

Deep learning con Python (autore: Vittorio Albertoni)

Premessa

Abbiamo machine learning (apprendimento automatico) quando un computer, analizzando le caratteristiche di un set di dati o le interrelazioni presenti in un set di dati, elabora autonomamente modelli in grado di riprodurre o prevedere le variazioni dei dati stessi al variare dello scenario di riferimento.

Si tratta di una importante branca dell'intelligenza artificiale, strettamente collegata alla statistica.

Possiamo capire meglio di che cosa stiamo parlando se ripercorriamo le definizioni di talune sfaccettature del machine learning dal punto di vista del tipo di analisi.

Possiamo avere un set di dati costituito da dati relazionati tra di loro e l'apprendimento consiste nel capire quale sia questa relazione in modo da poterla applicare a dati che non fanno parte del set stesso. In questo caso si parla di apprendimento supervisionato.

E' il caso, per esempio, della regressione, dove si stabilisce il più probabile legame funzionale tra una o più variabili indipendenti e una variabile dipendente presenti in un data set, in modo da trovare un modello che ci consenta di prevedere quale sia il più probabile valore della variabile dipendente in corrispondenza a variabili indipendenti non contemplate nel data set. Altro esempio quello della classificazione, dove, data una certa etichettatura attribuita ad una certa conformazione di dati nel data set, se ne ricavano elementi per etichettare casistiche non contemplate nel data set stesso. Per esempio per classificare come truffaldino un messaggio di posta elettronica dopo aver assimilato quali siano le caratteristiche di una certa serie di messaggi classificati truffaldini in un data set.

A volte il data set contiene semplicemente dei dati, non relazionati tra loro e non etichettati e, in questo caso, l'apprendimento consiste nel capire talune caratteristiche di questi dati. In questo caso si parla di apprendimento non supervisionato.

E', per esempio, il caso del clustering, dove si cerca di capire come i dati del data set possano essere raggruppati in sotto-insiemi di dati che presentino tra di loro determinate omogeneità.

Per estensione, infine, rientra nel machine learning anche l'apprendimento che avviene senza che vi sia alcun dato per il condizionamento. E' il caso dell'apprendimento con rinforzo, dove l'apprendimento avviene attraverso un meccanismo di ricompensa (per incentivare il compimento di azioni virtuose) e penalità (per scoraggiare azioni indesiderate) in un contesto di interazione con l'ambiente esterno.

E' per esempio ciò che avviene per far funzionare gli assistenti al parcheggio presenti su certe vetture. In questo caso l'apprendimento avviene attraverso i sensori montati sulla vettura che individuano la presenza di ostacoli al parcheggio e attraverso la penalizzazione delle azioni tendenti alla collisione con questi o la ricompensa per le azioni tendenti a schivarli.

Dal punto di vista del metodo con cui vengono effettuate analisi e apprendimento possiamo utilizzare altre definizioni.

Normalmente, fino a che lavoriamo su dataset composti da numeri o da stringhe di caratteri, le metodologie con cui abbiamo a che fare sono quelle dell'algebra lineare, che sfrutta proprietà e operazioni su vettori e matrici, dell'analisi matematica, con lo studio di funzione e di ottimizzazione, e del calcolo delle probabilità.

Se il dataset è composto da immagini, suoni o video dobbiamo ricorrere a metodologie diverse, dove in luogo dei soliti vettori a una o due dimensioni abbiamo vettori di dimensione superiore, detti tensori, e dove la modellizzazione di funzioni complesse può convenientemente avvenire utilizzando reti neurali artificiali.

Ed è qui che il machine learning diventa deep learning.

Indice

1	Cosa ci serve	3
2	Reti neurali artificiali	3
3	Strumenti	4
3.1	Scikit-learn	4
3.2	Theano	4
3.3	TensorFlow	4
3.4	PyTorch	5

1 Cosa ci serve

Nel dicembre 2021, sul mio blog all'indirizzo www.vittal.it, ho pubblicato l'articolo «Python per la Data Science» con allegato un manualetto in formato PDF nel quale ho presentato alcuni moduli del linguaggio Python concepiti per trattare e fare machine learning con dati numerici o testuali.

In quel manualetto troviamo un esempio di apprendimento supervisionato, un esempio di apprendimento non supervisionato e un esempio di sentiment analysis su dati testuali utilizzata per instaurare un piccolo dialogo con il computer in cui il computer si esprime tenendo conto di ciò che gli si dice.

Piccole cose che possiamo fare con un computer qualsiasi di media potenza. Ma già in queste piccole cose c'è il germe dell'intelligenza artificiale, anche se l'apprendimento del computer è piuttosto superficiale e non profondo.

Con il deep learning i metodi di analisi dei dati e di apprendimento vanno più in profondità e si avvalgono dello strumento della rete neurale artificiale, accorgimento informatico che imita il funzionamento del cervello umano.

Una rete neurale artificiale, come le applicazioni tradizionali per computer, è un software. Il relativo codice, anche se scritto con gli stessi linguaggi usati per sviluppare le applicazioni tradizionali, non è un insieme di istruzioni fisse che permettono ai programmi di elaborare richieste specifiche in modo standard. Esso è costituito da algoritmi che permettono alle reti di imparare quali output fornire a fronte di input che possono essere molto grandi, complessi e inattesi.

La grande comunità di Python ha prodotto e produce moduli di questo software innovativo e, nei successivi capitoli, ne propongo una carrellata.

Si tratta di moduli che possiamo caricare sul nostro computer con il solito pip.

Dal momento, tuttavia, che il funzionamento di questo tipo di software richiede spesso una notevole capacità elaborativa e normalmente diventa difficile o impossibile utilizzarlo su computer casalinghi da dilettanti principianti, suggerisco un primo avvicinamento ad esso attraverso Google Colaboratory.

A chi non conosca questo formidabile strumento suggerisco la lettura del manualetto «google_colab» allegato all'articolo «Se il nostro computer non ce la fa» che ho pubblicato sul mio blog all'indirizzo www.vittal.it lo scorso maggio.

2 Reti neurali artificiali

Il neurone è l'unità cellulare del sistema nervoso in grado di ricevere, elaborare e trasmettere impulsi nervosi.

L'elaborazione consiste nella "somma pesata" di segnali in ingresso: se il risultato supera un certo valore di soglia il neurone si attiva e genera un potenziale di azione, se il risultato rimane sotto un certo valore di soglia il neurone rimane in uno stato di riposo.

Un gruppo di neuroni che si influenzano a vicenda grazie alle connessioni che li collegano per svolgere una determinata funzione fisiologica è una rete neurale.

Come l'informatica abbia trovato modo di imitare tutto ciò e come tutto ciò si possa implementare utilizzando il linguaggio di programmazione Python comporta un notevole sforzo di comprensione. A chi si voglia cimentare suggerisco i Capitoli 2 e 12 del testo di Sebastian Raschka, Machine Learning con Python, edito da Apogeo.

Le reti neurali artificiali vengono utilizzate in applicazioni nei campi più disparati, dalla finanza alla medicina, dalla sicurezza alla multimedialità. Soprattutto in applicazioni ove vi sia necessità di riconoscere ed elaborare immagini, riconoscere voci e interpretare il parlato.

Nell'allegato «impro_visor.pdf» al mio articolo «Intelligenza artificiale per l'improvvisazione musicale» pubblicato nel settembre 2018 sul mio blog all'indirizzo www.vittal.it si trova la presentazione di un software musicale che utilizza il deep learning per generare improvvisazioni jazzistiche secondo lo stile di un certo interprete.

Le reti neurali artificiali non sono da considerare sempre un'alternativa alle applicazioni tradizionali.

Ha senso usarle quando la complessità dei dati e le difficoltà di elaborazione renderebbero inefficace o impossibile ricorrere a metodi di analisi tradizionale, che sono comunque sempre efficaci quando i dati da elaborare sono ben chiari e di natura numerica, anche se la mole dei dati da elaborare è molto consistente.

3 Strumenti

La comunità Python ha elaborato molti strumenti per il deep learning e ne propongo una rassegna con utili indicazioni per chi voglia approfondire la conoscenza e fare qualche prova.

3.1 Scikit-learn

Rilasciato nel 2007, è il modulo Python più anziano e completo per il machine learning.

E' basato sui moduli NumPy, SciPy e Matplotlib.

All'indirizzo <https://scikit-learn.org/stable/> troviamo tutto ciò che c'è da sapere su Scikit-learn.

Scegliendo la pagina USER GUIDE possiamo consultare un'ottima guida in lingua inglese, ricca di esemplificazioni.

In questa guida sono dedicati al deep learning i capitoli NEURAL NETWORK MODELS delle sezioni SUPERVISED LEARNING e UNSUPERVISED LEARNING.

Veniamo preavvisati sul fatto che l'implementazione deep learning di Scikit-learn non è pensata per applicazioni su larga scala, in quanto Scikit-learn non offre supporto per elaborazioni GPU o TPU.

3.2 Theano

A questa debolezza di Scikit-learn supplisce il modulo Theano, che, rilasciato nel 2008, si autodefinisce il nonno dei moduli Python dedicati al deep learning.

Basato sui moduli NumPy e SciPy, con sintassi molto simile a quella di NumPy introduce la possibilità di operare con array multidimensionali (tensori) alla base della costruzione di reti neurali, che, a differenza di quelli producibili con NumPy, possono essere elaborati sia su CPU sia su GPU.

All'indirizzo

<https://www.projectpro.io/data-science-in-python-tutorial/theano-deep-learning-tutorial/> troviamo un'ottima descrizione di cosa possiamo fare con Theano.

Purtroppo, nel momento in cui scrivo (giugno 2023), la versione di Theano disponibile è la 1.0.5 e non funziona con versioni di Python superiori alla 3.8.

Dal momento che Google Colab utilizza ormai versioni Python 3.9 non ci offre attualmente la possibilità di provare Theano con questo strumento.

Per fare qualche prova possiamo tuttavia installarlo sul nostro computer casalingo con pip e farlo girare normalmente su CPU, rinunciando a sperimentarne la grande velocità con cui funziona girando su GPU.

Siamo sempre nel campo degli strumenti di machine learning che fanno anche deep learning.

3.3 TensorFlow

E' la prima libreria open source di calcolo numerico pensata e prodotta per il deep learning.

Creata e rilasciata nel 2015 da Google è utilizzabile con i linguaggi Python e Javascript, attraverso i quali si creano applicazioni che vengono eseguite in C++ ad alta prestazione.

Il suo utilizzo con il linguaggio Python è agevolato dal frontend **Keras**, rilasciato nel 2015 dalla comunità Python.

All'indirizzo <https://www.tensorflow.org/overview?hl=it> troviamo tutto ciò che c'è da sapere su TensorFlow e non solo.

Seguendo quanto ci viene proposto nelle pagine TUTORIAL e GUIDA siamo infatti introdotti al deep learning in maniera che più facile non si può, con molti esempi di ciò che si può fare con TensorFlow e Keras.

Tutti questi esempi, di cui si può copiare il codice negli appunti, possono essere convenientemente eseguiti utilizzando Google Colab.

Anche con tutte queste agevolazioni imparare a lavorare con TensorFlow è molto difficile.

3.4 PyTorch

Ottima alternativa a TensorFlow, rispetto al quale è un tantino più facile, rilasciata dalla comunità Python nel 2016.

E' una libreria Python basata su Torch, a sua volta scritta con i linguaggi Lua, C e C++, dotata di due moduli: `torch` per trattare tensori (array multidimensionali) e `nn` per creare reti neurali.

Per lavorare con dati audio e video possiamo disporre dei moduli `torchaudio` e `torchvideo`.

Lavorando con Google Colab abbiamo accesso a tutto quanto: basta importare quel che serve con `import`.

Troviamo tutto ciò che riguarda PyTorch all'indirizzo <https://pytorch.org/>

Nella pagina TUTORIAL abbiamo interessanti esercitazioni che possiamo far funzionare utilizzando Google Colab e nella pagina DOCS abbiamo la documentazione ufficiale.

Ai principianti suggerisco il corso accelerato in lingua italiana che si trova all'indirizzo <https://www.intelligenzaartificialeitalia.net/post/deep-learning-con-python-e-pytorch-la-guida-pratica>