



ICT Training Center

Il tuo partner per la Formazione e la Trasformazione digitale della tua azienda



Note



SPRING AI

GENERATIVE ARTIFICIAL INTELLIGENCE CON JAVA

Simone Scannapieco

Corso base per Venis S.p.A, Venezia, Italia

Settembre 2025

Note

MACHINE LEARNING

CENNI ED ESEMPI

Note



Arthur L. Samuel.
Some studies in Machine Learning using the game of checkers.
IBM Journal of Research and Development, 1959.



- Nato all'inizio del primo declino della AI
 - Riprende l'eredità di Rosenblatt (algoritmo di apprendimento del *perceptron*)
 - Insieme di metodologie che
 - scoprono **modelli** (regolarità o **pattern**) nei dati
 - predicono comportamenti o prendono decisioni sulla base di nuovi dati

Si dice che un programma **apprende** dall'esperienza E con riferimento ad alcune classi di compiti T e con misurazione della performance P , se le sue performance nel compito T , come misurato da P , migliorano con l'esperienza E .

A black and white portrait photograph of James R. Thompson. He is a middle-aged man with dark hair, wearing thick-rimmed glasses and a dark suit jacket over a white shirt and tie. He is smiling slightly and looking towards the camera.

Arthur Samuel
©HistoryOfInformation

 Simone Scannapieco

Spring AI - Corso base

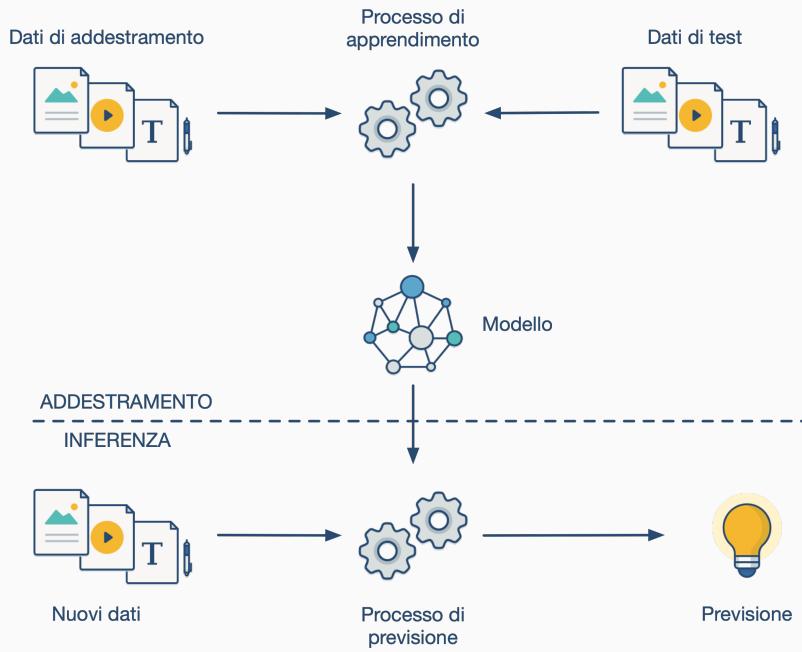
Venis S.p.A, Venezia, IT

3 / 9

Note

WORKFLOW PROCESSO DI MACHINE LEARNING

VISIONE SEMPLIFICATA

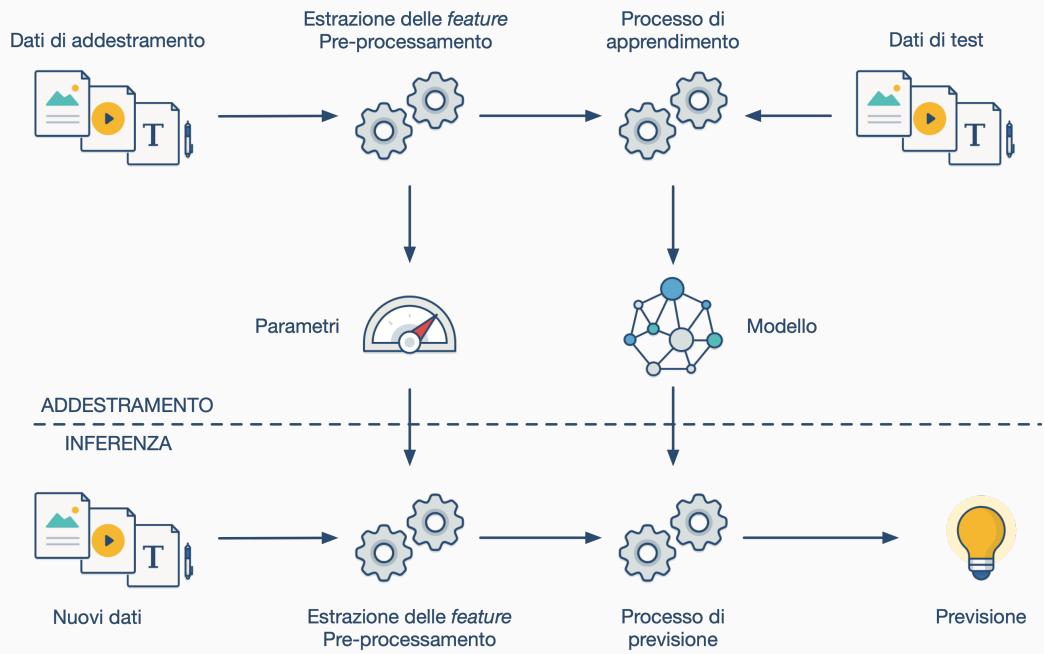


©Simone Scannapieco

Note

WORKFLOW PROCESSO DI MACHINE LEARNING

VISIONE DETTAGLIATA

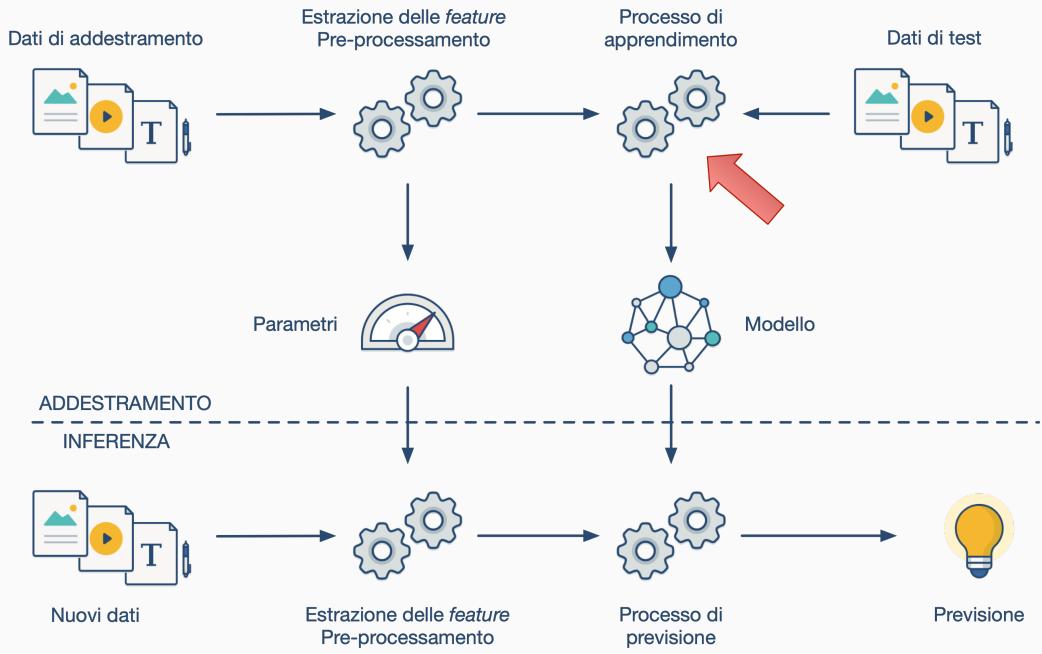


©Simone Scannapieco

Note

WORKFLOW PROCESSO DI MACHINE LEARNING

COME AVVIENE L'APPRENDIMENTO?

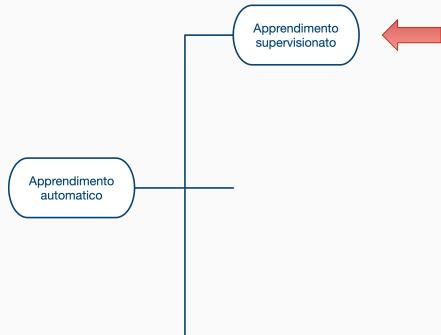


©Simone Scannapieco

Note

PROCESSI DI APPRENDIMENTO

CLASSIFICAZIONE CANONICA



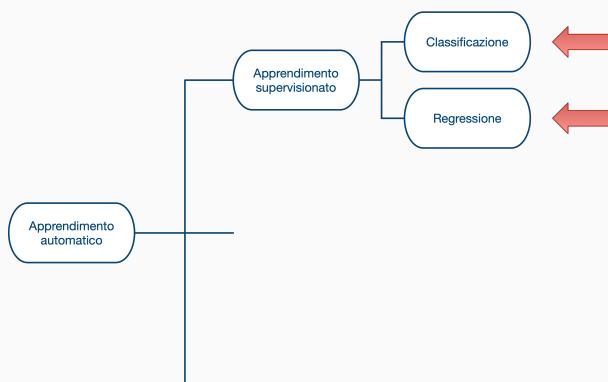
©Simone Scannapieco

- Per ogni dato abbiamo il valore che desideriamo venisse predetto dal modello (etichetta)
 - Supervisore che etichetta ogni dato di addestramento
 - Ricerca di un modello che realizzi la corrispondenza dato-etichetta corrispondente

Note

PROCESSI DI APPRENDIMENTO

CLASSIFICAZIONE CANONICA

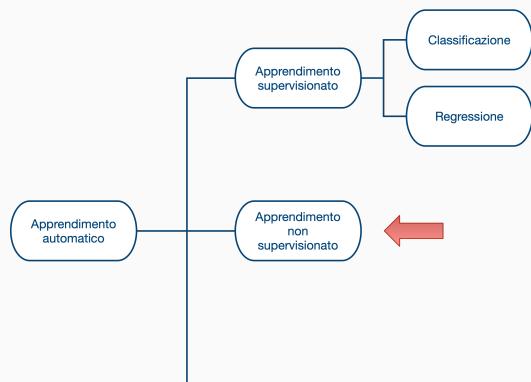


©Simone Scannapieco

- ➡ **Classificazione** quando il numero di etichette è finito (es. identificazione di oggetti)
 - ➡ **Regressione** quando l'etichetta ha un numero infinito di valori (es. previsione di un numero reale)

Note

PROCESSI DI APPRENDIMENTO CLASSIFICAZIONE CANONICA



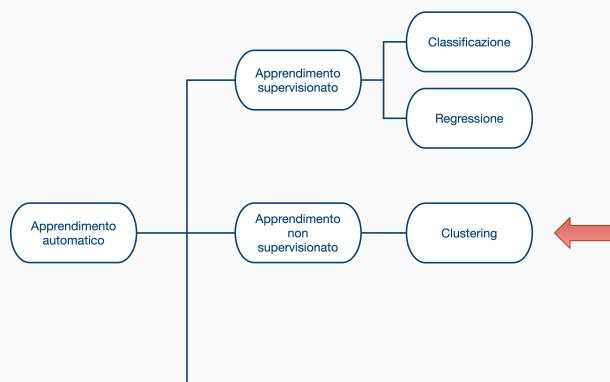
©Simone Scannapieco

- Nessun supervisore etichetta i dati
 - Non esiste una etichetta da predire
 - Il *focus* è su come (e quali) dati sono relazionati fra loro (**similarità**)

Note

PROCESSI DI APPRENDIMENTO

CLASSIFICAZIONE CANONICA

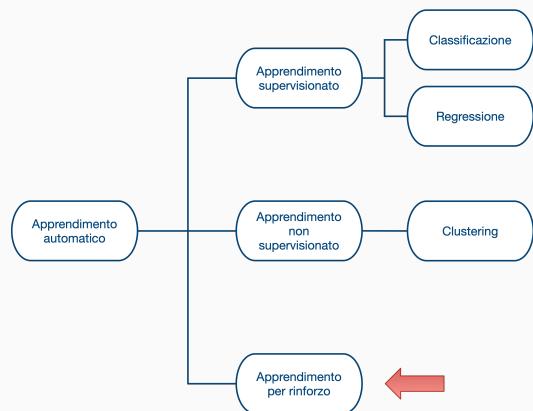


©Simone Scannapieco

- ### → Clustering per identificare gruppi disgiunti (o meno)

Note

PROCESSI DI APPRENDIMENTO CLASSIFICAZIONE CANONICA



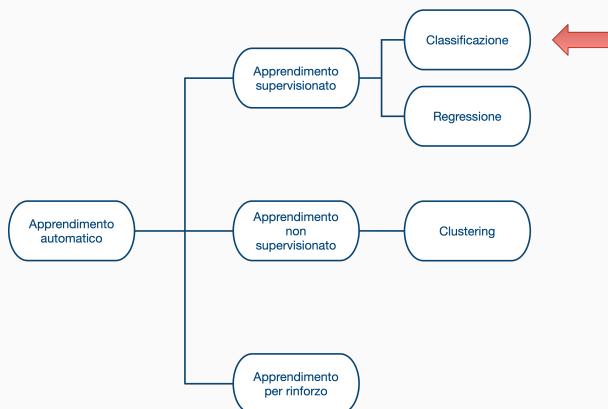
©Simone Scannapieco

- ➡ Problemi di decisione sequenziali (decidere l'azione futura in base allo stato attuale)
 - ➡ Meccanismo interno che valuta l'efficacia dell'azione (rispetto a parametri)
 - ➡ Azioni efficaci vengono premiate
 - ➡ Azioni inefficaci vengono penalizzate

Note

PROCESSI DI APPRENDIMENTO

CLASSIFICAZIONE CANONICA



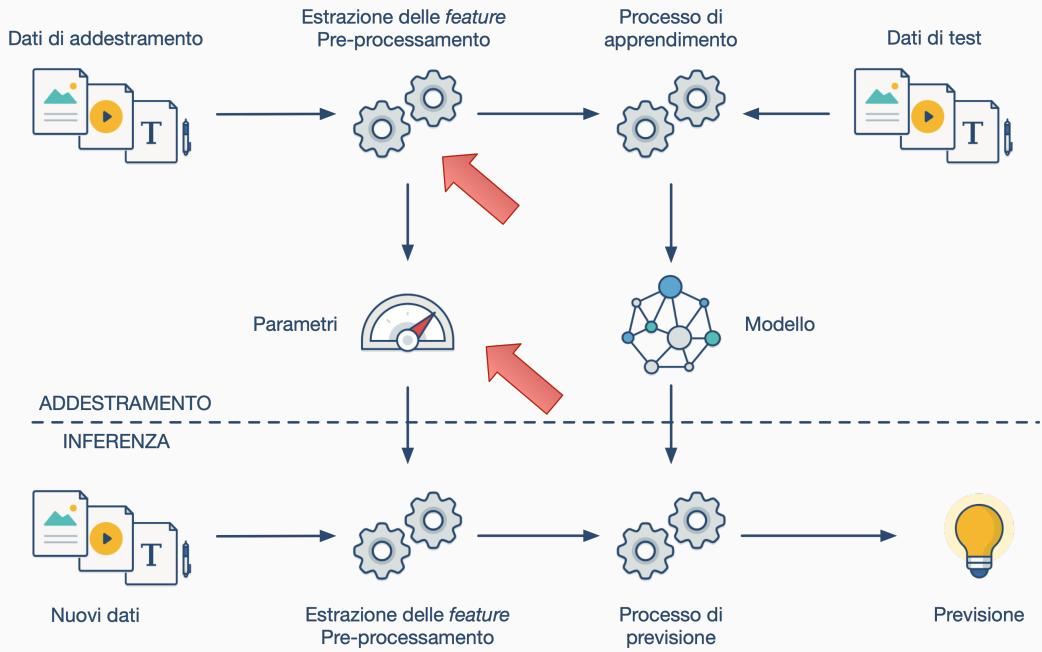
©Simone Scannapieco

- Ci concentriamo su **addestramento supervisionato**
 - Classificazione
 - Rilevamento entità (caso speciale di classificazione)

Note

RAPPRESENTAZIONE DELLA CONOSCENZA

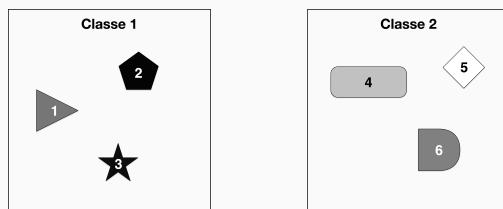
COSA SONO FEATURES E PARAMETRI? PERCHÉ PRE-PROCESSARE?



©Simone Scannapieco

Note

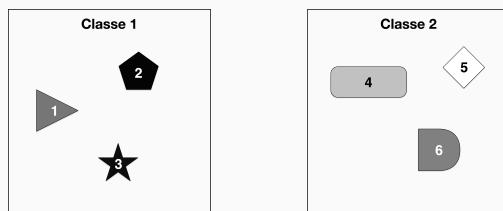
RAPPRESENTAZIONE DELLA CONOSCENZA ESTRAZIONE DELLE FEATURES



- Come distinguere gli oggetti delle due classi?
 - Definire delle caratteristiche peculiari (**features**)
 - Colore
 - Numero dei vertici
 - Forma dei vertici

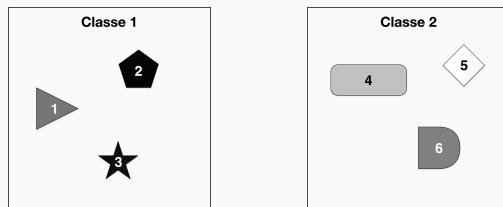
Note

RAPPRESENTAZIONE DELLA CONOSCENZA ESTRAZIONE DELLE FEATURES



Elemento	Colore	Vertici	Vertici arrotondati?	Classe
1	Grigio medio	3	No	1
2	Grigio scuro	5	No	1
3	Grigio scuro	10	No	1
4	Grigio chiaro	4	Sí	2
5	Bianco	4	No	2
6	Grigio medio	4	Sí	2

Note

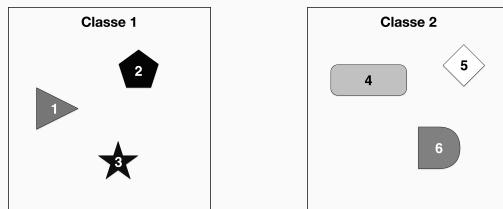


- ➔ Il linguaggio macchina è **numerico**
 - ➔ Tradurre i valori delle *features*
 - ➔ Colore: intero in $[0, 255]$ (0 = nero, 255 = bianco)
 - ➔ Numero dei vertici: intero in $[3, n]$ (n fissato a priori)
 - ➔ Forma dei vertici: booleano in $[0, 1]$ (0 = No, 1 = Sí)

Note

RAPPRESENTAZIONE DELLA CONOSCENZA

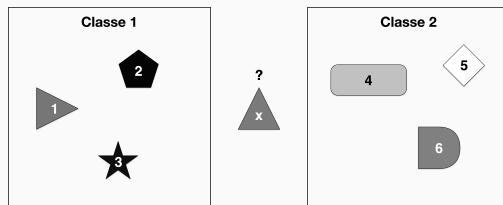
ESTRAZIONE DELLE FEATURES



Elemento	Colore	Vertici	Vertici arrotondati?	Classe	Feature vector
1	120	3	0	1	[120, 3, 0]
2	5	5	0	1	[5, 5, 0]
3	10	10	0	1	[10, 10, 0]
4	196	4	1	2	[196, 4, 1]
5	255	4	0	2	[255, 4, 0]
6	128	4	1	2	[128, 4, 1]

Note

RAPPRESENTAZIONE DELLA CONOSCENZA PERCHÉ PRE-PROCESSARE I DATI?



→ Esempio di algoritmo di classificazione

Data: Un nuovo elemento x

Result: La classe C di appartenenza di x

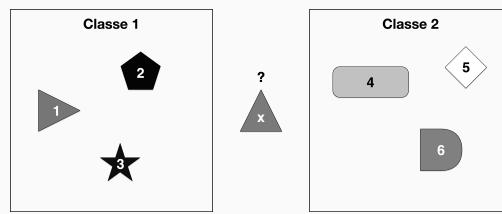
```

1 for elemento y in {Classe 1, Classe 2} do
2   | Calcola la distanza d fra y ed x;
3 end for
4 C = classe dell'elemento y che minimizza d;

```

→ Distanza intesa come euclidea $d(y, x) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2}$

Note

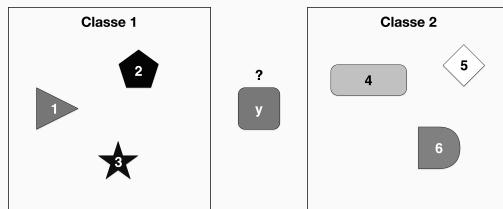


💡 Con $x = [123, 3, 0]$

$$\left. \begin{array}{l} d(1, x) = \sqrt{(-3)^2 + 0^2 + 0^2} = 3 \\ d(2, x) = \sqrt{(-118)^2 + 2^2 + 0^2} = 118.01 \\ d(3, x) = \sqrt{(-113)^2 + 7^2 + 0^2} = 113.21 \\ d(4, x) = \sqrt{73^2 + 1^2 + 1^2} = 73.01 \\ d(5, x) = \sqrt{132^2 + 1^2 + 0^2} = 132.003 \\ d(6, x) = \sqrt{5^2 + 1^2 + 1^2} = 5.19 \end{array} \right\} C = 1 \quad \text{⌚️ Previsione plausibile}$$

Note

RAPPRESENTAZIONE DELLA CONOSCENZA PERCHÉ PRE-PROCESSARE I DATI?



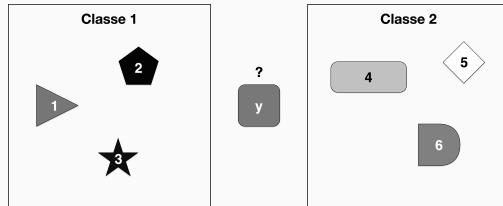
→ Con $y = [120, 4, 1]$

$$\left. \begin{array}{l} d(1, y) = \sqrt{0^2 + (-1)^2 + (-1)^2} = 1.41 \\ d(2, y) = \sqrt{(-115)^2 + 1^2 + (-1)^2} = 115.008 \\ d(3, y) = \sqrt{(-110)^2 + 6^2 + (-1)^2} = 110.16 \\ d(4, y) = \sqrt{76^2 + 0^2 + 0^2} = 73.01 \\ d(5, y) = \sqrt{135^2 + 0^2 + (-1)^2} = 135.003 \\ d(6, y) = \sqrt{8^2 + 0^2 + 0^2} = 8 \end{array} \right\} C = 1 \quad \text{Previsione non plausibile}$$

Note

RAPPRESENTAZIONE DELLA CONOSCENZA

PERCHÉ PRE-PROCESSARE I DATI?

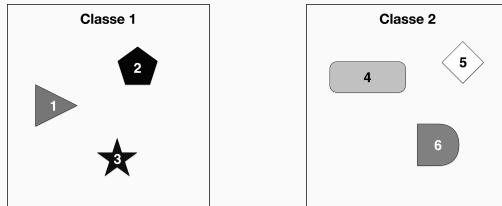


→ La feature colore maschera le altre due

$$\left. \begin{aligned} d(1, y) &= \sqrt{0^2 + (-1)^2 + (-1)^2} = 1.41 \\ d(2, y) &= \sqrt{(-115)^2 + 1^2 + (-1)^2} = 115.008 \\ d(3, y) &= \sqrt{(-110)^2 + 6^2 + (-1)^2} = 110.16 \\ d(4, y) &= \sqrt{76^2 + 0^2 + 0^2} = 73.01 \\ d(5, y) &= \sqrt{135^2 + 0^2 + (-1)^2} = 135.003 \\ d(6, y) &= \sqrt{8^2 + 0^2 + 0^2} = 8 \end{aligned} \right\} d(n, y) \approx y_1$$

Note

RAPPRESENTAZIONE DELLA CONOSCENZA PERCHÉ PRE-PROCESSARE I DATI?

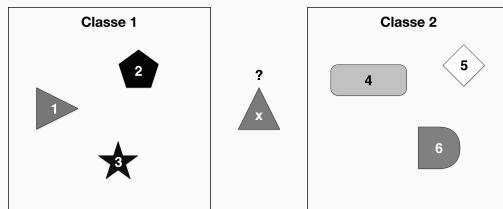


- Necessaria **normalizzazione** per rendere paragonabili tutte le *features* (es. $\bar{n}_i = \frac{n_i - n_{min}}{n_{max} - n_{min}}$)
 - Per colore ($n_{min} = 0, n_{max} = 255$), per numero vertici ($n_{min} = 3, n_{max} = 10$)

Elemento	Feature vector	Normalizzazione
1	[120, 3, 0]	[0.47, 0, 0]
2	[5, 5, 0]	[0.01, 0.28, 0]
3	[10, 10, 0]	[0.03, 1, 0]
4	[196, 4, 1]	[0.76, 0.14, 0]
5	[255, 4, 0]	[1, 0.14, 0]
6	[128, 4, 1]	[0.5, 0.14, 1]

Note

RAPPRESENTAZIONE DELLA CONOSCENZA PERCHÉ PRE-PROCESSARE I DATI?

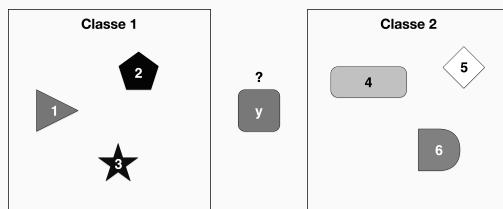


→ Con $x = [0.48, 0, 0]$

$$\left. \begin{aligned} d(1, x) &= \sqrt{(-0.01)^2 + 0^2 + 0^2} = 0.01 \\ d(2, x) &= \sqrt{(-0.47)^2 + 0.28^2 + 0^2} = 0.547 \\ d(3, x) &= \sqrt{(-0.45)^2 + 1^2 + 0^2} = 1.096 \\ d(4, x) &= \sqrt{0.28^2 + 0.14^2 + 1^2} = 0.49 \\ d(5, x) &= \sqrt{0.52^2 + 0.14^2 + 0^2} = 0.538 \\ d(6, x) &= \sqrt{0.02^2 + 0^2 + 0^2} = 0.02 \end{aligned} \right\} C = 1$$

Note

RAPPRESENTAZIONE DELLA CONOSCENZA PERCHÉ PRE-PROCESSARE I DATI?



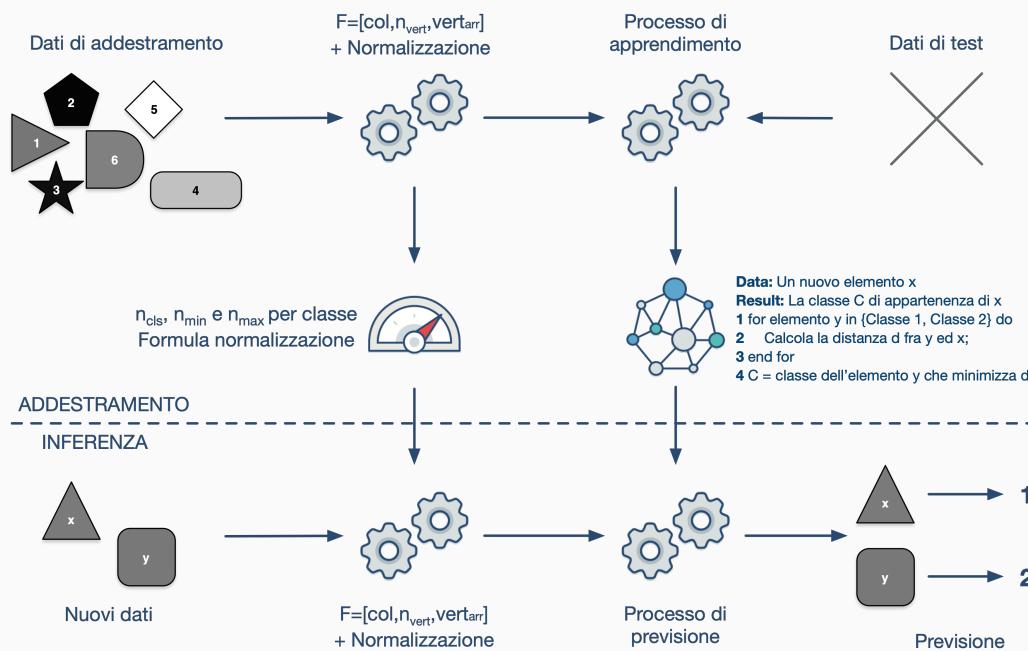
→ Con $y = [0.47, 0.14, 1]$

$$\left. \begin{aligned} d(1, y) &= \sqrt{0^2 + (-0.14)^2 + (-1)^2} = 1.009 \\ d(2, y) &= \sqrt{(-0.46)^2 + 0.14^2 + (-1)^2} = 1.109 \\ d(3, y) &= \sqrt{(-0.44)^2 + 0.86^2 + (-1)^2} = 1.73 \\ d(4, y) &= \sqrt{0.29^2 + 0^2 + 0^2} = 0.289 \\ d(5, y) &= \sqrt{0.53^2 + 0^2 + (-1)^2} = 1.131 \\ d(6, y) &= \sqrt{0.03^2 + 0^2 + 0^2} = \textcolor{red}{0.03} \end{aligned} \right\} C = 2$$

Note

WORKFLOW PROCESSO DI MACHINE LEARNING

ESEMPIO FORME GEOMETRICHE



©Simone Scannapieco

Note