Politecnico di Torino

Pre-Lab 3 & 4

Business Intelligence per Big Data

Applicare algoritmi di clustering per segmentare utenti

Clustering	**K-means**	**Agglomerative Clustering & Nominal to Numerical**
Principal Component Analysis, PCA	**Linear Discrimination Analysis, LDA**	**DBSCAN**

Eleonora Poeta, Adriano de Cesare, Meryem Ennadi

Politecnico di Torino

Pre-Lab 3

Business Intelligence per Big Data

Applicare algoritmi di clustering per segmentare utenti

Clustering | **K-means** | **Agglomerative Clustering & Nominal to Numerical**

Eleonora Poeta, Adriano de Cesare

Obiettivo 1 (ripasso Lab 3)

Clustering Introduzione

Comprendere i motivi per cui si fa clustering

Clustering – Definizione e Applicazioni

Cos'è il clustering?

Il clustering è una tecnica di **apprendimento non supervisionato**: raggruppa i dati in cluster senza usare etichette predefinite.

Obiettivo:

- **Massimizzare** la **similarità intra-cluster** (*elementi simili nello stesso gruppo*)
- **Massimizzare** la **dissimilarità inter-cluster** (*gruppi diversi tra loro*)

Differenza da classificazione:

- Classificazione: le classi sono note a priori (supervisionato)
- Clustering: nessuna etichetta, si scoprono le strutture nei dati (non supervisionato)

Applicazioni reali

Segmentazione clienti

Raggruppare clienti per comportamento d'acquisto → campagne mirate.

Musica / contenuti

Raggruppare brani o utenti simili → raccomandazioni.

Bioinformatica

Clusterizzare geni con pattern di espressione simili.

Text clustering

Raggruppare documenti per argomento (es. articoli Wikipedia).

Nel Lab: dataset UsersSmall – raggruppare persone con caratteristiche anagrafiche/lavorative simili (segmentazione clienti).

Preprocessing per il Clustering – Passi Chiave

1. Read Excel

Caricare UsersSmall.xls. Analizzare semantica degli attributi.

2. Missing + Outlier

Declare Missing ('?') → Replace (moda). Filter Examples per rimuovere outlier sull'attributo Age.

3. Select Attributes

Escludere Response: il clustering è non supervisionato, non deve conoscere la label.

4. Normalize [0,1]

Range transformation su Age. Necessario per evitare che attributi con range ampio dominino la distanza.

5. Multiply

Replica l'input per alimentare in parallelo più algoritmi di clustering da confrontare.

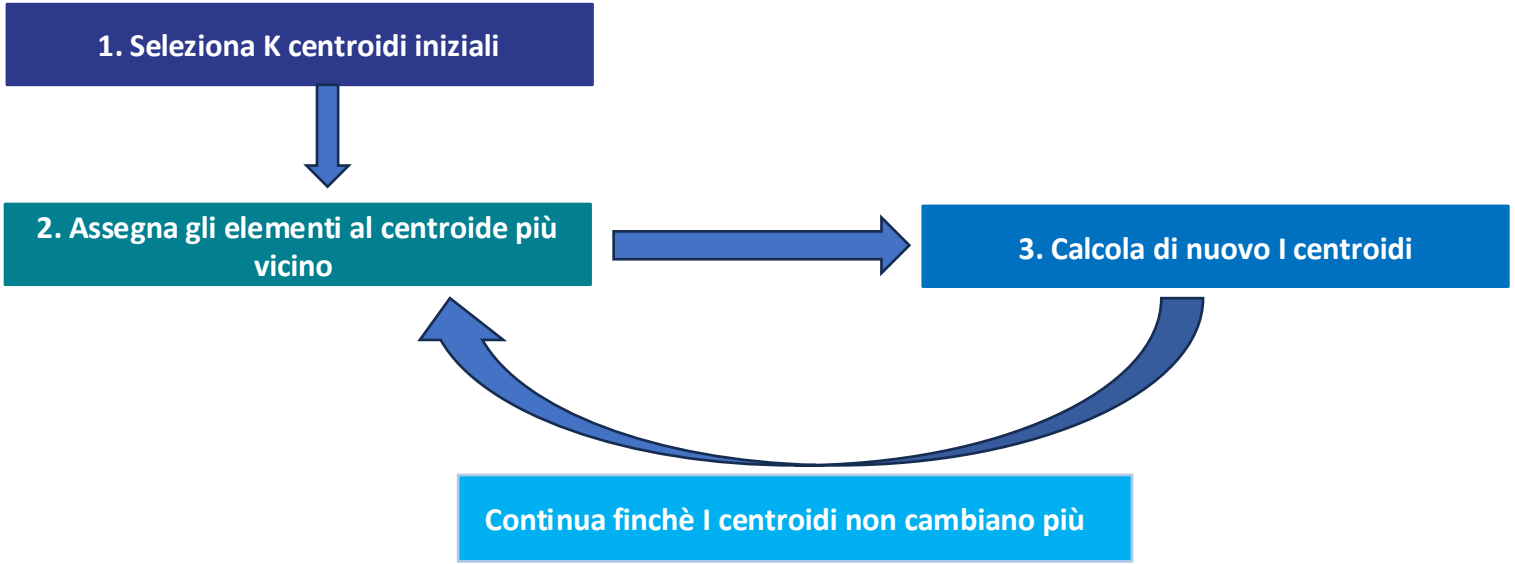
📋 Nel Lab: pipeline Read Excel → Declare Missing → Replace Missing → Select Attributes (escludi Response) → Normalize → Multiply.

Obiettivo 2 (ripasso Lab 3)

K-Means

Comprendere come funziona l'algoritmo di clustering k-means

K-means algoritmo base



Nel Lab: usiamo k-medoids

K-Means vs K-Medoids – Differenze Fondamentali

Entrambi assegnano K centri ai cluster, ma in modo diverso

K-Means

Centro: Centroide: media aritmetica dei punti del cluster (punto 'virtuale', può non esistere nel dataset).

Distanza: Minimizza la somma delle distanze euclidee al centro.

✓ **Pro:** Veloce, efficiente su grandi dataset numerici

✕ **Con:** Sensibile agli outlier. Richiede attributi numerici. Il centroide può non essere un punto reale.

K-Medoids

Centro: Medoide: il punto del dataset più rappresentativo del cluster (esiste sempre nel dataset).

Distanza: Minimizza la somma delle dissimilarità al medoide (MixedMeasures in RapidMiner).

✓ **Pro:** Robusto agli outlier. Funziona con distanze miste (numeriche + nominali).

✕ **Con:** Più lento di K-Means ($O(n^2)$ per iterazione). Meno scalabile.

Nel Lab: K-Medoids con K=2, MixedMeasures, MixedEuclideanDistance. Analizzare distribuzione Marital-Status nei cluster (Bar chart).

Obiettivo 3 (ripasso Lab3)

Agglomerative clustering

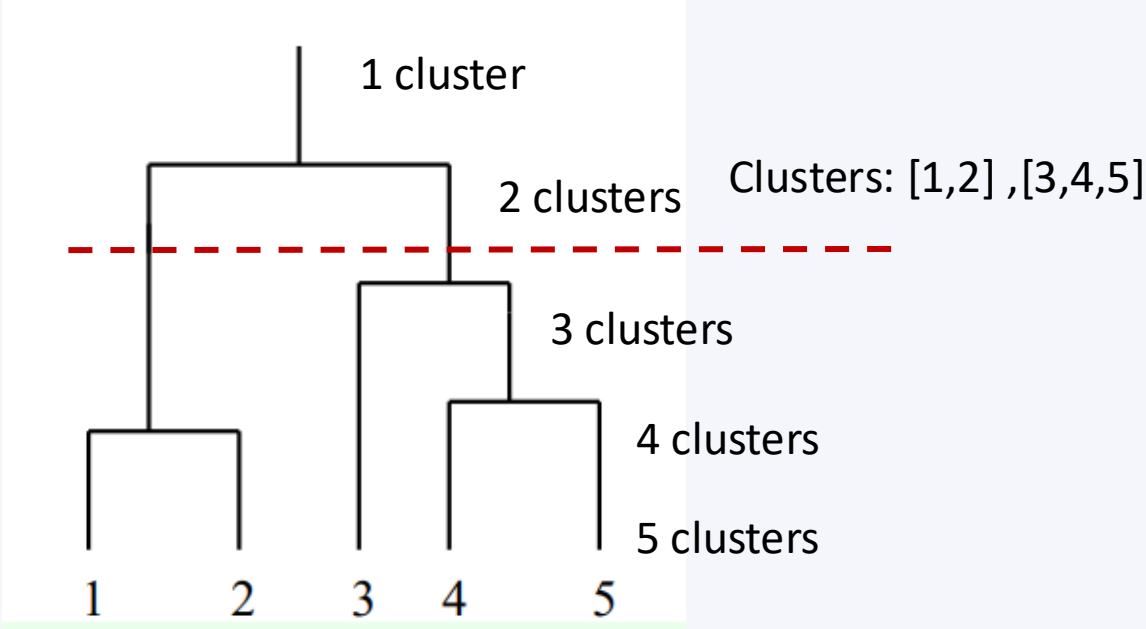
Comprendere come funziona l'agglomerative clustering

Hierarchical clustering

Produce clusters annidati visualizzabili come un albero gerarchico (**dendrogramma).**

✓ Pro:

- ***Non serve assumere 'k'*** come numero di clusters;
- Il K può essere deciso a posteriori *'tagliando'* l'albero;
- Può essere *agglomerativo* o *divisivo*.



Nel Lab: **Flatten Clustering estrae k gruppi dal dendrogramma generato dal blocco Agglomerative clustering 'tagliando' l'albero**

Agglomerative clustering algoritmo base

1. Calcola matrice di prossimità
2. Ogni punto inizialmente è un diverso cluster
3. Unisce I due clusters più "vicini"
4. Aggiorna la matrice di prossimità

Continua finchè non si ha un unico cluster

	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00

Matrice di prossimità:

Cella= distanza tra coppia di elementi o clusters

extra: Il parametro 'mode' dell'operatore agglomerative clustering definisce come calcola l'Inter-Cluster Similarity nella matrice di prossimità.

Cluester Similarity

Nell'agglomerative clustering serve stabilire quali siano I cluster simili per poi unirli

Come valutare quali sono I due cluster più vicini (simili) ?

Cluster Similarity *= Quanto sono simili tra loro 2 clusters?*

Può essere calcolata in vari modi :

- MIN o Single linkage = distanza minima tra 2 punti in 2 cluster diversi
- MAX o Complete linkage =distanza massima tra 2 punti in 2 cluster diversi
- Group Average o Average linkage = distanza media tra tutti I punti di un cluster con tutti I punti dell'altro
- Distance Between Centroid= distanza tra I due centroidi
- Ward method

Nel Lab: per stabilire I 2 cluster più vicini si usa average Linkage dell'operatore 'Agglomerative clustering' definito nel parametro 'mode'

Differenze tra algoritmi di clustering

K-means

Algoritmo partizionale basato sui centroidi.
Richiede di definire a priori il numero esatto di cluster (k)

✓ Pro: computazionalmente meno dispendioso
✗ Contro: clusters non globulari , clusters con densità o dimensioni diverse, I cluster dipendono dall' inizializzazione centroidi

Agglomerative clustering

Algoritmo gerarchico basato sull'aggregazione di cluster simili fino ad averne solo 1.
Dendrogramma rappresenta I vari clusters.
Il numero di cluster si decide a posteriori "tagliando" l'albero al livello desiderato.

✓ Pro: Non serve assumere k(n clusters) Puo modellare forme più complesse (ellissi)

✗ Contro: computazionalmente più dispendioso

DBSCAN

Algoritmo basato sulla densità.

Raggruppa punti vicini in aree ad alta densità e identifica automaticamente i dati isolati come rumore (outlier).
Non richiede di definire il numero di cluster (k) a priori

✓ Pro: Identifica cluster di forma arbitraria e gestisce il rumore.
✗ Contro: Molto sensibile alla scelta dei parametri (ϵ e *min points*).

Nel Lab: per mitigare la dipendenza del k-means dai centroidi iniziali, settando max runs a 10, il tool fa 10 tentativi e salva il clustering ottimale.

Obiettivo 4 (ripasso Lab3)

Valutazione dei Cluster

Come misurare la qualità di un clustering?

Valutazione dei Cluster – Metriche Interne

Senza etichette note, valutiamo il clustering con metriche interne (basate solo sulla struttura dei dati)

Cluster Density

Avg. distanza intra-cluster

Misura la compattezza dei cluster: media delle distanze tra i punti del cluster e il suo centroide/medoide.

Valore più basso = cluster più compatti = migliore clustering (in RapidMiner: Cluster Density Performance).

Davies-Bouldin Index

DBI = (1/K) Σ max(Ri,j)

Rapporto tra dispersione interna e separazione tra cluster. Considera sia compattezza che separazione.

Valore più basso = migliore. Idealmente DBI < 1.

Silhouette Score

s(i) = (b-a) / max(a,b)

Per ogni punto: a = distanza media intra-cluster, b = distanza media al cluster più vicino.

Valore tra -1 e 1. Più vicino a 1 = punto ben assegnato al suo cluster.

📋 Nel Lab: K-Medoids (clu) → Data To Similarity (MixedMeasures) → Performance (Cluster Density). Confrontare con diversi K.

Come Scegliere K – Il Problema del Numero di Cluster

K-Medoids richiede K come input. Non esiste un valore 'giusto' a priori. Bisogna trovarlo sperimentalmente.

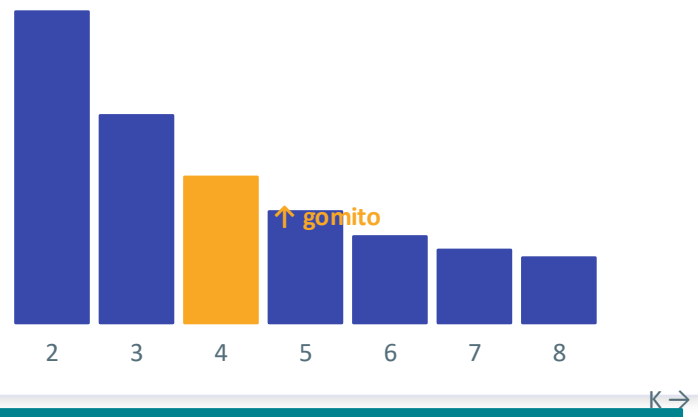
Metodo del gomito (Elbow Method)

- Calcola la metrica di performance (es. densità, inerzia) per K = 2, 3, 4, ..., n.
- Aumentando K, la qualità migliora sempre (con K=n ogni punto è il proprio cluster → densità 0).
- Si cerca il 'gomito': il punto in cui il miglioramento diventa marginale. Quel K è il migliore trade-off.

Attenzione:

- Con K=N (un cluster per punto) la densità è sempre 0 → non è un buon clustering! Serve interpretabilità.

Andamento tipico della densità



Nel Lab: Rieseguire K-Medoids con K=2,3,4,5... Osservare come cambia la Cluster Density Performance. Identificare il gomito.

Obiettivo 5 (ripasso Lab3)

Nominal to Numerical

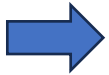
Comprendere tecniche di encoding per convertire da formato testuale a numerico

Nominal to numerical

Dummy coding (o one hot encoding)

Per ogni possibile valore di un attributo nominale , crea un **nuovo attributo** che indica la presenza o assenza di tal valore .

Race ...
White
Black
Asian
White



Race=White	Race=Black	Race=Asian
1	0	0
0	1	0
0	0	1
1	0	0

✓ Pro: non assume ordinamento dei valori
✗ Contro: esplosione del numero di attributi, aumentando memoria e tempi di calcolo

Label encoding

Converte ogni possibile valore di un attributo nominale in un **diverso intero** progressivo.

Race
White
Black
Asian
White

Race
0
1
2
0

✓ Pro: non cambia il numero di attributi
✗ Contro: assume ordinamento dei valori (in questo caso senza senso)

Nel Lab: si usa nominal to numerical operator per usare dummy coding , senza quel blocco l'operatore clustering applica label encoding

Politecnico di Torino

Pre-Lab 4

Business Intelligence per Big Data

Applicare algoritmi di clustering per segmentare utenti

Principal Component Analysis, PCA

Linear Discrimination Analysis, LDA

DBSCAN

Eleonora Poeta, Meryem Ennadi

Obiettivo 1.1 Lab4

Principal Component Analysis

Comprendere l'uso della tecnica PCA tramite Singular Value Decompositioon

Principal Component Analysis via Singular Value Decomposition (SVD)

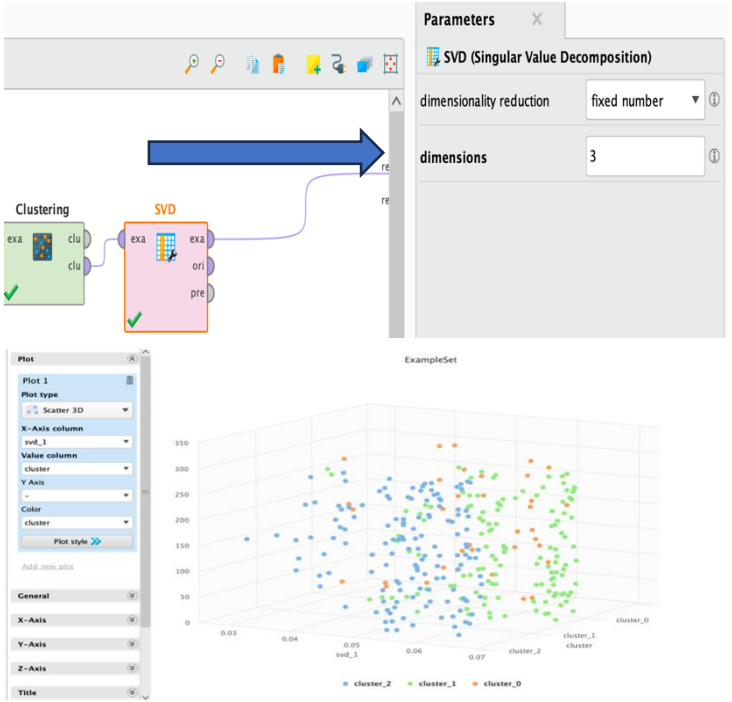
Principal Component Analysis

Il **PCA** è la più nota tecnica algebrica per effettuare la decomposizione matriciale Singular Value Decomposition (SVD) per la **riduzione della dimensionalità dei dati**.

Obiettivo: trasformare linearmente i dati originali in un set di componenti non correlate (valori singolari).

Utilizzo nel Lab: Ridurre il dataset a **K=3** dimensioni **per permettere la visualizzazione 3D dei cluster** ottenuti precedentemente

✓ Pro: stabilità e riduzione del rumore.
✗ Contro: perdita di interpretabilità.



Nel Lab: si usa K=3

Obiettivo 1.2 Lab4

Linear Discriminant Analysis (LDA)

Comprendere l'uso della tecnica LDA e l'uso dell'operatore "Apply Model"

Linear Discriminant Analysis + Apply Model

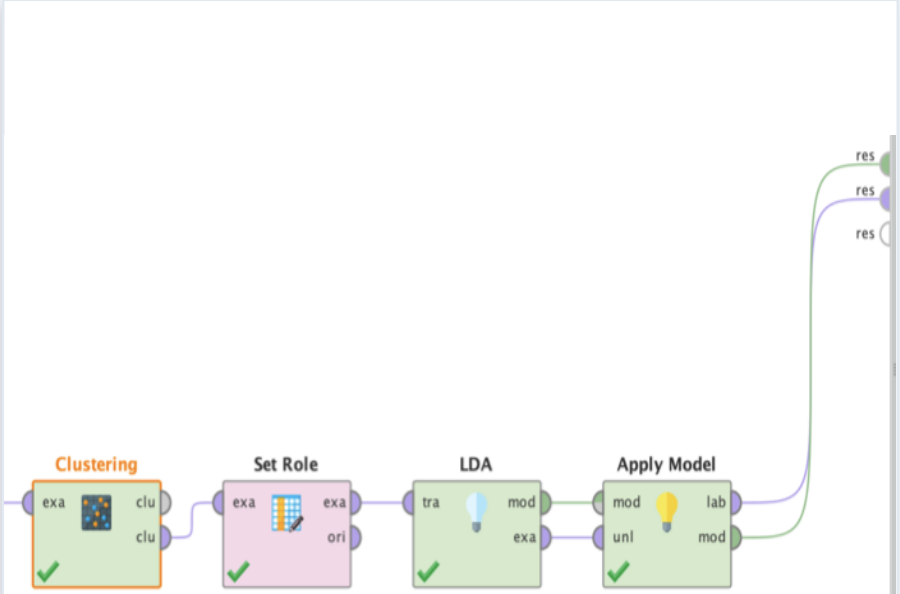
Che cosa è LDA?

A differenza del PCA (che è ***unsupervised***), la LDA è una tecnica **supervised** che serve ad **analizzare la qualità del clustering** trovando **le variabili che meglio separano le classi** (i ***cluster***).

Obiettivo: massimizzare il rapporto tra la varianza tra le classi (**between-class**) e la varianza all'interno delle classi (**within-class**)

Utilizzo nel Lab: verificare se i cluster trovati sono effettivamente distinguibili in base agli attributi del dataset

✓ Pro: Considera esplicitamente la struttura delle classi.
✗ Contro: assume che I dati seguano una distribuzione normale.



Nel Lab: si usa l'operatore LDA seguito dall'operatore "Apply Model" per **proiettare** o **trasformare** i dati correnti.

Obiettivo 2 Lab4

DBSCAN

Comprendere l'uso dell'algoritmo DBSCAN

DBSCAN

Che cosa è il DBSCAN?

A differenza del K-Means, il **DBSCAN non richiede di specificare il numero di cluster (K) a priori**. Raggruppa i punti basandosi sulla **densità spaziale**, identificando aree ad alta concentrazione separate da zone vuote.

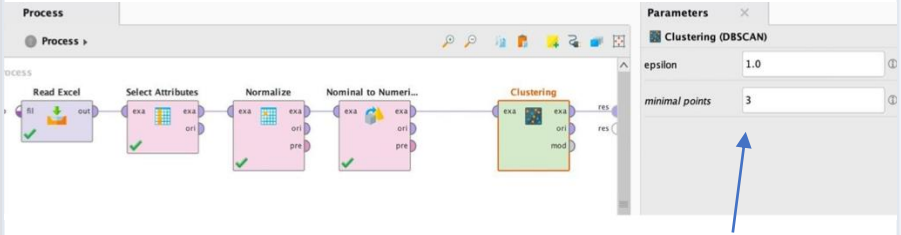
Obiettivo: identificare cluster di forma arbitraria basati sulla densità locale, isolando automaticamente il rumore (outlier)

Utilizzo nel lab: applicare **DBSCAN** (minimal points = 3)per identificare raggruppamenti densi e analizzare i punti di rumore nello spazio 3D generato dalla **SVD**

✓ Pro: Identifica cluster di forma arbitraria e rileva automaticamente gli **outlier** (rumore).
✗ Contro: Estremamente sensibile alla scelta dei parametri (ϵ e *min points*) e alle diverse densità dei dati.

Parametri Fondamentali in RapidMiner:
Per configurare **l'operatore DBSCAN**, devi impostare due parametri critici:

- **Epsilon (ϵ):** Il raggio di vicinanza entro cui cercare altri punti. Definisce la "portata" della scansione.
- **Min Points:** Il numero minimo di punti necessari entro il raggio ϵ per formare un "core point" (nel tuo lab è impostato a **3**)



Nel Lab: si usa l'operatore "Clustering(DBSCAN)" settando min points = 3

Riepilogo del Lab

① Clustering Introduzione

② K-Means

③ Agglomerative clustering & Valutazione dei Cluster & Nominal to Numerical

④ Principal Component Analysis

⑤ Linear Discriminant Analysis

⑥ DBSCAN

Buon laboratorio!