

Introduction to machine learning

Giacomo Fantoni

Telegram: @GiacomoFantoni

Github: <https://github.com/giacThePhantom/intro2ml>

7 aprile 2021

Indice

1	Introduzione	2
1.1	Definizioni	2
1.2	Processo	2
1.2.1	Il processo di apprendimento	2
1.3	Modello	2
1.4	Deep learning	3
2	Machine learning basics	4
2.1	Introduzione	4
2.1.1	Processo di learning	4
2.2	Dati	4
2.2.1	Training e test set	4
2.3	Task	5
2.4	Modello	5
2.4.1	Target ideale	5
2.4.2	Target feasible	5
2.4.3	Target attuale	5
2.4.4	Funzione di errore	6
2.4.5	Tipi di errore	6
2.4.6	Stimare l'errore di generalizzazione	6
2.5	Tipi di learning	6
2.5.1	Supervised learning	6
2.5.2	Unsupervised learning	7
2.5.3	Reinforcement learning	8

Capitolo 1

Introduzione

1.1 Definizioni

Si intende per machine learning lo studio di algoritmi che migliorano autonomamente attraverso l'esperienza. È un campo dell'intelligenza artificiale. Stravolge il paradigma convenzionale della programmazione: un algoritmo di machine learning infatti prende come input un insieme di dati e risultati in modo da produrre un programma che fornisce un risultato appropriato. Coinvolge pertanto la scoperta automatica di regolarità nei dati attraverso algoritmi in modo da poter compiere azioni basate su di essi.

1.2 Processo

Il machine learning permette ai computer di acquisire conoscenza attraverso algoritmi che inferiscono e imparano da dati. Questa conoscenza viene rappresentata da un modello che può essere utilizzato su nuovi dati.

1.2.1 Il processo di apprendimento

Il processo di apprendimento in particolare coinvolge diversi passaggi:

- Acquisizione dei dati dal mondo reale attraverso dispositivi di misurazione come sensori o database.
- Preprocessamento dei dati: filtraggio del rumore, estrazione delle feature e normalizzazione.
- Riduzione dimensionale: selezione e proiezione delle feature.
- Apprendimento del modello: classificazione, regressione, clustering e descrizione.
- Test del modello: cross-validation e bootstrap.
- Analisi dei risultati.

1.3 Modello

Un algoritmo di machine learning impara dall'esperienza E in rispetto di una classe di compiti T e di misurazione delle performance P , se la P di T aumenta con E . Si nota pertanto come un compito

di machine learning ben definito possiede una tripla:

$$\langle T, P, E \rangle$$

1.4 Deep learning

Il deep learning è un sottoinsieme del machine learning che permette a modelli computazionali composti di multipli strati di imparare la rappresentazione di dati con multipli livelli di astrazione. Si utilizza pertanto una rete neurale con diversi strati di nodi tra input e output. Questa serie di strati tra input e output computa caratteristiche rilevanti automaticamente in una serie di passaggi. Questi algoritmi sono resi possibili da:

- Enorme mole di dati disponibili.
- Aumento del potere computazionale.
- Aumento del numero di algoritmi di machine learning e della teoria sviluppata dai ricercatori.
- Aumento del supporto dall'industria.

Capitolo 2

Machine learning basics

2.1 Introduzione

Il machine learning permette ai computer di acquisire conoscenza attraverso algoritmi che imparano e inferiscono dai dati. Tale conoscenza viene rappresentata da un modello che viene poi utilizzato su dati futuri.

2.1.1 Processo di learning

Si individua un processo di learning:

- Acquisizione di dati dal mondo reale attraverso sensori.
- Preprocessamento dei dati: eliminazione del rumore, estrazione delle features e normalizzazione.
- Riduzione di dimensionalità attraverso selezione e proiezione di features.
- Learning del modello: classification, regression, clustering e description.
- Test del modello attraverso cross-validation e bootstrap.
- Analisi dei risultati.

2.2 Dati

I dati disponibili ad un algoritmo di machine learning sono tipicamente un insieme di esempi. Questi esempi sono tipicamente rappresentati come un array di features, caratteristiche dei dati di interesse per lo studio in atto.

2.2.1 Training e test set

In particolare per questi algoritmi si assume sempre che il training e il test set siano distribuiti secondo variabili indipendenti e identicamente distribuite (*i.i.d*) La distribuzione P_{data} è tipicamente sconosciuta ma si può campionare, attraverso un modello probabilistico di learning. In particolare la distribuzione di probabilità di coppie di esempio e label viene detta data generating distribution e sia il training data che il test set sono generati basandosi su di essa.

2.3 Task

Si intende per task una rappresentazione del tipo di predizione che viene svolta per risolvere un problema su dei dati. Viene identificata con un insieme di funzioni che possono potenzialmente risolverla. In generale consiste di una funzione che assegna ogni input $x \in \mathcal{X}$ a un output $y \in \mathcal{Y}$:

$$f : \mathcal{X} \rightarrow \mathcal{Y} \quad \mathcal{F}_{task} \subset \mathcal{Y}^{\mathcal{X}}$$

La natura di $\mathcal{X}, \mathcal{Y}, \mathcal{F}_{task}$ dipende dal tipo di task.

2.4 Modello

Un modello è un programma per risolvere un problema. È cioè l'implementazione di una funzione $f \in \mathcal{F}_{task}$ che può essere computata. UN insieme di modelli formano uno spazio di ipotesi:

$$\mathcal{H} \subset \mathcal{F}_{task}$$

L'algoritmo cerca una soluzione nello spazio di ipotesi.

2.4.1 Target ideale

Il target ideale del modello è quello di minimizzare una funzione di errore (generalizzazione)

$$E(f; P_{data})$$

Questa funzione determina quanto bene una soluzione $f \in \mathcal{F}_{task}$ fitta dei dati. Guida pertanto la selezione della migliore soluzione in \mathcal{F}_{task} . Pertanto:

$$f^* \in \arg \min_{f \in \mathcal{F}_{task}} E(f; P_{data})$$

2.4.2 Target feasible

Si deve restringere il focus sul trovare funzioni che possono essere implementate e valutate in maniera trattabile. Si definisce pertanto uno spazio di ipotesi del modello $\mathcal{H} \subset \mathcal{F}_{task}$ e si cerca la soluzione all'interno di quello spazio:

$$f_{\mathcal{H}}^* \in \arg \min_{f \in \mathcal{H}} E(f; P_{data})$$

Si noti come questa funzione non possa essere computata correttamente in quanto P_{data} è sconosciuta.

2.4.3 Target attuale

Per trovare il target attuale si deve lavorare su un campione di dati o il training set

$$\mathcal{D}_n = \{z_1, \dots, z_n\}$$

Dove

$$z_i = (x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$$

$$z_i \sim P_{data}$$

Pertanto in:

$$f_{\mathcal{H}}^* \in \arg \min_{f \in \mathcal{H}} E(f; P_{data})$$

$E(f; P_{data})$ è il training error.

2.4.4 Funzione di errore

Le funzioni di generalizzazione e di training error possono essere scritte in termini di una pointwise loss $\downarrow(f; z)$ che misura l'errore che avviene a f su un esempio di training z .

$$E(f; P_{data}) = \mathbb{E}_{z \sim P_{data}} [\downarrow(f; z)]$$
$$E(f; \mathcal{D}_n) = \frac{1}{n} \sum_{i=1}^n \downarrow(f; z_i)$$

Si nota pertanto come l'algoritmo di learning risolve il problema di ottimizzazione con target:

$$f_{\mathcal{H}}^*(\mathcal{D}_n)$$

2.4.5 Tipi di errore

- Overfitting.
- Estimation error, indotto imparando da un campione di dati.
- Approximation error, indotto dallo spazio di ipotesi \mathcal{H} .
- Irreducible error a causa della variabilità intrinseca.

2.4.6 Stimare l'errore di generalizzazione

L'errore di generalizzazione può essere stimato utilizzando diversi insiemi di training, validation e test.

2.4.6.1 Migliorare la generalizzazione

La generalizzazione può essere migliorata:

- Evitando di ottenere il minimo sul training error.
- Riducendo la capacità del modello.
- Cambiando l'obiettivo con un termine di regolarizzazione.
- Iniettando rumore nell'algoritmo.
- Fermando l'algoritmo prima che converga.
- Aumentando la quantità di dati.
- Aggiungendo più campioni di training.
- Aumentando il training set con trasformazioni.
- Combinando predizioni da più modelli decorrelati o ensembling.

2.4.6.1.1 Regolarizzazione Si intende per regolarizzazione la modifica della funzione di training error con un termine $\Omega(f)$ che penalizza soluzioni complesse:

$$E_{reg}(f; \mathcal{D}_n) = E(f; \mathcal{D}_n) + \lambda_n \Omega(f)$$

2.5 Tipi di learning

2.5.1 Supervised learning

Nel supervised learning vengono dati in input a un modello o predittore un insieme di esempi che possiedono una label. Il modello poi impara a creare delle predizioni su un nuovo esempio.

2.5.1.1 Dati

Nel caso del supervised learning i dati creano una distribuzione:

$$p_{data} \in \Delta(\mathcal{X} \times \mathcal{Y})$$

2.5.1.2 Classificazione

In un problema di classificazione si trova un insieme finito di label discrete. In particolare dato un training set $\mathcal{T} = \{(x_1, u_1), \dots, (x_m, y_m)\}$, si deve imparare una funzione f per predire y dato x . f sarà pertanto:

$$f : \mathbb{R}^d$$

Dove d è la dimensionalità di x e k il numero di labels distinte.

2.5.1.2.1 Task Si deve pertanto trovare una funzione $f \in \mathcal{Y}^{\mathcal{X}}$ che assegna ogni input $x \in \mathcal{X}$ a una label discreta.

$$f(x) \in \mathcal{Y} = \{c_1, \dots, c_k\}$$

2.5.1.3 Regression

Un problema di regressione presenta un insieme di label continue. Dato un training set $\mathcal{T} = \{(x_1, y_1), \dots, (x_m, y_m)\}$, si deve imparare una funzione f per predire y dato x . f sarà pertanto:

$$f : \mathbb{R}^d \rightarrow \mathbb{R}$$

Dove d è la dimensionalità di x .

2.5.1.3.1 Task Si deve trovare una funzione $f(x) \in \mathcal{Y}$ che assegna ogni input a una label continua.

2.5.1.4 Ranking

Il ranking è un tipo particolare di classificazione in cui una label è un ranking.

2.5.2 Unsupervised learning

Nel unsupervised learning vengono dati in input a un modello o predittore un insieme di esempi senza label. Il modello impara a creare delle predizioni su un nuovo esempio.

2.5.2.1 Dati

Nel caso del supervised learning i dati creano una distribuzione:

$$p_{data} \in \Delta(\mathcal{X})$$

2.5.2.2 Clustering

Nel clustering, data $\mathcal{T} = \{x_1, \dots, x_m\}$ si deve trovare la struttura nascosta che intercorre tra le x o i clusters.

2.5.2.2.1 Task Si deve trovare una funzione $f \in \mathbb{N}^{\mathcal{X}}$ che assegna ogni input $x \in \mathcal{X}$ a un indice di cluster $f(x) \in \mathbb{N}$. Tutti i punti mappati sullo stesso indice formano un cluster.

2.5.2.3 Dimensionality reduction

Nella dimensionality reduction si tenta di ridurre il numero di variabili sotto considerazione ottenendo un insieme di variabili principali.

2.5.2.3.1 Task Si deve trovare una funzione $f \in \mathcal{Y}^{\mathcal{X}}$ che mappa ogni input di molte dimensioni $x \in \mathcal{X}$ a un output a dimensione minore $f(x) \in \mathcal{Y}$, dove $\dim(\mathcal{Y}) \ll \dim(\mathcal{X})$

2.5.3 Reinforcement learning

Nel reinforcement learning un agente impara dall'ambiente interagendo con esso e ricevendo premi per lo svolgimento di azioni particolari. In particolare, data una sequenza di esempi o stati e una reward dopo il completamento di tale sequenza si impara a predire l'azione da svolgere per uno stato o esempio individuale.