Introduzione al Machine Learning

Come i computer imparano dall’esperienza

UNIVERSITÀ
DEGLI STUDI DI TRIESTE

ARTIFICIAL INTELLIGENCE
& DATA ANALYTICS

ai.units.it

Cos’è il Machine Learning?

“**Machine Learning** is the science of getting computer to learn without being explicitly programmed.”

Cosa può imparare un computer?

- Classificare
- Calcolare
- Predire
- Raggruppare e visualizzare
- Decidere

Supervised Learning

Supervised Learning: insegnare ad un computer a classificare o calcolare a partire da un insieme di esempi forniti da un insegnante.

Riconoscere la presenza di cani in un immagine

Identificare le email di spam

Prediction

Fare predizioni sul futuro o su scenari nuovi

Prevedere l'andamento della borsa

Predire il tempo atmosferico

Predire in anticipo l'insorgenza di una patologia

Unsupervised Learning

Unsupervised Learning: il computer identifica autonomamente informazione e struttura nei dati.

Identificare profili di marketing

Dare raccomandazioni sugli acquisti

Reinforcement Learning

Reinforcement Learning: il computer impara a prendere decisioni ottimali in condizioni di incertezza.

Vincere a giochi multiplayer

Guidare auto a guida autonoma

Un amico botanico

Un nostro amico ama i fiori di iris. Ci sono tre specie preferite: Iris setosa, Iris virginica e Iris versicolor.
Come possiamo riconoscere la specie in modo automatico?

Iris setosa

Iris verginica

Iris versicolor

I dati a disposizione

Abbiamo a disposizione alcune misurazioni della lunghezza e larghezza di sepali e petali.

Possiamo usarli per riconoscere il tipo di iris?

	sepal_length	sepal_width	petal_length	petal_width	species
116	6.5	3.0	5.5	1.8	virginica
23	5.1	3.3	1.7	0.5	setosa
88	5.6	3.0	4.1	1.3	versicolor
122	7.7	2.8	6.7	2.0	virginica
19	5.1	3.8	1.5	0.3	setosa
25	5.0	3.0	1.6	0.2	setosa

Uno sguardo ai dati

Uno sguardo ai dati

Risolviamo il problema a mano

Risolviamo il problema a mano

Risolviamo il problema a mano

Alberi di decisione

Radice dell'albero

Nodo interno:
decisione del tipo
petal_length > 2
sepal_width < 3.2

Foglia:
Scelta di una classe di iris

Alberi di decisione

Processo di decisione:

Fissa un datapoint

Ad ogni nodo
se la condizione è verificata, scendi a sinistra,
altrimenti a destra

Quando arrivi ad una foglia, rispondi con la classe corrispondente

Esempio:

petal_length	petal_width	species
5.5	1.8	virginica
1.7	0.5	setosa

Alberi di decisione: visualizzazione delle scelte

Alberi di decisione: come si costruiscono

Si parte da un dataset, e da un albero con solo la radice

1. Si sceglie la condizione migliore (che divide meglio le classi):

petal_length < 2.0

	sepal_length	sepal_width	petal_length	petal_width	species
116	6.5	3.0	5.5	1.8	virginica
23	5.1	3.3	1.7	0.5	setosa
88	5.6	3.0	4.1	1.3	versicolor
122	7.7	2.8	6.7	2.0	virginica
19	5.1	3.8	1.5	0.3	setosa
25	5.0	3.0	1.6	0.2	setosa

Alberi di decisione: come si costruiscono

Si parte da un dataset, e da un albero con solo la radice

2. Si divide il dataset in due gruppi: chi soddisfa la condizione e chi non la soddisfa

	sepal_length	sepal_width	petal_length	petal_width	species	
116	6.5	3.0	5.5	1.8	virginica	NO
23	5.1	3.3	1.7	0.5	setosa	SI
88	5.6	3.0	4.1	1.3	versicolor	NO
122	7.7	2.8	6.7	2.0	virginica	NO
19	5.1	3.8	1.5	0.3	setosa	SI
25	5.0	3.0	1.6	0.2	setosa	SI

Alberi di decisione: come si costruiscono

Si parte da un dataset, e da un albero con solo la radice

3. Si creano due nodi, uno a SX ed uno a DX, e si associa loro uno dei due gruppi (punti che soddisfano a SX)

	sepal_length	sepal_width	petal_length	petal_width	species	
116	6.5	3.0	5.5	1.8	virginica	NO
23	5.1	3.3	1.7	0.5	setosa	SI
88	5.6	3.0	4.1	1.3	versicolor	NO
122	7.7	2.8	6.7	2.0	virginica	NO
19	5.1	3.8	1.5	0.3	setosa	SI
25	5.0	3.0	1.6	0.2	setosa	SI

Alberi di decisione: come si costruiscono

Si parte da un dataset, e da un albero con solo la radice

4. Se in un nuovo nodo, tutti i datapoints sono della stessa classe, il nodo diventa una foglia

	sepal_length	sepal_width	petal_length	petal_width	species	
116	6.5	3.0	5.5	1.8	virginica	NO
23	5.1	3.3	1.7	0.5	setosa	SI
88	5.6	3.0	4.1	1.3	versicolor	NO
122	7.7	2.8	6.7	2.0	virginica	NO
19	5.1	3.8	1.5	0.3	setosa	SI
25	5.0	3.0	1.6	0.2	setosa	SI

Alberi di decisione: come si costruiscono

Si parte da un dataset, e da un albero con solo la radice

5. Altrimenti, si ripete il procedimento sul nodo, per la porzione di dati rimasta

	sepal_length	sepal_width	petal_length	petal_width	species	
116	6.5	3.0	5.5	1.8	virginica	SI
~~23~~	~~5.1~~	~~3.3~~	~~1.7~~	~~0.5~~	~~setosa~~	
88	5.6	3.0	4.1	1.3	versicolor	NO
122	7.7	2.8	6.7	2.0	virginica	SI
~~19~~	~~5.1~~	~~3.8~~	~~1.5~~	~~0.3~~	~~setosa~~	
~~25~~	~~5.0~~	~~3.0~~	~~1.6~~	~~0.2~~	~~setosa~~	

Alberi di decisione: come si costruiscono

Si parte da un dataset, e da un albero con solo la radice

6. Quando non ci sono più nodi interni, ci fermiamo

	sepal_length	sepal_width	petal_length	petal_width	species	
116	6.5	3.0	5.5	1.8	virginica	SI
~~23~~	~~5.1~~	~~3.3~~	~~1.7~~	~~0.5~~	~~setosa~~	
88	5.6	3.0	4.1	1.3	versicolor	NO
122	7.7	2.8	6.7	2.0	virginica	SI
~~19~~	~~5.1~~	~~3.8~~	~~1.5~~	~~0.3~~	~~setosa~~	
~~25~~	~~5.0~~	~~3.0~~	~~1.6~~	~~0.2~~	~~setosa~~	

Iris dataset

Il dataset del nostro amico ha 150 punti, tutti etichettati. Costruiamo l'albero guardando solo lunghezza e larghezza dei petali.

Iris dataset

Generalizzazione

Come faccio a capire se il mio albero funziona bene?

Generalizzazione: Come funziona il mio albero su dati che non ho ancora visto?

Si stima dividendo il dataset in train e test sets (70%-30%)

Generalizzazione

Come faccio a capire se il mio albero funziona bene?

Generalizzazione:
Si misura con la percentuale di punti del test set predetti correttamente

```
Test accuracy: 0.96
```

Overfitting

Costruiamo l'albero con lunghezza e larghezza del sepalo

```
Train accuracy: 0.99
Test accuracy: 0.68
```

Qui le cose vanno malino: l'albero è troppo calibrato sul dataset di train e generalizza male

Questo è **overfitting**

Pruning

L’albero di decisione è troppo complesso: questo causa overfitting.

Possiamo combattere questo comportamento usando il rasoio di Occam.

Possiamo per esempio limitare la profondità dell’albero.

Pruning

Profondità limitata a 3

Questa è una forma di **regolarizzazione**, che diminuisce il gap tra errore di train e di test

Train accuracy: 0.82 ↓

Test accuracy: 0.74 ↑

Pruning

Profondità limitata a 3

Questa è una forma di **regolarizzazione**, che diminuisce il gap tra errore di train e di test

`Train accuracy: 0.82` ↓

`Test accuracy: 0.74` ↑

Pruning

Perchè non potare di più? Profondità limitata a 1!

`Train accuracy: 0.65`

`Test accuracy: 0.58`

Il modello è troppo debole!

La saggezza delle folle

Gli alberi di profondità 1 sono troppo deboli.
Ma se invece di costruire un albero ne costruissimo tanti?
Ognuno consapevole di una porzione diversa dei dati e delle features?

Su un nuovo dato, ogni albero vota, e la maggioranza vince…

Foreste Casuali

Ogni albero è costruito su una porzione di dataset e features scelta in modo casuale.
E quindi abbiamo una foresta di alberi casuali…

```
Train accuracy: 0.95

Test accuracy: 0.96
```

Con 100 alberi di profondità 1 (e tutte le features) raggiungiamo una test accuracy del 96%

Ci sono più cose in cielo ed in terra…

Oggi abbiamo visto gli alberi di decisione (un ponte tra AI simbolica e statistica) e accennato alle foreste casuali (un tipico metodo ad ensemble di tipo bagging).

Ci sono moltissime altre tecniche di machine learning, ognuna con le sue forze e debolezze.

Comprenderle richiede solide basi di matematica, probabilità, statistica ed algoritmica.

Gli alberi di decisione appaiono a pagina 543 di 1096…

Intelligenza Artificiale Moderna

Informatica | Matematica | Statistica | Fisica

UNIVERSITÀ
DEGLI STUDI DI TRIESTE

ai.units.it

ARTIFICIAL INTELLIGENCE
& DATA ANALYTICS