

DENCUS

DBNsity-based CLUstEring

DBSCAN e OPTICS usano concetti di densità basati sul contatore i vicini entro un raggio ϵ , quindi sono molto dipendenti da ϵ .

DENCUS cerca di descrivere la densità in modo più rigoroso, ovvero usando **funzioni di distribuzione della densità**.

Ogni oggetto è usato come un indicatore della alta probabilità che ci sia alta densità nella regione circostante al punto sotto analisi. Questa probabilità dipende dalla distanza del punto in osservazione e i punti precedentemente osservati.

Franciamo $x_1 \dots x_n$ campioni indipendenti di una certa variabile aleatoria f . Il **Kernel density approximation** della funzione di probabilità di densità è:

$$\hat{f}_h(x) = \frac{1}{n \cdot h} \sum_{i=1}^n K\left(\frac{x-x_i}{h}\right)$$

dove oppunto $K(\cdot)$ è il Kernel, una funzione non negativa, definita in \mathbb{R} e integrabile che soddisfa le seguenti proprietà

$$3) \int_{-\infty}^{+\infty} K(u) du = 1 \quad 2) K(-u) = K(u)$$

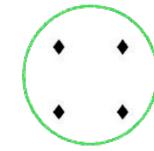
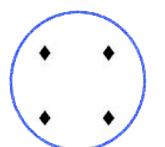
Esempio di una funzione Kernel:

$$K\left(\frac{x-x_i}{h}\right) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-x_i)^2}{2h^2}}$$

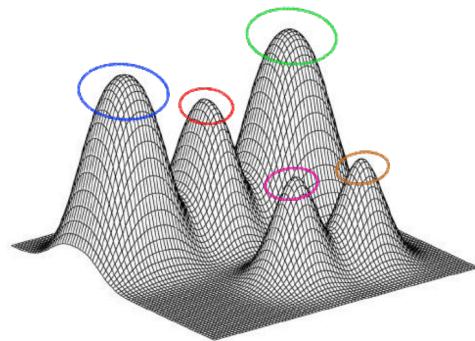
Kernel GAUSSIANO.

INTUIZIONE DELLA FORMULA

Noi abbiamo un campione di punti indipendenti e egualmente distribuiti. Per ogni punto del campione valutiamo il contributo degli altri punti nell'area. Infatti $f_h(x)$ valuta l'altezza in x nello spazio che è data dalla somma dei contributi degli altri punti del campione. Questo fa sì che punti vicini si influenzino di più rispetto a punti distanti da x : il cui contributo è basso se non 0.



Set of 12 points.

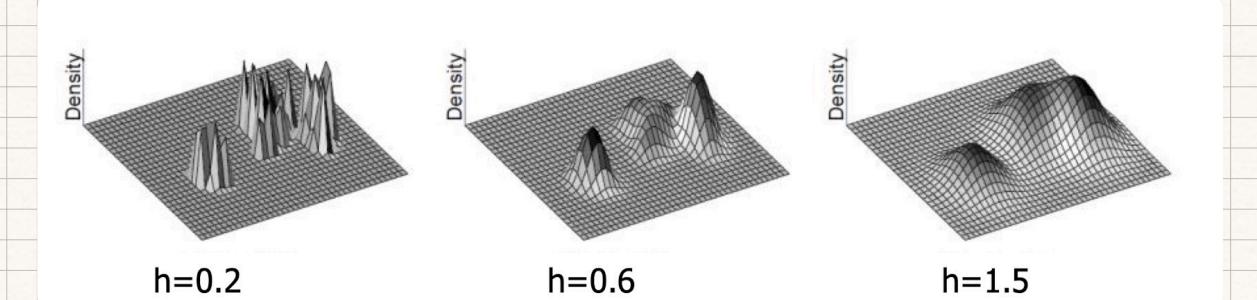


Overall density—surface plot.

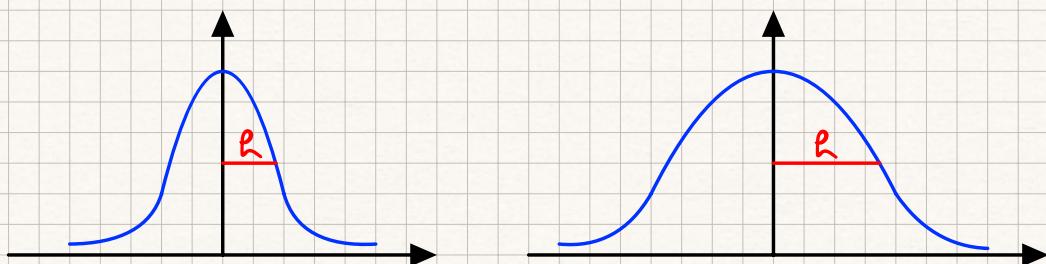
Più alta è la montagna, più alta è la probabilità di avere una regione densa di punti.

PARAMETRO h

h è un parametro da settare da cui poi deriverà la forma della $f_h(x)$. Se prendiamo la GAUSSIANA, con h si indica la deviazione standard della Gaussiana stessa e quindi, esteticamente $\sigma=h$ fa riferimento a quando "larga" sia la curva e anche quando dettaglio si vede.



Più h è alto più il contributo di ogni punto da un contributo maggiore perché la sua Gaussiane comprendrà uno spazio maggiore:

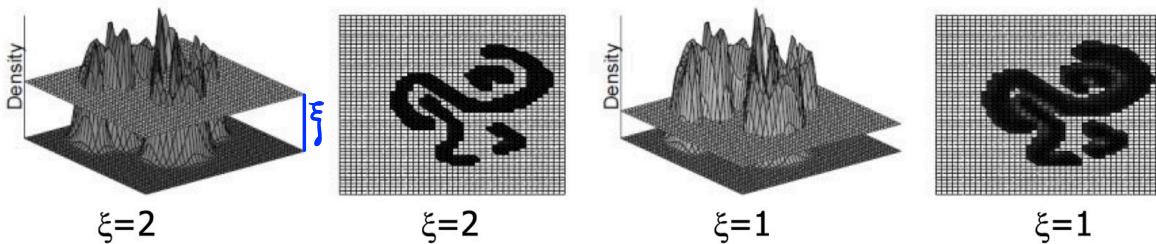


DENSITY ATTRACTOR

I veri cluster possono essere matematicamente determinati tramite i **density attractor**, che sono i massimi locali della $\hat{f}_h(x)$, la vetta delle montagne.

PARAMETRO ξ (Noise threshold)

Il parametro ξ definisce come DENCUD crea i cluster, perché ci dice quali density attractor unire per generare un vero cluster. È una questione di togliere le "montagne" trovate.



Quindi vengono considerati solo i density attractor che soddisfano

$$\hat{f}(x^*) \geq \xi$$

di fatti escludendo tutti quei punti, o quei density attractor sotto lo threshold.

I density attractor fungono da centro del cluster e i veri oggetti vengono attribuiti a un density attractor tramite una procedura di tipo hill-climbing

outlier: $x \in \text{density attractor} : \hat{f}(x) < \xi$

ALGORITMO

Bisogna trovare i massimi della $f_h(x)$ che poi corrispondono al centro delle aree ad alta densità:

1) Finché ci sono sample x :

Partiamo da $x_0 = x$ per cercare i punti detti density attractor (i massimi in f_h) tramite una procedura di hill-climbing tramite il gradiente:

$$x^{i+1} = x^i + \delta \frac{\nabla \hat{f}(x_i)}{|\nabla \hat{f}(x_i)|}$$

Dove δ è un parametro per controllare i passi da fare e:

$$\nabla \hat{f}(x) = \frac{1}{h^{d+2} \cdot n \sum_{i=1}^n k\left(\frac{|x-x_i|}{h}\right)(x-x_i)}$$

con d che rappresenta lo spazio dimensionale in cui giace K .

2) Questa procedura continua finché:

$$\hat{f}(x^{k+1}) < \hat{f}(x^k)$$

e conseguente x^k come density attractor.

Ottimizzazione

L'algoritmo salva tutti i punti x' : $\text{dis}(x_j, x') < \frac{h}{2}$ con $0 < j < K$. Ovvero associa tutti i punti vicini $\frac{h}{2}$ al punto corrente al density attractor x^* senza applicare per questi punti tutta la procedura di hill-climbing.

Merge density attractor

Settata una threshold, possono venire uniti i cluster che fanno riferimento a quel density attractor che sono connessi da percorsi altamente densi. Così da avere clusters di forme arbitraria.

$$\forall x \in C \exists x^* \in X : \hat{f}(x) > \xi$$

$\forall x_1^*, x_2^* \in X : \exists$ un path $P \in F^d$ che parte da x_1^* e arriva a x_2^* con $\forall p \in P : \hat{f}(p) \geq \xi$.

Come scegliere h ?

Testando e procedendo h nel massimo intervallo fra h_{\max} e h_{\min} in cui il merge li density-attractor è costante

Come scegliere ξ ?

Assumendo il nostro dataset "noise free", tutti i density attractor sono significativi (Nessun outlier). In queste condizioni:

$$0 \leq \xi \leq \min(f^{D_c}(x^*))$$

Pro

- 1) Basato sulla matematica.
- 2) Buona gestione del rumore
- 3) Descrive matematicamente le forme più disperate.
- 4) Veloce, **COMPRESSENTE** $O(n \log n)$

Centro

- 1) Scelta accurata di h che determina quanto un punto influenza il suo vicino
- 2) Scelta accurata di ϵ_g .

IMPLEMENTAZIONE

- 1) Iniziere clusterizzando tramite un griglia per velocizzare la procedura.

Il minimo rettangolo che contiene il DB è suddiviso in sotto cubi di altezza $2h$. Si prendono in considerazione i soli cubi che contengono punti.

31	32	33	34	35	36
25	26	27	28	29	30
19	20	21	22	23	24
13	14	15	16	17	18
7	8	9	10	11	12
1	2	3	4	5	6

Per ogni cubo abbiamo

- 1) Key : Id incrementale
- 2) N_c : # punti nel cubo
- 3) Puntatori ai punti
- 4) $\sum X_i$: somma linea dei punti

Queste info sono usate per velocizzare i calcoli.

Ogni cubo viene connesso a un cubo vicino se:

$$d(\text{mean}(C_1), \text{mean}(C_2)) < k\sigma$$

dove si usano il Kernel Gaussiano $\hat{G} = h$.

Se due cubi vengono connessi : $C_1, C_2 \in C_p$

2) Determinazione dei cluster

Vengono usati solo i cubi altamente densi : C_{sp}
e i cubi connessi a C_{sp} : C_r

$$C_{sp} = \{c \in C_p \mid N_c > \xi_c\}$$

$$C_r = C_{sp} \cup \{c \in C_p \mid \exists c_s \in C_{sp} \text{ and } \exists \text{ connessione}(c_s, c)\}$$

Per $x \in C$ e $c_1, c_2 \in C_r$ settiamo

$$\text{near}(x) = \{x_i \in C_r \mid d(\text{mean}(c_i), x) \leq k\sigma \vee \exists \text{ connessione}(c_i, c)\}$$

Dove il limite $k\sigma$ è scelto per far sì che solo i punti immediatamente vicini a x influenzino l'altezza del risultato.

La funzione di densità totale risultante sarà :

$$\hat{f}_{GAUSS}(x) = \sum_{x_i \in \text{near}(x)} e^{-\frac{d(x, x_i)^2}{2\sigma^2}}$$

Come prima eseguiamo un algoritmo di tipo Hill-Climbing con un certo passo d .