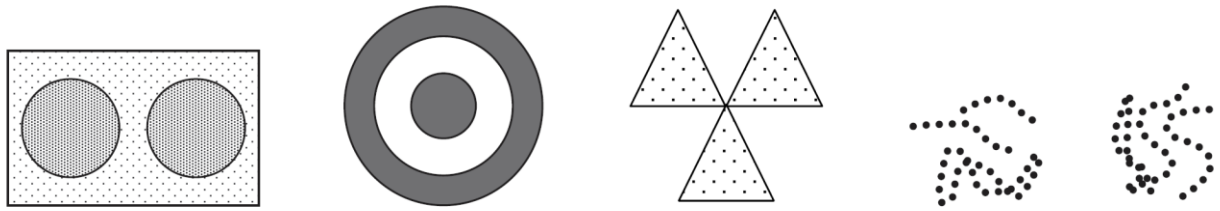
Tipi di clustering

Esercizio 1

Identificare i cluster nella figura seguente utilizzando le tecniche: center-based (o prototype-based), contiguity-based e density based.



(a) (b) (c) (d)

Center-based = un cluster è dato dall'insieme degli oggetti che risultano più vicini (o simili) al un prototipo del cluster in oggetto, rispetto al prototipo di ogni altro cluster.
Prototipo = centro del cluster.
Es. K-means

(a) 2 cluster. La regione rettangolare viene suddivisa in due parti e gli elementi in essa contenuti verranno ripartiti tra i due cluster. Nota: del rumore viene aggiunto ad entrambi i cluster.

(b) 1 cluster che include entrambi gli anelli. Infatti, il prototipo del cluster è dato dal centro degli anelli concentrici.

(c) 3 cluster, uno per ogni regione triangolare. In alternativa si può pensare anche ad un unico cluster il cui prototipo è rappresentato dal punto di incontro tra i tre triangoli.

(d) 2 cluster: viene creato un cluster per ciascun gruppo di "linee".

Contiguity-based = due oggetti sono connessi solo se si trovano entro una certa distanza.
Es. in una rappresentazione graph-based di un cluster con criterio MIN (= single link)

(a) 1 cluster, perché le due regioni circolari sono unite dal rumore.

(b) 2 cluster, uno per ogni anello.

(c) 1 cluster, poiché i tre triangoli si incontrano nel punto centrale.

(d) 5 cluster: le linee che si intrecciano vengono posizionate all'interno di uno stesso cluster, mentre le linee disgiunte vengono considerate cluster separati.

Density based = un cluster è una zona densa di oggetti che è contornata da una regione a bassa densità.
Es. DB-SCAN

(a) 2 cluster, uno per ogni regione circolare. Il rumore viene eliminato.

(b) 2 cluster, uno per ogni anello.

(c) 3 cluster, uno per ogni triangolo. Anche se i tre triangoli si toccano, la densità della regione nel punto di incontro è minore rispetto alla densità all'interno dei triangoli.

(d) 2 cluster, i due gruppi di linee costituiscono ciascuno un cluster, cioè un'area ad alta densità separata da un'area a bassa densità.

K-MEANS

Algoritmo

1. Selezionare K punti come centroidi iniziali
2. Ripetere
4. Formare K clusters assegnagno ciascun punto al centroide più vicino
5. Ricalcolare i centroidi
6. Finchè i centroidi non rimangono invariati

Esercizio 2

Si consideri l'insieme dei seguenti punti 1-dimensionali: {6, 12, 18, 24, 30, 42, 48}.

Per ciascuno dei seguenti insiemi di centroidi iniziali, creare i due cluster assegnando i punti al suo centroide più vicino, and quindi calcolare l'errore quadratico totale (SSE) per i due cluster.

(a) C = {18, 45}

	6	12	18	24	30	42	48
dist(x, 18)	12	6	0	6	12	4	30
dist(x, 45)	39	33	27	21	15	3	3

C1 = {6, 12, 18, 24, 30}

$SSE(C1) = (6-18)^2 + (12-18)^2 + (18-18)^2 + (24-18)^2 + (30-18)^2 = 360$

C2 = {42, 48}

$SSE(C2) = (42-45)^2 + (48-45)^2 = 18$

Total SSE = 360 + 18 = 378

(b) C = {15, 40}

	6	12	18	24	30	42	48
dist(x, 15)	9	3	3	9	15	27	33
dist(x, 40)	34	28	22	16	10	2	8

C1 = {6, 12, 18, 24}

$SSE(C1) = (6-15)^2 + (12-15)^2 + (18-15)^2 + (24-15)^2 = 180$

C2 = {30, 42, 48}

$SSE(C2) = (30-40)^2 + (42-40)^2 + (48-40)^2 = 168$

Total SSE = 180 + 168 = 348

(c) e due soluzioni rappresentano una situazione stabile?

Si, perché in entrambi i casi i centroidi calcolati sui nuovi cluster non cambiano:

Caso (a):

C1 = {6, 12, 18, 24, 30} → punto medio = 6 + (30-6)/2 = 18

C2 = {42, 48} → punto medio = 42 + (48-42)/2 45

Caso (b):

C1 = {6, 12, 18, 24} → punto medio = 6 + (24-6)/2 = 15

C2 = {30, 42, 48} → punto medio = 30 + (48-30)/2= 39

Il secondo centroide viene modificato, ma all'iterazione successiva i cluster non cambiano, quindi la situazione diventa stabile entro 2 iterazioni.

Hierarchical Clustering

Basic agglomerative algorithm

1. Calcolare la matrice di similarità
2. Ciascun punto diventa un cluster
3. Ripetere
4. Eseguire il merge di **due cluster vicini** → definizione di «prossimità» tra due cluster
5. Aggiornare la matrice di similarità
6. Finchè non rimane un unico cluster

Prossimità (cluster graph-based)

- MIN (single link): cluster proximity = vicinanza tra i punti più vicini in due cluster diversi
- MAX (complete link) : cluster proximity = vicinanza tra i due punti più distanti in due cluster diversi.
- GROUP AVERAGE : cluster proximity = vicinanza media tra tutte le coppie di punti in due cluster diversi.

Prossimità (prototype-based)

- Prossimità = distanza tra i centroid
- Ward's method: usa il parametro SSE = la prossimità tra due cluster è data dall'incremento in SSE che risulta dall'eseguire il merge di due cluster.

SSE = $\sum_{i=1}^{K} \sum_{x \in C_i} (c_i, x)^2$

Esercizio 3

Usare la matrice di similarità seguente per eseguire un cluster gerarchico con la tecnica single link e complete link. Mostrare i risultati attraverso un dendrogramma.

Similarity matrix

	p1	p2	p3	p4	p5
p1	1.00	0.10	0.41	0.55	0.35
p2	0.10	1.00	0.64	0.47	0.98
p3	0.41	0.64	1.00	0.44	0.85
p4	0.55	0.47	0.44	1.0	0.76
p5	0.35	0.98	0.85	0.76	1.00

Single link

Passo 0: ciascun punto rappresenta un cluster

C = {{p1}, {p2}, {p3}, {p4}, {p5}}

	p1	p2	p3	p4	p5
p1	1.00	0.10	0.41	0.55	0.35
p2	0.10	1.00	0.64	0.47	0.98
p3	0.41	0.64	1.00	0.44	0.85
p4	0.55	0.47	0.44	1.00	0.76
p5	0.35	0.98	0.85	0.76	1.00

Passo 1: determinare due cluster vicini → cerco la coppia di punti (cluster) che sono più vicini = hanno la similarità massima (vedi cella in rosa nella tabella sopra).

Attenzione che nelle slide abbiamo usato la matrice di distanza (che va quindi minimizzata), qui c'è la similarità (che va invece massimizzata)

C = {{p1}, {{p2},{p5}}, {p3}, {p4}} → altezza del dendrogramma = 1 – 0.98

Devo ricalcolare la matrice di similarità

	p1	p2, p5	p3	p4
p1	1.00	0.35	0.41	0.55
p2, p5	0.35	1.00	0.85	0.76
p3	0.41	0.85	1.00	0.44
p4	0.55	0.76	0.44	1.00

La similarità tra {p1} e {p2, p5} è data dalla similarità tra i punti più vicini nei due cluster.

es . Similarity({p1}, {p2, p5}) = max (similarity(p1, p2), similarity(p1, p5)).

Passo 2: i cluster più vicini sono {p2, p5} e {p3}.

C = {{p1}, {{{p2},{p5}},p3}, {p4}} → altezza del dendrogramma = 1 – 0.85

Devo ricalcolare la matrice di similarità:

es. Similarity({p1}, {p2, p5, p3}) = max (similarity(p1, p2), similarity(p1, p5), similarity(p1,p3)).

	p1	p2, p5, p3	p4
p1	1.00	0.41	0.55
p2, p5, p3	0.41	1.00	0.76
p4	0.55	0.76	1.00

Passo 3: i cluster più vicini sono {p2, p5} e {p3}.

C = {{p1}, {{{{p2},{p5}},p3},{p4}} → altezza del dendrogramma = 1 – 0.76

Devo ricalcolare la matrice di similarità:

	p1	p2, p5, p3, p4
p1	1.00	0.55
p2, p5, p3, p4	0.55	1.00

Passo 4: posso unire tutti i punti in un cluster unico

C = {{{p1}, {{{{p2},{p5}},p3},{p4}} → altezza del dendrogramma = 1 – 0.55

Complete link

Passo 0: ciascun punto rappresenta un cluster

C = {{p1}, {p2}, {p3}, {p4}, {p5}}

	p1	p2	p3	p4	p5
p1	1.00	0.10	0.41	0.55	0.35
p2	0.10	1.00	0.64	0.47	0.98
p3	0.41	0.64	1.00	0.44	0.85
p4	0.55	0.47	0.44	1.00	0.76
p5	0.35	0.98	0.85	0.76	1.00

Passo 1: determinare due cluster vicini → cerco la coppia di punti (cluster) che sono più vicini = hanno la similarità massima (vedi cella in rosa nella tabella sopra) → questo passo è uguale al single link perché i cluster sono formati da un solo elemento.

C = {{p1}, {{p2},{p5}}, {p3}, {p4}} → altezza del dendrogramma = 1 – 0.98

Devo ricalcolare la matrice di similarità

	p1	p2, p5	p3	p4
p1	1.00	0.10	0.41	0.55
p2, p5	0.10	1.00	0.64	0.47
p3	0.41	0.64	1.00	0.44
p4	0.55	0.47	0.44	1.00

La similarità tra {p1} e {p2, p5} è data dalla similarità tra i punti più distanti nei due cluster.

es. Similarity({p1}, {p2, p5}) = min (similarity(p1, p2), similarity(p1, p5)).

Passo 2: i cluster più vicini sono {p2, p5} e {p3}.

C = {{p1}, {{{p2},{p5}},p3}, {p4}} → altezza del dendrogramma = 1 – 0.64

Devo ricalcolare la matrice di similarità:

Similarity({p1}, {p2, p5, p3}) = max (similarity(p1, p2), similarity(p1, p5), similarity(p1,p3)).

	p1	p2, p5, p3	p4
p1	1.00	0.10	0.55
p2, p5, p3	0.10	1.00	0.44
p4	0.55	0.44	1.00

Passo 3: i cluster più vicini sono {p1} e {p4}.

C = {{{p1},{p4}}, {{{p2},{p5}},p3}} → altezza del dendrogramma = 1 – 0.55

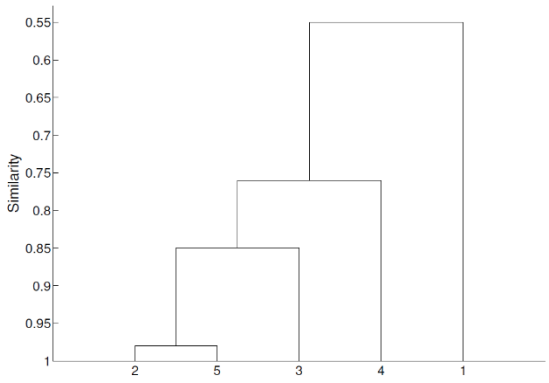
Devo ricalcolare la matrice di similarità:

	p1, p4	p2, p5, p3
p1, p4	1.00	0.10
p2, p5, p3	0.10	1.00

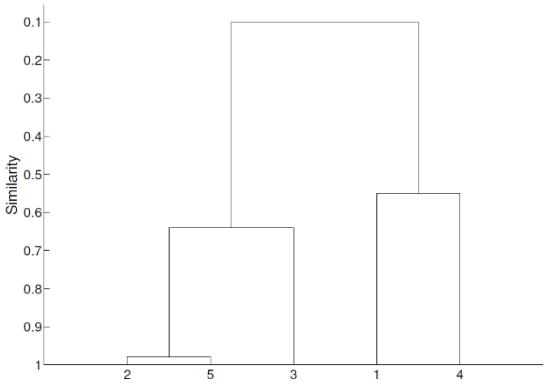
Passo 4: posso unire tutti i punti in un cluster unico

C = {{{{p1}, {p4}}, {{{p2}, {p5}}, {p3}}} → altezza del dendrogramma = 1 – 0.1

Attenzione che nei dendrogrammi seguenti nell'asse delle y è stata messo il valore di similarità che parte da 1 e arriva a 0, si può leggere anche come distanza da 0 a 1 considerando i valori come 1 - valore riportato nel grafico (vedi parte testuale del testo).



(a) Single link.



(b) Complete link.

DB-SCAN

Density-based clustering algorithm: usa una nozione di densità invece che di similarità.

La densità di un certo punto è determinata dal numero di punti che si trovano entro una certa distanza eps da esso.

- Core point: interno ad un'area densa → ci sono almeno MinPts punti ad una distanza eps da esso
- Border point: al confine di un'area densa → non è un core point, ma ricade nelle vicinanze di un core point.
- Noise point: in un'area occupata in modo sparso → non è ne un core point, ne un border point.

Algortmo:

1. Etichettare tutti i punti come core, border o noise point.
2. Eliminare i noise point.
3. Tracciare un arco tra tutti i core point che si trovano ad una distanza eps tra loro.
4. Inserire tutti i core point connessi tra loro nello stesso cluster.
5. Assegnare i border point al cluster associato ad uno dei suoi core point.

Esercizio 4

Qual è il comportamento dell'algoritmo DBSCAN su un dataset uniforme? E su un dataset casuale?

Nel caso di un dataset uniformemente distribuito, DBSCAN assegnerebbe tutti i punti allo stesso cluster, oppure ciascun punto ad un cluster diverso, a seconda della distanza *eps*.

DBSCAN sarebbe in grado di determinare dei cluster anche in dati casuali, basta che ci siano delle variazioni di densità.

Validazione della clusterizzazione

Unsupervised classification – silhouette

- cohesion(C_i) = $\sum_{x \in C_i} proximity(x, c_i)$
- total cohesion = $\sum_i \sum_{x \in C_i} (x - c_i)^2$

c_i is the centroid of C_i

- separation(C_i) = $proximity(c_i, c)$
- total separation = $\sum_i |C_i| (c - c_i)^2$

c_i is the centroid of C_i and c is the overall centroid

- silhouette: it combines cohesion and separation into a unique measure
- silhouette = $\frac{b(o) - a(o)}{\max (a(o), b(o))}$

a(o) = average distance of object o to all the other objects in its cluster C_i
b(o) = minimum of the average distance of object o to all the other objects in other clusters C_j different from C_i

silhouette of a cluster C_i = average of the silhouettes of the points in C_i
overall silhouette for a clustering = average of the silhouettes of all points

Supervised classification – classification oriented

- Entropy: grado con cui ciascun cluster contiene oggetti di una sola classe

  Entropy(C_i) = $-\sum_{j=1}^{L} p_{ij} log_2 p_{ij}$ L = number of classes
  Total entropy = $\sum_{i=1}^{K} \frac{m_i}{m} Entropy(C_i)$

  m_i = number of objects in C_i
  m = total number of objects

- Purity: misura in cui un cluster contiene oggetti di una sola classe.

  Purity(C_i) = $max_j p_{ij}$

Total purity = $\sum_{i=1}^{K} \frac{m_i}{m} Purity(C_i)$

- Precision: misura la frazione di un cluster che è formata da oggetti di una certa classe.

  Precision(C_i, j) = p_{ij}

- Recall: misura in cui un cluster è costituito completamente da oggetti di una classe specifica

  Recall(C_i, j) = $\frac{m_{ij}}{m_j}$

  m_{ij} = number of objects in C_i of class j
  m_j = number of objects of class jù

- F1(C_i, j) = (2 * Precision(C_i, j) * Recall(C_i, j)) / (Precision(C_i, j) + Recall(C_i, j))

Supervised classification – similarity oriented

Rand index = $\frac{f_{00}+f_{11}}{f_{00}+f_{10}+f_{10}+f_{11}}$
f00 = number of pairs of objects having a different class and a different cluster
f10 = number of pairs of objects having a different class and the same cluster
f01 = number of pairs of objects having the same class and a different cluster
f11= number of pairs of objects having the same class and the same cluster

Esercizio 5

Dati i punti nella seguente tabella che riporta anche la loro distanza reciproca

	P1	P2	P3	P4
P1	0	0.20	0.35	0.45
P2	0.20	0	0.30	0.40
P3	0.35	0.30	0	0.10
P4	0.45	0.40	0.10	0

Calcolare il coefficiente silhouette per ciascun di essi, per ciascun cluster e per la clusterizzazione complessiva considerando i seguenti cluster

C1 = {p1, p2}
C2 = {p3, p4}

a(p1) = 0.20
b(p1) = (0.35 + 0.45) / 2 = 0.40
Silhouette(p1) = (0.40 – 0.20)/0.40 = 0.50

a(p2) = 0.20
b(p2) = (0.30 + 0.40) / 2 = 0.35
Silhouette(p2) = (0.35 – 0.20)/0.35 = 0.4286

a(p3) = 0.10
b(p3) = (0.35 + 0.30) / 2 = 0.325
Silhouette(p2) = (0.325 – 0.10)/0.325 = 0.6923

a(p4) = 0.10
b(p4) = (0.45 + 0.40) / 2 = 0.425
Silhouette(p2) = (0.425 – 0.10)/0.425 = 0.7647

Silhouette(C1) = (0.50 + 0.4286)/2 = 0.4643

Silhouette(C2) = (0.6923 + 0.7647)/2 = 0.7285

Silhouette totale = (0.50 + 0.428 + 0.6923 + 0.7647)/4 = 0.59625

Esercizio 6

Calcolare l'entropia e la purezza dei cluster rappresentati dalla seguente matrice di confusione.

Cluster	Entertainment	Financial	Foreign	Metro	National	Sports	Total
#1	1	1	0	11	4	676	693
#2	27	89	333	827	253	33	1562
#3	326	465	8	105	16	29	949
Total	354	555	341	943	273	738	3204

$p_{c1,entertainment} = 1/693 = 0.00144$
$p_{c1,financial} = 1/693 = 0.00144$
$p_{c1,foreign} = 0/693 = 0$
$p_{c1,metro} = 11/693 = 0.01587$
$p_{c1,national} = 4/693 = 0.00558$
$p_{c1,sports} = 676/693 = 0.97547$

$$\text{Entropy}(C1) = -p_{c1,entertainment}\log_2 p_{c1,entertainment} - p_{c1,financial}\log_2 p_{c1,financial} - p_{c1,foreign}\log_2 p_{c1,foreign} - p_{c1,metro}\log_2 p_{c1,metro} - p_{c1,national}\log_2 p_{c1,national} - p_{c1,sports}\log_2 p_{c1,sports}$$
$$= 0.01359 + 0.01359 + 0 + 0.09486 + 0.04176 + 0.03495$$
$$= 0.19875$$

Purity(C1) = 0.97547

$p_{c2,entertainment} = 27/1562 = 0.01729$
$p_{c2,financial} = 89/1592 = 0.05590$
$p_{c2,foreign} = 333/1592 = 0.20917$
$p_{c2,metro} = 827/1592 = 0.51947$
$p_{c2,national} = 253/1592 = 0.15892$
$p_{c2,sports} = 33/1592 = 0.02073$

$$\text{Entropy}(C2) = -p_{c2,entertainment}\log_2 p_{c2,entertainment} - p_{c2,financial}\log_2 p_{c2,financial} - p_{c2,foreign}\log_2 p_{c2,foreign} - p_{c2,metro}\log_2 p_{c2,metro} - p_{c2,national}\log_2 p_{c2,national} - p_{c2,sports}\log_2 p_{c2,sports}$$
$$= 0.10121 + 0.23260 + 0.47215 + 0.49084 + 0.42171 + 0.11592$$
$$= 1.83443$$

Purity(C2) = 0.51947

$p_{c3,entertainment}$ = 326/949 = 0.34351
$p_{c3,financial}$ = 465/949 = 0.49000
$p_{c3,foreign}$ = 8/949 = 0.00843
$p_{c3,metro}$ = 105/949 = 0.11064
$p_{c3,national}$ = 16/949 = 0.01686
$p_{c3,sports}$ = 29/949 = 0.03056

Entropy(C3) = $- p_{c3,entertainment} \log_2 p_{c3,entertainment} - p_{c3,financial} \log_2 p_{c3,financial} - p_{c3,foreign} \log_2 p_{c3,foreign}$
$- p_{c3,metro} \log_2 p_{c3,metro} - p_{c3,national} \log_2 p_{c3,national} - p_{c3,sports} \log_2 p_{c3,sports}$
= 0.52955 + 0.50429 + 0.05808 + 0.35140 + 0.09931 + 0.15378
= 1.69641

Purity(C3) = 0.49000

Total entropy = 693/3204 * 0.02073 + 1562/3204 * 1.83443 + 949/3204 * 1.69641 = 1.40

Total purity = 693/3204 * 0.97547 + 1562/3204 * 0.51947+ 949/3204 * 0.49000 = 0.61

Per un esempio di calcolo di Precision, Recall e F1 si possono vedere le slides.

Per un esempio di rand index vedere le slides.