



Sapere utile

IFOA

Istituto Formazione Operatori Aziendali

BIG DATA e Analisi dei Dati

Mauro Bellone,
Robotics and AI researcher

bellonemauro@gmail.com
www.maurobellone.com

Obiettivo

- ✓ Introduzione agli algoritmi di data mining
- ✓ Elementi di classificazione
- ✓ tutorial python – Regole associative, classificazione k-NN

Data mining



Data mining

il processo di rivelazione di patterns di valore da grandi data sets

anche noto come
knowledge discovery in data (KDD)



Data mining

il processo di rivelazione di patterns di valore da grandi data sets

anche noto come
knowledge discovery in data (KDD)

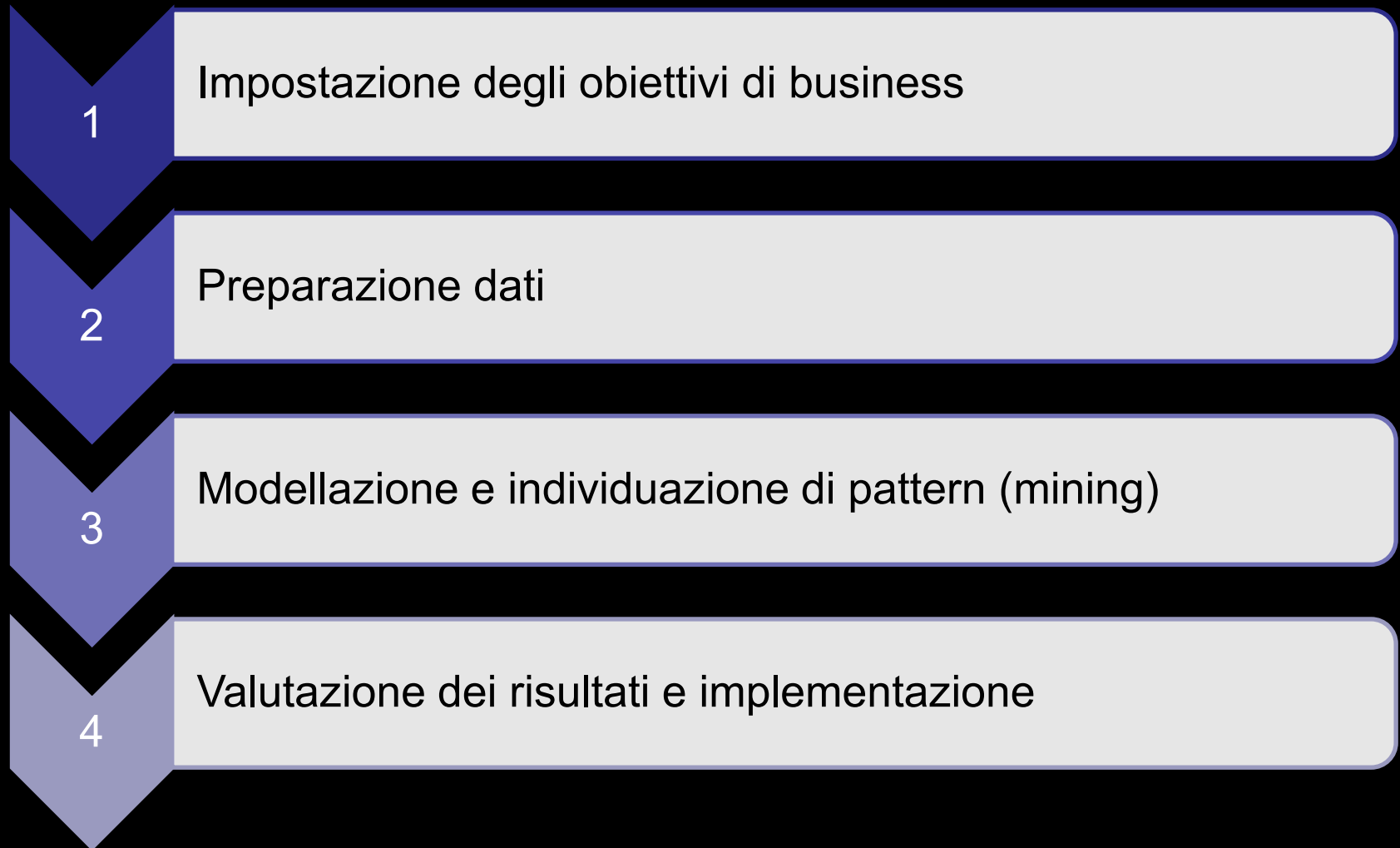
Il termine corretto potrebbe essere “mining in data” e non “data mining”
“estrazione -di informazione- dai dati” e non “estrazione di dati”

Data mining: Motivazioni

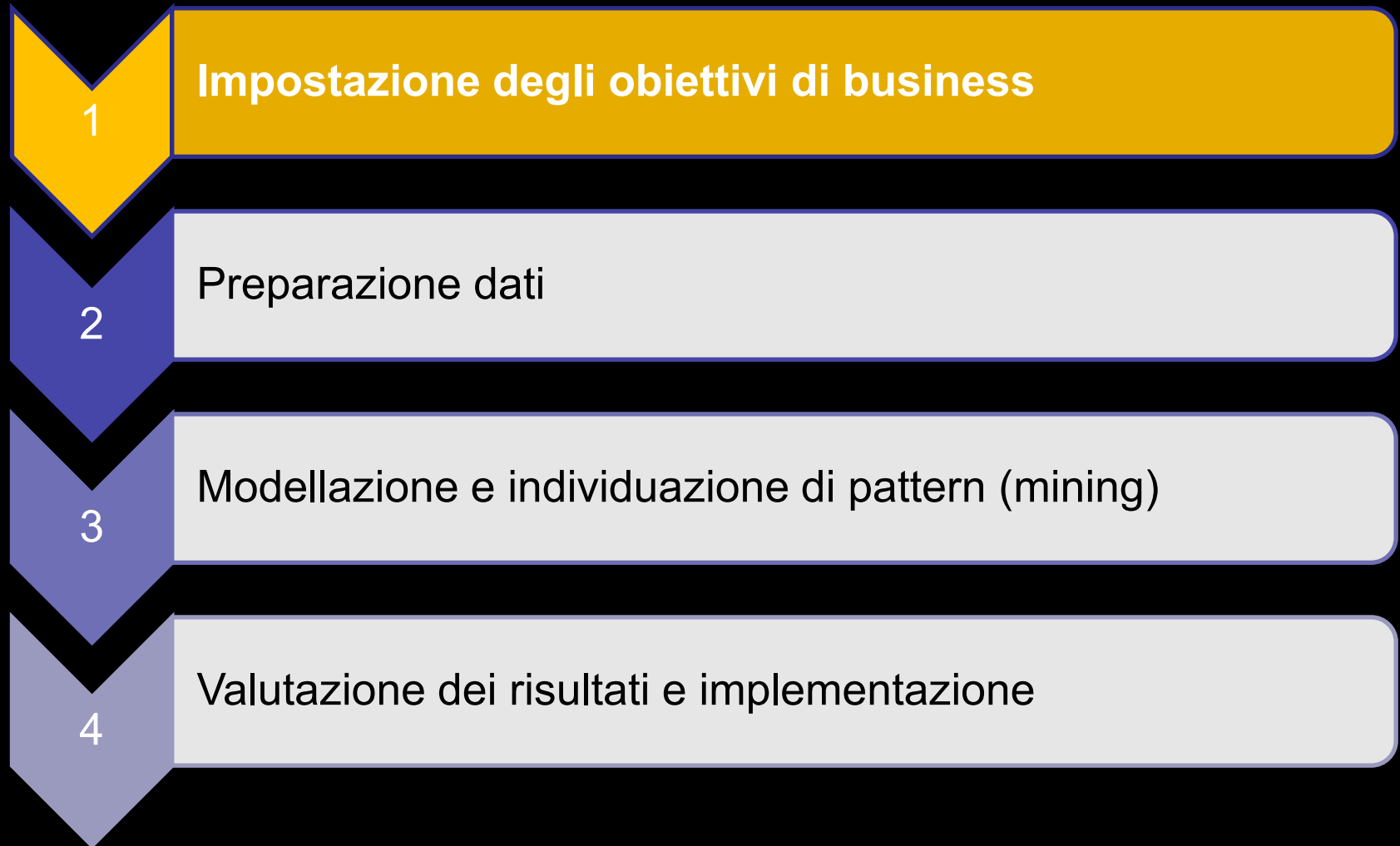
1. Descrizione di grandi dataset
2. Predizione del comportamento di un sistema (predictive modeling)



Processi di data mining



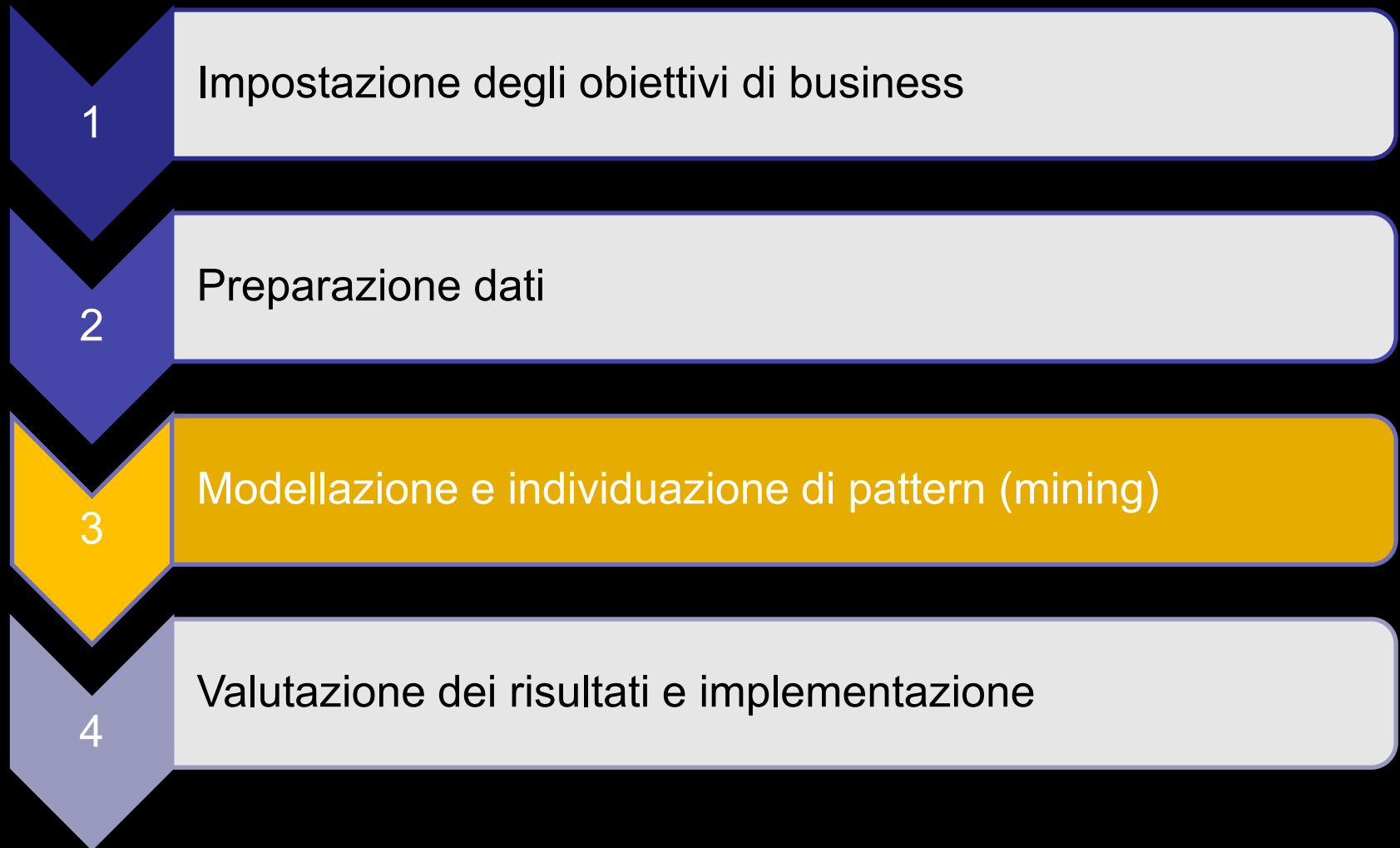
Processi di data mining



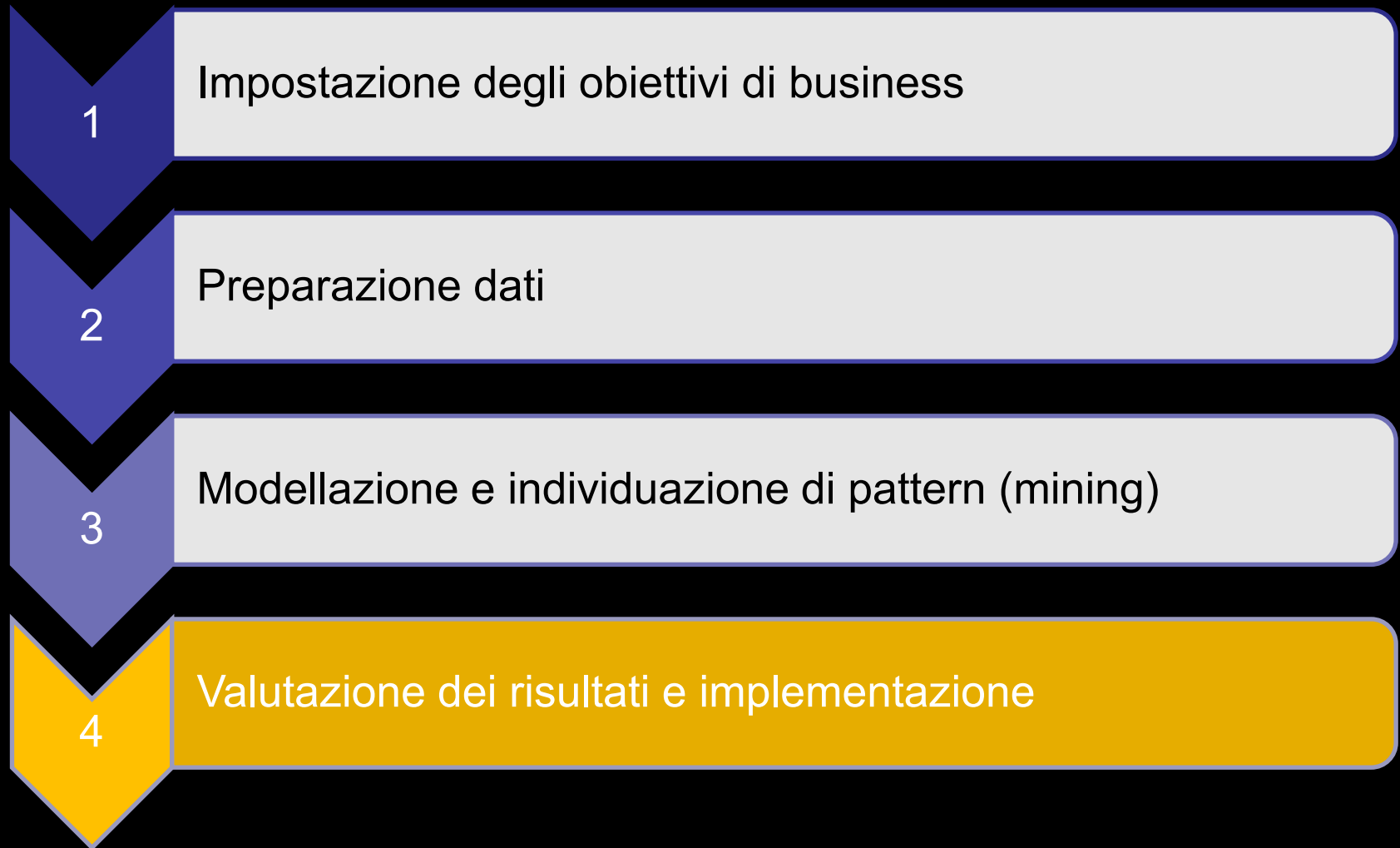
Processi di data mining



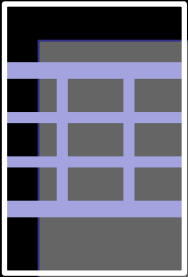
Processi di data mining



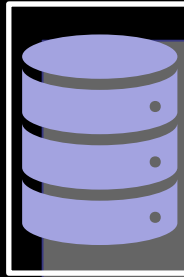
Processi di data mining



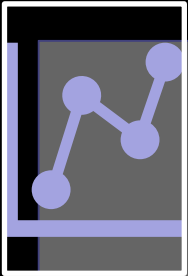
Task di data mining



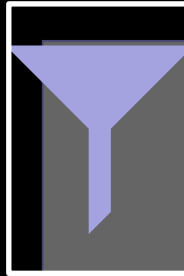
Clusterizzazione



Classificazione

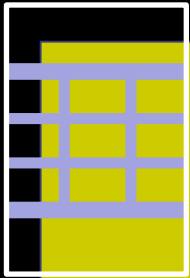


Regressione



Aggregazione

Task di data mining



Clusterizzazione



Regressione

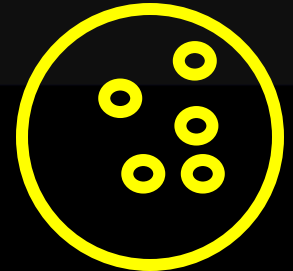
Individuare gruppi di dati
somiglianti all'interno di un dataset



Classificazione

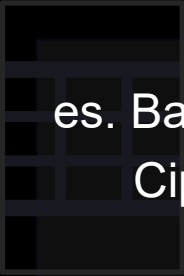


Aggregazione

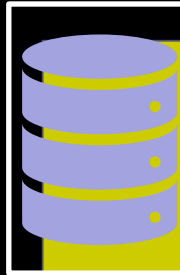


Task di data mining

La classificazione consiste nel task di assegnare ad un dato una classe all'interno di un insieme



es. Banana → Frutta
Cipolla → Verdura



Classificazione



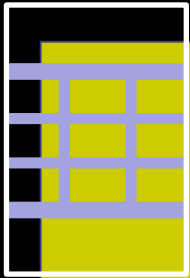
Regressione



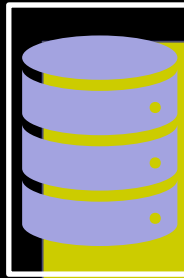
Aggregazione

Task di data mining

La classificazione consiste nel task di assegnare ad un dato una classe all'interno di un insieme



Clusterizzazione



Classificazione



Regressione



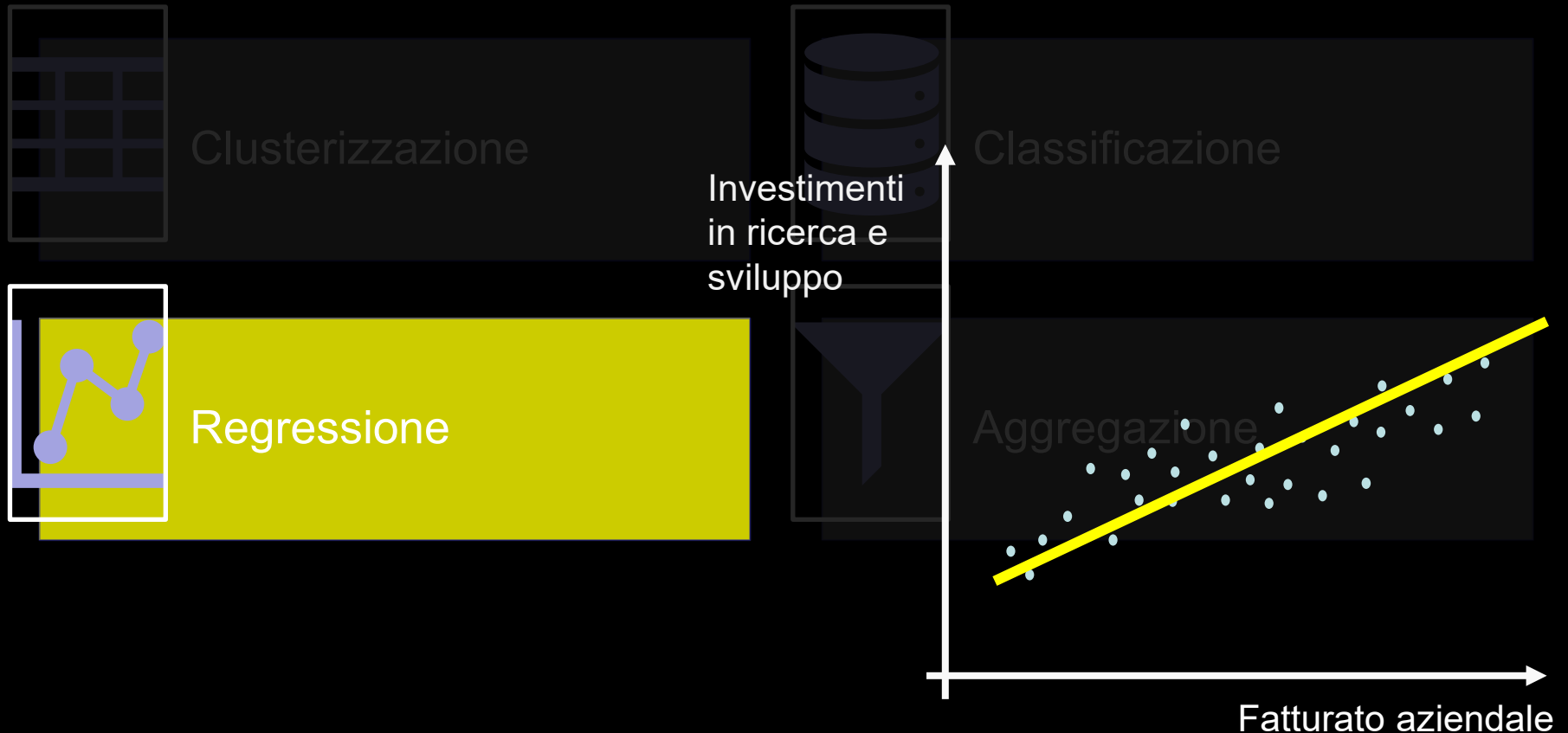
Aggregazione

La differenza sottesa tra clustering e classificazione:

- ❖ obiettivo del clustering → **gruppi risultanti** dalla divisione dei dati
- ❖ obiettivo della classificazione → il **potere discriminativo**

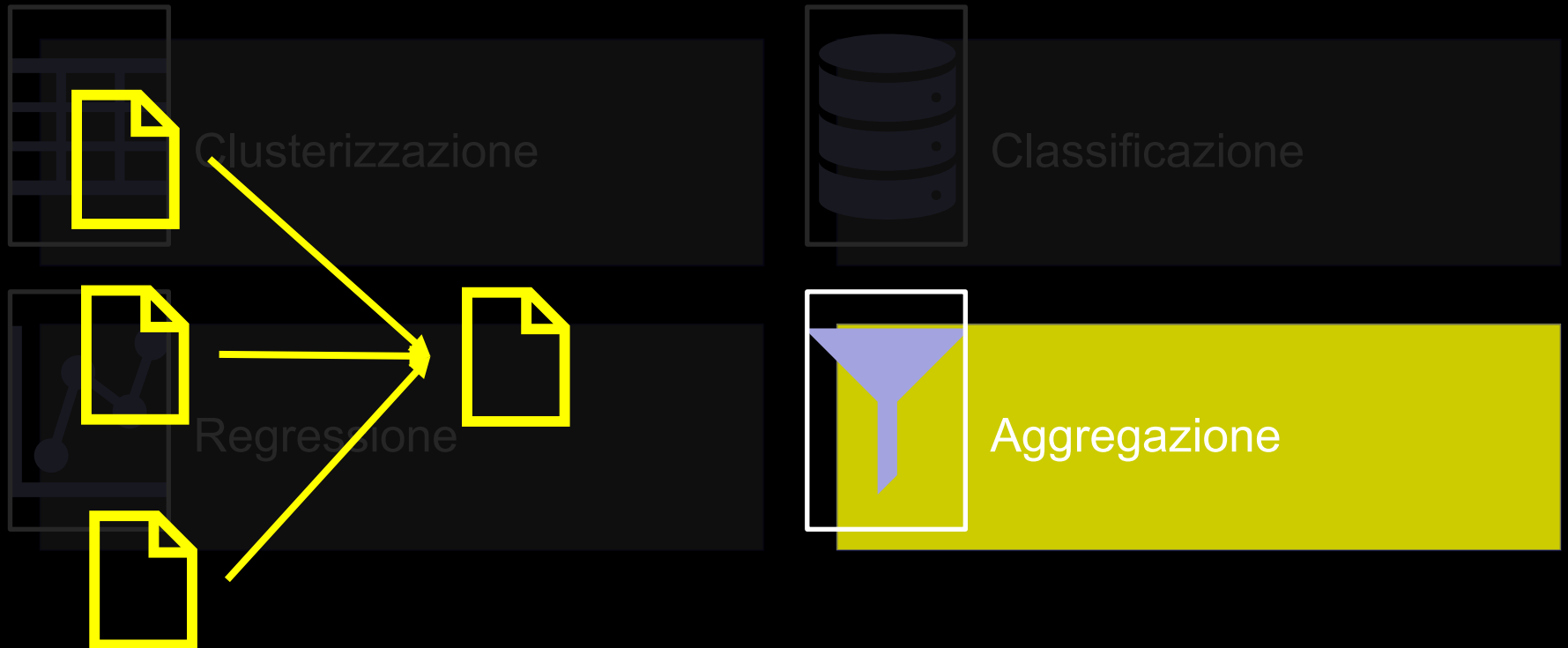
Task di data mining

La regressione consiste nella generazione di un modello statistico usato per predire relazioni tra una variabile dipendente e una variabile indipendente



Task di data mining

L'aggregazione consiste in un processo di ricerca, raccolta e presentazione di dati in un formato riassunto, tipicamente utile per la visualizzazione



Tecniche di data mining



Alberi decisionali

Regole di associazione

k-nearest neighbor (KNN)

Reti neurali

Tecniche di data mining



Alberi decisionali

Regole di associazione

k-nearest neighbor (KNN)

Reti neurali

Tecniche di data mining



Alberi decisionali

Regole di associazione

k-nearest neighbor (KNN)

Reti neurali

Tecniche di data mining



Alberi decisionali

Regole di associazione

k-nearest neighbor (KNN)

Reti neurali

Tecniche di data mining



Alberi decisionali

Regole di associazione

k-nearest neighbor (KNN)

Reti neurali

Tecniche di data mining



Alberi decisionali

Regole di associazione

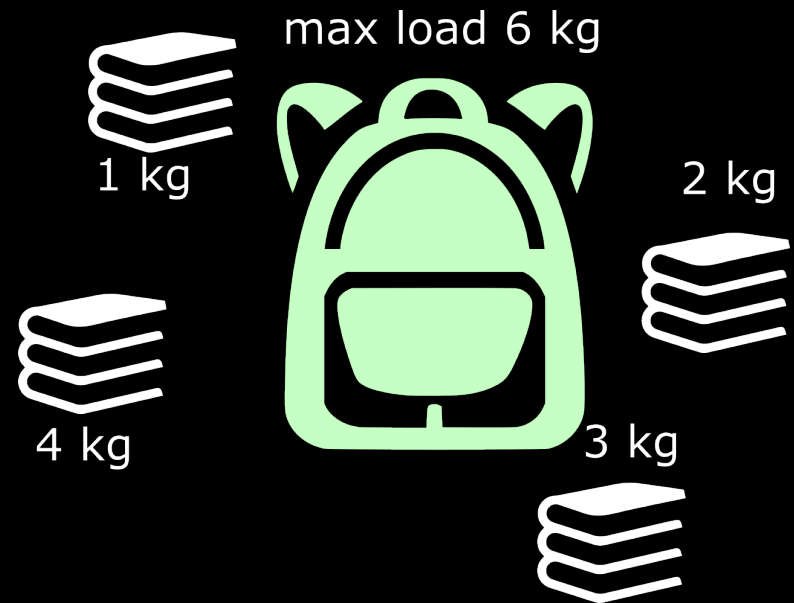
k-nearest neighbor (KNN)

Reti neurali

Il problema dello zaino

Dato uno zaino con una massica capacità di carico pari a x kg, e dati n oggetti aventi pesi diversi.

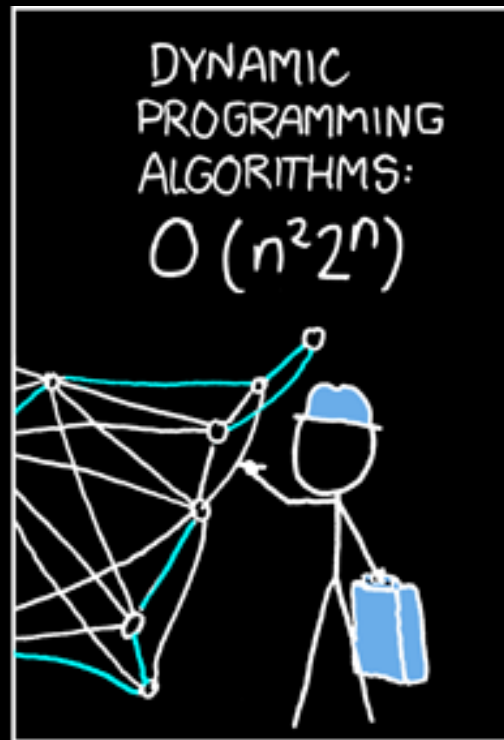
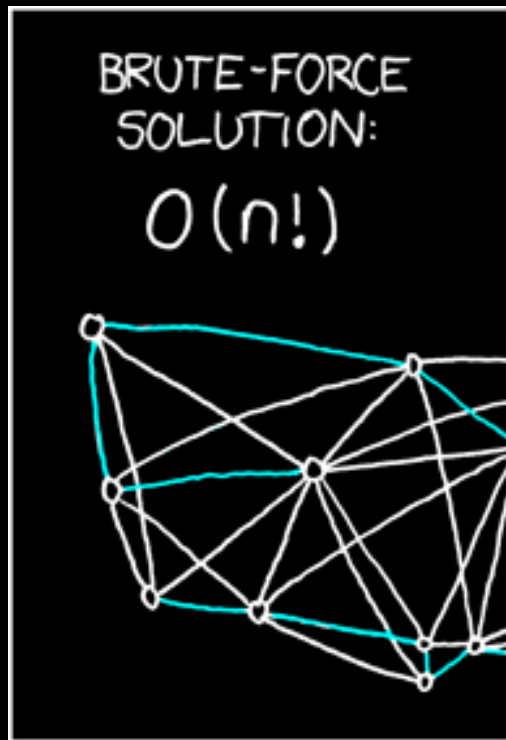
Qual è la combinazione di oggetti dentro lo zaino che massimizza il carico di peso senza eccedere il limite assegnato?



Travelling Salesman Problem (TSP)

Dato un insieme compost da città
e distanze per ogni coppia

Qual è la **minima** strata per visitare tutte le
destinazioni e tornare alla città iniziale?



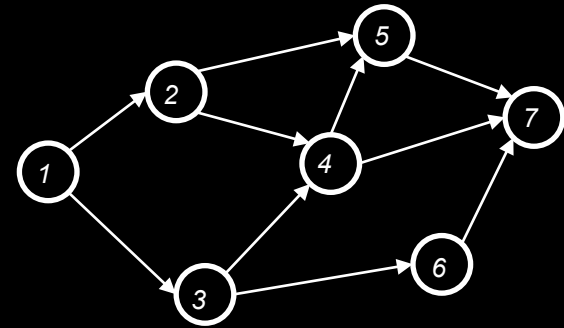
Schemi a grafo

un grafo è una coppia ordinate $G = (V, E)$

Con:

un insieme di n vertici $V = \{v_i, \forall i = 1, \dots, n\}$

un insieme di m rami $E = \{v_j, \forall j = 1, \dots, m\}$



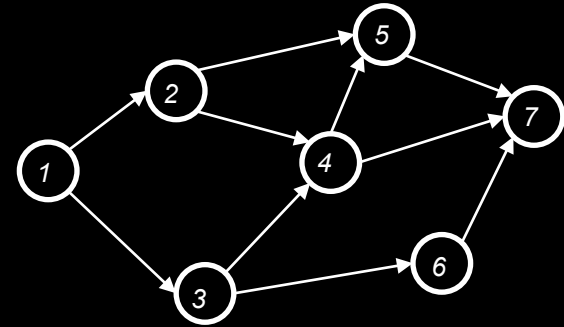
Schemi a grafo

un grafo è una coppia ordinate $G = (V, E)$

Con:

un insieme di n vertici $V = \{v_i, \forall i = 1, \dots, n\}$

un insieme di m rami $E = \{v_j, \forall j = 1, \dots, m\}$



Numerosi sistemi reali adottano questa rappresentazione:

- Fisici;
- Biologici;
- Economici;
- Sociali;
- Computer science;
- Robotica;

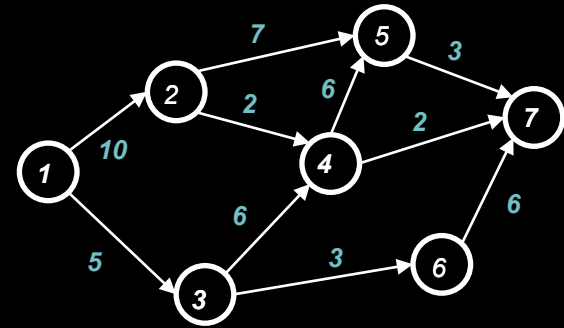
Schemi a grafo

un grafo è una coppia ordinate $G = (V, E)$

Con:

un insieme di n vertici $V = \{v_i, \forall i = 1, \dots, n\}$

un insieme di m rami $E = \{v_j, \forall j = 1, \dots, m\}$



Nei grafi tipicamente se assegna un costo ad ogni ramo

$$C = \{c_1, c_2, \dots, c_m\}$$

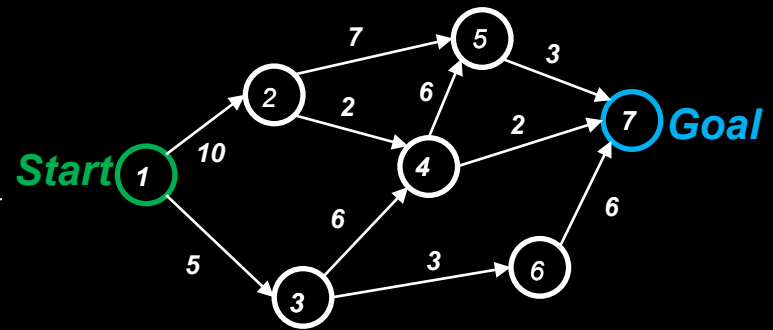
Schemi a grafo

un grafo è una coppia ordinate $G = (V, E)$

Con:

un insieme di n vertici $V = \{v_i, \forall i = 1, \dots, n\}$

un insieme di m rami $E = \{v_j, \forall j = 1, \dots, m\}$



Nei grafi tipicamente se assegna un costo ad ogni ramo

$$C = \{c_1, c_2, \dots, c_m\}$$

Problema di ottimizzazione: Cercare il percorso dal punto iniziale **start**, al punto finale **goal** cge minimizza il costo

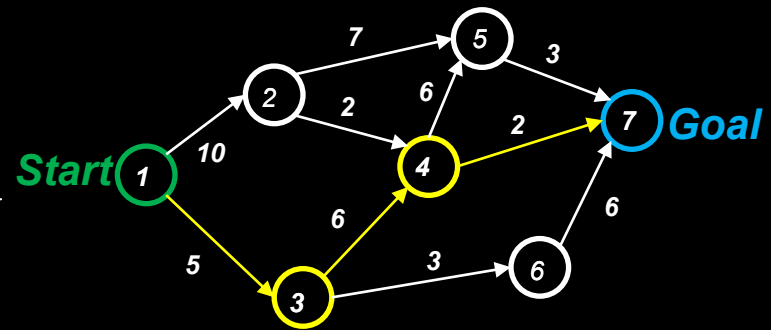
Schemi a grafo

un grafo è una coppia ordinate $G = (V, E)$

Con:

un insieme di n vertici $V = \{v_i, \forall i = 1, \dots, n\}$

un insieme di m rami $E = \{v_j, \forall j = 1, \dots, m\}$



Nei grafi tipicamente si assegna un costo ad ogni ramo

$$C = \{c_1, c_2, \dots, c_m\}$$

Problema di ottimizzazione: Cercare il percorso dal punto iniziale **start**, al punto finale **goal** che minimizza il costo

$$J = \sum_{k=1}^l c_k(v_i, e_j)$$

Funzione di costo

Problemi di ottimizzazione

$$\begin{array}{ll} \min_x & c(x) \\ \text{soggetto a} & g_i(x) \leq 0 \text{ con } i = 1, 2, \dots, n \\ & h_j(x) = 0 \text{ con } j = 1, 2, \dots, m \end{array}$$

Dove

$$c: \mathbb{R}^n \rightarrow \mathbb{R}$$

si dice funzione obiettivo (cost function)

$$g_i(x) \leq 0 \text{ e } h_j(x) = 0$$

sono vincoli

Alberi decisionali

Gli alberi decisionali sono una tecnica usata per classificare o effettuare regressione basate su un insieme di decisioni.

Alberi decisionali

Gli alberi decisionali sono una tecnica usata per classificare o effettuare regressione basate su un insieme di decisioni.

Essenzialmente è un diagramma ad albero usato per determinare un corso di azioni. Ogni ramo dell'albero rappresenta una possibile decisione, la sua occorrenza e la possibile reazione

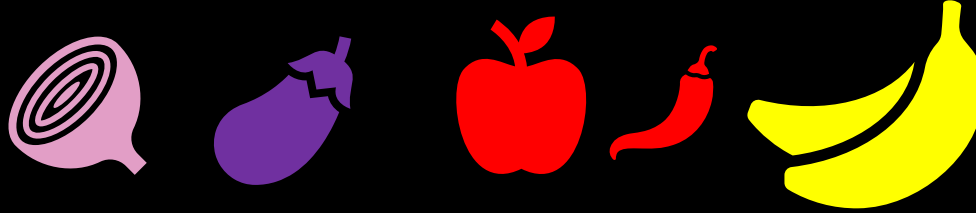
Alberi decisionali

Gli alberi decisionali sono una tecnica usata per classificare o effettuare regressione basate su un insieme di decisioni.

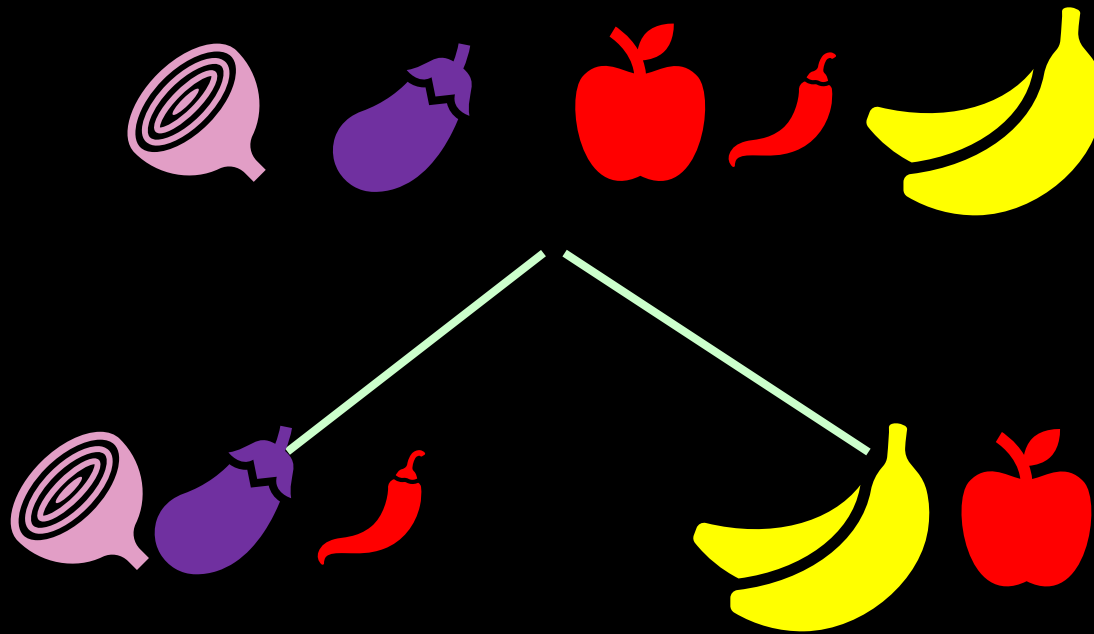
Essenzialmente è un diagramma ad albero usato per determinare un corso di azioni. Ogni ramo dell'albero rappresenta una possibile decisione, la sua occorrenza e la possibile reazione

Come suggerito dal nome è possibile usare una visualizzazione ad albero che rappresenta il risultato potenziale di ogni decisione

Alberi decisionali - esempio

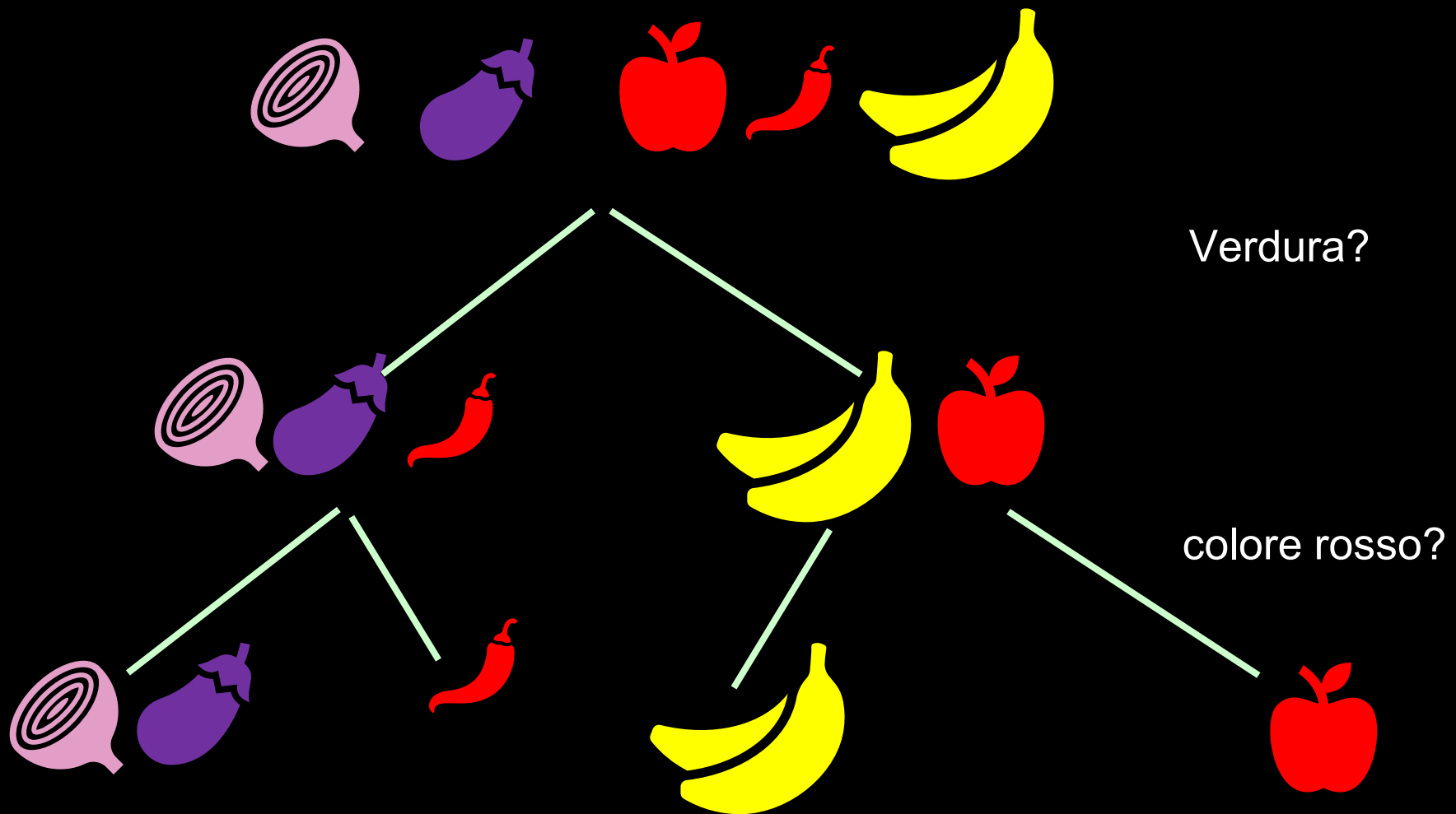


Alberi decisionali - esempio

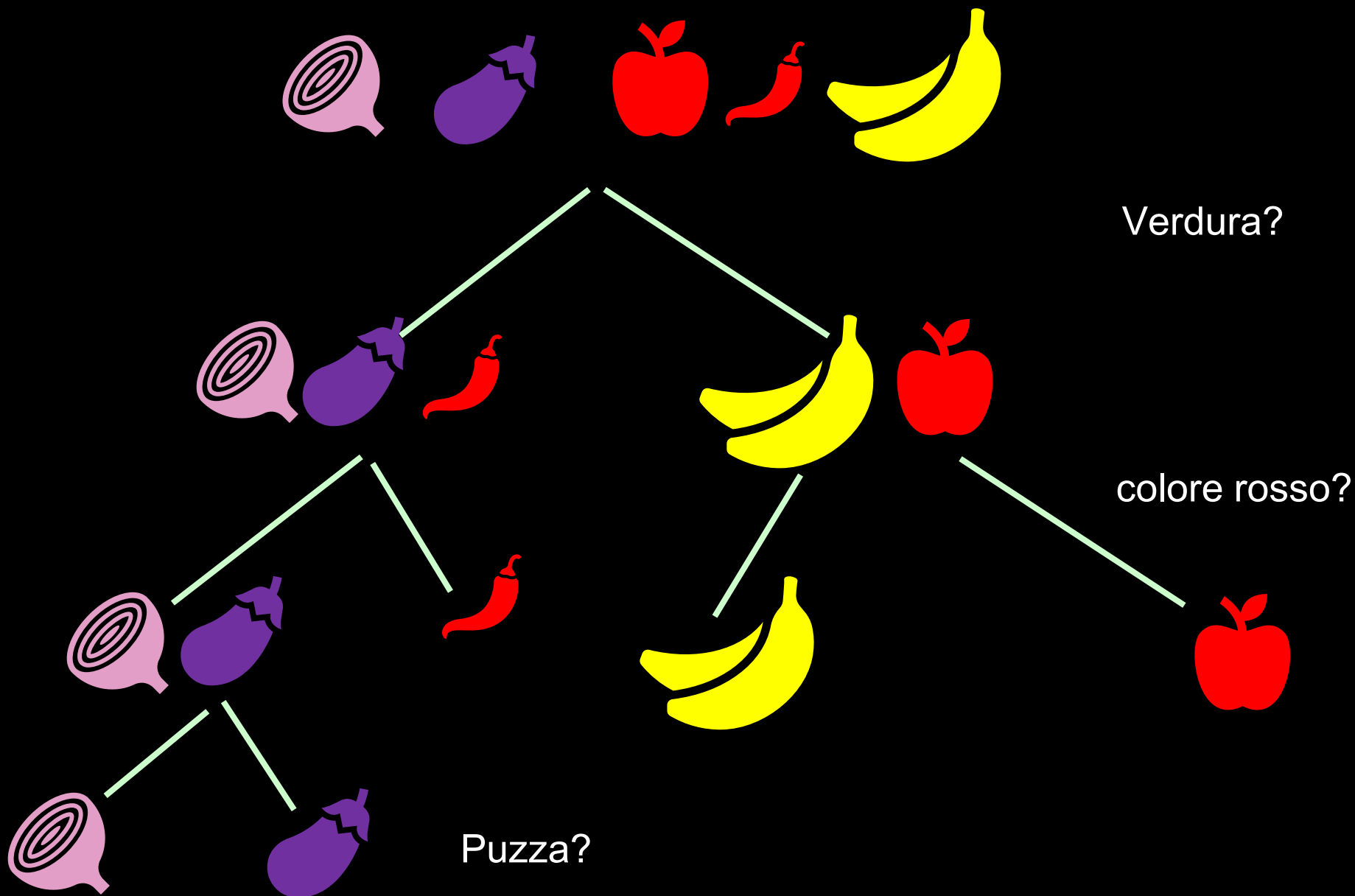


Verdura?

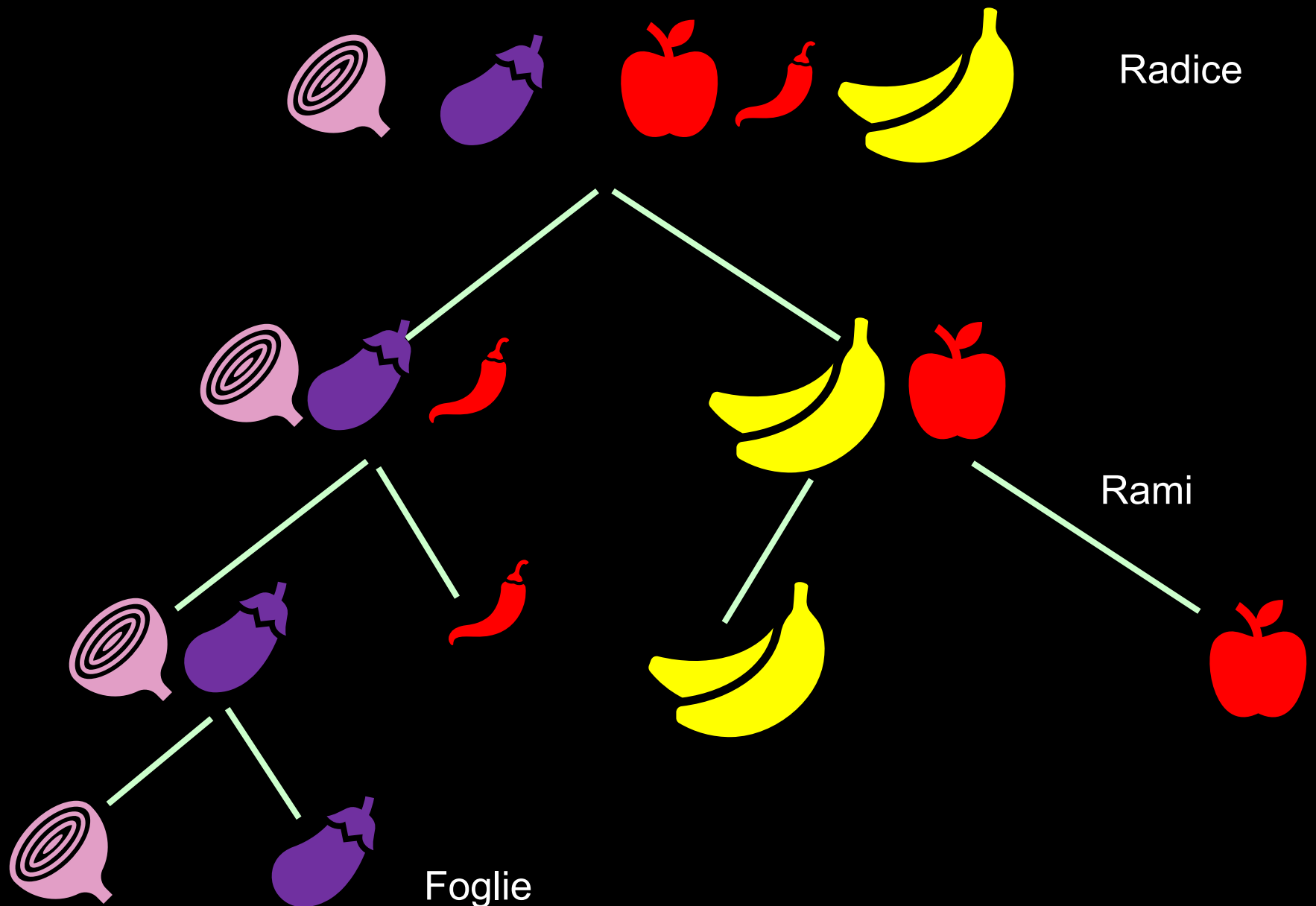
Alberi decisionali - esempio



Alberi decisionali - esempio



Alberi decisionali - esempio



Alberi decisionali

Abbiamo una tabella di features, per ogni feature abbiamo un valore e l'albero funziona filtrando di volta in volta tutte le features

| Oggetto | Numero | Tipo | Colore | Puzza |
|-------------|--------|---------|--------|-------|
| Mela | 5 | Frutta | Rosso | NO |
| Banana | 12 | Frutta | Giallo | NO |
| Cipolla | 3 | Verdura | Rosa | SI |
| Melanzana | 4 | Verdura | Viola | NO |
| Peperoncino | 2 | Verdura | Rosso | NO |

Alberi decisionali

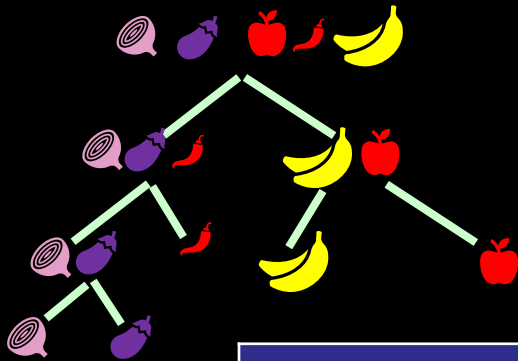
Abbiamo una tabella di features, per ogni feature abbiamo un valore e l'albero funziona filtrando di volta in volta tutte le features

$$C = \sum_{i=0}^n P(o_i) \log P(o_i) \quad \rightarrow \quad \text{Entropia}$$

| Oggetto | Numero | Tipo | Colore | Puzza |
|-------------|--------|---------|--------|-------|
| Mela | 5 | Frutta | Rosso | NO |
| Banana | 12 | Frutta | Giallo | NO |
| Cipolla | 3 | Verdura | Rosa | SI |
| Melanzana | 4 | Verdura | Viola | NO |
| Peperoncino | 2 | Verdura | Rosso | NO |

Alberi decisionali

Abbiamo una tabella di features, per ogni feature abbiamo un valore e l'albero funziona filtrando di volta in volta tutte le features



$$C = \sum_{i=0}^n P(o_i) \log P(o_i)$$



Entropia

| Oggetto | Numero | Tipo | Colore | Puzza |
|-------------|--------|---------|--------|-------|
| Mela | 5 | Frutta | Rosso | NO |
| Banana | 12 | Frutta | Giallo | NO |
| Cipolla | 3 | Verdura | Rosa | SI |
| Melanzana | 4 | Verdura | Viola | NO |
| Peperoncino | 2 | Verdura | Rosso | NO |

Alberi decisionali

Vantaggi:

- ✓ Semplici da capire e visualizzare
- ✓ La preparazione dei dati è tipicamente semplice (no normalizzazione, no scaling etc.)
- ✓ Possiamo gestire facilmente dati numerici e categorie
- ✓ Parametri non lineari non influiscono sulle performance

Alberi decisionali

Svantaggi:

- ✓ Forte overfitting
- ✓ Alta varianza (il modello può diventare instabile)
- ✓ Bassa capacità di astrazione con nuovi dati

Regole di associazione

Regole di associazione

Nel campo del data mining la tecnica delle regole di associazione si riferisce alla ricerca di regole e relazioni tra variabili in un grande dataset.

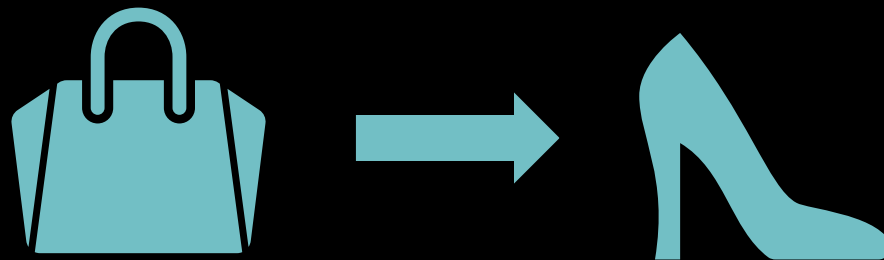
Tipicamente è usato per identificare delle forti relazioni in funzione di qualche metrica di interesse.

Regole di associazione

Nel campo del data mining la tecnica delle regole di associazione si riferisce alla ricerca di regole e relazioni tra variabili in un grande dataset.

Tipicamente è usato per identificare delle forti relazioni in funzione di qualche metrica di interesse.

es. acquisto prodotto A → acquisto prodotto B

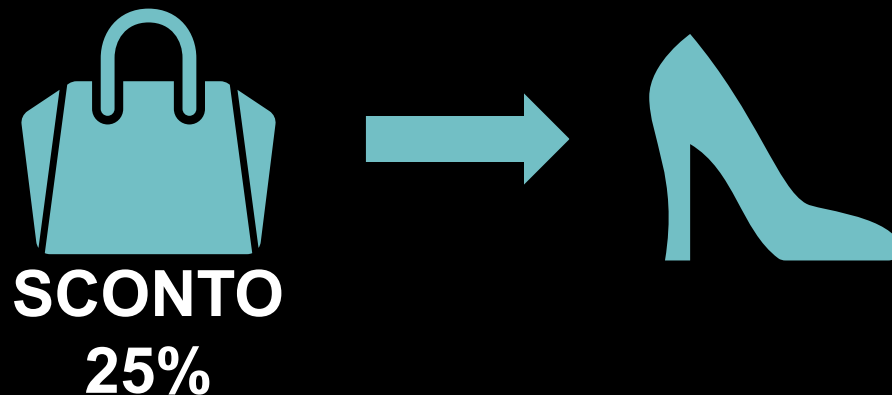


Regole di associazione

Nel campo del data mining la tecnica delle regole di associazione si riferisce alla ricerca di regole e relazioni tra variabili in un grande dataset.

Tipicamente è usato per identificare delle forti relazioni in funzione di qualche metrica di interesse.

es. acquisto prodotto A → acquisto prodotto B



Regole di associazione

$$A \Rightarrow B$$

if then

A è detto antecedente o precedente

B è detto conseguente

Misure di associazione

$A \Rightarrow B$
if then

- Supporto – è una indicazione di quante volte il nostro oggetto compare nel dataset

$$\begin{aligned} SUP(A) &= \frac{freq(A)}{N} & SUP(A) &= P(A) \\ SUP(A \cup B) &= \frac{freq(A, B)}{N} & SUP(A \cup B) &= P(A \cup B) \end{aligned}$$

Misure di associazione

$A \Rightarrow B$
if then

- Supporto
- Confidenza – ci dice quanto spesso una regola è vera

$$CONF(A, B) = \frac{freq(A, B)}{freq(A)} \quad CONF(A, B) = \frac{P(A \cup B)}{P(A)} = \boxed{?}$$

Probabilità condizionata

Sia B un evento tale che $P(B) > 0$

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Nel caso in cui gli eventi A e B siano indipendenti:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A) * P(B)}{P(B)} = P(A)$$

Misure di associazione

$A \Rightarrow B$
if then

- Supporto
- Confidenza – ci dice quanto spesso una regola è vera

$$CONF(A, B) = \frac{freq(A, B)}{freq(A)} \quad CONF(A, B) = \frac{P(A \cup B)}{P(A)} = P(B|A)$$

Misure di associazione

$$\begin{array}{ccc} A & \Rightarrow & B \\ \text{if} & & \text{then} \end{array}$$

- Supporto
- Confidenza
- Lift – è il rapporto tra la probabilità dell'unione e la probabilità attesa se A e B fossero indipendenti

$$LIFT(A \Rightarrow B) = \frac{SUP(A \cup B)}{SUP(A) * SUP(B)} = \frac{P(A \cup B)}{P(A) * P(B)}$$

Misure di associazione

| | | |
|----------|---|--|
| Lift = 1 | → | perfetta antecedenza $A \Rightarrow B$ |
| Lift > 1 | → | A può implicare B |
| Lift < 1 | → | A può sostituire B |

- Supporto
- Confidenza
- Lift – è il rapporto tra la probabilità dell'unione e la probabilità attesa se A e B fossero indipendenti

$$LIFT(A \Rightarrow B) = \frac{SUP(A \cup B)}{SUP(A) * SUP(B)} = \frac{P(A \cup B)}{P(A) * P(B)}$$

Misure di associazione

$A \Rightarrow B$
if then

- Supporto
- Confidenza
- Lift $Conv(A \Rightarrow B) = \frac{1 - SUP(B)}{1 - CONF(A, B)} = \frac{1 - P(B)}{1 - P(B|A)}$
- Conviction – è una misura di quanto la regola è incoretta (come se A e B fossero sostitutivi)

Misure di associazione

Conv = 1.2 → la regola sarebbe errata il 20% in più delle volte rispetto ad una associazione X,Y completamente casuale

- Supporto

- Confidenza

- Lift
$$Conv(A \Rightarrow B) = \frac{1 - SUP(B)}{1 - CONF(A, B)} = \frac{1 - P(B)}{1 - P(B|A)}$$

- Conviction – è una misura di quanto la regola è incoretta (come se A e B fossero sostitutivi)

Mining delle regole di associazione

Supponiamo di avere una serie di prodotti A, B, C, D, E e di aver registrato una serie di transazioni in un database relazionale

| Tr ID | Prod | Prod | Prod |
|-------|------|------|------|
| T1 | A | B | C |
| T2 | A | C | D |
| T3 | B | C | D |
| T4 | A | D | E |
| T5 | B | C | E |

Mining delle regole di associazione

Supponiamo di avere una serie di prodotti A, B, C, D, E e di aver registrato una serie di transazioni in un database relazionale

| Tr ID | Prod | Prod | Prod |
|-------|------|------|------|
| T1 | A | B | C |
| T2 | A | C | D |
| T3 | B | C | D |
| T4 | A | D | E |
| T5 | B | C | E |

Regole di associazione

1. $A \rightarrow D$
2. $C \rightarrow A$
3. $A \rightarrow C$
4. $B \& C \rightarrow A$

| Reg | Sup | Conf | Lift | Conv |
|-----|-----|------|------|------|
| 1 | 2/5 | 2/3 | 10/9 | 6/5 |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |

Mining delle regole di associazione

Supponiamo di avere una serie di prodotti A, B, C, D, E e di aver registrato una serie di transazioni in un database relazionale

Il grande problema di questa procedura è che per numero di oggetti e transazioni molto grande il numero di possibili regole di associazione diventa computazionalmente intrattabile

| Tr ID | Prod | Prod | Prod |
|-------|------|------|------|
| T1 | A | B | C |
| T2 | A | C | D |
| T3 | B | C | D |
| T4 | A | D | E |
| T5 | B | C | E |

Regole di associazione

1. $A \rightarrow D$
2. $C \rightarrow A$
3. $A \rightarrow C$
4. $B \& C \rightarrow A$

| Reg | Sup | Conf | Lift | Conv |
|-----|-----|------|------|------|
| 1 | 2/5 | 2/3 | 10/9 | 6/5 |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |

Algoritmi di generazione delle regole di associazione

- Apriori
- Max-miner
- FPgrowth (Frequency Pattern growth)
- FPmax

Algoritmo di generazione regole di associazione apriori

Def: un insieme di **oggetti frequenti** è un insieme di oggetti che hanno un valore di supporto maggiore di una certa soglia



Algoritmo di generazione regole di associazione apriori

Def: un insieme di **oggetti frequenti** è un insieme di oggetti che hanno un valore di supporto maggiore di una certa soglia



Significa semplicemente che nella lista della spesa ci sono oggetti acquistati con una frequenza più alta e che quindi possono avere una associazione

Per calcolarli ho bisogno di calcolare il valore di supporto dell'intero dataset.

Scelgo un valore di supporto soglia e scarto tutti quelli che non sono rilevanti

Algoritmo di generazione regole di associazione apriori

Def: un insieme di **oggetti frequenti** è un insieme di oggetti che hanno un valore di supporto maggiore di una certa soglia



- ✓ L'algoritmo fa uso degli insiemi di oggetti frequenti per generare regole di associazione.

Algoritmo di generazione regole di associazione apriori

Def: un insieme di **oggetti frequenti** è un insieme di oggetti che hanno un valore di supporto maggiore di una certa soglia



- ✓ L'algoritmo fa uso degli insiemi di oggetti frequenti per generare regole di associazione.
- ✓ Si basa sull'idea che un sottoinsieme di un insieme di oggetti frequenti deve essere un insieme di oggetti frequenti

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1} | 3 |
| {2} | 3 |
| {3} | 4 |
| {4} | 1 |
| {5} | 4 |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1} | 3 |
| {2} | 3 |
| {3} | 4 |
| {4} | 1 |
| {5} | 4 |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di supporto

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1} | 3 |
| {2} | 3 |
| {3} | 4 |
| {5} | 4 |

| OggFreq | Sup |
|---------|-----|
| {1,2} | 1 |
| {1,3} | 3 |
| {1,5} | 2 |
| {2,3} | 2 |
| {2,5} | 3 |
| {3,5} | 3 |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di support

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1} | 3 |
| {2} | 3 |
| {3} | 4 |
| {5} | 4 |

| OggFreq | Sup |
|---------|-----|
| {1,2} | 1 |
| {1,3} | 3 |
| {1,5} | 2 |
| {2,3} | 2 |
| {2,5} | 3 |
| {3,5} | 3 |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di support

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1,3} | 3 |
| {1,5} | 2 |
| {2,3} | 2 |
| {2,5} | 3 |
| {3,5} | 3 |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di support

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1,3} | 3 |
| {1,5} | 2 |
| {2,3} | 2 |
| {2,5} | 3 |
| {3,5} | 3 |

| OggFreq | Sup |
|---------|-----|
| {1,2,3} | |
| {1,2,5} | |
| {1,3,5} | |
| {2,3,5} | |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di support

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Pruning: calcolo i supporti dei sottoinsiemi ed elimino i branch che hanno supporto minore della soglia

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1,2,3} | |
| {1,2,5} | |
| {1,3,5} | |
| {2,3,5} | |

| OggFreq | sottoinsiemi | In F2? |
|---------|--------------------|--------|
| {1,2,3} | {1,2},{1,3}, {2,3} | NO |
| {1,2,5} | {1,2},{1,5}, {2,5} | NO |
| {1,3,5} | {1,3},{1,5}, {3,5} | SI |
| {2,3,5} | {2,3},{2,5}, {3,5} | SI |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di support

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Pruning: calcolo i supporti dei sottoinsiemi ed elimino i branch che hanno supporto minore della soglia

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1,2,3} | |
| {1,2,5} | |
| {1,3,5} | |
| {2,3,5} | |

{1,2} era già stato scartato

| OggFreq | sottoinsiemi | In F2? |
|---------|--------------------|--------|
| {1,2,3} | {1,2},{1,3}, {2,3} | NO |
| {1,2,5} | {1,2},{1,5}, {2,5} | NO |
| {1,3,5} | {1,3},{1,5}, {3,5} | SI |
| {2,3,5} | {2,3},{2,5}, {3,5} | SI |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di support

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Pruning: calcolo i supporti dei sottoinsiemi ed elimino i branch che hanno supporto minore della soglia

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1,3,5} | 2 |
| {2,3,5} | 2 |

A questo punto dovrei solo iterare pruning e calcolo del supporto finchè la tabella ha valori

| OggFreq | sottoinsiemi | In F2? |
|---------|--------------------|--------|
| {1,3,5} | {1,3},{1,5}, {3,5} | SI |
| {2,3,5} | {2,3},{2,5}, {3,5} | SI |

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di support

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Pruning: calcolo i supporti dei sottoinsiemi ed elimino i branch che hanno supporto minore della soglia

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| OggFreq | Sup |
|---------|-----|
| {1,3,5} | 2 |
| {2,3,5} | 2 |

| OggFreq | Sup |
|-----------|-----|
| {1,2,3,5} | 1 |

Non continuiamo a iterare e torniamo alla tabella precedente

Algoritmo di generazione regole di associazione a-priori

Passo 1 – Crea il valore di supporto di tutti gli elementi

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Passo 2 - prendiamo le coppie di valori e calcoliamo il loro valore di support

Tutti gli elementi con supporto minore della soglia sono eliminati (es. <2)

Pruning: calcolo i supporti dei sottoinsiemi ed elimino i branch supporto minore della soglia

| OggFreq | Sup |
|---------|-----|
| {1,3,5} | 2 |
| {2,3,5} | 2 |

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}$, $\{1,5\}$, $\{3,5\}$, $\{1\}$, $\{3\}$, $\{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}$, $\{2,5\}$, $\{3,5\}$, $\{2\}$, $\{3\}$, $\{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

Algoritmo di generazione regole di associazione a-priori

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}$, $\{1,5\}$, $\{3,5\}$, $\{1\}$, $\{3\}$, $\{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}$, $\{2,5\}$, $\{3,5\}$, $\{2\}$, $\{3\}$, $\{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

| OggFreq | Sup |
|-------------|-----|
| $\{1,3,5\}$ | 2 |
| $\{2,3,5\}$ | 2 |

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

Regola 1: $\{1,3\} \rightarrow \{1,3,5\} - \{1,3\}$ significa che $1 \ \& \ 3 \rightarrow 5$

Algoritmo di generazione regole di associazione a-priori

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}$, $\{1,5\}$, $\{3,5\}$, $\{1\}$, $\{3\}$, $\{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}$, $\{2,5\}$, $\{3,5\}$, $\{2\}$, $\{3\}$, $\{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

| OggFreq | Sup |
|-------------|-----|
| $\{1,3,5\}$ | 2 |
| $\{2,3,5\}$ | 2 |

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

Regola 1: $\{1,3\} \rightarrow \{1,3,5\} - \{1,3\}$ significa che $1 \ \& \ 3 \rightarrow 5$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1,3) = 2/3 = 66\%$

Algoritmo di generazione regole di associazione a-priori

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}, \{1,5\}, \{3,5\}, \{1\}, \{3\}, \{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}, \{2,5\}, \{3,5\}, \{2\}, \{3\}, \{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

| OggFreq | Sup |
|-------------|-----|
| $\{1,3,5\}$ | 2 |
| $\{2,3,5\}$ | 2 |

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

Regola 1: $\{1,3\} \rightarrow \{1,3,5\} - \{1,3\}$ significa che $1 \ \& \ 3 \rightarrow 5$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1,3) = 2/3 = 66\%$

Regola 2: $\{1,5\} \rightarrow \{1,3,5\} - \{1,5\}$ significa che $1 \ \& \ 5 \rightarrow 3$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1,5) = 2/2 = 100\%$

Algoritmo di generazione regole di associazione a-priori

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}$, $\{1,5\}$, $\{3,5\}$, $\{1\}$, $\{3\}$, $\{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}$, $\{2,5\}$, $\{3,5\}$, $\{2\}$, $\{3\}$, $\{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

| OggFreq | Sup |
|-------------|-----|
| $\{1,3,5\}$ | 2 |
| $\{2,3,5\}$ | 2 |

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

Regola 1: $\{1,3\} \rightarrow \{1,3,5\} - \{1,3\}$ significa che $1 \ \& \ 3 \rightarrow 5$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1,3) = 2/3 = 66\%$

Regola 2: $\{1,5\} \rightarrow \{1,3,5\} - \{1,5\}$ significa che $1 \ \& \ 5 \rightarrow 3$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1,5) = 2/2 = 100\%$

Regola 3: $\{3,5\} \rightarrow \{1,3,5\} - \{3,5\}$ significa che $3 \ \& \ 5 \rightarrow 1$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(3,5) = 2/3 = 66\%$

Algoritmo di generazione regole di associazione a-priori

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}$, $\{1,5\}$, $\{3,5\}$, $\{1\}$, $\{3\}$, $\{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}$, $\{2,5\}$, $\{3,5\}$, $\{2\}$, $\{3\}$, $\{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

| OggFreq | Sup |
|-------------|-----|
| $\{1,3,5\}$ | 2 |
| $\{2,3,5\}$ | 2 |

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

Regola 1: $\{1,3\} \rightarrow \{1,3,5\} - \{1,3\}$ significa che $1 \ \& \ 3 \rightarrow 5$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1,3) = 2/3 = 66\%$

Regola 2: $\{1,5\} \rightarrow \{1,3,5\} - \{1,5\}$ significa che $1 \ \& \ 5 \rightarrow 3$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1,5) = 2/2 = 100\%$

Regola 3: $\{3,5\} \rightarrow \{1,3,5\} - \{3,5\}$ significa che $3 \ \& \ 5 \rightarrow 1$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(3,5) = 2/3 = 66\%$

Algoritmo di generazione regole di associazione a-priori

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}$, $\{1,5\}$, $\{3,5\}$, $\{1\}$, $\{3\}$, $\{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}$, $\{2,5\}$, $\{3,5\}$, $\{2\}$, $\{3\}$, $\{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

| OggFreq | Sup |
|-------------|-----|
| $\{1,3,5\}$ | 2 |
| $\{2,3,5\}$ | 2 |

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

Regola 4: $\{1\} \rightarrow \{1,3,5\} - \{1\}$ significa che $1 \rightarrow 3 \text{ \& } 5$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1) = 2/3 = 66\%$

Regola 5: $\{3\} \rightarrow \{1,3,5\} - \{3\}$ significa che $3 \rightarrow 1 \text{ \& } 5$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(3) = 2/4 = 50\%$

Regola 6: $\{5\} \rightarrow \{1,3,5\} - \{5\}$ significa che $5 \rightarrow 1 \text{ \& } 3$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(5) = 2/4 = 50\%$

Algoritmo di generazione regole di associazione a-priori

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}, \{1,5\}, \{3,5\}, \{1\}, \{3\}, \{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}, \{2,5\}, \{3,5\}, \{2\}, \{3\}, \{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

| OggFreq | Sup |
|-------------|-----|
| $\{1,3,5\}$ | 2 |
| $\{2,3,5\}$ | 2 |

Se min_conf = 60% → regola 5 e 6 saranno rigettate

Regola 4: $\{1\} \rightarrow \{1,3,5\} - \{1\}$ significa che $1 \rightarrow 3 \& 5$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(1) = 2/3 = 66\%$

Regola 5: $\{3\} \rightarrow \{1,3,5\} - \{3\}$ significa che $3 \rightarrow 1 \& 5$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(3) = 2/4 = 50\%$

Regola 6: $\{5\} \rightarrow \{1,3,5\} - \{5\}$ significa che $5 \rightarrow 1 \& 3$

valore di confidenza = $\text{supporto}(1,3,5)/\text{supporto}(5) = 2/4 = 50\%$

Algoritmo di generazione regole di associazione a-priori

Per $I = \{1,3,5\}$ i sottoinsiemi sono $\{1,3\}, \{1,5\}, \{3,5\}, \{1\}, \{3\}, \{5\}$

Per $I = \{2,3,5\}$ i sottoinsiemi sono $\{2,3\}, \{2,5\}, \{3,5\}, \{2\}, \{3\}, \{5\}$

Per ogni sottoinsieme di I , le regole in uscita sono:

$S \rightarrow (I-S)$ quindi S raccomanda $(I-S)$ se $\frac{\text{supporto}(I)}{\text{supporto}(S)} \geq \text{min_conf}$

| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

Regola 1: $\{2,3\} \rightarrow \{2,3,5\} - \{2,3\}$ significa che $2 \ \& \ 3 \rightarrow 5$

valore di confidenza = $\text{supporto}(2,3,5)/\text{supporto}(2,3) = 2/2 = 100\%$

Regola 2: $\{2,5\} \rightarrow \{2,3,5\} - \{2,5\}$ significa che $2 \ \& \ 5 \rightarrow 3$

valore di confidenza = $\text{supporto}(2,3,5)/\text{supporto}(2,5) = 2/3 = 66\%$

Regola 3: $\{3,5\} \rightarrow \{2,3,5\} - \{3,5\}$ significa che $3 \ \& \ 5 \rightarrow 2$

valore di confidenza = $\text{supporto}(2,3,5)/\text{supporto}(3,5) = 2/3 = 66\%$

Regola 4: $\{2\} \rightarrow \{2,3,5\} - \{2\}$ significa che $2 \rightarrow 3 \ \& \ 5$

valore di confidenza = $\text{supporto}(2,3,5)/\text{supporto}(2) = 2/3 = 66\%$

Regola 5: $\{3\} \rightarrow \{2,3,5\} - \{3\}$ significa che $3 \rightarrow 2 \ \& \ 5$

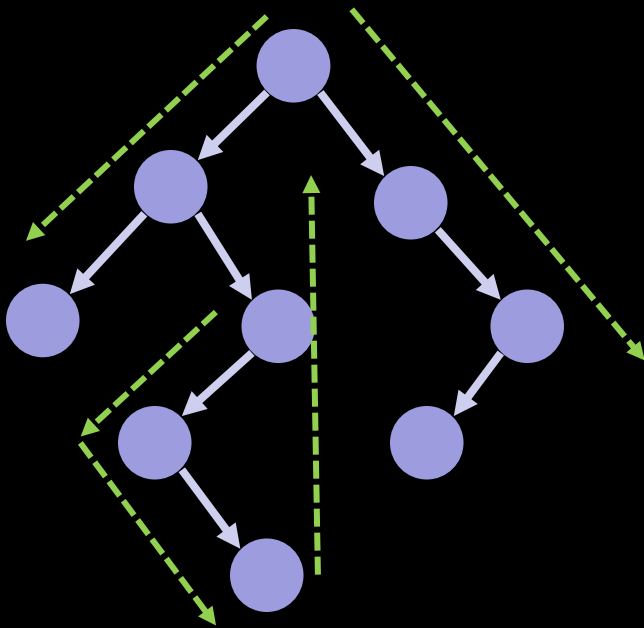
valore di confidenza = $\text{supporto}(2,3,5)/\text{supporto}(3) = 2/4 = 50\%$

Regola 6: $\{5\} \rightarrow \{2,3,5\} - \{5\}$ significa che $5 \rightarrow 2 \ \& \ 3$

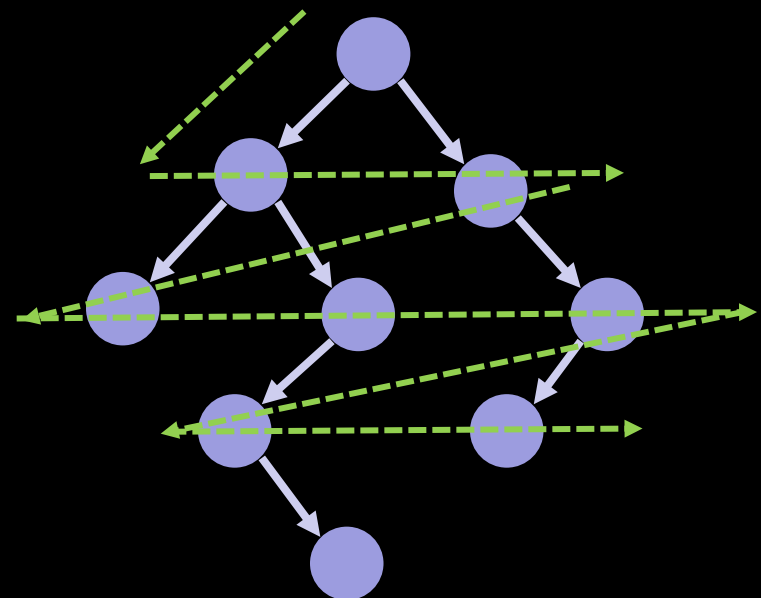
valore di confidenza = $\text{supporto}(2,3,5)/\text{supporto}(5) = 2/3 = 66\%$

Algoritmi di generazione delle regole di associazione

- Apriori
- Max-miner
- FPgrowth (Frequency Pattern growth)
- FPmax
- FPclose



Depth first search

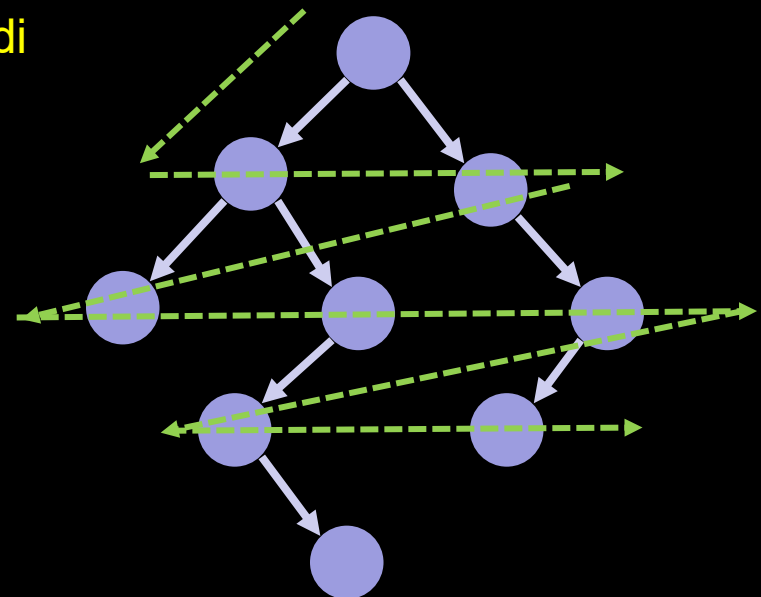


Breadth first search

Algoritmi di generazione delle regole di associazione

- Apriori
- Max-miner
- FPgrowth (Frequency Pattern growth)
- FPmax
- FPclose

apriori genera di volta in volta tutti i valori di supporto (generazione candidati) e scarta quelli poco rilevanti secondo l'algoritmo di ricerca breadth first

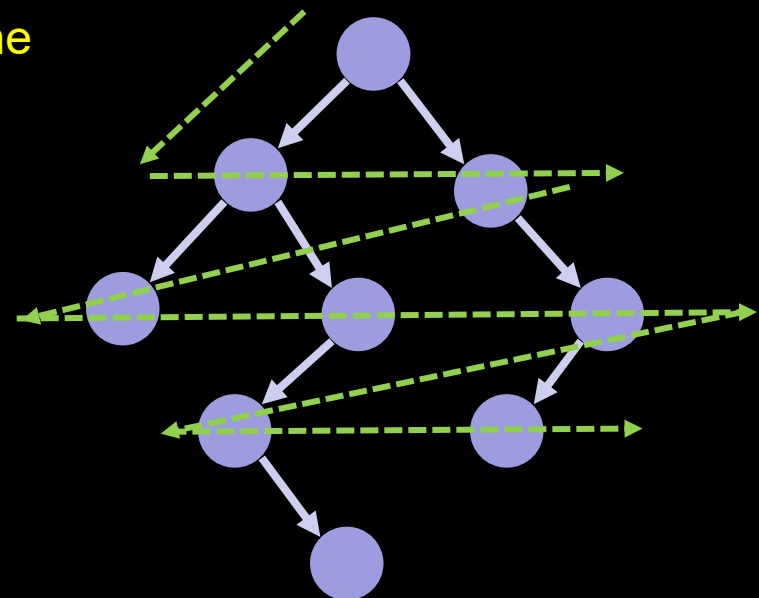


Breadth first search

Algoritmi di generazione delle regole di associazione

- Apriori
- Max-miner
- FPgrowth (Frequency Pattern growth)
- FPmax
- FPclose

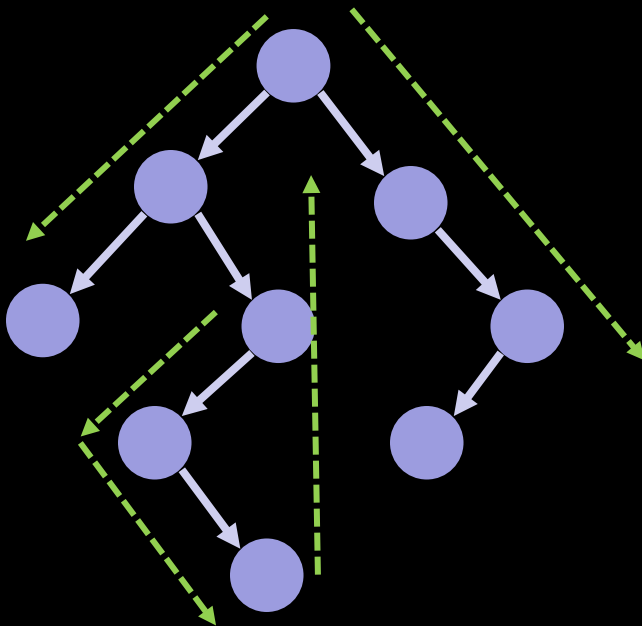
max-miner genera l'albero di enumerazione completo un livello alla volta ma espande solo i rami con il massimo supporto



Breadth first search

Algoritmi di generazione delle regole di associazione

- Apriori
- Max-miner
- FPgrowth (Frequency Pattern growth)
- FPmax
- FPclose



Depth first search

gli algoritmi basati su FP-trees cercano nell'albero usando delle informazioni per scegliere la direzione, es. max supporto

FP-tree

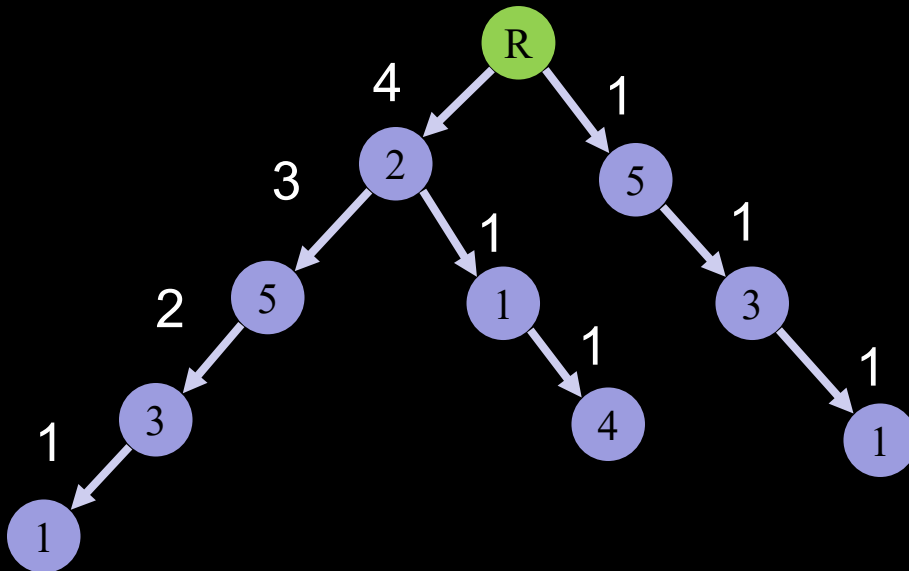
Nel caso di FP-tree

1. Si comprime l'informazione del database che contiene le frequenze degli oggetti in un «frequent-pattern tree» anche detto «FP-tree», che contiene l'informazione relativa alle associazioni tra gli oggetti
2. Si divide il dataset in un insieme di dataset condizionati ognuno associato con un oggetto frequente o un suo sottoinsieme detto «frammento» e si cerca in ogni dataset in maniera separata

FP-tree

FP-tree

Nell'albero ogni nodo rappresenta un oggetto e ogni ramo rappresenta il numero di transazioni che coinvolgono quell'oggetto

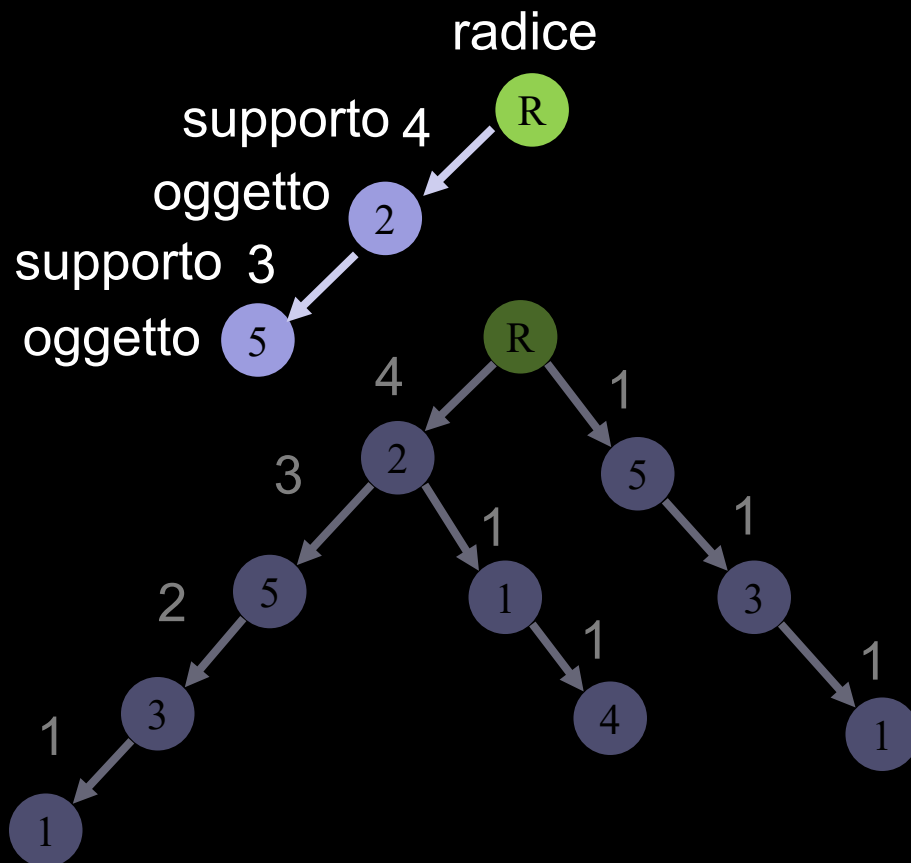


| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 2 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| Oggetto | Supp |
|---------|------|
| 1 | 3 |
| 2 | 4 |
| 3 | 3 |
| 4 | 1 |
| 5 | 4 |

FP-tree

Nell'albero ogni nodo rappresenta un oggetto e ogni ramo rappresenta il numero di transazioni che coinvolgono quell'oggetto



| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 2 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

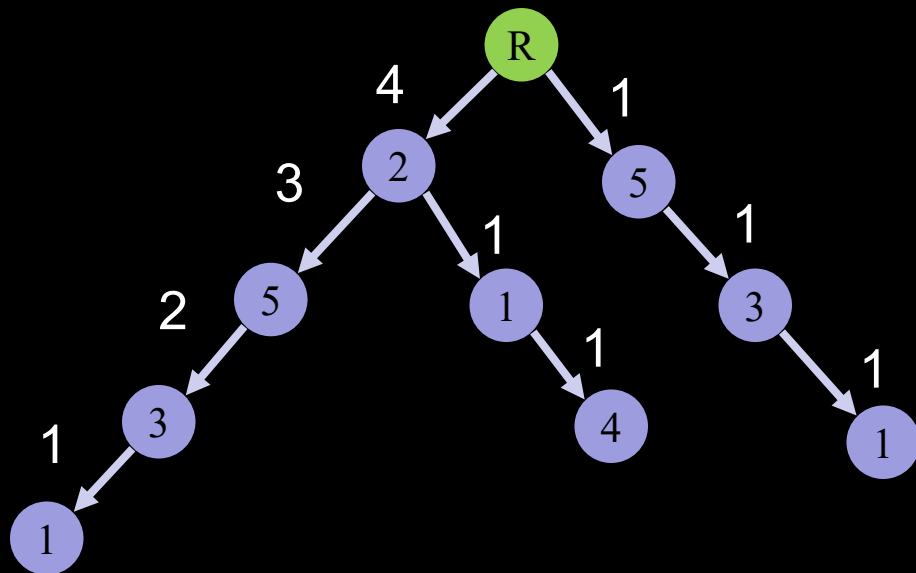
| Oggetto | Supp |
|---------|------|
| 1 | 3 |
| 2 | 4 |
| 3 | 3 |
| 4 | 1 |
| 5 | 4 |

FP-tree

Nell'albero ogni nodo rappresenta un oggetto e ogni ramo rappresenta il numero di transazioni che coinvolgono quell'oggetto

Per costruire l'albero:

1. Ordino tutti gli oggetti in ordine decrescente per supporto
2. Aggiorno il contatore per ogni transazione

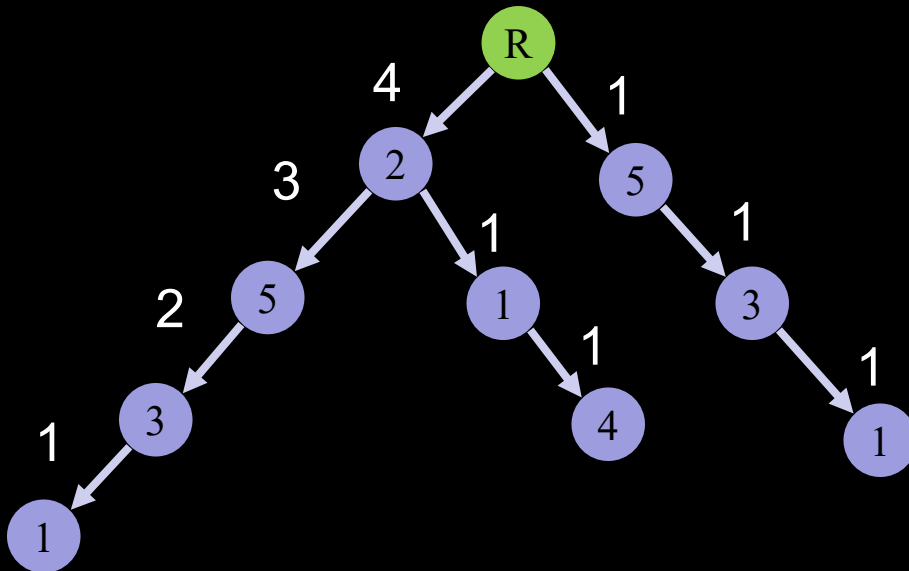


| Tr ID | Oggetti |
|-------|---------|
| T1 | 2 1 4 |
| T2 | 2 5 3 |
| T3 | 2 5 3 1 |
| T4 | 2 5 |
| T5 | 5 3 1 |
| | |

| Oggetto | Supp |
|---------|------|
| 2 | 4 |
| 5 | 4 |
| 3 | 3 |
| 1 | 3 |
| 4 | 1 |

FP-tree – Pattern condizionati

| Oggetto | Base pattern Cond. {Percorso_nodi:supporto} | FT-tree Cond <nodi:supporto> | Pattern frequenti <nodi_frequenti:supp> |
|---------|--|---------------------------------|--|
| 4 | {2,1:1} | <2:1> <1:1> supp<2 | |
| 1 | {2,5,3:1} {5,3:1} {2:1} | <2:2> | <2,1:2> |
| 3 | {2,5:2} {5:1} | <5:2> | <5,3:2> |
| 5 | {2:3} | <2:3> | <2,5:3> |
| 2 | - | - | - |

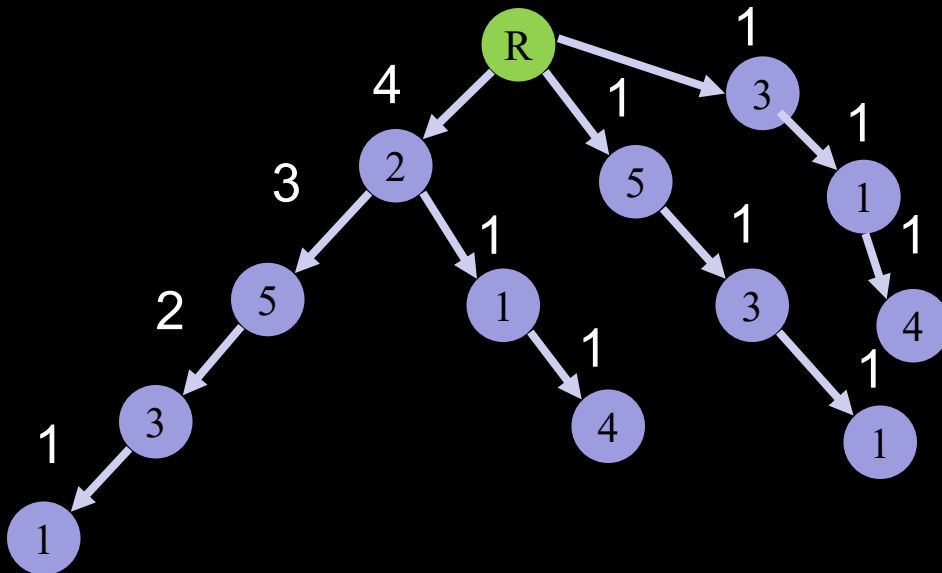


| Tr ID | Oggetti |
|-------|---------|
| T1 | 2 1 4 |
| T2 | 2 5 3 |
| T3 | 2 5 3 1 |
| T4 | 2 5 |
| T5 | 5 3 1 |

| Oggetto | Supp |
|---------|------|
| 2 | 4 |
| 5 | 4 |
| 3 | 3 |
| 1 | 3 |
| 4 | 1 |

FP-tree – modifica proposta in aula

Per costruire l'albero:

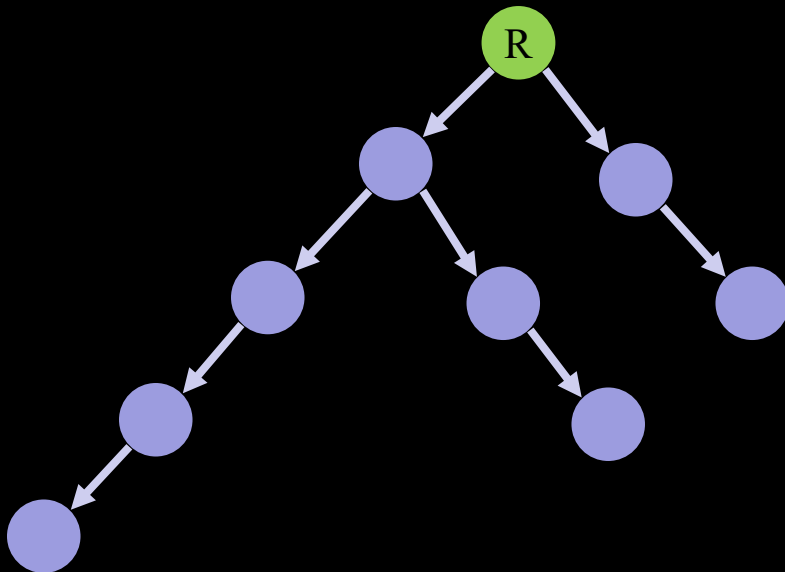


| Tr ID | Oggetti |
|-------|---------|
| T1 | 2 1 4 |
| T2 | 2 5 3 |
| T3 | 2 5 3 1 |
| T4 | 2 5 |
| T5 | 5 3 1 |
| T6 | 3 1 4 |

| Oggetto | Supp |
|---------|------|
| 2 | 4 |
| 5 | 4 |
| 3 | 4 |
| 1 | 4 |
| 4 | 2 |

FP-tree – esercizio fatto in classe

Nell'albero ogni nodo rappresenta un oggetto e ogni ramo rappresenta il numero di transazioni che coinvolgono quell'oggetto

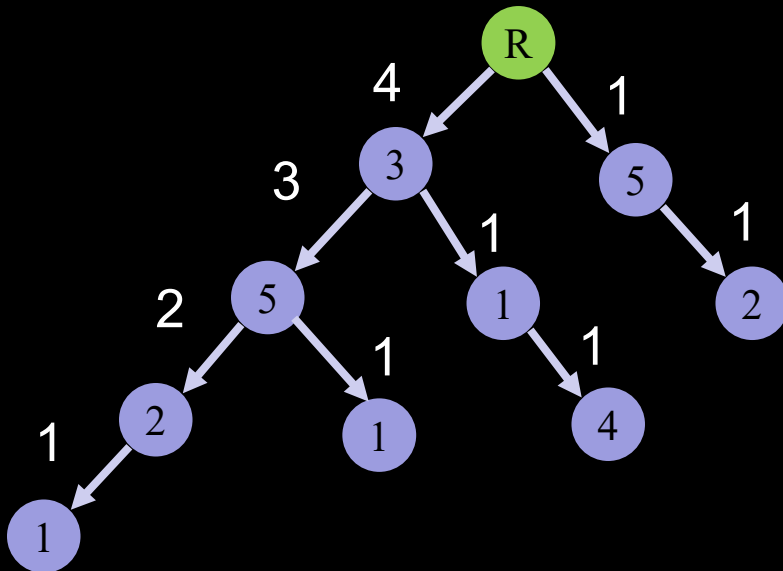


| Tr ID | Oggetti |
|-------|---------|
| T1 | 1 3 4 |
| T2 | 2 3 5 |
| T3 | 1 2 3 5 |
| T4 | 2 5 |
| T5 | 1 3 5 |

| Oggetto | Supp |
|---------|------|
| 1 | 3 |
| 2 | 3 |
| 3 | 4 |
| 4 | 1 |
| 5 | 4 |

FP-tree – esercizio fatto in classe

Nell'albero ogni nodo rappresenta un oggetto e ogni ramo rappresenta il numero di transazioni che coinvolgono quell'oggetto



| Tr ID | Oggetti |
|-------|---------|
| T1 | 3 1 4 |
| T2 | 3 5 2 |
| T3 | 3 5 2 1 |
| T4 | 5 2 |
| T5 | 3 5 1 |

| Oggetto | Supp |
|---------|------|
| 5 | 4 |
| 3 | 4 |
| 2 | 3 |
| 1 | 3 |
| 4 | 1 |

Fast Algorithms for Frequent Itemset Mining Using FP-Trees

Gösta Grahne, *Member, IEEE*, and Jianfei Zhu, *Student Member, IEEE*

Abstract—Efficient algorithms for mining frequent itemsets are crucial for mining association rules as well as for many other data mining tasks. Methods for mining frequent itemsets have been implemented using a prefix-tree structure, known as an FP-tree, for storing compressed information about frequent itemsets. Numerous experimental results have demonstrated that these algorithms perform extremely well. In this paper, we present a novel FP-array technique that greatly reduces the need to traverse FP-trees, thus obtaining significantly improved performance for FP-tree-based algorithms. Our technique works especially well for sparse data sets. Furthermore, we present new algorithms for mining all, maximal, and closed frequent itemsets. Our algorithms use the FP-tree data structure in combination with the FP-array technique efficiently and incorporate various optimization techniques. We also present experimental results comparing our methods with existing algorithms. The results show that our methods are the fastest for many cases. Even though the algorithms consume much memory when the data sets are sparse, they are still the fastest ones when the minimum support is low. Moreover, they are always among the fastest algorithms and consume less memory than other methods when the data sets are dense.

Index Terms—Data mining, association rules.

Online retail dataset



<https://archive.ics.uci.edu/ml/datasets/online+retail>

A questo indirizzo potete trovare un dataset di esempio per provare i sistemi di generazione alberi decisionali e regole associative

Online Retail Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: This is a transnational data set which contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail.

| | | | | | |
|----------------------------|---------------------------------------|-----------------------|--------|---------------------|------------|
| Data Set Characteristics: | Multivariate, Sequential, Time-Series | Number of Instances: | 541909 | Area: | Business |
| Attribute Characteristics: | Integer, Real | Number of Attributes: | 8 | Date Donated | 2015-11-06 |
| Associated Tasks: | Classification, Clustering | Missing Values? | N/A | Number of Web Hits: | 637175 |

Source:

Dr Daqing Chen, Director: Public Analytics group, chend '@' lsbu.ac.uk, School of Engineering, London South Bank University, London SE1 0AA, UK.

Data Set Information:

This is a transnational data set which contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail. The company mainly sells unique all-occasion gifts. Many customers of the company are wholesalers.

Attribute Information:

InvoiceNo: Invoice number. Nominal, a 6-digit integral number uniquely assigned to each transaction. If this code starts with letter 'c', it indicates a cancellation.
StockCode: Product (item) code. Nominal, a 5-digit integral number uniquely assigned to each distinct product.
Description: Product (item) name. Nominal.
Quantity: The quantities of each product (item) per transaction. Numeric.
InvoiceDate: Invoice Date and time. Numeric, the day and time when each transaction was generated.
UnitPrice: Unit price. Numeric, Product price per unit in sterling.
CustomerID: Customer number. Nominal, a 5-digit integral number uniquely assigned to each customer.
Country: Country name. Nominal, the name of the country where each customer resides.

Relevant Papers:

The evolution of direct, data and digital marketing, Richard Webber, Journal of Direct, Data and Digital Marketing Practice (2013) 14, 291â€“309.
Clustering Experiments on Big Transaction Data for Market Segmentation.
Ashishkumar Singh, Grace Rumantr, Annie South, Blair Bethwaite, Proceedings of the 2014 International Conference on Big Data Science and Computing.
A decision-making framework for precision marketing, Zhen You, Yain-Whar Si, Defu Zhang, XiangXiang Zeng, Stephen C.H. Leung c, Tao Li, Expert Systems with Applications, 42 (2015) 3357â€“3367.

Citation Request:

Daqing Chen, Sai Liang Sain, and Kun Guo, Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining, Journal of Database Marketing and Customer Strategy Management, Vol. 19, No. 3, pp. 197â€“208, 2012 (Published online before print: 27 August 2012, doi: 10.1057/dbm.2012.17).

Regole di associazione

Vantaggi:

- ✓ Semplici da capire e implementare
- ✓ Semplice inferenza

Regole di associazione

Svantaggi:

- ✓ Orverfitting
- ✓ No regressione
- ✓ Bassa capacità di astrazione
- ✓ Alto costo computazionale (devo girare l'intero dataset molte volte)

Regole associative – tutorial python

Mlxtend (machine learning extensions) è una libreria python contenente degli strumenti utili in diversi task di data science



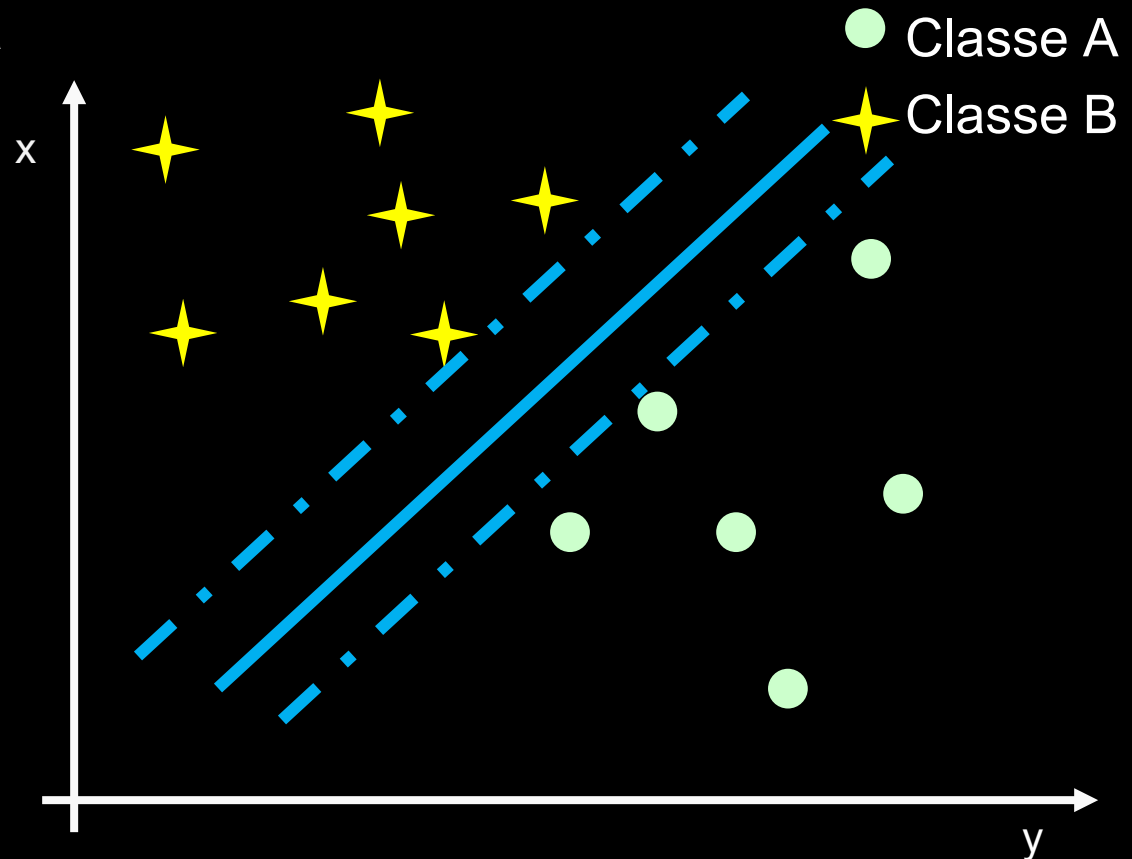
Support vector machines

Il support vector machine o SVM è un algoritmo di apprendimento che ordina i dati disponibili in categorie

Support vector machines

Il support vector machine o SVM è un algoritmo di apprendimento che ordina i dati disponibili in categorie

Massimizza la separazione tra le classi (distanza tra i dati e la retta di separazione)



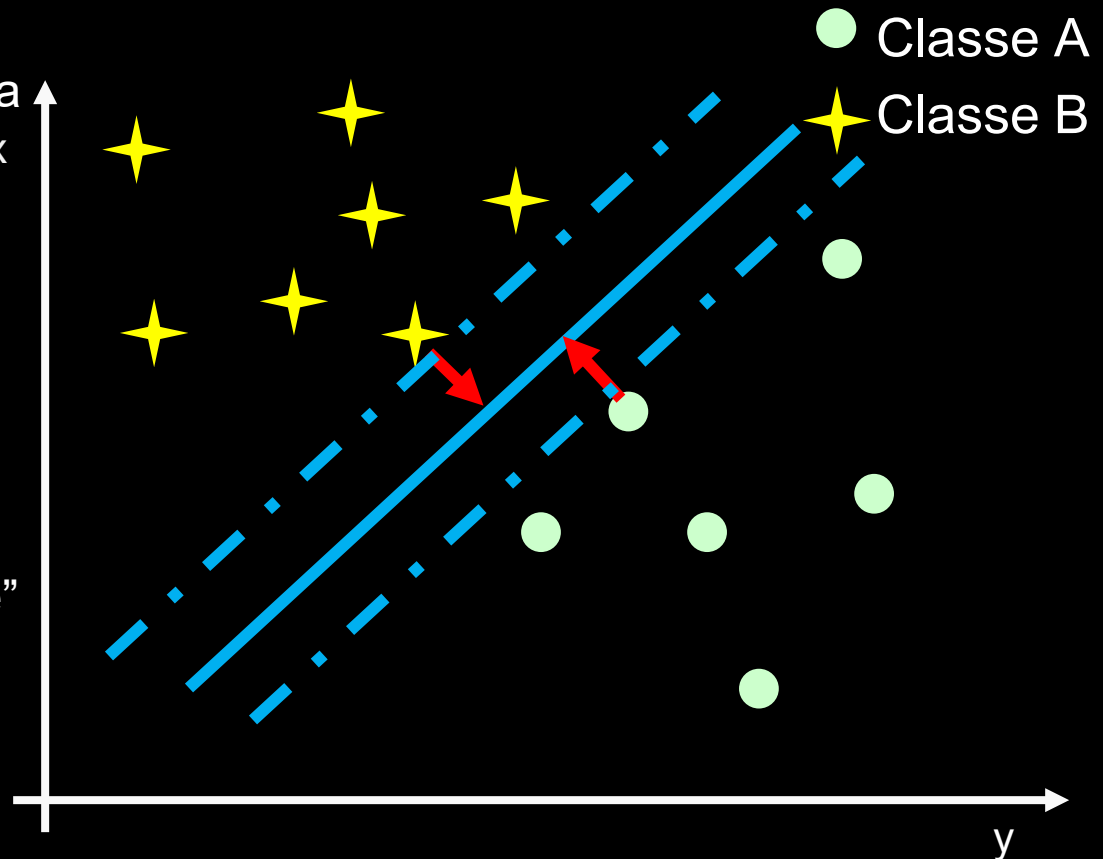
Support vector machines

Il support vector machine o SVM è un algoritmo di apprendimento che ordina i dati disponibili in categorie

In termini più tecnici si cerca la retta che massimizza la distanza tra i primi dati delle classi detti “support vectors”

D^+ e d^- sono le distanze tra il punto delle classi più vicino alla retta di separazione.

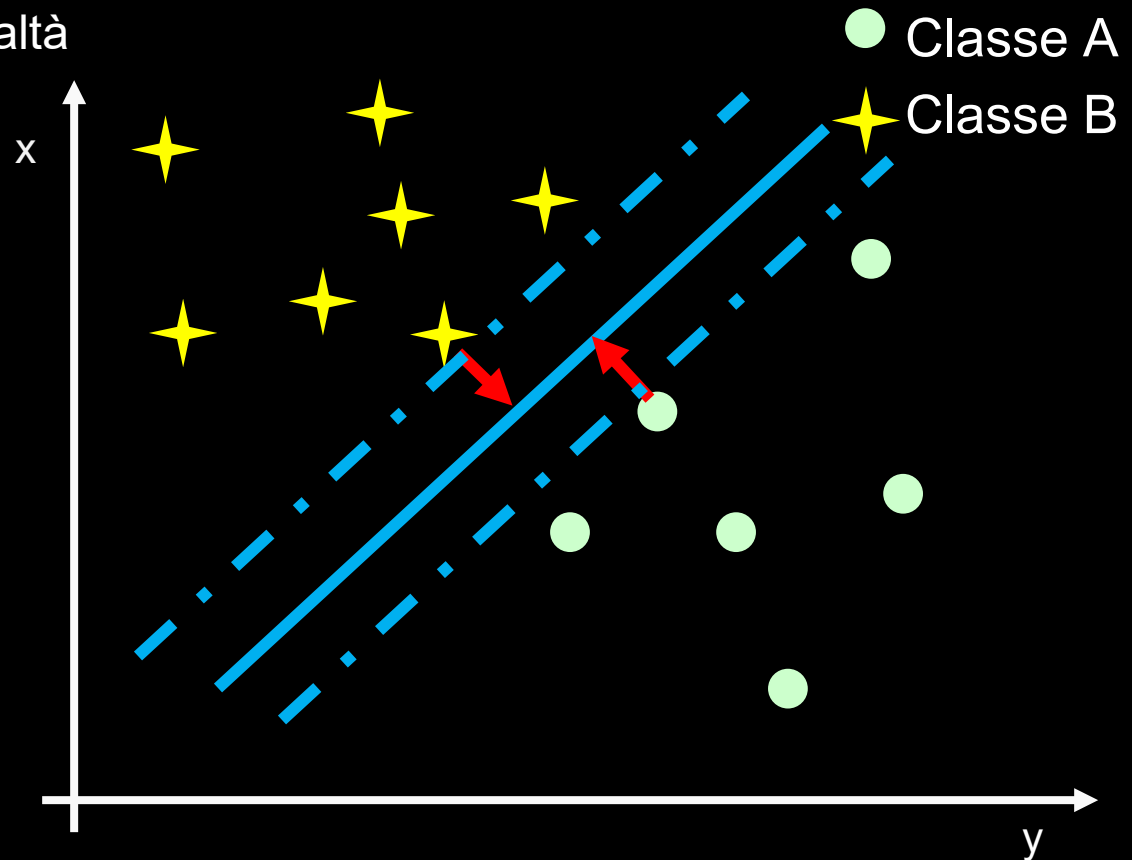
La somma è chiamata “margine”



Support vector machines

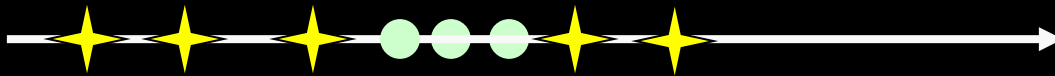
Il support vector machine o SVM è un algoritmo di apprendimento che ordina i dati disponibili in categorie

La retta di separazione è in realtà un piano o iperpiano, quando si parla di dati con più dimensioni



Support vector machines – kernels

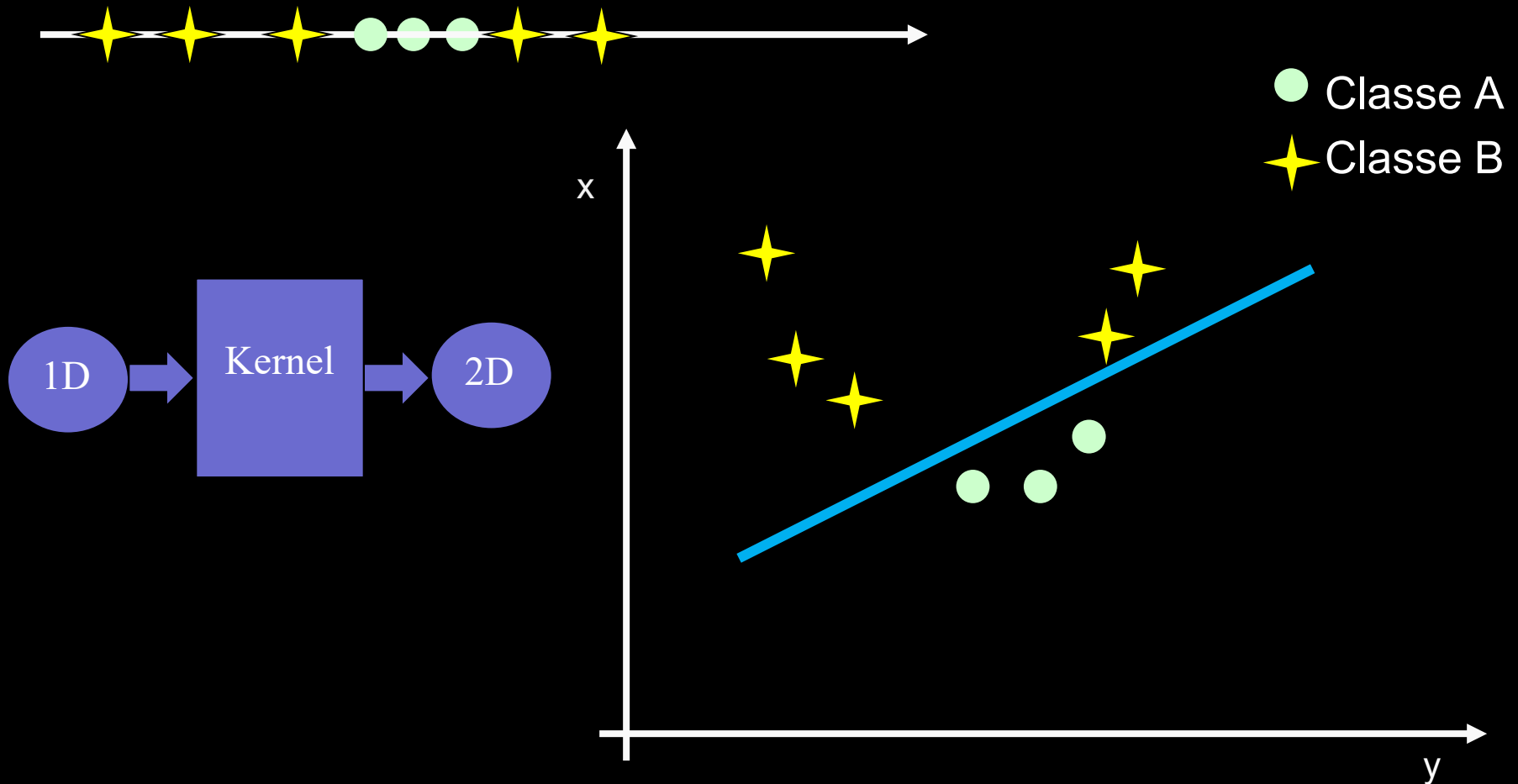
la separazione dei dati non è sempre semplice come potrebbe apparire



● Classe A
★ Classe B

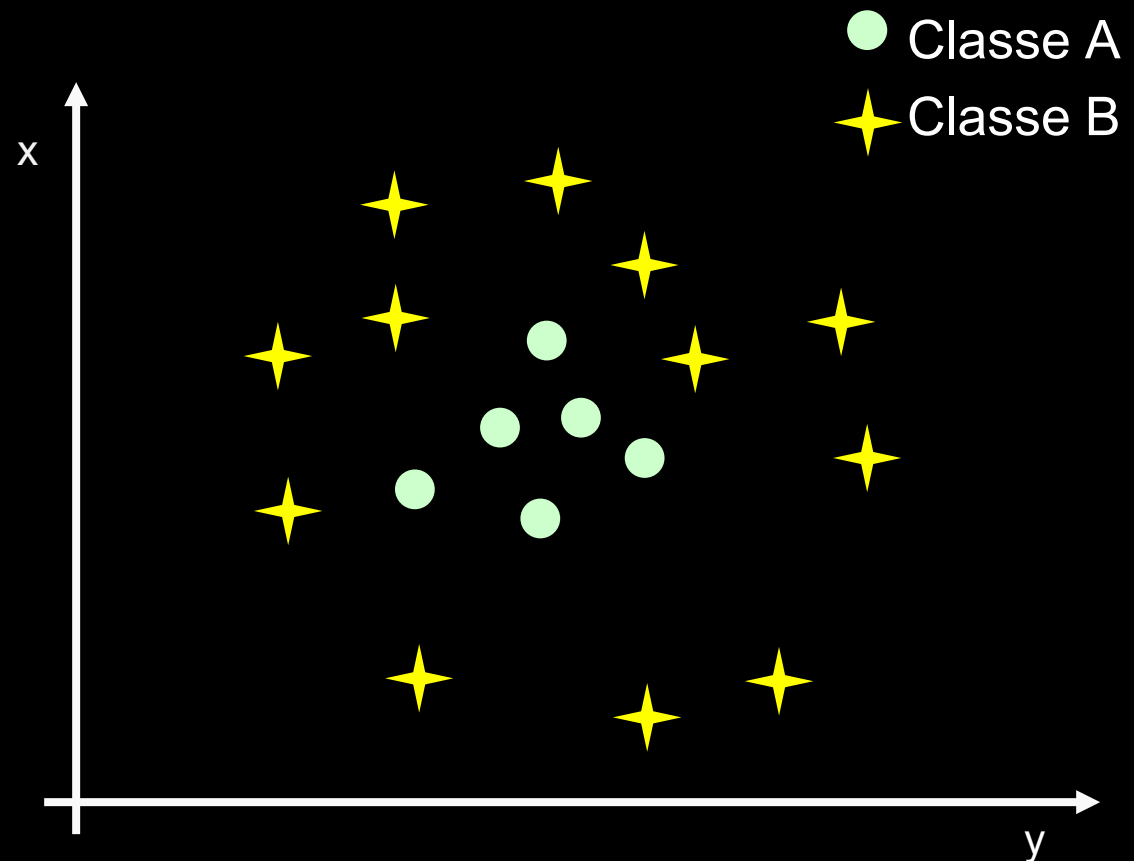
Support vector machines – kernels

la separazione dei dati non è sempre semplice come potrebbe apparire



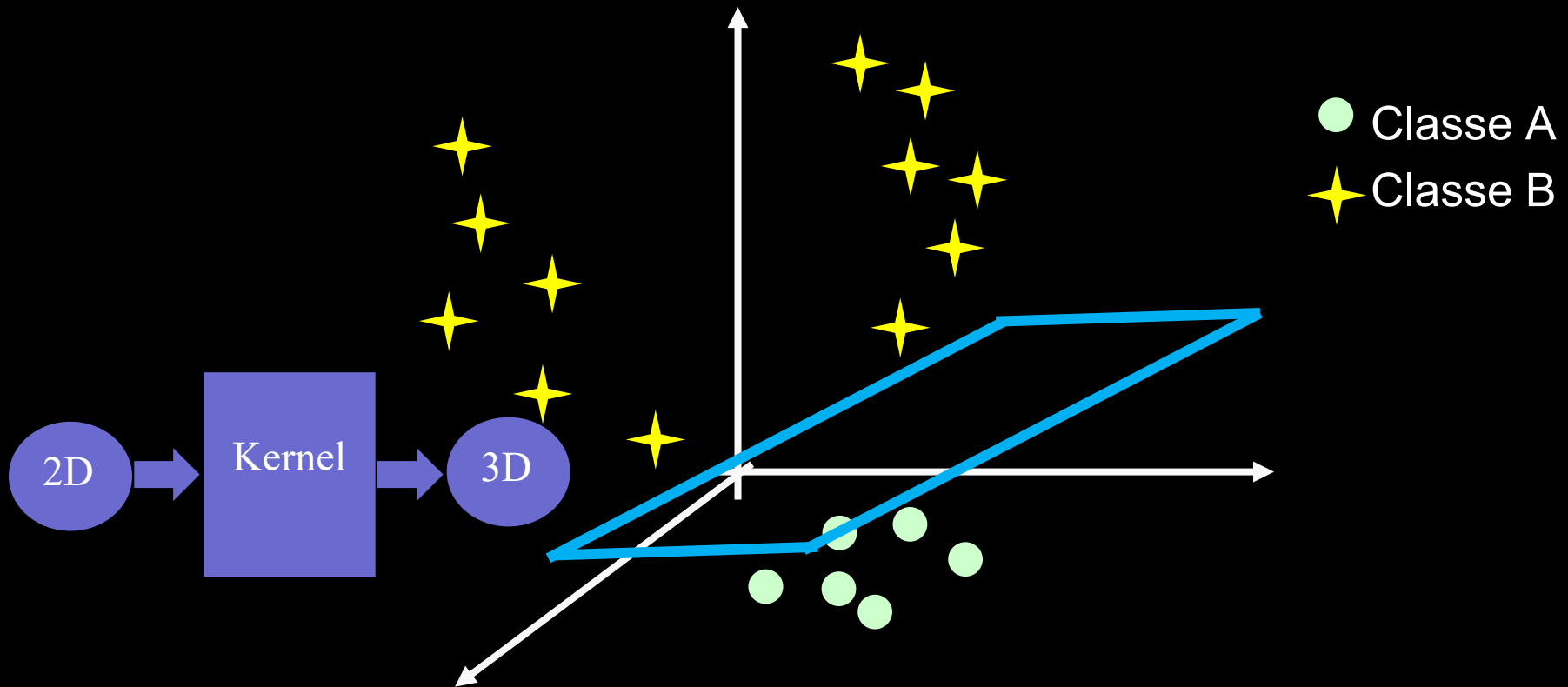
Support vector machines – kernels

la separazione dei dati non è sempre semplice come potrebbe apparire



Support vector machines – kernels

la separazione dei dati non è sempre semplice come potrebbe apparire



Support vector machines – pseudocode

input:

`N_in` #input, `N_sv` #support vectors, `N_ft` #features

`SV[N_sv]` array di support vectors, `IN[N_in]` dati input, `b*` bias

output:

`F` (funzione decisionale)

Support vector machines – pseudocode

input:

N_in #input, N_sv #support vectors, N_ft #features

SV[N_sv] array di support vectors, IN[N_in] dati input, b* bias

output:

F (funzione decisionale)

for **i** ← 1 to **N_in** do

F=0

Consideriamo i support vectors e
poniamo a zero la distanza iniziale

end

Support vector machines – pseudocode

input:

N_in #input, **N_sv #support vectors**, N_ft #features
SV[N_sv] array di support vectors, IN[N_in] dati input, b* bias

output:

F (funzione decisionale)

for i←1 to N_in **do**

F=0

for j←1 to **N_sv** **do**

dist = 0

end

end

Consideriamo i support vectors e
poniamo a zero la distanza iniziale

Support vector machines – pseudocode

input:

N_in #input, N_sv #support vectors, **N_ft #features**
SV[N_sv] array di support vectors, IN[N_in] dati input, b* bias

output:

F (funzione decisionale)

for i←1 to N_in **do**

F=0

for j←1 to N_sv **do**

dist = 0

for k←1 to **N_ft** **do**

dist+= (SV[j].feature[k] - IN[i].feature[k])^2

end

end

end

Consideriamo le features che abbiamo nei dati e calcoliamo la distanza tra la feature generata e i vettori di supporto

Support vector machines – pseudocode

input:

N_in #input, N_sv #support vectors, N_ft #features
SV[N_sv] array di support vectors, IN[N_in] dati input, b* bias

output:

F (funzione decisionale)

for i←1 **to** N_in **do**

F=0

for j←1 **to** N_sv **do**

dist = 0

for k←1 **to** N_ft **do**

dist+=(SV[j].feature[k] - IN[i].feature[k])^2

end

kk = e^{-gamma x dist} -- radial basis function kernel

end

end

Trasformazione del kernel

Support vector machines – pseudocode

input:

N_in #input, N_sv #support vectors, N_ft #features
SV[N_sv] array di support vectors, IN[N_in] dati input, b* bias

output:

F (funzione decisionale)

for i←1 **to** N_in **do**

F=0

for j←1 **to** N_sv **do**

dist = 0

for k←1 **to** N_ft **do**

dist+=(SV[j].feature[k] - IN[i].feature[k])^2

end

kk = e^(-gamma x dist) -- radial basis function kernel

F += SV[j].alpha x kk

end

F = F+b*

end

Aggiornamento della funzione
decisionale

Support vector machines – pseudocode

input:

N_in #input, N_sv #support vectors, N_ft #features
SV[N_sv] array di support vectors, IN[N_in] dati input, b* bias

output:

F (funzione decisionale)

for i←1 **to** N_in **do**

 F=0

for j←1 **to** N_sv **do**

 dist = 0

for k←1 **to** N_ft **do**

 dist+=(SV[j].feature[k] - IN[i].feature[k])^2

end

 kk = e^(-gamma x dist) -- radial basis function kernel

 F += SV[j].alpha x kk

end

 F = F+b*

end

Questo algoritmo è la funzione
«fit» per calcolare il modello
tramite la minimizzazione di una
funzione di distanza tra classi

Support vector machines – pseudocode

input:

```
Input_data #input,  
F (funzione decisionale)
```

output:

```
classe di appartenenza
```

```
classe = F(input_data)
```

Per stimare la classe devo semplicemente usare il modello stimato, tipicamente indicato nelle librerie come funzione «predict»

SVM tutorial

Codice



Metriche di valutazione della classificazione

Dati 2 classi A (positivi) e B (negativi), e un insieme di dati,

Si definiscono:

- Veri positivi: numero di campioni positivi inseriti nella classe corretta
- Veri negativi: numero di campioni negativi inseriti nella classe corretta
- Falsi positivi: numero di campioni negativi, inseriti nella classe positivi
- Falsi negativi: numero di campioni positivi, inseriti nella classe negativi

Metriche di valutazione della classificazione

Matrice di confusione 2 classi:



| | | Stima | |
|--------------|----------|----------|----------|
| | | positivi | negativi |
| Ground truth | positivi | VP | FN |
| | negativi | FP | VN |


Metriche di valutazione della classificazione

Matrice di confusione 2 classi:



| | | Stima | |
|--------------|----------|----------|----------|
| | | positivi | negativi |
| Ground truth | positivi | TP | FN |
| | negativi | FP | TN |

Matrice di confusione n classi:



| | | Stima | | |
|--------------|----------|-----------|-----------|-----------|
| | | Classe 1 | Classe 2 | Classe n |
| Ground truth | Classe 1 | $C_{1,1}$ | $C_{1,2}$ | $C_{1,n}$ |
| | Classe 2 | $C_{2,1}$ | $C_{2,2}$ | $C_{1,1}$ |
| | Classe n | $C_{n,1}$ | $C_{n,2}$ | $C_{n,n}$ |

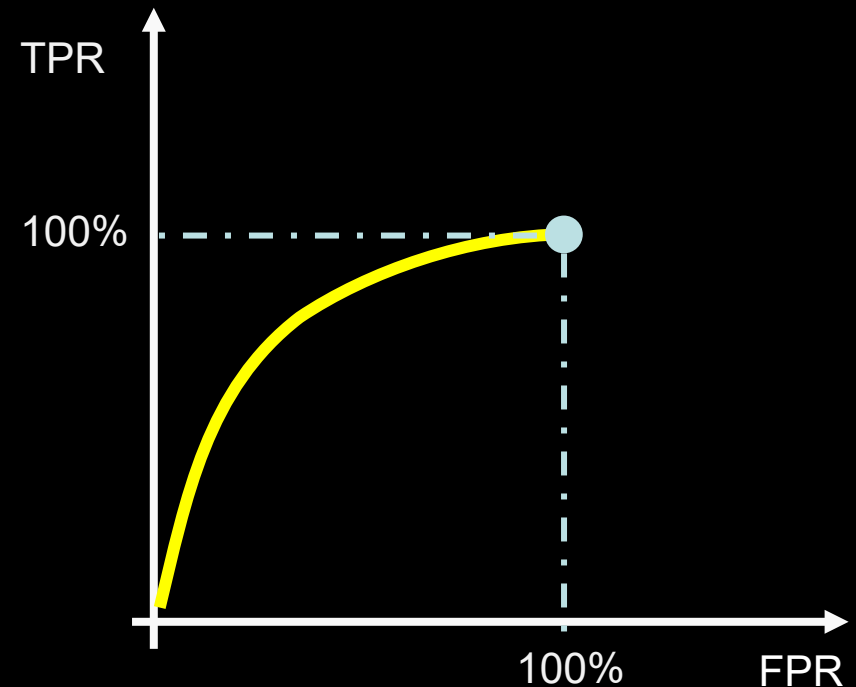
Metriche di valutazione della classificazione – Analisi ROC

Receiver operating characteristic meglio nota come Curva ROC è usata per illustrare le abilità di un classificatore binario di discriminare due classi

Metriche di valutazione della classificazione – Analisi ROC

Receiver operating characteristic meglio nota come Curva ROC è usata per illustrare le abilità di un classificatore binario di discriminare due classi

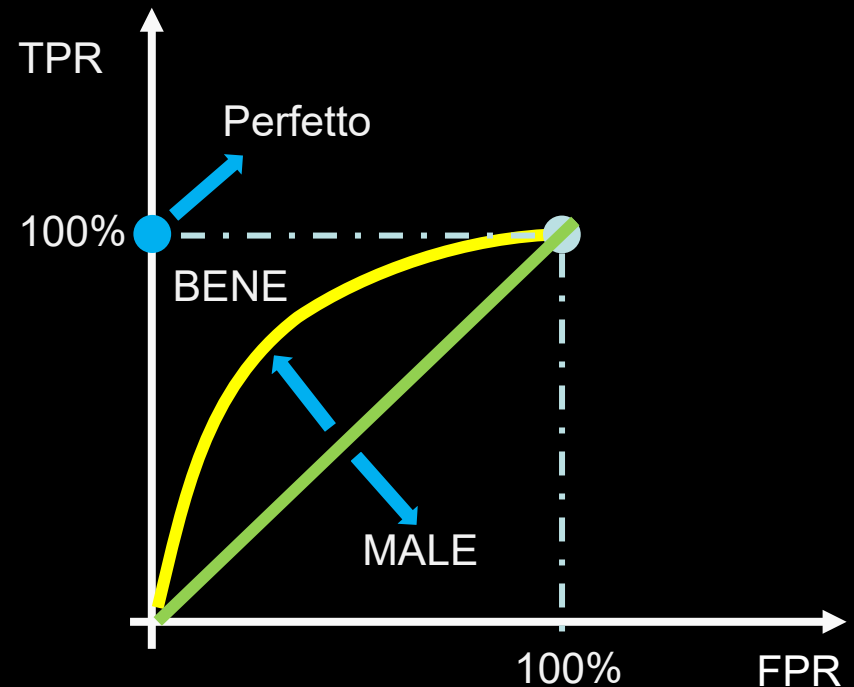
La curva si ottiene plottando il TPR contro il FPR



Metriche di valutazione della classificazione – Analisi ROC

Receiver operating characteristic meglio nota come Curva ROC è usata per illustrare le abilità di un classificatore binario di discriminare due classi

La curva si ottiene plottando il TPR contro il FPR



Metriche di valutazione della classificazione – Analisi ROC

Receiver operating characteristic meglio nota come Curva ROC è usata per illustrare le abilità di un classificatore binario di discriminare due classi

La curva si ottiene plottando il TPR contro il FPR

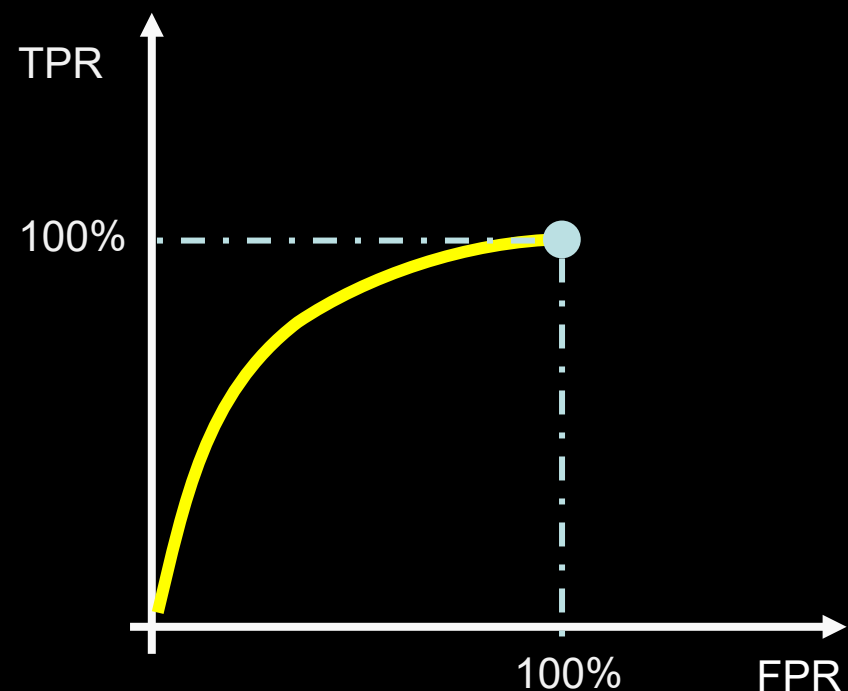
True positive rate (sensibilità)

True negative rate (specificità)

False positive rate

False negative rate

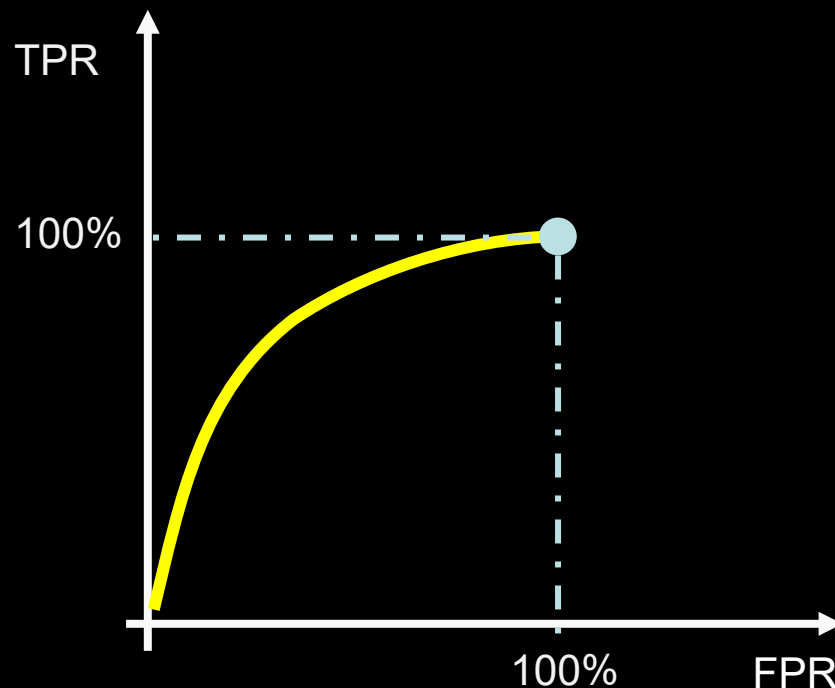
| | | Stima | |
|--------------|----------|----------|----------|
| | | positivi | negativi |
| Ground truth | positivi | TP % | FN % |
| | negativi | FP % | TN % |



Metriche di valutazione della classificazione – Analisi ROC

AUC under the curve o in italiano, area sottesa dalla curva ROC.

Usando delle unità normalizzate, l'area sottesa dalla curva ROC (semplicemente indicata come AUC) è la probabilità che un classificatore scelta un esempio random come positivo piuttosto che negativo



Metriche di valutazione della classificazione – Analisi ROC

$$\text{TPR} = \text{TP} / \text{P}$$

$$\text{TNR} = \text{TN} / \text{N}$$

$$\text{FPR} = \text{FP} / \text{N}$$

$$\text{FNR} = \text{FN} / \text{P}$$

$$\text{ACC} = (\text{TP} + \text{TN}) / (\text{P} + \text{N})$$

$$\text{F1-score} = 2\text{TP} / (2\text{TP} + \text{FP} + \text{FN})$$

Reti neurali

Saranno coperte nelle prossime lezioni