİĞİ

Demetleme Yöntemleri

Prof. Dr. Şule Gündüz Öğüdücü

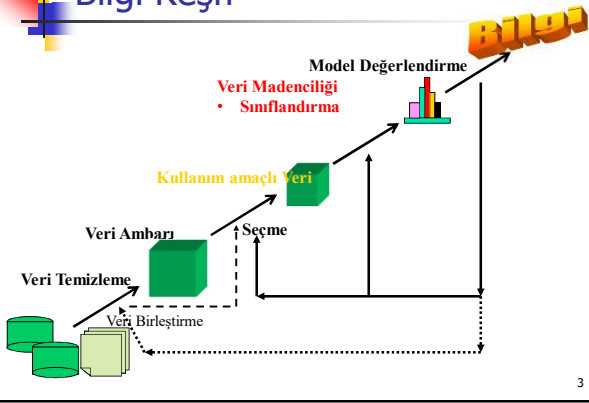
1

Konular

- Demetleme işlemleri
 - Demetleme tanımı
 - Demetleme uygulamaları
- Demetleme Yöntemleri
 - Bölünmeli Yöntemler
 - Hiyerarşik Yöntemler
 - Yoğunluk Tabanlı Yöntemler
 - Model Tabanlı Yöntemler

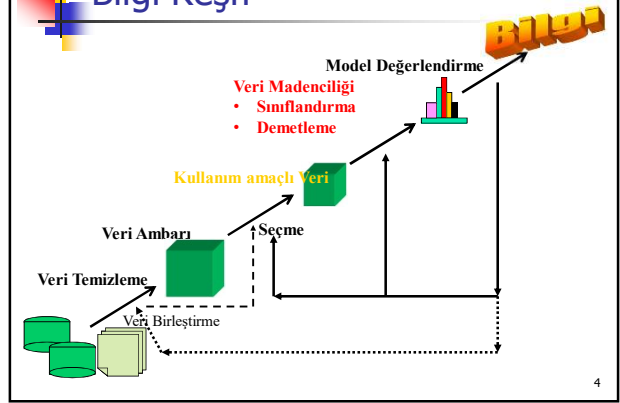
2

Bilgi Keşfi



3

Bilgi Keşfi



4

Demetleme

- Nesneleri demetlere (gruplara) ayırma
- Demet: birbirine benzeyen nesnelerden oluşan grup
 - Aynı demetdeki nesneler birbirine daha çok benzer
 - Farklı demetlerdeki nesneler birbirine daha az benzer

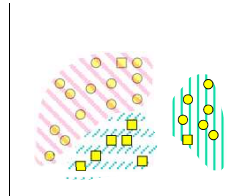
Aynı demet içindeki nesneler arasındaki uzaklığı en küçültme

Farklı demetlerdeki nesneler arasındaki uzaklığı en büyütme

5

Demetleme

- Gözetimsiz öğrenme: Hangi nesnenin hangi sınıfa ait olduğu ve sınıf sayısı belli değil
- Uygulamaları:
 - verinin dağılımını anlama
 - başka veri madenciliği uygulamaları için ön hazırlık



6

Demetleme Uygulamaları

- Örüntü tanıma
- Görüntü işleme
- Ekonomi
- Aykırılıkları belirleme
- WWW
 - Doküman demetleme
 - Kullanıcı davranışlarını demetleme
 - Kullanıcıları demetleme
- Diğer veri madenciliği algoritmaları için bir ön işleme adımı
 - Veri azaltma – demet içindeki nesnelerin temsil edilmesi için demet merkezlerinin kullanılması

7

Veri Madenciliğinde Demetleme

- Ölçeklenebilirlik
- Farklı tipteki niteliklerden oluşan nesneleri demetleme
- Farklı şekillerdeki demetleri oluşturabilme
- En az sayıda giriş parametresi gereksinimi
- Hatalı veriler ve aykırılıklardan en az etkilenme
- Model oluşturma sırasında örneklerin sırasından etkilenmeme
- Çok boyutlu veriler üzerinde çalışma
- Kullanıcıların kısıtlarını göz önünde bulundurma
- Sonucun yorumlanabilir ve anlaşılabilir olması

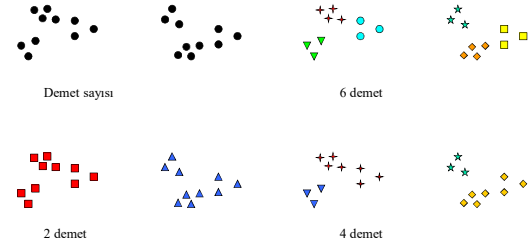
8

İyi Demetleme

- İyi demetleme yöntemiyle elde edilen demetlerin özellikleri
 - aynı demet içindeki nesneler arası benzerlik fazla
 - farklı demetlerde bulunan nesneler arası benzerlik az
- Oluşan demetlerin kalitesi seçilen benzerlik ölçütüne ve bu ölçütün gerçekleşmesine bağlı
 - Uzaklık / Benzerlik nesnelerin nitelik tipine göre değişir
 - Nesneler arası benzerlik: $s(i,j)$
 - Nesneler arası uzaklık: $d(i,j) = 1 - s(i,j)$
- İyi bir demetleme yöntemi veri içinde gizlenmiş örüntüleri bulabilmeli
- Veriyi gruplama için uygun demetleme kriteri bulunmalı
 - demetleme = aynı demetteki nesneler arası benzerliği en büyüyen, farklı demetlerdeki nesneler arası benzerliği en küçülten fonksiyon
- Demetleme sonucunun kalitesi seçilen demetlerin şekline ve temsil edilme yöntemine bağlı

9

Farklı Demetler



10

Temel Demetleme Yaklaşımları

- Bölünmeli yöntemler: Veriyi bölerek, her grubu belirlenmiş bir kritere göre değerlendirir
- Hiyerarşik yöntemler: Veri kümelerini (ya da nesneleri) önceden belirlenmiş bir kritere göre hiyerarşik olarak ayırır
- Yoğunluk tabanlı yöntemler: Nesnelerin yoğunluğuna göre demetleri oluşturur
- Model tabanlı yöntemler: Her demetin bir modele uyduğu varsayılır. Amaç bu modellere uyan verileri gruplamak

11

Konular

- Demetleme işlemleri
 - Demetleme tanımı
 - Demetleme uygulamaları
- Demetleme Yöntemleri
 - Bölünmeli Yöntemler
 - K-means demetleme yöntemi
 - K-medoids demetleme yöntemi
 - Hiyerarşik Yöntemler
 - Yoğunluk Tabanlı Yöntemler
 - Model Tabanlı Yöntemler

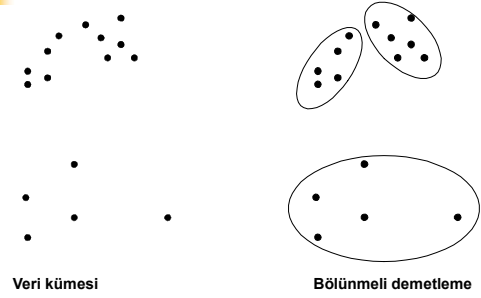
12

Bölünmeli Yöntemler

- Amaç: n nesneden oluşan bir veri kümesini (D) k ($k \leq n$) demete ayırmak
 - her demette en az bir nesne bulunmalı
 - her nesne sadece bir demette bulunmalı
- Yöntem: Demetleme kriterini en büyütmek şeklinde D veri kümesi k gruba ayırma
 - Global çözüm: Mümkün olan tüm gruplamaları yaparak en iyisini seçme (NP karmaşık)
 - Sezgisel çözüm: k-means ve k-medoids
 - k-means (MacQueen'67): Her demet kendi merkezi ile temsil edilir
 - k-medoids veya PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Her demet, demette bulunan bir nesne ile temsil edilir

13

Bölünmeli Demetleme



14

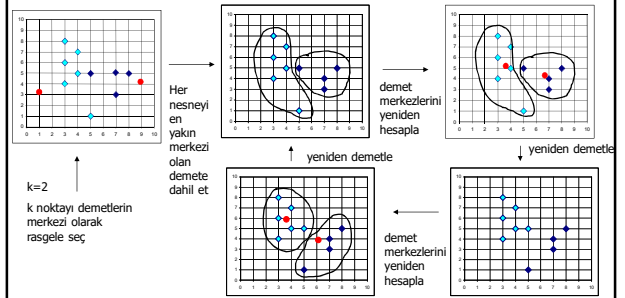
K-means Demetleme

- Bilinen bir k değeri için k-means demetleme algoritmasının 4 aşaması vardır:
 1. Veri kümesi k altkümeye ayrılır (her demet bir altküme)
 2. Her demetin ortalaması hesaplanır: merkez nokta (demetteki nesnelerin niteliklerinin ortalaması)
 3. Her nesne en yakın merkez noktanın olduğu demete dahil edilir
 4. Nesnelerin demetlenmesinde değişiklik olmayana kadar adım 2'ye geri dönlür.

15

K-means Demetleme Yöntemi

Örnek



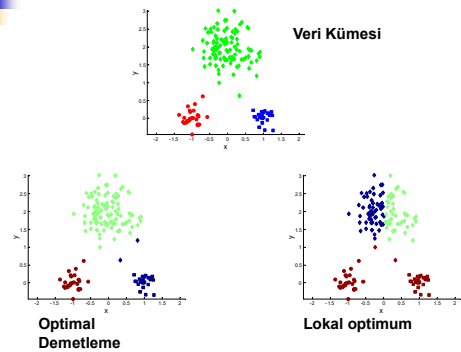
16

k-means Demetleme Yöntemi

- Demet sayısının belirlenmesi gerekir
 - Başlangıçta demet merkezleri rasgele belirlenir
 - Her uygulamada farklı demetler oluşabilir
 - Uzaklık ve benzerlik Öklid uzaklığı, kosinüs benzerliği gibi yöntemlerle ölçülebilir
 - Az sayıda tekrarda demetler oluşur
 - Yakınsama koşulu çoğunlukla az sayıda nesnenin demet değişmesi şekline dönüştürülür
 - Karmaşıklık:
 - Yer karmaşıklığı - $O((n+k) d)$
 - Zaman karmaşıklığı - $O(ktn d)$
- k : demet sayısı, t : tekrar sayısı, n : nesne sayısı, d : nitelik sayısı

17

K-Means: İki Farklı Demetleme



18

K-Means Demetleme Yöntemini Değerlendirme

- Yaygın olarak kullanılan yöntem hataların karelerinin toplamı (Sum of Squared Error SSE)

- Nesnelerin bulundukları demetin merkez noktalarına olan uzaklıklarının karelerinin toplamı

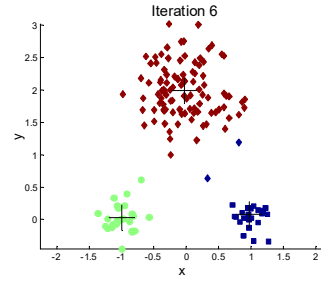
$$SSE = \sum_{i=1}^k \sum_{x \in C_i} dist^2(m_i, x)$$

x : C_i demetinde bulunan bir nesne, m_i : C_i demetinin merkez noktası

- Hataların karelerinin toplamını azaltmak için k demet sayısı artırılabilir
 - Çoğu durumda belirlenen bir k demet sayısı için SSE değeri azaltılmak istenir.
- Başlangıç için farklı merkez noktaları seçerek farklı demetlemeler oluşturulur
- En az SSE değerini sahip olan demetleme seçilir

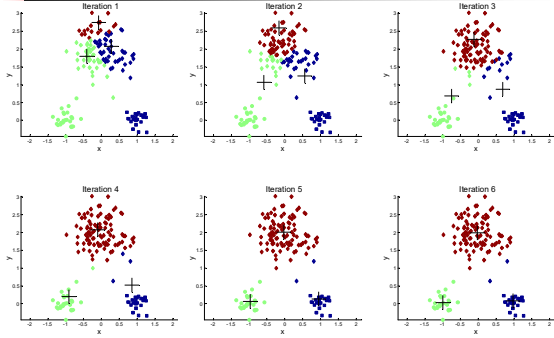
19

Merkez Noktaların Seçimi



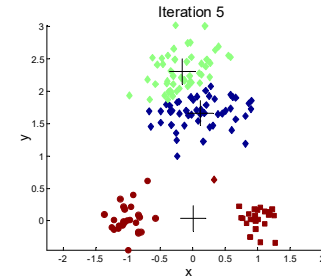
20

Merkez Noktaların Seçimi



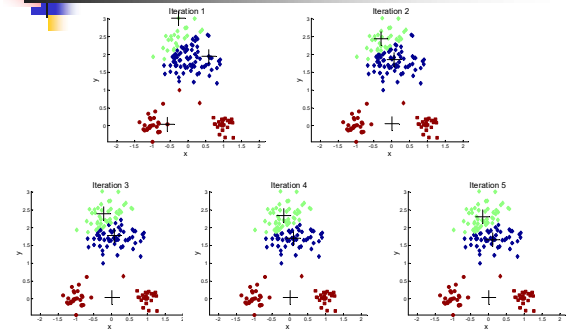
21

Merkez Noktaların Seçimi



22

Merkez Noktaların Seçimi



23

K-Means Demetleme Çeşitleri

- K-Means demetlemeye başlamadan önce yapılanlar
 - Veri kümesini örnekleyerek hiyerarşik demetleme yap. Oluşan k demetin ortalamasını başlangıç için merkez nokta seç
 - Başlangıçta k 'dan fazla merkez nokta seç. Daha sonra bunlar arasından k tane seç.
- K-Means demetleme işlemi sonrasında yapılanlar
 - Küçük demetleri en yakın başka demetlere dahil et
 - En büyük toplam karesel hataya sahip olan demeti böl
 - Merkez noktaları birbirine en yakın demetleri birleştir
 - Toplam karesel hatada en az artışa neden olacak iki demeti birleştir

24

K-Means Demetleme Algoritmasının Özellikleri

- Gerçeklemesi kolay
- Karmaşıklığı diğer demetleme yöntemlerine göre az
- K-Means algoritması bazı durumlarda iyi sonuç vermeyebilir
 - Veri grupları farklı büyüklüklerde ise
 - Veri gruplarının yoğunlukları farklı ise
 - Veri gruplarının şekli küresel değilse
 - Veri içinde aykırılıklar varsa

25

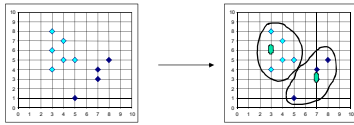
Konular

- Demetleme işlemleri
 - Demetleme tanımı
 - Demetleme uygulamaları
- Demetleme Yöntemleri
 - Bölünmeli Yöntemler
 - K-means demetleme yöntemi
 - K-medoids demetleme yöntemi
 - Hiyerarşik Yöntemler
 - Yoğunluk Tabanlı Yöntemler
 - Model Tabanlı Yöntemler

26

K-Medoids Demetleme Yöntemi

- Her demeti temsil etmek için demet içinde orta nokta olan nesne seçilir.
 - 1, 3, 5, 7, 9 ortalama: **5**
 - 1, 3, 5, 7, 1009 ortalama: **205**
 - 1, 3, 5, 7, 1009 orta nokta: **5**



27

K-Medoids Demetleme Yöntemi

- PAM (Partitioning Around Medoids 1987)
 1. Başlangıçta k adet nesne demetleri temsil etmek üzere rasgele seçilir x_{jk}
 2. Kalan nesneler en yakın merkez nesnenin bulunduğu demete dahil edilir
 3. Merkez nesne olmayan rasgele bir nesne seçilir x_{rk}
 4. x_{rk} merkez nesne olursa toplam karesel hatanın ne kadar değiştiği bulunur
$$TC_{ik} = \sum_{j=1}^{n_k} (x_{ik} - x_{jk})^2 - \sum_{j=1}^{n_k} (x_{rk} - x_{jk})^2$$

n_k : k demeti içindeki nesne sayısı
 x_{jk} : k demeti içindeki j . nesne

 5. $TC_{ik} > 0$ ise x_{rk} merkez nesne olarak atanır.
 6. Demetlerde değişiklik oluşmaya kadar 3. adıma geri gidilir.
- Küçük veri kümeleri için iyi sonuç verebilir, ancak büyük veri kümeleri için uygun değil
- CLARA (Kaufmann & Rousseeuw, 1990)
- CLARANS (Ng & Han, 1994)

28

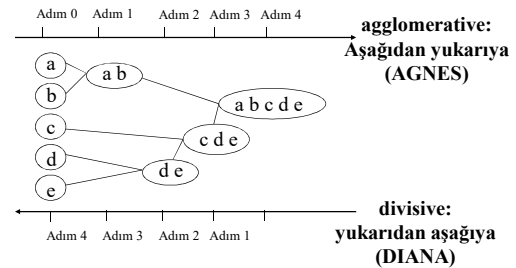
Konular

- Demetleme işlemleri
 - Demetleme tanımı
 - Demetleme uygulamaları
- Demetleme Yöntemleri
 - Bölünmeli Yöntemler
 - Hiyerarşik Yöntemler
 - Yoğunluk Tabanlı Yöntemler
 - Model Tabanlı Yöntemler

29

Hiyerarşik Demetleme

- Demet sayısının belirlenmesine gerek yok
 - Sonlanma kriteri belirlenmesi gerekiyor

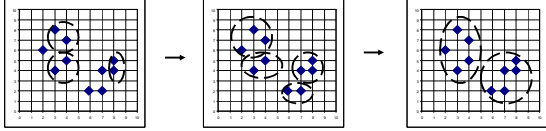


30

Hiyerarşik Yöntemler

■ AGNES (AGglomerative NESTing):

- Kaufmann ve Rousseeuw tarafından 1990 yılında önerilmiştir.
- Birinci adımda her nesne bir demet oluşturur.
- Aralarında en az uzaklık bulunan demetler her adımda birleştirilir.
- Bütün nesneler tek bir demet içinde kalana kadar ya da istenen sayıda demet elde edene kadar birleştirme işlemi devam eder.

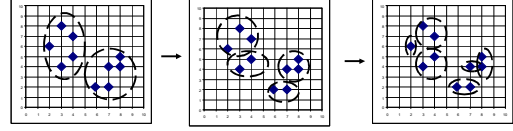


31

Hiyerarşik Yöntemler

■ DIANA (DIvisive ANALysis):

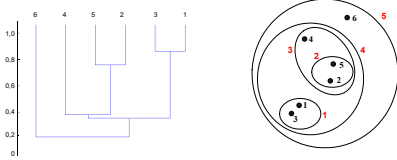
- Kaufmann ve Rousseeuw tarafından 1990 yılında önerilmiştir.
- AGNES'in yaptığı işlemlerin tersini yapar.
- Birinci adımda bütün nesneler tek bir demette yer alır.
- Aralarında en fazla uzaklık bulunan demetler her adımda ayrılır.
- Her nesne ayrı bir demet oluşturma ya da istenilen demet sayısı elde edene kadar ayrılma işlemi devam eder.



32

Hiyerarşik Demetleme

- Dendrogram: Demetler hiyerarşik olarak ağaç yapısı şeklinde görüntülenebilir
- Ara düğümler çocuk düğümlerdeki demetlerin birleşmesiyle elde edilir
 - Kök: bütün nesnelerden oluşan tek demet
 - Yapraklar: bir nesneden oluşan demetler
- Dendrogram istenen seviyede kesilerek demetler elde edilir



33

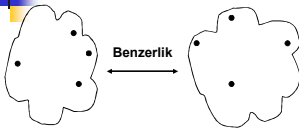
Aşağıdan Yukarıya Demetleme

■ Algoritma

1. Uzaklık matrisini hesapla
 2. Her nesne bir demet
 3. Tekrarla
 4. En yakın iki demeti birleştir
 5. Uzaklık matrisini yeniden hesapla
 6. Sonlanma: Tek bir demet kalana kadar
- Uzaklık matrisini hesaplarlarken farklı yöntemler farklı demetleme sonuçlarına neden olurlar

34

Demetler Arası Uzaklık



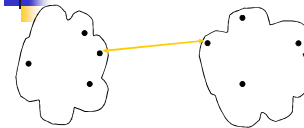
- MIN (Tek Bağ)
- MAX (Tam Bağ)
- Ortalama
- Merkezler arası uzaklık

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
...						

Uzaklık Matrisi

35

Demetler Arası Uzaklık



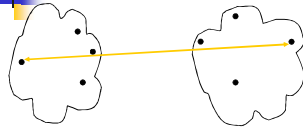
- MIN (Tek Bağ)
- MAX (Tam Bağ)
- Ortalama
- Merkezler arası uzaklık

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
...						

Uzaklık Matrisi

36

Demetler Arası Uzaklık



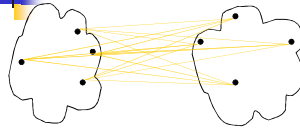
- MIN (Tek Bağ)
- MAX (Tam Bağ)
- Ortalama
- Merkezler arası uzaklık

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						

Uzaklık Matrisi

37

Demetler Arası Uzaklık



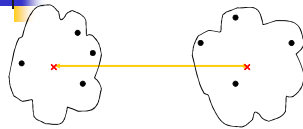
- MIN (Tek Bağ)
- MAX (Tam Bağ)
- Ortalama
- Merkezler arası uzaklık

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						

Uzaklık Matrisi

38

Demetler Arası Uzaklık



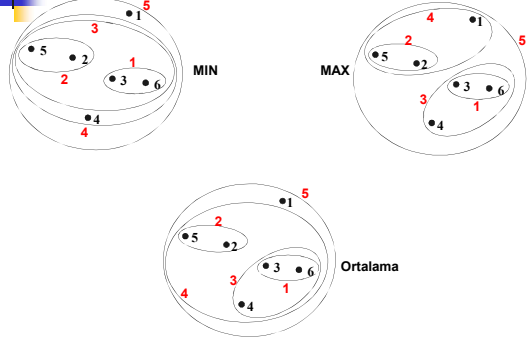
- MIN (Tek Bağ)
- MAX (Tam Bağ)
- Ortalama
- Merkezler arası uzaklık

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						

Uzaklık Matrisi

39

Farklı Uzaklık Yöntemlerinin Etkisi



40

Hiyerarşik Demetleme Yöntemlerinin Özellikleri

- Demetleme kriteri yok
- Demet sayılarının belirlenmesine gerek yok
- Aykırılıklardan ve hatalı verilerden etkilenir
- Farklı boyuttaki demetleri oluşturmak problemli olabilir
- Yer karmaşıklığı – $O(n^2)$
- Zaman karmaşıklığı – $O(n^2 \log n)$
n : nesne sayısı

41

Konular

- Demetleme işlemleri
 - Demetleme tanımı
 - Demetleme uygulamaları
- Demetleme Yöntemleri
 - Bölünmeli Yöntemler
 - Hiyerarşik Yöntemler
 - Yoğunluk Tabanlı Yöntemler
 - Model Tabanlı Yöntemler

42

Yoğunluk Tabanlı Yöntemler

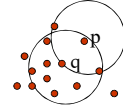
- Demetleme nesnelerin yoğunluğuna göre yapılır.
- Başlıca özellikleri:
 - Rastgele şekillerde demetler üretilebilir.
 - Aykırı nesnelerden etkilenmez.
 - Algoritmanın son bulması için yoğunluk parametresinin verilmesi gerekir.
- Başlıca yoğunluk tabanlı yöntemler:
 - DBSCAN**: Ester, et al. (KDD'96)
 - OPTICS**: Ankerst, et al (SIGMOD'99).
 - DENCLUE**: Hinneburg & D. Keim (KDD'98)
 - CLIQUE**: Agrawal, et al. (SIGMOD'98)

43

DBSCAN

- İki parametre:
 - Eps**: En büyük komşuluk yarıçapı
 - MinPts**: Eps yarıçaplı komşuluk bölgesinde bulunan en az nesne sayısı
- $N_{eps}(p) = \{q \in D \mid d(p,q) \leq Eps\}$
- Doğrudan erişilebilir nesne: **Eps** ve **MinPts** koşulları altında bir q nesnesinin doğrudan erişilebilir bir p nesnesi şu şartları sağlar:
 - $p \in N_{eps}(q)$
 - q nesnesinin çekirdek nesne koşulunu sağlaması

$$N_{eps}(q) \geq MinPts$$



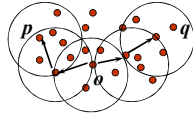
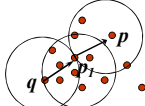
MinPts = 5

Eps = 1 cm

44

DBSCAN

- Erişilebilir nesne:
 - Eps** ve **MinPts** koşulları altında q nesnesinin erişilebilir bir p nesnesi olması için:
 - $p_1, p_2, \dots, p_n$ nesne zinciri olması,
 - $p_1 = q, p_n = p$,
 - p_i nesnesinin doğrudan erişilebilir nesnesi: p_{i+1}
- Yoğunluk bağlantılı Nesne:
 - Eps** ve **MinPts** koşulları altında q nesnesinin yoğunluk bağlantılı nesnesi p şu koşulları sağlar:
 - p ve q nesneleri **Eps** ve **MinPts** koşulları altında bir o nesnesinin erişilebilir nesnesidir.



45

Yoğunluk Tabanlı Yöntemler: DBSCAN

- Veri tabanındaki her nesnenin **Eps** yarıçaplı komşuluk bölgesi araştırılır.
- Bu bölgede **MinPts**'den daha fazla nesne bulunan p nesnesi çekirdek nesne olacak şekilde demetler oluşturulur.
- Çekirdek nesnelerin doğrudan erişilebilir nesneleri bulunur.
- Yoğunluk bağlantılı demetler birleştirilir.
- Hiçbir yeni nesne bir demete eklenmezse işlem sona erer.
- Yer karmaşıklığı – $O(n)$
- Zaman karmaşıklığı – $O(n \log n)$

n: nesne sayısı

46

Konular

- Demetleme işlemleri
 - Demetleme tanımı
 - Demetleme uygulamaları
- Demetleme Yöntemleri
 - Bölünmeli Yöntemler
 - Hiyerarşik Yöntemler
 - Yoğunluk Tabanlı Yöntemler
 - Model Tabanlı Yöntemler**

47

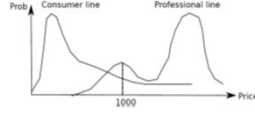
Model Tabanlı Demetleme Yöntemleri

- Veri kümesi için öngörülen matematiksel model en uygun hale getiriliyor.
- Verinin genel olarak belli olasılık dağılımlarının karışımından geldiği kabul edilir.
- Model tabanlı demetleme yöntemi
 - Modelin yapısının belirlenmesi
 - Modelin parametrelerinin belirlenmesi
- Örnek **EM (Expectation Maximization)** Algoritması

48

Olasılıksal Model Tabanlı Demetleme

- Demetlemede amaç gizli değişkenleri bulmak: sınıf bilgisi
- Gizli değişkenler veri üzerinde bir olasılık fonksiyonu ile matematiksel olarak ifade edilebilen bir olasılıksal dağılımdır
- Örnek: Satılan dijital kameraların sınıfı
 - Profesyonel veya amatör (C1 ve C2)
- Her sınıf için iki olasılık yoğunluk fonksiyonu: f1 ve f2
- Olasılıksal karışım modelleri gözlenen verinin farklı olasılık dağılımları ile temsil edilen demetlerin örnekleri olduğunu varsayar
- Gözlenen verideki her örnek birbirinden bağımsız



49

Model Tabanlı Demetleme Yöntemleri

- İstatistiksel yaklaşım:
 - K nesneden oluşan bir veri kümesi $D=\{x_1, x_2, \dots, x_K\}$
 - her $x_i (i \in [1, \dots, K])$ nesnesi θ parametre kümesiyle tanımlanan bir olasılık dağılımından oluşturulur.
 - Olasılık dağılımının, $c_g \in C = \{c_1, c_2, \dots, c_G\}$ şeklinde G adet bileşeni vardır.
 - Her $\theta_g, g \in [1, \dots, G]$ parametre kümesi g bileşeninin olasılık dağılımını belirleyen, θ kümesinin ayrışık bir alt kümesidir.
 - Herhangi bir x_i nesnesi öncelikle, $p(c_g | \theta) = \tau_g$ ($\sum_g \tau_g = 1$ olacak şekilde) bileşen katsayısına (ya da bileşenin seçilme olasılığına) göre bir bileşene atanır.
 - Bu bileşen $p(x_i | c_g, \theta_g)$ olasılık dağılımına göre x_i değişkenini oluşturur.
 - Böylece bir x_i nesnesinin bu model için olasılığı bütün bileşenlerin olasılıklarının toplamıyla ifade edilebilir:

$$p(x_i | \theta) = \sum_{g=1}^G p(c_g | \theta) p(x_i | c_g, \theta_g)$$

$$p(x_i | \theta) = \sum_{g=1}^G \tau_g p(x_i | c_g, \theta_g)$$

50

Model Tabanlı Demetleme Problemi

- Model parametrelerinin belirlenmesi
 - Maximum Likelihood (ML) yaklaşımı

$$\hat{\theta}_{ML} = \arg \max_{\theta} \prod_{i=1}^K p(x_i | \theta) = \prod_{i=1}^K \sum_{g=1}^G \tau_g p(x_i | c_g, \theta_g)$$

- Maximum Aposteriori (MAP) yaklaşımı

$$\hat{\theta}_{MAP} = \arg \max_{\theta} p(\theta | D) = \arg \max_{\theta} \frac{p(D | \theta) p(\theta)}{p(D)} = \frac{\prod_{i=1}^K \sum_{g=1}^G \tau_g p(x_i | c_g, \theta_g) p(\theta)}{\prod_{i=1}^K \sum_{g=1}^G \tau_g p(x_i | c_g, \theta_g) p(\theta)}$$

posterior likelihood prior

- Uygulamada her ikisinin logaritması

$$\arg \max_{\theta} \ln p(\theta | D) = \sum_{i=1}^K \ln \sum_{g=1}^G (\tau_g p(x_i | c_g, \theta_g))$$

$$\arg \max_{\theta} \ln p(D | \theta) = \sum_{i=1}^K \ln \sum_{g=1}^G (\tau_g p(x_i | c_g, \theta_g)) + \ln p(\theta)$$

51

EM Algoritması

- Veri kümesi: $D=\{x_1, x_2, \dots, x_K\}$
- Gizli değişkenler $H=\{z_1, z_2, \dots, z_K\}$ (her nesnenin hangi demete dahil olduğu bilgisi)
 - $z_j = \{x_j, y_1, y_2, \dots, y_G\}$
 - x_j verisi c_i demetinde ise $y_{ij}=1$
- Verinin eksik olduğu durumda, tam verinin beklenen değeri hesaplanır:

$$Q(\theta, \theta') = E[L_c(D, H | \theta) | D, \theta']$$

$$= \sum_{i=1}^K \sum_{g=1}^G p(c_g | x_i) [\ln p(x_i | c_g) + \ln \tau_g]$$

52

EM Algoritması

veri kümesi eksik: verinin sonrasal olasılığı (posterior probabilities) kullanılarak tamamlanmaya çalışılıyor

x	y	z			
x_1	?	$P(y_{11}=1 x_1, \theta)$	$P(y_{21}=1 x_1, \theta)$	$P(y_{31}=1 x_1, \theta)$	...
x_2	?	$P(y_{12}=1 x_2, \theta)$	$P(y_{22}=1 x_2, \theta)$	$P(y_{32}=1 x_2, \theta)$	...
x_3	?	$P(y_{13}=1 x_3, \theta)$	$P(y_{23}=1 x_3, \theta)$	$P(y_{33}=1 x_3, \theta)$	...
:	:	:	:	:	:

53

E ve M Adımları

- İki adım arasında yinelemeli
- E adımı: θ' parametrelerine dayanarak eksik veri için en iyi olası kestirimler
- M adımı: Eksik veri atandığında yeni parametrelere ilişkin kestirimler
- EM Algoritmasının adımları:
 - θ' için başlangıç değerleri atama
 - (E) Expectation: $Q(\theta | \theta')$ hesaplanması
 - (M) Maximization: $\arg \max Q(\theta | \theta')$

54

Normal Dağılım için EM Algoritması

- x_j verisi c_i demetinde ise $y_{ij}=1$ değilse $y_{ij}=0$

- E adımı:

$$P(c_i | x_j, \Theta^l) = P(y_{ij} = 1 | x_j, \Theta^l) = \frac{P(x_j | c_i) P(c_i)}{P(x_j)} = \frac{P(x_j | c_i, \mu_i^l, \sigma_i^l) P(c_i)}{\sum_{g=1}^G P(c_g) P(x_j | c_g, \mu_g^l, \sigma_g^l)} \equiv h_{ij}$$

- Normal dağılım için: $P(x_j | c_i, \mu_i, \sigma) = \frac{1}{(2\pi)^{d/2} \sigma^d} \exp \frac{-(x_j - \mu_i)^2}{2\sigma^2}$

$$P(c_i) = \frac{\sum_j h_{ij}}{K} \quad \mu_i^{l+1} = \frac{\sum_j h_{ij} x_j}{\sum_j h_{ij}}$$

- M adımı

$$\sigma_i^{l+1} = \frac{\sum_j h_{ij} (x_j - \mu_i^l)(x_j - \mu_i^l)^T}{\sum_j h_{ij}}$$

55

Örnek

- 51 örnekten oluşan tek boyutlu bir veri kümesi var. Bu veri kümesi iki farklı normal dağılımdan alınmış.

51	62	64	48	39	51
43	47	51	64	62	48
62	52	52	51	64	64
64	64	62	63	52	42
45	51	49	43	63	48
42	65	48	65	64	41
46	48	62	66	48	
45	49	43	65	64	
45	46	40	46	48	

56

Örnek - Devam

- EM algoritmasının çalışması
- $D = \{x_1, x_2, \dots, x_K\}$ veri kümesinden yararlanarak model parametrelerini belirlemek
- Bu örnek için 5 parametre belirlenmesi gerekiyor:

$$\mu_1, \sigma_1, \mu_2, \sigma_2, P(c_1).$$

- Bu parametreler ilk adımda tahmin ediliyor

57

Örnek - Devam

- Örnek için $P(c_1)=0,5$ olarak tahmin ediliyor ve veri kümesi ikiye ayrılarak normal dağılımın parametreleri için başlangıç değerleri atanıyor

$$\{51(*2.5), 52(*3), 62(*5), 63(*2), 64(*8), 65(*3), 66(*1)\};$$

$$\{51(*2.5), 49(*2), 48(*7), 47(*1), 46(*3), 45(*3), 43(*3), 42(*2), 41(*1), 40(*1), 39(*1)\};$$

$$\mu_1 = 47.63;$$

$$\sigma_1 = 3.79;$$

$$\mu_2 = 58.53;$$

$$\sigma_2 = 5.57.$$

58

Örnek - Devam

- x_i örneğinin demetlere dahil olma olasılığı:

$$h_{1i} = P(c_1 | x_i) = \frac{P(x_i | c_1) P(c_1)}{P(x_i)} = \frac{P(c_1)}{P(x_i)} \cdot \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x_i - \mu_1)^2}{2\sigma_1^2}}$$

$$h_{2i} = P(c_2 | x_i) = \frac{P(x_i | c_2) P(c_2)}{P(x_i)} = \frac{P(c_2)}{P(x_i)} \cdot \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(x_i - \mu_2)^2}{2\sigma_2^2}}$$

$$P(x_i) = \sum_{g=1}^2 P(c_g) P(x_i | c_g)$$

$$= P(c_1) \cdot \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x_i - \mu_1)^2}{2\sigma_1^2}} + P(c_2) \cdot \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(x_i - \mu_2)^2}{2\sigma_2^2}}$$

59

Örnek - Devam

- E - Adımı: Olasılıkların hesaplanması

- Örnek: Değeri 52 olan bir örnek için olasılıklar:

$$h_{1j} = P(c_1 | x_j = 52) = \frac{0.5}{P(x = 52)} \cdot \frac{1}{\sqrt{2\pi} \cdot (3.79)} e^{-\frac{(52 - 47.63)^2}{2 \cdot (3.79)^2}}$$

$$h_{2j} = P(c_2 | x_j = 52) = \frac{0.5}{P(x = 52)} \cdot \frac{1}{\sqrt{2\pi} \cdot 5.57} e^{-\frac{(52 - 58.53)^2}{2 \cdot (5.57)^2}}$$

60

Örnek - Devam

- Parametrelerin yeni değerleri

$$\mu_1^2 = \frac{\sum_{j=1}^n h_{1j} x_j}{\sum_{j=1}^n p_{1j}}; \mu_2^2 = \frac{\sum_{j=1}^n (1 - h_{1j}) x_j}{\sum_{j=1}^n (1 - h_{1j})}$$

$$\sigma_1^2 = \frac{\sum_{j=1}^n h_{1j} (x_j - \mu_1)^2}{\sum_{j=1}^n p_{1j}}; \sigma_2^2 = \frac{\sum_{j=1}^n (1 - h_{1j}) (x_j - \mu_2)^2}{\sum_{j=1}^n (1 - h_{1j})}$$

$$P(C_1) = \frac{\sum_{j=1}^n h_{1j}}{n}$$

- İşlem demetleme sonucu yakınsayana kadar devam ediyor

61