

Lecture 11

Density-Based Clustering

Mercoledì, 24 novembre 2004

Giuseppe Manco

Readings:

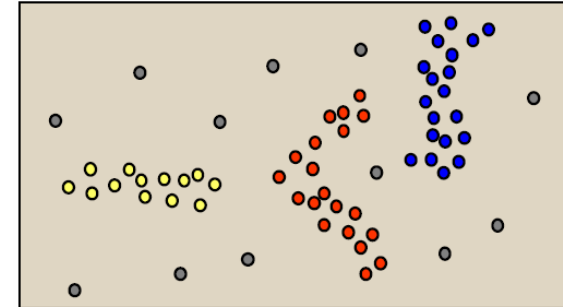
Chapter 8, Han and Kamber

Chapter 14, Hastie , Tibshirani and Friedman

Clustering basato su densità

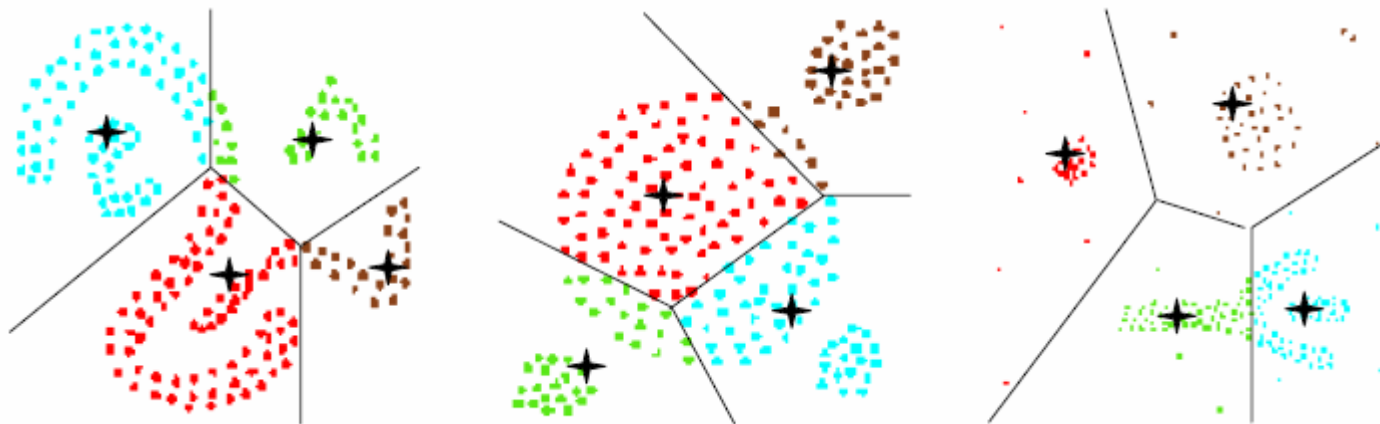
- **Idea di base:**

- I cluster sono regioni ad alta densità
- I cluster sono separati da regioni a bassa densità



- **Perché clustering basato su densità?**

- Algoritmo k-means e varianti inadeguati con clusters strutturati diversamente

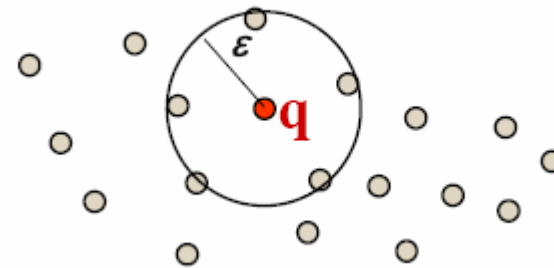


Concetti di base

- **Idea**
 - Ogni punto in un cluster è caratterizzato da una densità locale
 - L'insieme dei punti in un cluster è connesso spazialmente
- **Densità locale**
 - Intorno ε
$$N_{\varepsilon}(x) = \{y \in D \mid \text{dist}(x, y) \leq \varepsilon\}$$
 - **MinPoints**
 - Numero minimo di punti richiesti in $N_{\varepsilon}(x)$
- **Un oggetto q è un core-object se, dati ε e **MinPts**,**

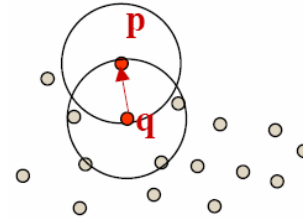
$$\left| N_{\varepsilon(q)} \right| \geq \text{MinPts}$$

MinPts=5

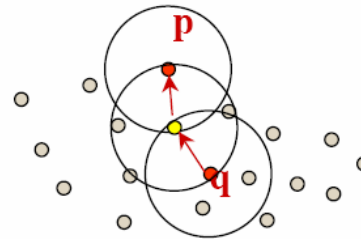


Concetti di base [2]

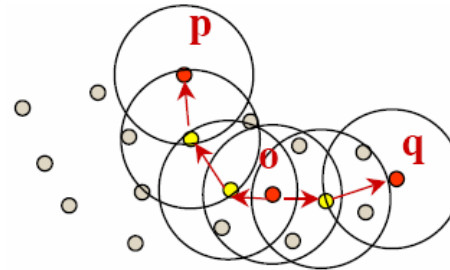
- **p è raggiungibile da q se**
 - q è un core-object
 - $p \in N_\epsilon(q)$



- **Chiusura transitiva**



- **p connesso a q se**
 - Esiste o
 - p e q sono raggiungibili da o



Concetti di Base [3]

- **Cluster**

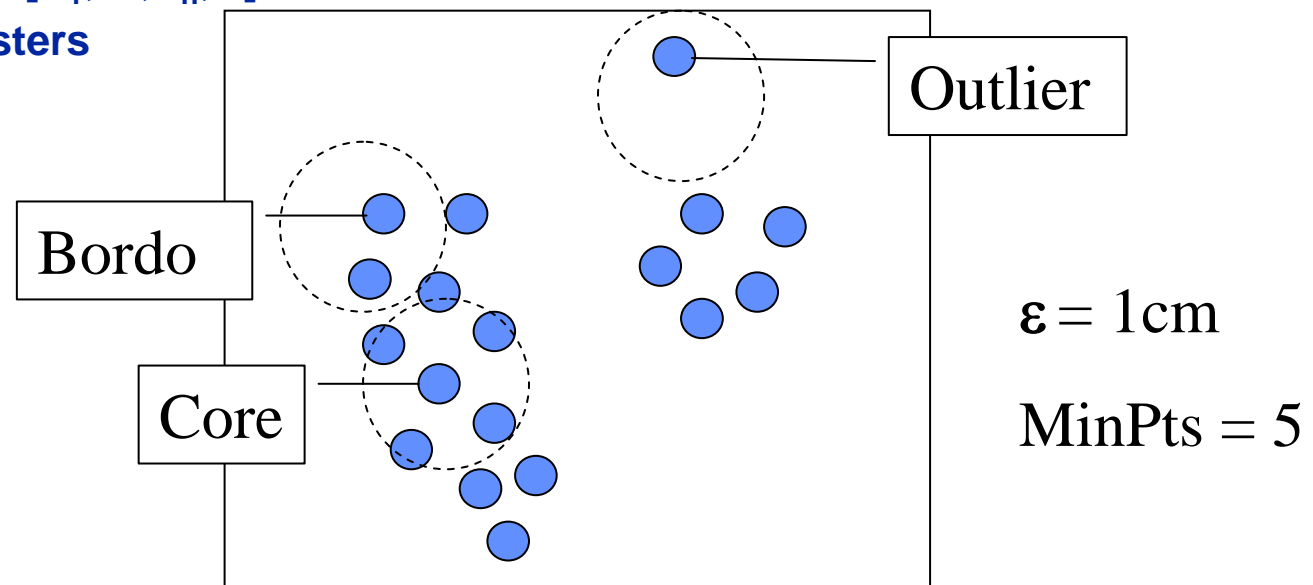
- Sottinsieme $S \subseteq D$

- Massimale: se $p \in S$ e q è raggiungibile da p allora $q \in S$
- Connesso: ogni oggetto in S è connesso agli altri oggetti in S

- **Clustering**

- Partizione di $D = [S_1; \dots; S_n; N]$

- $S_1; \dots; S_n$ clusters
- N outliers



L'algoritmo DBScan

- **Assunzione fondamentale**

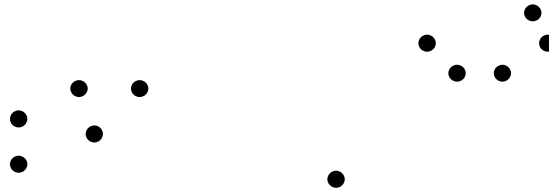
- Ogni oggetto in un cluster S è raggiungibile da un qualsiasi core-object
- Gli outliers non sono raggiungibili dai core-objects

```
1.FOR EACH  $x \in D$ 
2.     IF  $x$  non appartiene a nessun cluster
3.         IF  $x$  è un core-object
4.              $S_x = \{y \mid y \text{ raggiungibile da } x\}$ 
           ELSE
5.              $N = N \cup \{x\}$ 
```

- Passo 4: gli y raccolti tramite una serie di queries di neighborhood

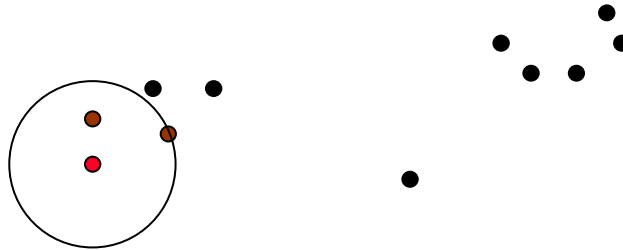
DBScan: esempio

- $\varepsilon=2cm$
- **MinPts =3**



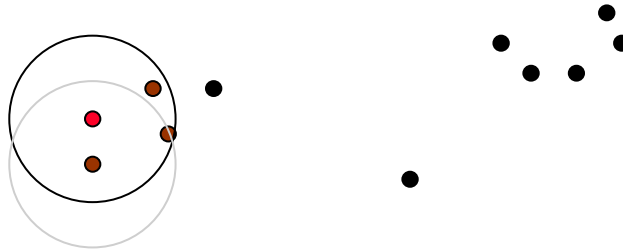
DBScan: esempio

- $\varepsilon=2\text{cm}$
- $\text{MinPts}=3$



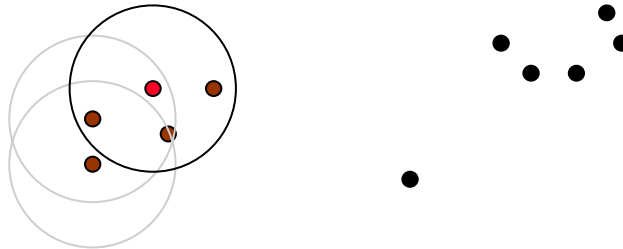
DBScan: esempio

- $\varepsilon=2\text{cm}$
- $\text{MinPts}=3$



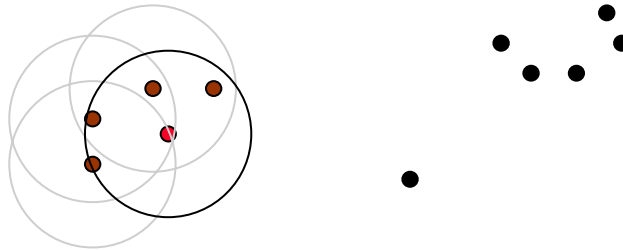
DBScan: esempio

- $\varepsilon=2cm$
- **MinPts =3**



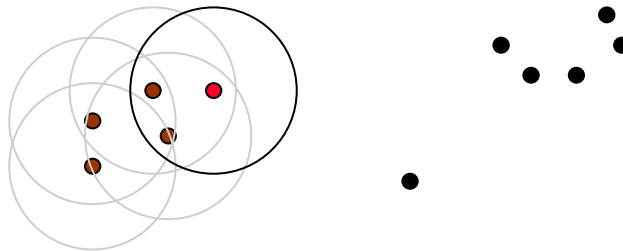
DBScan: esempio

- $\varepsilon=2cm$
- **MinPts =3**



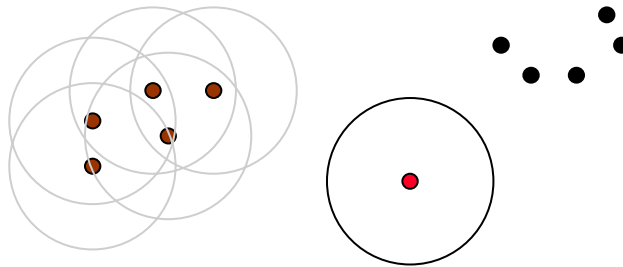
DBScan: esempio

- $\varepsilon=2cm$
- **MinPts =3**



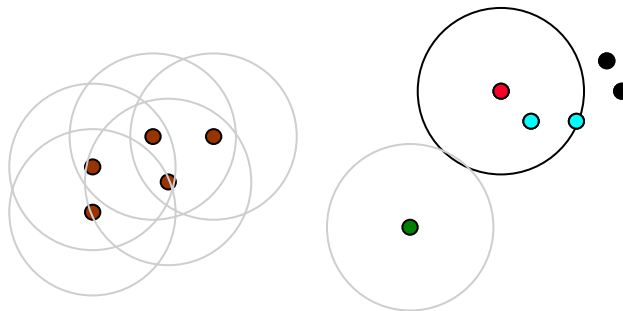
DBScan: esempio

- $\epsilon=2cm$
- **MinPts =3**



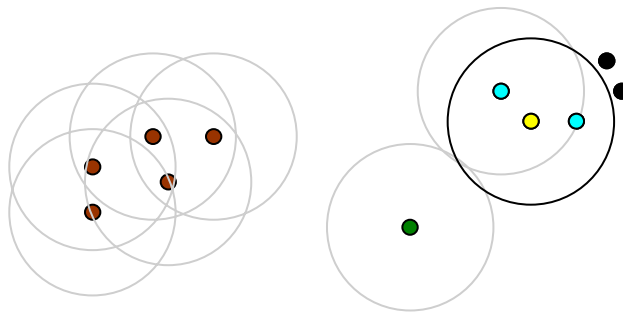
DBScan: esempio

- $\epsilon=2cm$
- **MinPts =3**



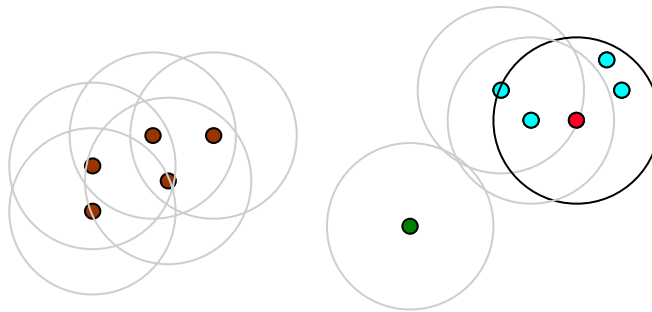
DBScan: esempio

- $\epsilon=2cm$
- **MinPts =3**



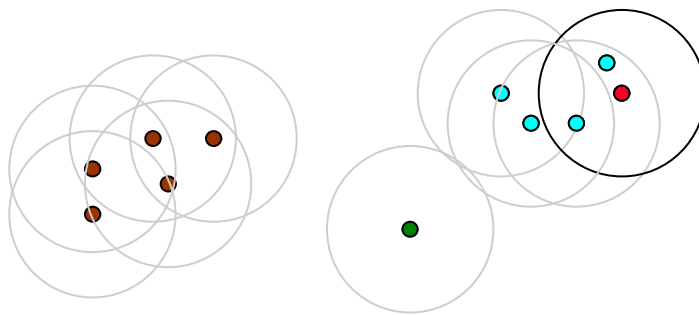
DBScan: esempio

- $\varepsilon=2\text{cm}$
- $\text{MinPts}=3$



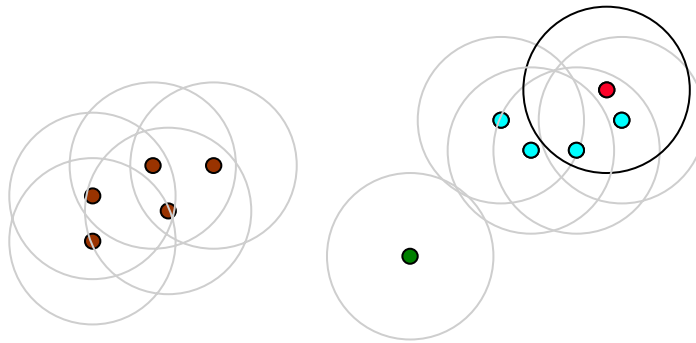
DBScan: esempio

- $\varepsilon=2cm$
- **MinPts =3**

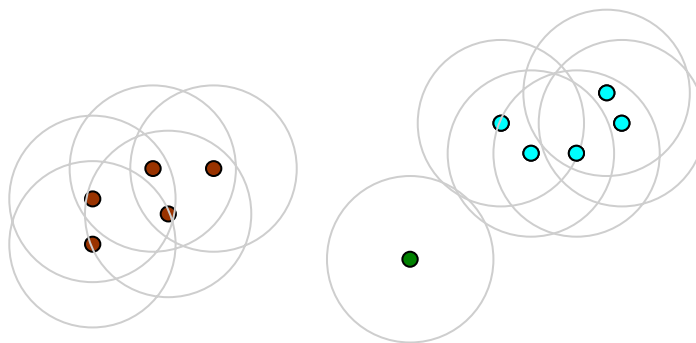


DBScan: esempio

- $\varepsilon=2cm$
- **MinPts =3**



DBScan: esempio



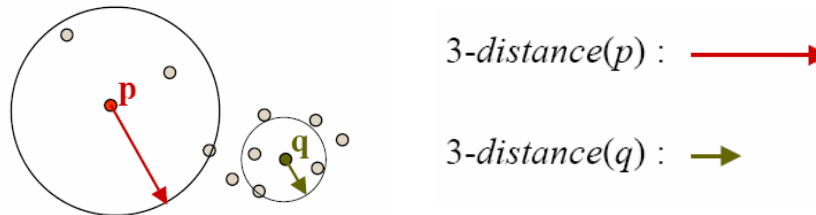
DBScan: Proprietà

- **Complessità computazionale**
 - Senza supporto alle queries di neighborhood: $O(n^2)$
 - $O(n)$ per ogni query
 - Supporto basato su strutture ad albero (alberi metrici, R-tree, ecc): $O(n \log n)$
 - $O(\log n)$ per ogni query
 - Accesso diretto ai Neighbors: $O(n)$
 - $O(1)$ per ogni query
- **Confronti Run-time**

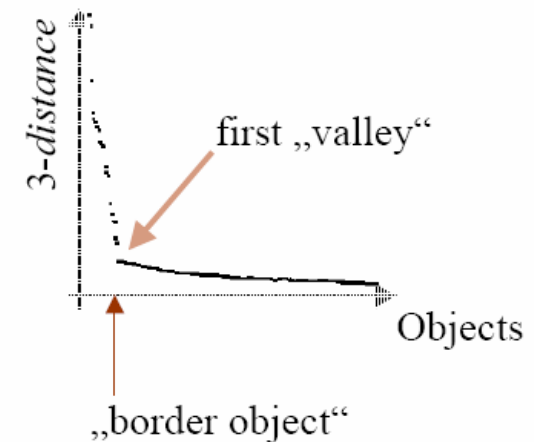
	Time (sec.)									
No. of Points	1,252	2,503	3,910	5,213	6,256	7,820	8,937	10,426	12,512	62,584
DBSCAN	3	7	11	16	18	25	28	33	42	233
CLARANS	758	3,026	6,845	11,745	18,029	29,826	39,265	60,540	80,638	?????

Come si determinano i parametri ottimali?

- **Cluster**
 - Insieme di oggetti di densità alta
 - Dipende da ε e MinPts
- **Idea: Stimiamo i parametri con la densità di ogni oggetto**
- **k-distance(x): la distanza dal suo k-esimo neighbor**



- **Euristica: fissiamo MinPts, scegliamo ε come punto di “taglio”**



Sommario

- **Clustering basato su densità**
 - **Vantaggi**
 - Clusters di dimensione e forma arbitraria
 - Numero di clusters non predeterminato
 - Ben supportato da indici spaziali
 - Determina automaticamente il rumore e gli outliers
 - **Svantaggi**
 - Difficile determinare i parametri giusti
 - In alcuni casi molto sensitivo ai parametri