Introduzione

Probabilmente non ho bisogno di dirvi che il machine learning è una delle tecnologie più stimolanti della nostra epoca. Tutte le grandi aziende, fra cui Google, Facebook, Apple, Amazon, IBM e molte altre ancora investono grandi somme nella ricerca e nelle applicazioni nell'ambito del machine learning e questo per ottimi motivi. Anche se può sembrare che il "machine learning" sia uno degli elementi chiave per comprendere i nostri tempi, certamente il termine non è sulla bocca di tutti. Questo interessante campo apre la via a nuove possibilità ed è diventato un elemento indispensabile nella nostra vita quotidiana. Interviene ogni volta che parliamo con l'assistente vocale dello smartphone, che un sito invia consigli per gli acquisti ai clienti, viene bloccata una truffa su una carta di credito, viene eliminato un messaggio di spam dalle caselle di posta elettronica, viene diagnosticata una malattia. L'elenco sarebbe potrebbe continuare a lungo.

Se volete occuparvi di machine learning, se volete imparare a risolvere al meglio i problemi o anche considerare una carriera nell'ambito del machine learning, questo libro fa proprio per voi, ma per chi è alle prime armi, i concetti teorici su cui si basano gli algoritmi di machine learning potrebbero risultare piuttosto ostici. Tuttavia, nel corso degli ultimi anni sono stati pubblicati ottimi libri pratici, utili per iniziare la pratica del machine learning implementando potenti algoritmi di apprendimento. È mia opinione che la presenza di esempi pratici di codice sia fondamentale. Il codice presentato illustra i concetti, consentendo di mettere all'opera il materiale appreso. Tuttavia, considerate anche il fatto che: da un grande potere derivano grandi responsabilità! I concetti su cui si basa il machine learning sono troppo importanti per rimanere nascosti in una scatola. La mia personale missione è quella di fornirvi un libro differente, un libro nel quale troverete tutti i dettagli relativi ai concetti su cui si basa il machine learning e che offre spiegazioni intuitive e rigorose sul funzionamento degli algoritmi di machine learning, sul loro uso e, soprattutto, sul modo in cui evitare gli errori più comuni.

Se digitate i termini "machine learning" in Google Scholar, otterrete l'impressionante numero di 3.600.000 pubblicazioni. Naturalmente, potremmo anche trattare ogni singolo dettaglio dei vari algoritmi e delle relative applicazioni che sono emersi nel corso degli ultimi sessant'anni. Tuttavia, in questo libro, ci imbarcheremo in un interessante viaggio che tratterà gli argomenti e i concetti essenziali necessari per iniziare a comprendere questa branca dell'informatica. Se sentite che la vostra sete di conoscenza non è ancora soddisfatta, vi troverete molte altre risorse utili, che potrete utilizzare per approfondire questi argomenti.

Se avete già alle spalle qualche studio teorico nel campo del machine learning, questo libro vi aiuterà a mettere in pratica le vostre conoscenze. Se avete già impiegato tecniche di machine learning e volete capire qualcosa di più sul modo in cui funzionano tali algoritmi, questo libro per voi! Ma non preoccupatevi anche se siete completamente all'oscuro dell'argomento: avete ancora più ragioni per interessarvi. Vi prometto che il machine learning cambierà il modo in cui rifletterete sulla soluzione dei problemi e vi mostrerà come affrontarli, sfruttando tutte le potenzialità insite nei dati.
Prima di addentrarci nel campo del machine learning, cominciamo a rispondere alla domanda più importante. Perché Python?
La risposta è semplice: si tratta di un linguaggio potente e anche accessibile. Python è diventato il linguaggio di programmazione più diffuso per l'elaborazione dei dati, poiché consente di trascurare tutta la parte noiosa della programmazione, offrendo un ambiente in cui è possibile mettere in pratica le idee e vedere in azione i concetti.
Riflettendo sul mio personale percorso professionale, posso davvero dire che lo studio del machine learning mi ha reso migliore come scienziato, pensatore e risolutore di problemi. In questo libro vorrei condividere questa conoscenza con voi. La conoscenza si acquisisce con l'apprendimento e la chiave per farlo è il nostro entusiasmo, ma la padronanza dei concetti può arrivare solo con la pratica. La strada che avete davanti può essere talvolta irta di ostacoli e alcuni argomenti possono essere più complessi di altri, ma spero che cogliate questa opportunità e vi concentriate sulla ricca ricompensa. Ricordate che in questo viaggio siamo imbarcati insieme: nel corso di questo libro aggiungerò al vostro arsenale tante tecniche potenti, che vi aiuteranno a risolvere anche i problemi più difficili, sulla base dei dati disponibili.

Struttura del libro

Il Capitolo 1, *Dare ai computer la capacità di apprendere dai dati*, introduce i principali aspetti dell'uso di tecniche di machine learning per affrontare vari tipi di problemi. Inoltre, si occupa dei passi essenziali per la creazione di un tipico modello di machine learning, costruendo la catena (pipeline) che vi guiderà nel corso dei capitoli successivi.
Il Capitolo 2, *Addestrare gli algoritmi a compiti di classificazione*, torna alle origini del machine learning e introduce i classificatori binari perceptron e i neuroni lineari adattativi. Questo capitolo rappresenta una semplice introduzione alle basi della classificazione e si concentra sull'interazione esistente fra gli algoritmi di ottimizzazione e il machine learning.
Il Capitolo 3, *I classificatori di machine learning di scikit-learn*, descrive i principali algoritmi di machine learning rivolti alla classificazione, fornendo esempi pratici che sfruttano le funzionalità di una delle più note e ricche librerie di machine learning open source: scikit-learn.
Il Capitolo 4, *Costruire buoni set di addestramento: la pre-elaborazione*, insegna a gestire i più comuni problemi nei dataset grezzi, per esempio i dati mancanti. Inoltre discute vari approcci per identificare le caratteristiche più informative nei dataset e insegna a preparare le variabili di vari tipi, in modo che rappresentino input corretti per gli algoritmi di machine learning.
Il Capitolo 5, *Compressione dei dati tramite la riduzione della dimensionalità*, descrive le tecniche essenziali per ridurre il numero di caratteristiche di un dataset a insiemi più piccoli, mantenendo la maggior parte delle informazioni utili e discriminanti. Discute l'approccio

standard alla riduzione della dimensionalità tramite l'analisi del componente principale, confrontandolo con le tecniche di trasformazione con supervisione e non lineari.
Il Capitolo 6, *Valutazione dei modelli e ottimizzazione degli iperparametri*, discute i pro e i contro della stima delle prestazioni nei modelli predittivi. Inoltre discute varie metriche per la misurazione delle prestazioni dei modelli e le tecniche di ottimizzazione degli algoritmi di machine learning.
Il Capitolo 7, *Combinare più modelli: l'apprendimento d'insieme*, introduce vari concetti che prevedono di combinare efficacemente più algoritmi di apprendimento. Insegna a realizzare insiemi di "esperti" per superare i punti deboli dei singoli algoritmi, ottenendo previsioni più accurate e affidabili.
Il Capitolo 8, *Tecniche di machine learning per l'analisi del sentiment*, si occupa dei passi essenziali necessari per trasformare i dati testuali in rappresentazioni significative tramite algoritmi di machine learning, con lo scopo di prevedere l'opinione del pubblico sulla base dei testi che produce.
Il Capitolo 9, *Embedding di un modello in un'applicazione web*, prosegue la discussione sul modello predittivo del capitolo precedente e accompagna attraverso i passi essenziali dello sviluppo di applicazioni web con modelli di machine learning embedded.
Il Capitolo 10, *Previsioni di variabili target continue: l'analisi a regressione*, discute le tecniche essenziali di modellazione delle relazioni lineari fra variabili target e variabili di risposta, per effettuare previsioni su una scala continua. Dopo aver introdotto vari modelli lineari, parla di regressione polinomiale e approcci basati su una struttura ad albero.
Il Capitolo 11, *Lavorare con dati senza etichette: l'analisi a cluster*, sposta l'attenzione su una branca diversa del machine learning: l'apprendimento senza supervisione. Applichiamo gli algoritmi di tre diverse famiglie di algoritmi a clustering, per trovare gruppi di oggetti che condividono un certo grado di similarità.
Il Capitolo 12, *Reti neurali artificiali per il riconoscimento delle immagini*, estende i concetti dell'ottimizzazione basata su gradienti che abbiamo introdotto nel Capitolo 2, *Addestrare gli algoritmi a compiti di classificazione*, per realizzare potenti reti neurali multilivello, sulla base di algoritmi a retropropagazione.
Il Capitolo 13, *Parallelizzare l'addestramento delle reti neurali con Theano*, estende le conoscenze trattate nel capitolo precedente offrendo una guida pratica all'addestramento efficiente di reti neurali. L'argomento del capitolo è principalmente Theano, una libreria open source Python che consente di utilizzare i core delle moderne GPU.

Dotazione software necessaria

L'esecuzione degli esempi di codice forniti in questo libro richiede l'installazione di Python 3.4.3 o una versione successiva su un sistema Mac OS X, Linux o Microsoft Windows. Faremo frequentemente uso delle librerie essenziali di Python dedicate al calcolo scientifico, fra cui SciPy, NumPy, scikit-learn, matplotlib e pandas.
Il primo capitolo fornirà tutte le istruzioni e i suggerimenti necessari per configurare l'ambiente Python e queste librerie di base. Successivamente aggiungeremo ulteriori librerie al nostro repertorio, e le relative istruzioni di installazione verranno fornite nei capitoli successivi: la libreria NLTK, dedicata all'elaborazione del linguaggio naturale (Capitolo 8, *Tecniche di machine learning per l'analisi del sentiment*), il framework web Flask (Capitolo 9, *Embedding di un modello in un'applicazione web*), la libreria seaborn per la

visualizzazione dei dati statistici (Capitolo 10, *Previsioni di variabili target continue: l'analisi a regressione*) e Theano per l'addestramento efficiente di reti neurali su unità GPU (Capitolo 13, *Parallelizzare l'addestramento delle reti neurali con Theano*).

A chi è rivolto questo libro

A chi intende scoprire come utilizzare Python per iniziare a rispondere a domande critiche basate sui dati. Che vogliate iniziare da zero o vogliate estendere le vostre conoscenze nell'ambito dell'elaborazione dei dati, questa è una risorsa essenziale e imperdibile.

Convenzioni

In questo libro, troverete vari stili di testo per distinguere diversi tipi di informazioni. Ecco alcuni esempi degli stili utilizzati e del loro significato.
Gli elementi di codice nel testo, i nomi di tabelle di un database, i nomi di cartelle, i nomi di file, le estensioni di file, i percorsi, gli indirizzi URL, gli input da inserire sono tutti scritti utilizzando il carattere monospaziato, per esempio: "I pacchetti già installati possono essere aggiornati con il flag `--upgrade`".
Un blocco di codice ha il seguente aspetto:

```
>>> import matplotlib.pyplot as plt
>>> import numpy as np

>>> y = df.iloc[0:100, 4].values
>>> y = np.where(y == 'Iris-setosa', -1, 1)
>>> X = df.iloc[0:100, [0, 2]].values
>>> plt.scatter(X[:50, 0], X[:50, 1],
...             color='red', marker='x', label='setosa')
>>> plt.scatter(X[50:100, 0], X[50:100, 1],
...             color='blue', marker='o', label='versicolor')
>>> plt.xlabel('sepal length')
>>> plt.ylabel('petal length')
>>> plt.legend(loc='upper left')
>>> plt.show()
```

Ogni input da inserire nella riga di comando e ogni output è scritto nel seguente modo:

```
> dot -Tpng tree.dot -o tree.png
```

I nuovi termini e le parole chiave di un discorso sono indicate in *corsivo*. Le parole che compaiono sullo schermo, per esempio nei menu o nelle finestre di dialogo, sono sempre indicate in *corsivo*.

> **NOTA**
>
> Avvertimenti e note importanti sono evidenziate in questo modo.

Approfondimento

Approfondimenti e chiarificazioni ai concetti e agli argomenti trattati nel corso del testo vengono mostrati in questo modo.

Scarica i file degli esempi

Sul sito dell'editore originale inglese, Packt Publishing, potete scaricare i file degli esempi presentati del testo. Per farlo è necessario registrarsi gratuitamente all'indirizzo `https://www.packtpub.com/register`. Quindi andate sulla scheda del libro all'indirizzo `https://www.packtpub.com/big-data-and-business-intelligence/python-machine-learning` (per comodità in forma abbreviata `http://bit.ly/packt-pml`) e fate clic su *Code Files*. Per problemi nel download potete contattare la nostra redazione all'indirizzo `libri@apogeonline.com`.

L'autore

Sebastian Raschka, dottorando presso la Michigan State University, sviluppa modelli di calcolo computerizzato applicato alla biologia. È stato valutato da Analytics Vidhya come lo scienziato più influente nell'ambito della *data science* su GitHub. Vanta un'esperienza pluriennale di programmazione in Python e ha tenuto diversi seminari sulle applicazioni pratiche della scienza dei dati e sull'apprendimento automatico. Il fatto di parlare e scrivere di scienza dei dati, machine learning e Python lo ha motivato nella scrittura di questo libro per aiutare anche altri a sviluppare soluzioni basate sui dati, rivolgendosi anche a coloro che non sono dotati di un background specifico in questi ambiti.
Ha anche contribuito attivamente allo sviluppo di progetti e metodi open source che ha implementato e che ora vengono utilizzati con successo nelle competizioni di machine learning, come Kaggle. Nel tempo libero, progetta modelli per le previsioni dei risultati sportivi e quando non è davanti a un computer, si dedica allo sport.

I revisori

Richard Dutton ha iniziato a programmare su un Sinclair ZX Spectrum all'età di otto anni e le sue ossessioni lo hanno trasportato attraverso una serie confusa di tecnologie e incarichi nei campi della tecnologia e della finanza.
Ha lavorato con Microsoft e come direttore in Barclays, ma la sua attuale ossessione è un mix di Python, machine learning e concatenamento a blocchi.
Quando non è seduto a un computer, lo si può trovare in palestra o a casa, dietro a un calice di vino mentre osserva il display del suo iPhone. Lui chiama tutto questo "equilibrio".

Dave Julian è consulente informatico e docente con oltre quindici anni di esperienza. Ha lavorato come tecnico, project manager, programmatore e sviluppatore web. I suoi attuali progetti comprendono lo sviluppo di uno strumento di analisi della serricoltura nell'ambito delle strategie di lotta integrata ai parassiti. È particolarmente interessato alle

interazioni fra biologia e tecnologie, con una forte convinzione che, in futuro, macchine intelligenti potranno aiutare a risolvere i più grandi problemi che affliggono il mondo.

Vahid Mirjalili ha conseguito la laurea in ingegneria meccanica presso la Michigan State University, dove ha sviluppato innovative tecniche per il raffinamento della struttura proteica, utilizzando simulazioni molecolari dinamiche. Combinando la sua conoscenza nei campi della statistica, del data mining e della fisica ha sviluppato potenti approcci basati sui dati che hanno aiutato lui e il suo team di ricercatori a vincere due recenti competizioni a livello mondiale sulla predizione e il raffinamento della struttura proteica, CASP, nel 2012 e nel 2014.
Nel corso del suo dottorato, ha deciso di unirsi al Computer Science and Engineering Department della Michigan State University, per specializzarsi nel campo dell'apprendimento automatico. I suoi attuali progetti di ricerca riguardano lo sviluppo di algoritmi di machine learning senza supervisione per il mining di grossi data set. È anche un appassionato programmatore Python e condivide le sue implementazioni degli algoritmi di clustering sul suo sito web personale, `http://vahidmirjalili.com`.

Hamidreza Sattari è un professionista informatico che è stato coinvolto in varie aree dell'ingegneria software, dalla programmazione all'architettura, fino alla amministrazione. È dotato di un master in ingegneria del software conseguito presso la Herriot-Watt University, in Gran Bretagna e di una laurea in ingegneria elettronica conseguita presso la Azad University di Teheran, in Iran. Negli ultimi anni, ha rivolto la sua attenzione all'elaborazione di grosse quantità di dati e all'apprendimento automatico. È coautore del libro *Spring Web Services 2 Cookbook* e cura il blog `http://justdeveloped-blog.blogspot.com`.

Dmytro Taranovsky è un ingegnere software che manifesta un grande interesse e background nell'ambito di Python, Linux e del machine learning. Originario di Kiev, Ucraina, si è trasferito negli Stati Uniti nel 1996. Fin dalla più tenera età, ha dimostrato una passione per la scienza e la conoscenza, vincendo competizioni in campo matematico e fisico. Nel 1999 è stato scelto quale membro del U.S. Physics Team. Nel 2005 si è laureato presso il Massachusetts Institute of Technology, specializzandosi in Matematica. Successivamente ha lavorato come ingegnere software su un sistema di trasformazione del testo per trascrizioni mediche assistite al computer (eScription). Anche se originariamente ha lavorato in Perl, apprezza la potenza e la chiarezza di Python ed è stato in grado di scalare il proprio sistema in modo da farlo operare su grandi quantità di dati. Successivamente ha lavorato come ingegnere software e analista per una società di trading. Ha fornito significativi contributi alle basi matematiche, compresa la creazione e lo sviluppo di un'estensione al linguaggio della teoria degli insiemi e alla sua connessione con gli assiomi a grande cardinalità, sviluppando un concetto di verità costruttiva e creando un sistema di notazione ordinale e poi implementando il tutto in Python. Gli piace anche leggere, andare a spasso e tenta di rendere il mondo un luogo migliore.