

Python

mašinsko učenje

Mašinsko učenje i duboko učenje pomoću Pythona, scikit-learn biblioteke i TensorFlowa 2

Treće izdanje - uključuje TensorFlow 2, GAN i učenje uslovljavanjem

**Sebastian Raschka
i Vahid Mirjalili**

Prevod III izdanja

Python mašinsko učenje

Mašinsko učenje i duboko učenje pomoću Pythona,
scikit-learn biblioteke i TensorFlowa 2

Sebastian Raschka

Vahid Mirjalili



 kompiuter
biblioteka

Packt>

Izdavač:



Obalskih radnika 4a, Beograd

Tel: 011/2520272

e-mail: kombib@gmail.com

internet: www.kombib.rs

Urednik: Mihailo J. Šolajić

Za izdavača, direktor:

Mihailo J. Šolajić

Autor: Sebastian Raschka
Vahid Mirjalili

Prevod: Slavica Prudkov

Lektura: Miloš Jevtović

Slog: Zvonko Aleksić

Znak Kompjuter biblioteke:
Miloš Milosavljević

Štampa: „Pekograf“, Zemun

Tiraž: 500

Godina izdanja: 2020.

Broj knjige: 526

Izdanje: Prvo

ISBN: 978-86-7310-549-9

Python Machine Learning

Third Edition

Sebastian Raschka

Vahid Mirjalili

ISBN 978-1-78995-575-0

Copyright © 2019 Packt Publishing

All right reserved. No part of this book may be reproduced or transmitted in any form or by means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher. Autorizovani prevod sa engleskog jezika edicije u izdanju „Packt Publishing“, Copyright © 2019.

Sva prava zadržana. Nije dozvoljeno da nijedan deo ove knjige bude reprodukovan ili snimljen na bilo koji način ili bilo kojim sredstvom, elektronskim ili mehaničkim, uključujući fotokopiranje, snimanje ili drugi sistem presnimavanja informacija, bez dozvole izdavača.

Zaštitni znaci

Kompjuter Biblioteka i „Packt Publishing“ su pokušali da u ovoj knjizi razgraniče sve zaštitne oznake od opisnih termina, prateći stil isticanja oznaka velikim slovima.

Autor i izdavač su učinili velike napore u pripremi ove knjige, čiji je sadržaj zasnovan na poslednjem (dostupnom) izdanju softvera. Delovi rukopisa su možda zasnovani na predizdanju softvera dobijenog od strane proizvođača. Autor i izdavač ne daju nikakve garancije u pogledu kompletnosti ili tačnosti navoda iz ove knjige, niti prihvataju ikakvu odgovornost za performanse ili gubitke, odnosno oštećenja nastala kao direktna ili indirektna posledica korišćenja informacija iz ove knjige.

O AUTORIMA

Sebastian Raschka je doktorirao na Michigan State Universityju, gde se fokusirao na razvoj metoda na preseku računске biologije i mašinskog učenja. U leto 2018. godine dobio je posao na Univerzitetu Viskonsin u Medisonu, SAD, kao asistent profesora statistike. Njegove aktivnosti u istraživanju uključuju razvoj novih arhitektura dubokog učenja za rešavanje problema u oblasti biometrije.

Sebastian ima mnogo godina iskustva u kodiranju u Python jeziku i održao je nekoliko seminara, uključujući i tutoriale za mašinsko učenje na SciPyju, vodećoj konferenciji za naučno računarstvo u Pythonu.

Među Sebastianovim dostignućima je njegova knjiga „Python mašinsko učenje“, koja je najprodavaniji naslov u „Packtu“ i Amazon.com-u. Ta knjiga, za koju je dobio ACM Best of Computing nagradu 2016. godine, prevedena je na mnoge jezike, uključujući nemački, korejski, kineski, japanski, ruski, poljski i italijanski.

U slobodno vreme Sebastian voli da saraduje u projektima otvorenog koda. Metodi koje je implementirao se sada uspešno koriste u takmičenjima mašinskog učenja, kao što je Kaggle.

Zahvaljujem se odličnoj Python zajednici i programerima paketa otvorenog koda koji su mi pomogli da kreiram savršeno okruženje za naučna istraživanja i istraživanje podataka. Takođe se zahvaljujem mojim roditeljima koji su me uvek ohrabivali i podržavali u trasiranju profesionalne karijere kojoj sam strastveno težio. Posebno se zahvaljujem programerima scikit-learna i TensorFlowa. Kao saradnik i korisnik, imao sam zadovoljstvo da radim sa odličnim ljudima koji ne samo da imaju veliko znanje kada je reč o mašinskom učenju i dubokom učenju, već su i izvršni programeri.

Vahid Mirjalili je doktorirao mašinsko inženjerstvo, radeći na novim metodima za velike, računarske simulacije molekularnih struktura na Michigan State Universityju. Veoma je zainteresovan za oblast mašinskog učenja i pridružio se iPRoBe labu na Michigan State Universityju, gde je radio na primeni mašinskog učenja u domenima računarskog vida i biometrije. Nakon nekoliko produktivnih godina u iPRoBe labu i mnogo godina na studijama, Vahid se nedavno pridružio 3M kompaniji kao naučnik istraživač, gde može da iskoristi svoju stručnost i primeni najsavremenije tehnike mašinskog učenja i dubokog učenja za rešavanje problema iz realnog sveta u različitim aplikacijama da bi život učinio boljim.

Zahvaljujem se svojoj supruzi Taban Eslami, koja mi je pružala podršku i ohrabivala me na putu moje karijere. Takođe se posebno zahvaljujem svojim savetnicima Nikoli Priezjevu, Michaelu Feigu i Arunu Rossu što su me podržavali u toku doktorskih studija, kao i mojim profesorima Vishnu Boddeti, Leslie Kuhn i Xiaoming Liu, koji su me naučili mnogo i podsticali me da istrajem u svojoj strasti.

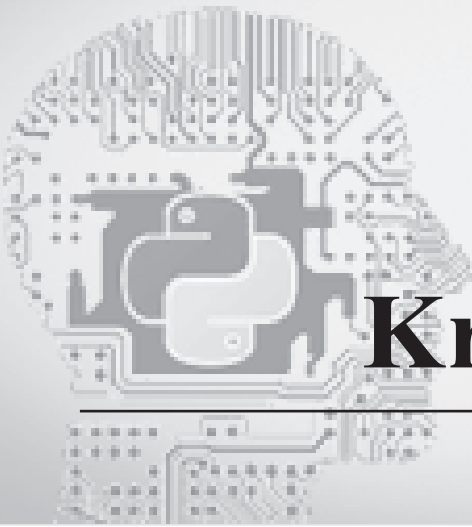
O RECENZENTIMA

Raghav Bali je stariji istraživač podataka u jednoj od svetski najvećih zdravstvenih organizacija. Njegov rad uključuje istraživanje i razvoj rešenja poslovnog nivoa zasnovanog na mašinskom učenju, dubokom učenju i obradi prirodnog jezika koji se koriste u oblastima zdravstva i osiguranja. U prethodnoj ulozi u „Intelu“ bio je uključen u omogućavanje proaktivnih IT inicijativa vođenih podacima upotrebom obrade prirodnog jezika, dubokog učenja i tradicionalnih statističkih metoda. Takođe je radio u oblasti finansija u American Expressu, rešavajući slučajeve digitalnog angažovanja i zadržavanja klijenata.

Raghav je takođe autor više knjiga kod vodećih izdavača, a najnovija knjiga posvećena je napretku u istraživanjima transfernog učenja.

Raghav je magistrirao (ima zlatnu medalju) u informacionim tehnologijama na Međunarodnom institutu za informacionu tehnologiju u Bangaloru, u Indiji. Voli da čita i fotografiše kada nije zauzet rešavanjem problema.

Motaz Saad je doktorirao računarske nauke na Univerzitetu Lorraine. Voli podatke i voli da se igra njima. Ima više od 10 godina profesionalnog iskustva u obradi prirodnih jezika, računarskoj lingvistici, istraživanju podataka i mašinskom učenju. Trenutno radi kao asistent na fakultetu Information Technology, IUG.



Kratak sadržaj

POGLAVLJE 1

Kako da računarima pružite mogućnost da uče iz podataka..... 1

POGLAVLJE 2

**Treniranje jednostavnih algoritama mašinskog učenja
za klasifikaciju 19**

POGLAVLJE 3

**Predstavljanje klasifikatora mašinskog učenja
pomoću scikit-learn biblioteke 53**

POGLAVLJE 4

**Izgradnja dobrih trening skupova podataka -
pretprocesiranje podataka 109**

POGLAVLJE 5

Kompresovanje podataka pomoću redukcije dimenzionalnosti 145

POGLAVLJE 6

**Učenje najbolje prakse za procenu modela i
fino podešavanje hiperparametara 191**

POGLAVLJE 7

Kombinovanje različitih modela za ansambl metode 223

POGLAVLJE 8

**Primena mašinskog učenja
na analizu sentimenta 259**

POGLAVLJE 9

Ugrađivanje modela mašinskog učenja u veb aplikaciju 285

POGLAVLJE 10

Predviđanje kontinualnih ciljnih promenljivih pomoću regresione analize 315

POGLAVLJE 11

Upotreba neoznačenih podataka - klaster analiza 353

POGLAVLJE 12

Implementiranje višeslojne veštačke neuronske mreže „od nule“ 383

POGLAVLJE 13

Paralelizacija treninga neuronske mreže pomoću TensorFlowa 425

POGLAVLJE 14

Detaljnije - mehanika TensorFlowa 471

POGLAVLJE 15

Klasifikovanje slika pomoću dubokih konvolutivnih neuronskih mreža 517

POGLAVLJE 16

Modelovanje sekvencijalnih podataka upotrebom rekurentnih neuronskih mreža 567

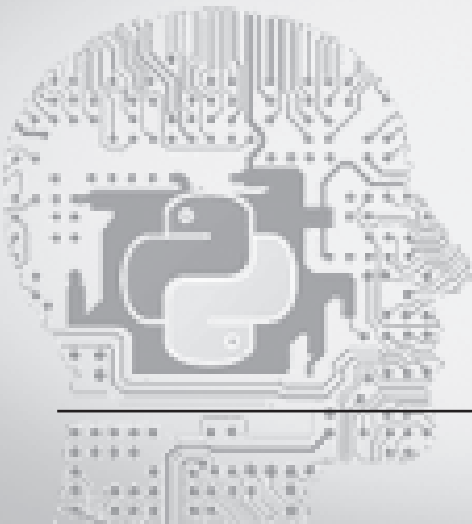
POGLAVLJE 17

Generativne suparničke mreže za sintetizovanje novih podataka 619

POGLAVLJE 18

Učenje uslovljavanjem za donošenje odluka u kompleksnim okruženjima 671

INDEKS



Sadržaj

UVOD

POGLAVLJE 1

Kako da računarima pružite mogućnost da uče iz podataka..... 1

Izgradnja inteligentnih mašina za transformisanje podataka u znanje.....	2
Tri različita tipa mašinskog učenja	2
Predviđanje budućnosti pomoću nadgledanog učenja.....	3
Klasifikacija za predviđanje oznaka klase	3
Regresija za predviđanje neprekidnog ishoda	4
Rešavanje interaktivnih problema pomoću učenja uslovljavanjem	6
Otkrivanje skrivenih struktura pomoću nenadgledanog učenja	7
Pronalaženje podgrupa pomoću klasterovanja.....	7
Redukcija dimenzionalnosti za kompresovanje podataka.....	8
Uvod u osnovnu terminologiju i notacije	8
Notacije i konvencije upotrebljene u ovoj knjizi	9
Terminologija mašinskog učenja	11
Mapa za izgradnju sistema mašinskog učenja.....	11
Pretprocesiranje - oblikovanje podataka	12
Trening i selektovanje prediktivnog modela.....	13
Procena modela i predviđanje neviđenih instanci podataka	14
Upotreba Pythona za mašinsko učenje.....	14
Instaliranje Pythona i paketa iz Python Package Indexa	15
Upotreba Anaconda Python distribucije i upravljača paketima.....	15
Paketi za naučna izračunavanja, istraživanje podataka i mašinsko učenje.....	16
Rezime	16

POGLAVLJE 2

Treniranje jednostavnih algoritama

mašinskog učenja za klasifikaciju 19

Veštački neuroni - kratak pregled istorije mašinskog učenja	20
Formalna definicija veštačkog neurona	21
Perceptron pravilo učenja	23
Implementiranje algoritma učenja perceptrona u Python	26
Objektno-orijentisan perceptron API	26
Treniranje perceptron modela u Iris skupu podataka	30
Prilagodljivi linearni neuroni i konvergencija učenja	36
Minimiziranje funkcija koštanja pomoću gradijentnog spusta.....	37
Implementiranje Adaline algoritma u Pythonu	40
Poboljšanje gradijentnog spusta pomoću skaliranja atributa	44
Mašinsko učenje visokog stepena i stohastički gradijentni spust.....	46
Rezime	51

POGLAVLJE 3

Predstavljanje klasifikatora mašinskog učenja

pomoću scikit-learn biblioteke 53

Biranje algoritma klasifikacije.....	54
Prvi koraci upotrebe scikit-learn biblioteke - treniranje perceptrona	54
Modelovanje verovatnoće klase pomoću logističke regresije.....	60
Logistička regresija i uslovne verovatnoće.....	60
Učenje težina logističke funkcije koštanja.....	65
Konvertovanje Adaline implementacije u algoritam za logističku regresiju	67
Treniranje modela logističke regresije pomoću scikit-learn biblioteke	72
Rešavanje problema prilagođavanja pomoću regularizacije.....	75
Klasifikacija maksimalne margine pomoću metoda potpornih vektora	79
Intuicija maksimalne margine	79
Rešavanje nelinearno razdvojitog slučaja upotrebom slack promenljivih.....	81
Alternativne implementacije u scikit-learn biblioteci.....	83
Rešavanje nelinearnih problema upotrebom kernel SVM-a.....	84
Kernel metodi za linearno razdvojive podatke	84
Upotreba kernel trika za pronalaženje razdvajajućih hiperravni u visokodimenzionalnom prostoru.....	86
Učenje stabla odlučivanja	90
Maksimiziranje IG-a - dobijanje maksimuma za naš novac	91
Izgradnja stabla odlučivanja	96
Kombinovanje više stabala odlučivanja pomoću slučajnih šuma	100
K-najbliži susedi - algoritam metoda lenjog učenja.....	103
Rezime	108

POGLAVLJE 4**Izgradnja dobrih trening skupova podataka -
pretprocesiranje podataka 109**

Rešavanje problema nedostajućih podataka	109
Identifikovanje nedostajućih vrednosti u tabelarnim podacima	110
Eliminisanje trening primera ili atributa sa nedostajućim vrednostima	111
Imputiranje nedostajućih vrednosti.....	112
Razumevanje scikit-learn API-a koji vrši procenu.....	113
Obrada kategorijskih podataka	115
Kodiranje kategorijskih podataka pomoću pandas biblioteke	116
Mapiranje rednih atributa.....	116
Kodiranje oznaka klase.....	117
Izvršavanje one-hot kodiranja na nominalnim atributima	118
Particionisanje skupa podataka u posebne skupove podataka za trening i testiranje.....	121
Dovođenje atributa na istu skalu.....	124
Selektovanje značajnih atributa.....	127
L1 i L2 regularizacija kao kazna, nasuprot kompleksnosti modela.....	128
Geometrijska interpretacija L2 regularizacije	128
Proređena rešenja L1 regularizacije	131
Algoritam sekvencijalne selekcije atributa.....	135
Procena važnosti atributa pomoću slučajnih šuma	141
Rezime	144

POGLAVLJE 5**Kompresovanje podataka pomoću redukcije dimenzionalnosti 145**

Nenadgledana redukcija dimenzionalnosti pomoću analize glavne komponente.....	146
Glavni koraci za analizu glavne komponente.....	146
Ekstrakcija glavnih komponenata korak po korak.....	148
Ukupna i objašnjena varijansa	151
Transformacija atributa.....	152
Analiza glavne komponente u scikit-learn implementaciji	155
Nadgledano kompresovanje podataka pomoću linearne diskriminantne analize	159
Analiza glavne komponente, nasuprot linearne diskriminantne analize.....	159
Izračunavanje matrica rasipanja.....	161
Selektovanje linearnih diskriminanti za novi potprostor atributa	164
Projektovanje primera u novi prostor atributa	167
LDA pomoću scikit-learn biblioteke.....	168
Upotreba kernel metode analize glavne komponente za nelinearna mapiranje	169
Kernel funkcije i kernel trik	170
Implementiranje kernel metoda analize glavne komponente u Python	175
Primer 1 - odvajanje oblika polumeseca.....	177
Primer 2 - razdvajanje koncentričnih krugova.....	180
Projektovanje novih tačaka podataka.....	183
Kernel metoda analize glavne komponente u scikit-learn implementaciji	187
Rezime	189

POGLAVLJE 6**Učenje najbolje prakse za procenu modela i fino podešavanje hiperparametara 191**

Pojednostavljanje procesa rada pomoću pipeline.....	191
Učitavanje skupa podataka Breast Cancer Wisconsin.....	192
Kombinovanje transformatora i procenjivača u pipelineu.....	193
Upotreba k-slojne unakrsne validacije za procenu performanse modela.....	195
Metod procene testnog uzorka.....	196
K-slojna unakrsna validacija.....	197
Algoritmi otklanjanje grešaka sa krivim učenja i validacije.....	201
Dijagnostikovanje problema biasa i varijanse pomoću krive učenja.....	201
Rešavanje problema prilagođavanja i nedovoljnog prilagođavanja pomoću krive validacije.....	205
Fino podešavanje modela mašinskog učenja pomoću grid search algoritma.....	207
Podešavanje hiperparametara pomoću grid search metoda.....	207
Selekcija algoritma pomoću ugneždene unakrsne validacije.....	209
Pregled različitih metrika procene performanse.....	211
Čitanje matrice konfuzije.....	211
Optimizacija preciznosti i opoziva modela klasifikacije.....	213
Isctavanje dijagrama operativne karakteristike primaoca.....	216
Metrike ocenjivanja za višeklasnu klasifikaciju.....	219
Rešavanje problema klasne neravnoteže.....	220
Rezime.....	222

POGLAVLJE 7**Kombinovanje različitih modela za ansambl metode 223**

Učenje pomoću ansambla.....	223
Kombinovanje klasifikatora pomoću većinskih glasova.....	227
Implementiranje jednostavnog klasifikatora većinskog glasanja.....	228
Upotreba principa većinskog glasanja za izvršavanje predviđanja.....	234
Procena i podešavanje ansambl klasifikatora.....	237
Bagging - izgradnja ansambla klasifikatora iz bootstrap uzoraka.....	243
Bagging ukratko.....	244
Primena bagginga za klasifikaciju primera u Wine skupu podataka.....	245
Iskorišćavanje slabih učenika pomoću adaptivnog boostinga.....	249
Kako funkcioniše boosting algoritam.....	250
Primena AdaBoost algoritma upotrebom scikit-learn biblioteke.....	254
Rezime.....	257

POGLAVLJE 8**Primena mašinskog učenja na analizu sentimenta..... 259**

Priprema podataka IMDb recenzija filmova za obradu teksta.....	260
Preuzimanje skupa podataka recenzija filmova.....	260
Preprocesiranje skupa podataka filmova u pogodniji format.....	261

Predstavljanje bag-of-words modela	262
Transformisanje reči u vektore atributa	263
Procena relevantnosti reči pomoću tehnike term frequency-inverse document frequency	265
Čišćenje tekstualnih podataka	267
Obrada dokumenata u tokene	269
Treniranje modela logističke regresije za klasifikaciju dokumenta	272
Upotreba većih podataka - online algoritmi i out-of-core učenje	274
Modelovanje teme pomoću Latent Dirichlet raspodele	278
Razlaganje tekstualnih dokumenata pomoću LDA-a	279
LDA pomoću scikit-learn biblioteke	279
Rezime	283

POGLAVLJE 9

Ugrađivanje modela mašinskog učenja u veb aplikaciju 285

Serijalizacija prilagođenih scikit-learn procenjivača	285
Podešavanje SQLite baze podataka za skladištenje podataka	289
Razvijanje veb aplikacije pomoću Flask radnog okvira	291
Naša prva Flask veb aplikacija	292
Validacija i renderovanje obrasca	294
Podešavanje strukture direktorijuma	295
Implementiranje makroa upotrebom Jinja2 templating mehanizma	296
Dodavanje stila pomoću CSS-a	296
Kreiranje stranice rezultata	298
Pretvaranje klasifikatora filmskih recenzija u veb aplikaciju	300
Fajlovi i direktorijum - pregled stabla direktorijuma	301
Implementiranje glavne aplikacije kao app.py	302
Podešavanje obrasca recenzije	305
Kreiranje šablona stranice rezultata	306
Raspoređivanje veb aplikacije na javni server	309
Kreiranje PythonAnywhere naloga	309
Slanje aplikacije klasifikatora filmova	310
Ažuriranje klasifikatora filma	311
Rezime	314

POGLAVLJE 10

Predviđanje kontinualnih ciljnih promenljivih pomoću regresione analize 315

Predstavljanje linearne regresije	316
Jednostavna linearna regresija	316
Višestruka linearna regresija	317
Istraživanje Housing skupa podataka	318
Učitavanje Housing skupa podataka u okvir podataka	318
Vizuelizacija važnih karakteristika skupa podataka	320
Pregled odnosa upotrebom matrice korelacije	322

Implementiranje modela linearne regresije običnih najmanjih kvadrata	325
Rešavanje regresije za parametre regresije pomoću gradijentnog spusta.....	325
Procenjivanje koeficijenta modela regresije pomoću scikit-learn biblioteke	330
Prilagođavanje robusnog modela regresije pomoću RANSAC-a.....	332
Procena performanse modela linearne regresije.....	334
Upotreba regularizovanih metoda za regresiju	337
Pretvaranje modela linearne regresije u krivu - polinomijalna regresija.....	339
Dodavanje polinomijalnih članova upotrebom scikit-learn biblioteke	340
Modelovanje nelinearnih odnosa u Housing skupu podataka	342
Rešavanje nelinearnih odnosa upotrebom slučajnih šuma.....	345
Regresija stabla odlučivanja.....	346
Regresija slučajne šume	348
Rezime	351

POGLAVLJE 11

Upotreba neoznačenih podataka - klaster analiza 353

Grupisanje objekata po sličnosti upotrebom k-srednjih vrednosti.....	354
Klasterizacija metodom k-srednjih vrednosti upotrebom scikit-learn biblioteke	354
Pametniji način postavljanja inicijalnih centroida klastera upotrebom algoritma k-means++	358
Tvrdo, nasuprot mekog klasterovanja.....	359
Upotreba elbow metoda za pronalaženje optimalnog broja klastera	361
Kvantifikovanje kvaliteta klasterovanja pomoću silhouette dijagrama.....	363
Organizovanje klastera kao hijerarhijskog stabla	367
Grupisanje klastera od dna ka vrhu.....	368
Izvršavanje hijerarhijskog klasterovanja na matrici rastojanja	369
Priključivanje dendrograma u toplotnu mapu	373
Primena algoritma sakupljajućeg klasterovanja pomoću scikit-learn biblioteke	375
Lociranje regiona visoke gustine pomoću DBSCAN-a	376
Rezime	382

POGLAVLJE 12

Implementiranje višeslojne veštačke neuronske mreže „od nule“ 383

Modelovanje kompleksnih funkcija pomoću veštačkih neuronskih mreža.....	384
Rekapitulacija jednoslojne neuronske mreže.....	385
Predstavljanje arhitekture višeslojne neuronske mreže	387
Aktiviranje neuronske mreže propagiranjem unapred.....	391
Klasifikovanje ručno pisanih cifara	393
Preuzimanje i pripremanje MNIST skupa podataka	394
Implementiranje višeslojnog perceptrona	400
Obučavanje veštačke neuronske mreže	412

Izračunavanje logističke funkcije koštanja	412
Bolje razumevanje backpropagation algoritma	415
Treniranje neuronskih mreža pomoću algoritma backpropagation.....	417
O konvergenciji u neuronskim mrežama.....	421
Još nekoliko reči o implementaciji neuronske mreže	422
Rezime	423

POGLAVLJE 13

Paralelizacija treninga neuronske mreže pomoću TensorFlowa 425

TensorFlow i performansa treninga.....	426
Izazovi performanse.....	426
Šta je TensorFlow?	427
Kako ćemo učiti TensorFlow	429
Prvi koraci upotrebe TensorFlowa.....	429
Instaliranje TensorFlowa	429
Kreiranje tenzora u TensorFlowu	430
Manipulisanje tipom podataka i oblikom tenzora.....	431
Primena matematičkih operacija na tenzore.....	432
Razdvajanje, slaganje i nadovezivanje tenzora.....	434
Izgradnja ulaznih pipelinea upotrebom funkcije tf.data -	
TensorFlow Dataset API	435
Kreiranje TensorFlow Dataseta iz postojećih tenzora	436
Kombinovanje dva tenzora u udruženi skup podataka.....	437
Mešanje, grupisanje i ponavljanje	439
Kreiranje skupa podataka iz fajlova na lokalnom disku za skladištenje	441
Preuzimanje dostupnih skupova podataka iz tensorflow_datasets biblioteke	445
Izgradnja NN modela u TensorFlowu.....	450
TensorFlow Keras API (tf.keras)	451
Izgradnja modela linearne regresije	451
Treniranje modela pomoću metoda .compile() i .fit().....	456
Izgradnja višeslojnog perceptrona za klasifikovanje cveća u Iris skupu podataka.....	457
Procena obučenog modela na test skupu podataka.....	461
Čuvanje i ponovno učitavanje obučenog modela	461
Biranje aktivacionih funkcija za višeslojne neuronske mreže.....	462
Rekapitulacija logističke funkcije	463
Procena verovatnoće klase u višeklasnoj klasifikaciji pomoću softmax funkcije	465
Proširenje spektra izlaza upotrebom hiperboličke tangente	466
Rectified linear unit aktivacija	468
Rezime	470

POGLAVLJE 14**Detaljnije - mehanika TensorFlowa 471**

Ključni atributi TensorFlowa	472
Grafovi izračunavanja TensorFlowa: prenos na TensorFlow v2.....	473
Razumevanje grafova izračunavanja	473
Kreiranje grafa u TensorFlow v1.x verziji.....	474
Prenos grafa u TensorFlow v2	475
Učitavanje ulaznih podataka u model: TensorFlow v1.x stil	476
Učitavanje ulaznih podataka u model: TensorFlow v2 stil.....	476
Poboljšanje performanse izračunavanja pomoću dekoratora funkcije	477
TensorFlow Variable objekti za skladištenje i ažuriranje parametara modela	479
Izračunavanje gradijenata korišćenjem automatske diferencijacije i GradientTapea.....	483
Izračunavanje gradijenata greške u odnosu na promenljive koje se mogu obučavati	483
Izračunavanje gradijenata u odnosu na tenzore koji se ne mogu obučavati	485
Čuvanje resursa za višestruka izračunavanja gradijenta	485
Pojednostavljenje implementacija uobičajenih arhitektura pomoću Keras API-a.....	486
Rešavanje problema XOR klasifikacije	489
Kako da učinite izgradnju modela fleksibilnijom pomoću Kerasovog funkcionalnog API-a	494
Implementiranje modela na osnovu Kerasove Model klase	496
Pisanje prilagođenih Keras slojeva	497
TensorFlow estimatori	501
Upotreba kolona atributa	501
Mašinsko učenje sa unapred kreiranim estimatorima.....	506
Upotreba estimatora za klasifikaciju skupa podataka MNIST ručno pisanih cifara	510
Kreiranje prilagođenog estimatora iz postojećeg Keras modela	512
Rezime	515

POGLAVLJE 15**Klasifikovanje slika pomoću dubokih konvolutivnih neuronskih mreža 517**

Gradivni blokovi CNN-a	518
Razumevanje CNN-a i hijerarhije atributa.....	518
Izvršavanje diskretnih konvolucija.....	520
Diskretna konvolucija u jednoj dimenziji	521
Dopunjavanje ulaza za kontrolu veličine mapa izlaznog atributa	523
Određivanje veličine izlaza konvolucije	525
Izvršavanje diskretne konvolucije u 2D-u.....	526
Slojevi za smanjivanje uzoraka	530
Spajanje svega - implementiranje CNN-a.....	532
Upotreba više ulaza ili kanala za boje.....	532
Regularizacija NN-e pomoću tehnike napuštanja.....	536

Funkcije greške za klasifikaciju.....	539
Implementiranje duboke CNN upotrebom TensorFlowa.....	542
Arhitektura višeslojne CNN-e.....	542
Učitavanje i pretprocesiranje podataka.....	543
Implementiranje CNN-e upotrebom TensorFlow Keras API-a.....	544
Konfigurisanje CNN slojeva u Kerasu.....	544
Konstruisanje CNN-a u Kerasu.....	545
Klasifikacija pola iz portreta upotrebom CNN-a.....	550
Učitavanje CelebA skupa podataka.....	551
Transformacija slike i proširenje podataka.....	552
Obučavanje klasifikatora pola CNN-e.....	558
Rezime.....	565

POGLAVLJE 16

Modelovanje sekvencijalnih podataka upotrebom rekurentnih neuronskih mreža.....567

Predstavljanje sekvencijalnih podataka.....	568
Modelovanje sekvencijalnih podataka - redosled je važan.....	568
Predstavljanje sekvenci.....	569
Drugačije kategorije modelovanja sekvence.....	570
RNN za modelovanje sekvenci.....	571
Razumevanje mehanizma ponavljanja u petlji RNN-e.....	571
Izračunavanje aktivacija u RNN-i.....	574
Skriveno ponavljanje, nasuprot izlaznog ponavljanja.....	577
Izazovi učenja interakcija dugog dometa.....	580
Long short-term memory cells.....	582
Implementiranje RNN-a za modelovanje sekvence u TensorFlowu.....	584
Prvi projekat - predviđanje sentimenta IMDb recenzija filmova.....	585
Priprema podataka recenzije filmova.....	585
Ugrađivanje slojeva za kodiranje rečenice.....	590
Izgradnja RNN modela.....	592
Izgradnja RNN modela za analizu sentimenta.....	594
Drugi projekat - modelovanje jezika na nivou karaktera u TensorFlowu.....	600
Pretprocesiranje skupa podataka.....	601
Izgradnja RNN modela nivoa karaktera.....	607
Faza evaluacije - generisanje novih odlomaka teksta.....	609
Razumevanje jezika pomoću Transformer modela.....	613
Razumevanje mehanizma samopažnje.....	614
Osnovna verzija samopažnje.....	614
Parametarizacija self-attention mehanizma pomoću upita, ključa i težina vrednosti.....	616
Multi-head attention i Transformer blok.....	617
Rezime.....	618

POGLAVLJE 17**Generativne suparničke mreže za sintetizovanje novih podataka 619**

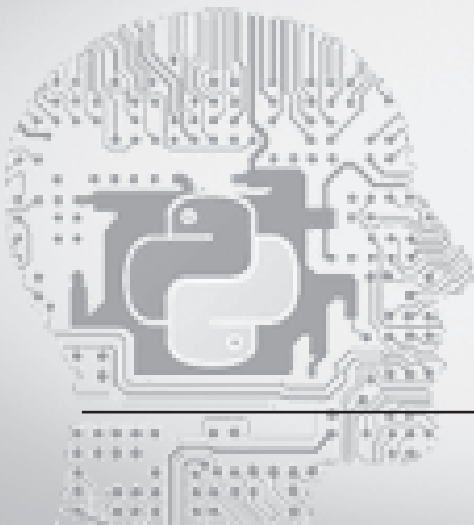
Predstavljanje generativnih suparničkih mreža	620
Autoenkodori.....	620
Generativni modeli za sintetizovanje novih podataka.....	623
Generisanje novih uzoraka pomoću GAN-a.....	624
Razumevanje funkcije greške generator i diskriminator mreža u GAN modelu.....	626
Implementiranje GAN modela „od nule“	628
Obučavanje GAN modela na Google Colabu	628
Implementiranje mreža generatora i diskriminatora.....	631
Definisanje trening skupa podataka	636
Obučavanje GAN modela	638
Poboljšanje kvaliteta sintetizovanih slika upotrebom konvolutivne i Wasserstein GAN-e.....	646
Transponovana konvolucija.....	647
Grupna normalizacija	648
Implementiranje generatora i diskriminatora	651
Mere različitosti između dve distribucije.....	657
Upotreba EM rastojanja u praksi za GAN modele	661
Kazna gradijenta.....	662
Implementiranje WGAN-GP modela za obučavanje DCGAN modela	663
Mode collapse	667
Ostali načini primene GAN modela.....	669
Rezime	670

POGLAVLJE 18**Učenje uslovljavanjem za donošenje odluka u kompleksnim okruženjima 671**

Uvod - učenje iz iskustva.....	672
Razumevanje učenja uslovljavanjem	672
Definisanje interfejsa agenta-okruženja za sistem učenja uslovljavanjem	674
Teoretske osnove RL-a.....	676
Markov procesi odlučivanja	676
Matematička formulacija Markov procesa odlučivanja	677
Vizuelizacija Markov procesa	679
Epizodni, nasuprot kontinualnih zadataka	680
RL terminologija: povraćaj, strategija i funkcija vrednosti.....	680
Povraćaj	680
Strategija	682
Funkcija vrednosti.....	683
Dinamičko programiranje upotrebom Bellman jednačine	685
Algoritmi učenja uslovljavanjem	686
Dinamičko programiranje.....	686
Procena strategije - predviđanje funkcije vrednosti pomoću dinamičkog programiranja	687
Poboljšanje strategije upotrebom procenjene funkcije vrednosti	688
Iteracija strategije.....	688

Iteracija vrednosti.....	689
Učenje uslovljavanjem pomoću Monte Carlo metoda	689
Procena funkcije vrednosti stanja upotrebom MC metoda.....	690
Procena funkcije vrednosti akcije upotrebom MC metoda.....	690
Pronalaženje optimalne strategije upotrebom MC kontrole.....	691
Poboljšanje strategije - izračunavanje pohlepne strategije iz funkcije vrednosti akcije	691
Učenje razlike u vremenu	691
TD predikcija	692
TD kontrola u skladu sa strategijom (SARSA).....	693
TD kontrola mimo strategije (Q-learning)	694
Implementiranje našeg prvog RL algoritma.....	694
Predstavljanje OpenAI Gym paketa alatki.....	695
Upotreba postojećih okruženja u OpenAI Gym paketu.....	695
Grid world primer.....	697
Implementiranje grid world okruženja u OpenAI Gym.....	698
Rešavanje grid world problema pomoću Q-learning algoritma.....	705
Implementiranje Q-learning algoritma	705
Provirimo u dubinu Q-learning algoritma	709
Treniranje DQN modela u skladu sa Q-learning algoritmom.....	710
Implementiranje deep Q-learning algoritma.....	712
Rezime poglavlja i knjige.....	717

INDEKS	721
---------------------	------------



Uvod

Verovatno vam je poznata činjenica da je mašinsko učenje postalo jedna od najuzbudljivijih tehnologija našeg vremena. Velike kompanije, kao što su „Google“, „Facebook“, „Apple“, „Amazon“ i IBM sasvim razumljivo investiraju u njegovo istraživanje i u njegovu primenu. Ova uzbudljiva oblast otvara put novim mogućnostima i postala je neophodna u svakodnevnom životu. Razmislite samo o razgovoru sa glasovnim pomoćnikom na pametnom telefonu, preporučivanju odgovarajućih proizvoda za kupce, sprečavanju prevare kreditnim karticama, filtriranju spam poruka iz elektronskog poštanskog sandučeta i detektovanju i dijagnostikovanju bolesti i tako dalje.

POČETAK UPOTREBE MAŠINSKOG UČENJA

Ako želite da postanete praktikant mašinskog učenja ili da bolje rešavate probleme koji se javljaju, ili možda čak razmišljate o karijeri u istraživanju mašinskog učenja, onda je ova knjiga za vas! Za nove korisnike teorijski koncepti mašinskog učenja mogu da budu previše teški, ali izdato je mnogo knjiga poslednjih godina koje će vam pomoći da započnete da koristite mašinsko učenje implementiranjem moćnih algoritama učenja.

PRAKSA I TEORIJA

Pregledanje praktičnih primera koda i izrada primera primene mašinskog učenja odlični su načini da započnete učenje ove oblasti. Osim toga, konkretni primeri pomažu da se ilustruju širi koncepti postavljanjem naučenog materijala o mašinskom učenju direktno u akciju. Međutim, ne zaboravite da sa velikom moći dolazi i veća odgovornost!

U ovoj knjizi, osim što ćemo vam pomoći da steknete praktično iskustvo u mašinskom učenju upotrebom Python programskog jezika i biblioteka mašinskog učenja zasnovanih na Pythonu, predstavimo i matematičke koncepte u algoritmima mašinskog učenja, koji su važni za uspešnu upotrebu mašinskog učenja. Prema tome, ova knjiga se razlikuje od čisto praktične knjige; u njoj su opisani potrebni detalji u vezi konceptata mašinskog učenja i obezbeđena su intuitivna i informativna objašnjenja kako funkcionišu algoritmi mašinskog učenja, kako da ih upotrebite i, najvažnije, kako da izbegnete najčešće zamke.

ZAŠTO PYTHON?

Pre nego što „zaronimo“ dublje u oblast mašinskog učenja, odgovorićemo na pitanje zašto Python. Odgovor je jednostavan: Python je moćan i veoma je pristupačan. Postao je najpopularniji programski jezik za istraživanje podataka, zato što omogućava da zaboravimo dosadne delove programiranja i obezbeđuje okruženje u kojem možemo brzo da unesemo svoje ideje i da koncepte direktno sprovedemo u delo.

ISTRAŽIVANJE OBLASTI MAŠINSKOG UČENJA

Ako ukucate „machine learning“ kao termin pretrage u Google Scholar, biće prikazan ogroman broj rezultata - 3.250.000 publikacija. Naravno, ne možemo da razgovaramo o svim sitnim detaljima svih algoritama i primene koji su se pojavili u poslednjih 60 godina. Međutim, u ovoj knjizi ćemo krenuti na uzbudljivo „putovanje“ i opisaćemo sve važne teme i koncepte da bismo vam pomogli da započnete rad u ovoj oblasti. Ako otkrijete da vaša „glad“ za znanjem nije zadovoljena, možete da upotrebite mnoge korisne resurse koji su referencirani u ovoj knjizi da biste pratili najnovija dostignuća u ovoj oblasti.

Mi, autori, možemo iskreno reći da nas je istraživanje mašinskog učenja učinilo boljim naučnicima - bolje razmišljamo i bolje rešavamo probleme. U ovoj knjizi želimo stečeno znanje da podelimo sa vama. Znanje se stiče učenjem. Ključ za to je entuzijazam, a stvarno ovladavanje veštinama može se postići samo kroz praksu.

„Put“ koji je pred nama može povremeno biti veoma težak i neke teme mogu predstavljati veći izazov od drugih, ali se nadamo da ćete iskoristiti ovu mogućnost i fokusirati se na nagradu - na znanje koje ćete steći. Ne zaboravite da smo zajedno na ovom „putovanju“ i pomoću ove knjige dodaćemo mnogo moćnih tehnika vašem arsenalu, koje će vam pomoći da rešavate čak i najteže probleme na način vođen podacima.

ZA KOGA JE OVA KNJIGA

Ako ste već detaljno istražili teoriju mašinskog učenja, ova knjiga će vam pokazati kako da stečeno znanje sprovedete u delo. Ako ste koristili ranije tehnike mašinskog učenja i želite da steknete uvid u način funkcionisanja mašinskog učenja, ova knjiga je takođe za vas.

Ne brinite ako ste potpuno novi u oblasti mašinskog učenja; čak imate zbog toga više razloga da budete uzbuđeni! Obećavamo da će mašinsko učenje promeniti način na koji razmišljate o problemima koje želite da rešite i pokazaće vam kako da ih rešite otkrivanjem moći podataka. Ako želite da saznate kako da upotrebite Python da biste započeli da odgovarate na važna pitanja o vašim podacima, kupite knjigu „Python mašinsko učenje“. Bez obzira da li želite da započnete „od nule“ ili da proširite svoje znanje o istraživanju podataka, ovo je važan resurs koji ne smete propustiti.

ŠTA OBUHVATA OVA KNJIGA

U Poglavlju 1, „Kako da računarima pružite mogućnost da uče iz podataka“, predstavimo glavne podoblasti mašinskog učenja koje se koriste za rešavanje različitih problema. Osim toga, opisaćemo osnovne korake za kreiranje tipične protočne obrade izgradnje modela mašinskog učenja, koji će nas pratiti u narednim poglavljima.

U Poglavlju 2, „Treniranje jednostavnih algoritama mašinskog učenja za klasifikaciju“, vraćamo se na početke mašinskog učenja i predstavljamo binarne klasifikatore perceptrona i adaptivne linearne neurone. Predstavimo osnove klasifikacije obrazaca i fokusiraćemo se na interakciju algoritama optimizacije i mašinskog učenja.

U Poglavlju 3, „Predstavljanje klasifikatora mašinskog učenja pomoću scikit-learn“, opisaćemo važne algoritme mašinskog učenja za klasifikaciju i obezbedićemo praktične primere upotrebom scikit-learn, jedne od najpopularnijih i sveobuhvatnijih biblioteka mašinskog učenja otvorenog koda.

U Poglavlju 4, „Izgradnja dobrih skupova podataka za trening - pretprocesiranje podataka“, opisaćemo kako se rešavaju najčešći problemi u neobrađenim skupovima podataka, kao što je podatak koji nedostaje. Takođe ćemo predstaviti nekoliko pristupa za identifikaciju najinformativnijih atributa u skupovima podataka i način kako se pripremaju promenljive različitih tipova kao pravilni unosi za algoritme mašinskog učenja.

U Poglavlju 5, „Kompresovanje podataka upotrebom redukcije dimenzionalnosti“, upoznaćete tehnike za redukciju broja atributa u skupovima podataka na manje skupove, uz zadržavanje većine njihovih korisnih i diskriminativnih informacija. Osim toga, opisaćemo standardni pristup za redukciju dimenzionalnosti

analizom glavnih komponenata i njihovim upoređivanjem sa nadgledanim tehnikama nelinearne transformacije.

U Poglavlju 6, „Učenje najbolje prakse za procenu modela i podešavanje hiperparametara“, saznaćete šta treba, a šta ne treba da radite za procenu performanse prediktivnih modela. Upoznaćete i različite metrike za merenje performanse modela i tehnika za fino podešavanje algoritama mašinskog učenja.

U Poglavlju 7, „Kombinovanje različitih modela za ansambl metode“, predstavimo različite koncepte efikasnog kombinovanja većeg broja algoritama učenja. Istražićemo kako se grade ansambl stručnjaka, koji će prevazići slabosti pojedinačnih učenika, što dovodi do tačnijih i pouzdanijih predviđanja.

U Poglavlju 8, „Primena mašinskog učenja na analizu sentimenta“, opisaćemo osnovne korake za transformisanje tekstualnih podataka u smislene reprezentacije za algoritme mašinskog učenja za predviđanje mišljenja ljudi na osnovu njihovog pisanja.

U Poglavlju 9, „Ugrađivanje modela mašinskog učenja u veb aplikacije“, nastavićemo upotrebu prediktivnog modela iz prethodnog poglavlja i vodićemo vas kroz osnovne korake razvoja veb aplikacija sa ugrađenim modelima mašinskog učenja.

U Poglavlju 10, „Predviđanje kontinualnih ciljnih promenljivih pomoću analize regresije“, opisaćemo osnovne tehnike za modelovanje linearnog odnosa između cilja i promenljivih odgovora za izvršavanje kontinualnog predviđanja. Nakon predstavljanja različitih linearnih modela, biće reči o polinomnoj regresiji i pristupima zasnovanim na stablu.

U Poglavlju 11, „Upotreba neoznačenih podataka - analiza grupisanja“, fokus prebacujemo na različite podoblasti mašinskog učenja, odnosno na nenadgledano učenje. Opisaćemo algoritme iz tri osnovne familije algoritama za grupisanje, koji pronalaze grupe objekata i dele određeni stepen sličnosti.

U Poglavlju 12, „Implementiranje višeslojnih veštačkih neuronskih mreža 'od nule'“, proširićemo koncept optimizacije zasnovane na gradijentu, koju smo predstavili u Poglavlju 2. Izgradićemo moćne višeslojne neuronske mreže (NN) na osnovu popularnog algoritma propagacije greške unazad u Pythonu.

Poglavlje 13, „Paralelizacija treninga neuronske mreže pomoću TensorFlowa“, nadovezuje se na znanje stečeno u prethodnom poglavlju za obezbeđivanje efikasnijeg praktičnog vodiča za trening NN-a. Fokus u ovom poglavlju je na TensorFlowu 2.0, Python biblioteci otvorenog koda koja omogućava da iskoristimo više jezgara modernih procesora (GPU-a) i konstruišemo duboke NN-e iz zajedničkih gradivnih blokova pomoću jednostavnog Keras API-a.

U Poglavlju 14, „Detaljnije - mehanika TensorFlowa“, nastavićemo razmatranje teme iz prethodnog poglavlja i predstavimo naprednije koncepte i funkcionalnosti TensorFlowa 2.0. TensorFlow je izuzetno velika i sofisticirana biblioteka i u

ovom poglavlju ćemo vas provesti kroz koncepte, kao što su kompajliranje koda u statičke grafove za brže izvršenje i definisanje parametara modela koji se mogu trenirati. Osim toga, obezbedićemo dodatnu praktičnu vežbu treniranja dubokih neuronskih mreža upotrebom Keras API-a TensorFlowa, kao i unapred definisanih Estimatora Tensor Flowa.

U Poglavlju 15, „Klasifikovanje slika pomoću dubokih konvolucionih neuronskih mreža“, predstavimo konvolucione neuronske mreže (CNN). CNN predstavlja određeni tip duboke NN arhitekture koja je posebno dobro prilagođena skupovima podataka slika. Zbog svoje superiorne performanse u odnosu na tradicionalne pristupe, CNN se sada koristi u računarskom vidu za postizanje vrhunskih rezultata za različite zadatke prepoznavanja slika. U ovom poglavlju ćete naučiti kako konvolucioni slojevi mogu da se upotrebe kao moćni ekstraktori atributa za klasifikaciju slika.

U Poglavlju 16, „Modelovanje sekvencijalnih podataka upotrebom rekurentnih neuronskih mreža“, upoznaćete još jednu popularnu NN arhitekturu za duboko učenje, koja je posebno dobro prilagođena za upotrebu teksta i drugih tipova sekvencijalnih podataka i podataka vremenskih serija. Kao vežbu zagrevanja, u ovom poglavlju predstavimo rekurentnu NN za predviđanje sentimenta recenzija filmova. Zatim ćemo opisati učenje rekurentnih mreža da prebacuju informacije iz knjiga da bi generisale potpuno novi tekst.

U Poglavlju 17, „Generativne suparničke mreže za sintetizovanje novih podataka“, predstavimo popularni suparnički trening režim za NN-e koji može da se upotrebi za generisanje novih slika realističnog izgleda. Poglavlje ćemo započeti kratkim uvodom u autoenkodere koji su poseban tip NN arhitekture koji može da se upotrebi za kompresovanje podataka. Zatim ćemo prikazati kako se kombinuje deo dekodera autoenkodera sa drugim NN, koji može da razlikuje stvarne i sintetizovane slike. Omogućavanjem nadmetanja dve NN u pristupu suparničkog treninga implementiraćemo generativnu suparničku mrežu koja generiše nove ručno pisane cifre. Na kraju, nakon predstavljanja osnovnih koncepata generativnih suparničkih mreža, predstavimo i poboljšanja koja mogu da stabilizuju suparnički trening, kao što je upotreba Wasserstein metrika udaljenosti.

U Poglavlju 18, „Učenje uslovljavanjem za donošenje odluka u kompleksnim okruženjima“, obuhvatićemo potkategoriju mašinskog učenja koja se često koristi za treniranje robota i drugih autonomnih sistema. Prvo ćemo predstaviti osnove učenja uslovljavanjem (RL) da biste upoznali interakcije agenta/okruženja procesom nagrađivanja RL sistema i konceptom učenja iz iskustva. Obuhvatićemo dve glavne kategorije RL-a: RL koji je zasnovan na modelu i RL bez modela. Nakon što naučite osnovne algoritamske pristupe, kao što su Monte Carlo i vremensko učenje zasnovano na udaljenosti, implementiraćete i trenirati agenta koji može da se kreće kroz mrežu okruženja upotrebom Q-learning algoritma. Na kraju ćemo predstaviti duboki Q-learning algoritam koji je varijanta Q-learning algoritma koji koristi duboke NN-e.

ŠTA VAM JE POTREBNO ZA OVU KNJIGU?

Izvršenje primera koda koji je obezbeđen u ovoj knjizi zahteva instalaciju Pythona 3.7.0 ili noviju verziju na macOS, Linux ili Microsoft Windows sistemu. U ovoj knjizi ćemo često za naučna izračunavanja koristiti Pythonove osnovne biblioteke, uključujući SciPy, NumPy, scikit-learn, Matplotlib i pandas.

U prvom poglavlju ćemo obezbediti instrukcije i korisne savete za podešavanje Python okruženja i ovih osnovnih biblioteka. Dodaćemo i neke dodatne biblioteke u repertoar, a instrukcije za instalaciju su obezbeđene u odgovarajućim poglavljima - na primer, NLTK biblioteka za obradu prirodnog jezika u Poglavlju 8, „Primenjena mašinskog učenja za analizu sentimenta“, Flask veb radni okvir u Poglavlju 9, „Ugrađivanje modela mašinskog učenja u veb aplikacije“, i TensorFlow za efikasan NN trening na GPU-ima u poglavljima od 13-18.

DA BISTE DOBILI MAKSIMUM IZ OVE KNJIGE

Sada, kada ste ponosni vlasnik „Packt“ knjige, imamo mnogo štošta da vam ponudimo da bismo vam pomogli da dobijete maksimum iz nje.

Preuzimanje primera koda

Kod za knjigu možete da preuzmete sa našeg sajta: <https://bit.ly/3aIMppS>

Kada su fajlovi preuzeti, raspakujte ili ekstrahujte direktorijum, koristeći najnoviju verziju:

- **WinRAR** / 7-Zip za Windows
- **Zipeg** / iZip / UnRarX za Mac
- **7-Zip** / PeaZip za Linux

Preuzimanje kolornih slika

Takođe smo obezbedili PDF fajl koji sadrži kolorne slike ekrana/dijagrama koji su upotrebljeni u knjizi. Kolorne slike će vam pomoći da bolje razumete promene u ispisima. Možete da preuzmete ovaj fajl sa adrese:

<https://bit.ly/2wYOM9x>

Pored toga, kolorne slike niže rezolucije su ugrađene u code notebook za ovu knjigu, koji se nalazi u fajlovima primera koda.

KONVENCIJE

U ovoj knjizi pronaći ćete više različitih stilova za tekst upotrebljenih za različite vrste informacija. Evo nekih primera tih stilova i objašnjenja njihovog značenja.

Reči koda u tekstu su prikazane na sledeći način: „A već instalirani paketi mogu da budu ažurirani pomoću oznake `-upgrade`“.

Blok koda je postavljen na sledeći način:

```
>>> 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()
```

Svi unosi ili ispisi komandne linije napisani su na sledeći način:

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

Novi termini i važne reči su napisani podebljanim slovima. Reči koje vidite na ekranu - na primer, u menijima ili okvirima za dijalog, prikazane su u tekstu na sledeći način: „Kliknite na dugme **Next** da biste se prebacili na sledeći ekran“.



Upozorenja ili važne **napomene** se prikazuju u ovakvom okviru.



Saveti i trikovi se prikazuju ovako.

STUPITE U KONTAKT

Povratne informacije naših čitalaca su uvek dobrodošle.

Opšte povratne informacije: Ako imate pitanja o bilo kom aspektu ove knjige, pošaljite nam email na adresu kombib@gmail.com.

Štamparske greške: Iako smo preduzeli sve mere da bismo obezbedili tačnost sadržaja, greške su moguće. Ako pronađete grešku u nekoj od naših knjiga - u tekstu ili u kodu, bili bismo zahvalni ako biste nam to javili. Na taj način možete da pomognete drugim čitaocima da izbegnu frustracije i pomognete nama da poboljšamo sledeće verzije ove knjige. Ako pronađete grešku, molimo vas da nas o tome obavestite na email kombib@gmail.com.

Piraterija: Ako pronađete ilegalnu kopiju naših knjiga, u bilo kojoj formi na Internetu, molimo vas da nas o tome obavestite i da nam pošaljete adresu lokacije ili naziv veb sajta. Pošaljite nam poruku na adresu kombib@gmail.com i pošaljite nam link ka sumnjivom materijalu.

PREGLEDI

Kada pročitate ovu knjigu i uradite vežbe, zašto ne biste napisali vaše mišljenje na sajtu sa kojeg ste je poručili? Potencijalni čitaoci mogu da pročitaju vaše mišljenje i upotrebe ga pri donošenju odluke o kupovini, mi u „Kompjuter biblioteci“ ćemo saznati šta mislite o našim proizvodima, a naši autori mogu da vide povratne informacije o svojoj knjizi.

Predlozi za prevod

Oni koji kupuju naša izdanja su nam, prethodnih godina, veoma pomagali da izaberemo knjigu za prevod na srpski jezik.

To možete da uradite i vi. Posetite stranu predloga za prevod:

<http://bit.ly/2NVhHRg>

i ukoliko je među knjigama koje smo ponudili i ona koja je vama potrebna, napišite komentar. Svaki komentar ćemo nagraditi.

A ukoliko među knjigama koje smo ponudili nema one koja je vama potrebna, pošaljite nam mail sa vašim predlogom na kombib@gmail.com. Ukratko objasnite zašto bi baš ta knjiga bila zanimljiva, a ukoliko je budemo objavili vi ćete dobiti knjigu na poklon.



Postanite član Kompjuter biblioteke

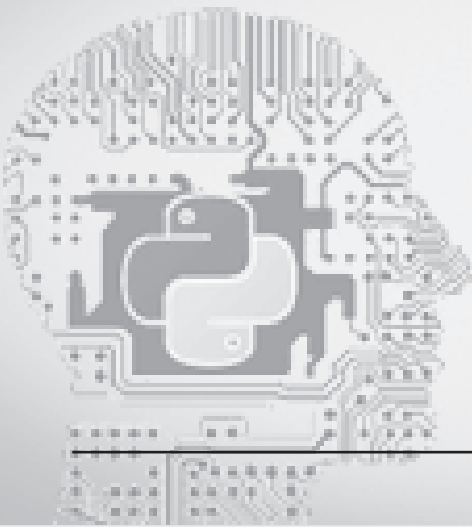
Kupovinom jedne naše knjige stekli ste pravo da postanete član Kompjuter biblioteke. Kao član možete da kupujete knjige u pretplati sa 40% popusta i učestvujete u akcijama kada ostvarujete popuste na sva naša izdanja.

Potrebno je samo da se prijavite preko formulara na našem sajtu.

Link za prijavu: <http://bit.ly/2TxeK5a>

Skenirajte QR kod
registrujte knjigu
i osvojite nagradu





1

Kako da računarima pružite mogućnost da uče iz podataka

Po mom mišljenju, mašinsko učenje, primena i nauka o algoritmima koji imaju smisla za podatke predstavljaju najuzbudljiviju oblast od svih računarskih nauka! Živimo u doba kada postoji ogroman broj podataka; upotrebom samostalno naučenih algoritama iz oblasti mašinskog učenja možemo da pretvorimo ove podatke u znanje. Zahvaljujući mnogobrojnim moćnim bibliotekama otvorenog koda koje su razvijene poslednjih godina, verovatno nikada nije bilo bolje vreme za proboj u oblast mašinskog učenja i za učenje kako da iskoristite moć algoritama za pronalaženje obrazaca u podacima i za predviđanja o budućim događajima.

U ovom poglavlju ćete učiti o glavnim konceptima i različitim tipovima mašinskog učenja. Zajedno sa osnovnim uvodom u relevantnu terminologiju, postavićemo temelj za uspešnu upotrebu tehnika mašinskog učenja za praktično rešavanje problema.

Obrađićemo sledeće teme:

- osnovni koncepti mašinskog učenja
- tri tipa učenja i osnovna terminologija
- gradivni blokovi za uspešno projektovanje sistema mašinskog učenja
- instaliranje i podešavanje Pythona za analizu podataka i za mašinsko učenje

IZGRADNJA INTELIGENTNIH MAŠINA ZA TRANSFORMISANJE PODATAKA U ZNANJE

U ovo doba moderne tehnologije dostupna nam je velika količina strukturiranih i nestrukturiranih podataka. U drugoj polovini 20. veka mašinsko učenje se razvijalo kao podoblast veštačke inteligencije (AI) i uključivalo je algoritme samoučenja koji izvode znanje iz podataka za predviđanja.

Umesto potrebe da ljudi ručno izvode pravila i grade modele analiziranjem velike količine podataka, mašinsko učenje obezbeđuje efikasniju alternativu za čuvanje znanja u podacima za postepeno poboljšanje performanse prediktivnih modela i donošenje odluka vođenih podacima.

Ne samo da mašinsko učenje postaje sve važnije u istraživanjima računarske nauke, već igra ogromnu ulogu i u svakodnevnom životu. Zahvaljujući mašinskom učenju, mi uživamo u robusnim filterima neželjene pošte, softverima za prepoznavanje teksta i glasa, pouzdanim veb pretraživačima i izazovnim programima za igranje šaha. Nadamo se da ćemo uskoro imati i bezbedne i efikasne samovozeće automobile na ovoj listi. Osim toga, primetan je napredak i u primeni u medicini; na primer, istraživači su predstavili da modeli dubokog učenja mogu da detektuju rak kože skoro ljudskom tačnošću (<https://www.nature.com/articles/nature21056>). Veliki napredak su nedavno postigli i istraživači u DeepMindu, koji su koristili duboko učenje za predviđanje 3D proteinskih struktura i prvi put su nadmašili rezultate pristupa zasnovanih na fizici (<https://deepmind.com/blog/alphafold/>).

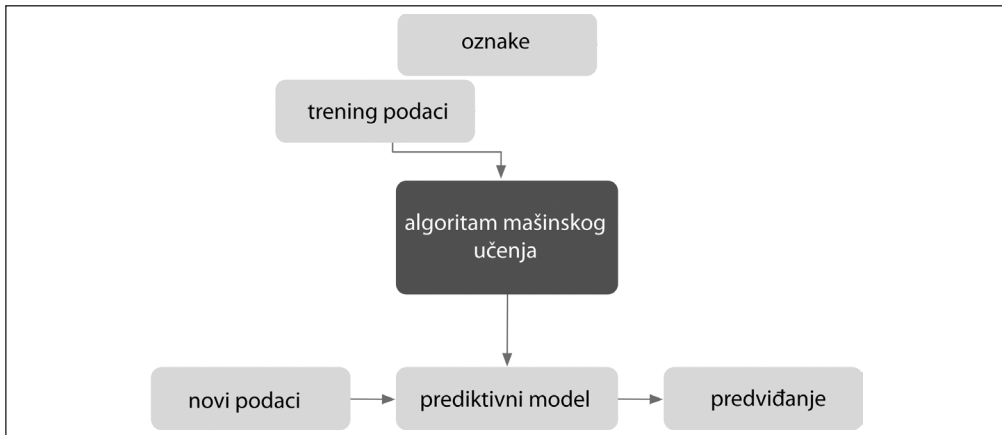
TRI RAZLIČITA TIPA MAŠINSKOG UČENJA

U ovom odeljku ćemo opisati tri tipa mašinskog učenja: **nadgledano učenje**, **nenadgledano učenje** i **učenje uslovljavanjem**. Učićemo o osnovnim razlikama između ta tri tipa i upotrebom konceptualnih primera ćemo razviti razumevanje domena praktičnih problema:

Nadgledano učenje	<ul style="list-style-type: none"> • označeni podacima • direktne povratne informacije • ishod/budućnost predviđanja
Nenadgledano učenje	<ul style="list-style-type: none"> • nema oznaka • nema povratnih informacija • otkrivena skrivena struktura u podacima
Učenje uslovljavanjem	<ul style="list-style-type: none"> • proces odlučivanja • sistem nagrađivanja • serije akcija za učenje

Predviđanje budućnosti pomoću nadgledanog učenja

Glavni cilj nadgledanog učenja je da obučimo model iz označenih trening podataka koji omogućavaju da izvršimo predviđanje o neviđenim ili budućim podacima. Ovde se termin „nadgledani“ odnosi na skup trening primera (ulaznih podataka) gde su signali željenog izlaza (oznake) već poznati. Na sledećoj slici rezimiran je tipičan tok rada nadgledanog učenja, u kojem su prosleđeni trening podaci u algoritam mašinskog učenja za usklađivanje prediktivnog modela koji može da izvrši predviđanje na novim, neoznačenim ulaznim podacima:

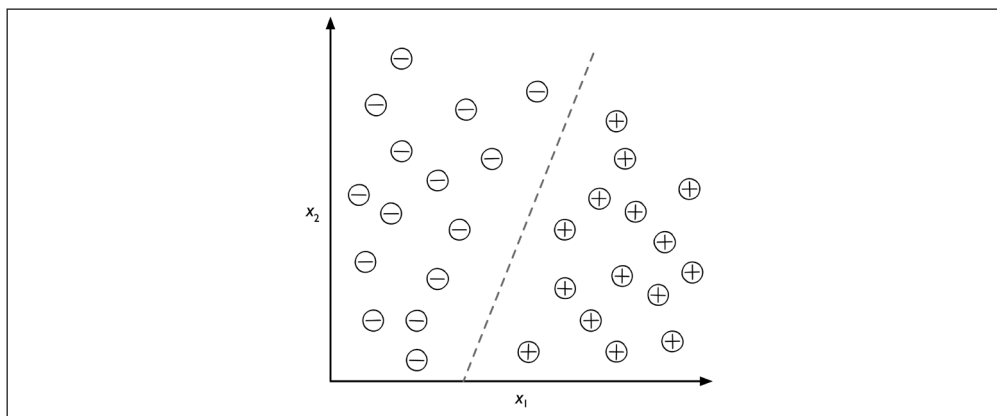


Razmatranjem primera filtriranja neželjene elektronske pošte možemo da obučimo model upotrebom algoritma nadgledanog mašinskog učenja u grupi označenih e-mailova, koji su korektno označeni kao spam ili ne-spam, za predviđanje da li novi e-mail pripada jednoj od ove dve kategorije. Zadatak nadgledanog učenja sa diskretnim oznakama klase, kao što je prethodni primer filtriranja neželjene pošte, naziva se i zadatak klasifikacije. Još jedna potkategorija nadgledanog učenja je **regresija**, gde je izlazni signal kontinualna vrednost.

Klasifikacija za predviđanje oznaka klase

Klasifikacija je potklasa nadgledanog učenja gde je cilj predvideti kategoričke klase oznake novih instanci na osnovu ranijih opažanja. Te oznake klase su diskretne, neuređene vrednosti koje mogu da se razumeju kao grupno članstvo instanci. Prethodno pomenuti primer detekcije neželjene pošte predstavlja tipičan primer zadatka binarne klasifikacije, gde algoritam mašinskog učenja uči skup pravila da bi razlikovao dve moguće klase: spam i ne-spam e-mail poruke.

Na sledećoj slici prikazan je koncept zadatka binarne klasifikacije gde je dato 30 trening primera: 15 je označeno kao negativna klasa (znak minus), a 15 kao pozitivna klasa (znak plus). U ovom scenariju naš skup podataka je dvodimenzionalan, što znači da svaki primer ima dve povezane vrednosti: x_1 i x_2 . Sada možemo da upotrebimo algoritam nadgledanog mašinskog učenja za učenje pravila (granica odluke predstavljena je isprekidanom linijom) koje može da razdvoji te dve klase i klasifikuje nove podatke u svaku od te dve kategorije, uzimajući u obzir njihove x_1 i x_2 vrednosti:



Međutim, skup oznaka klase ne treba da bude binaran. Prediktivni model obučen pomoću algoritma nadgledanog učenja može da dodeli bilo koju oznaku klase, koja je predstavljena u skupu podataka za trening, novoj i neoznačenoj instanci.

Tipičan primer zadatka **više-klasne klasifikacije** je prepoznavanje ručno pisane karaktera. Možemo da sakupimo skup podataka za trening, koji se sastoji od više ručno pisanih primera svakog slova u abecedi. Slova (A, B, C i tako dalje) će predstavljati različite neuređene kategorije ili oznake klase koje želimo da predvidimo. Ako korisnik unese novi ručno pisani karakter pomoću uređaja za unos, naš prediktivni model će moći da predvidi tačno slovo u abecedi sa određenom tačnošću. Međutim, naš sistem mašinskog učenja neće moći tačno da prepozna bilo koju od cifara između 0 i 9 ako nisu deo skupa podataka za trening.

Regresija za predviđanje neprekidnog ishoda

U prethodnom odeljku ste naučili da je zadatak klasifikacije dodela kategoričkih, neuređenih oznaka instancama. Drugi tip nadgledanog učenja je predviđanje neprekidnog ishoda, a naziva se i regresiona analiza. U **regresionoj analizi** dati su broj prediktor promenljivih (**istraživanje**) i neprekidna ciljna promenljiva (**ishod**) i pokušavamo da pronađemo odnos između tih promenljivih koje nam omogućavaju da predvidimo ishod.

Imajte na umu da se u oblasti mašinskog učenja prediktor promenljive obično nazivaju atributi, a promenljive odgovora se obično nazivaju ciljne promenljive. Mi ćemo usvojiti te konvencije u ovoj knjizi.

Na primer, pretpostavimo da smo zainteresovani za predviđanje uspešnosti studenata na ispitu. Ako postoji veza između vremena utrošenog u učenju za test i konačnih rezultata, možemo da je upotrebimo kao podatke za obučavanje modela koji koristi vreme učenja za predviđanje rezultata testa budućih studenata koji planiraju da ga polažu.

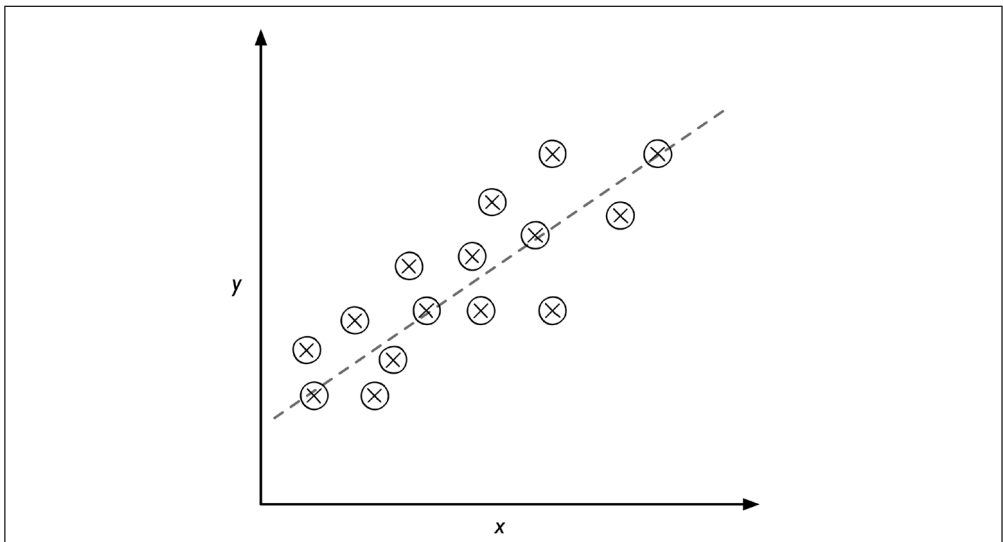
Regresija prema srednjoj vrednosti



Termin regresija osmislio je 1886. godine Francis Galton u svom članku „Regression towards Mediocrity in Hereditary Stature“. Galton je opisao biološki fenomen da se variranje visine populacije ne uvećava vremenom.

Primetio je da se visina roditelja ne prenosi na njihovu decu, već visina njihove dece napreduje ka srednjoj vrednosti populacije.

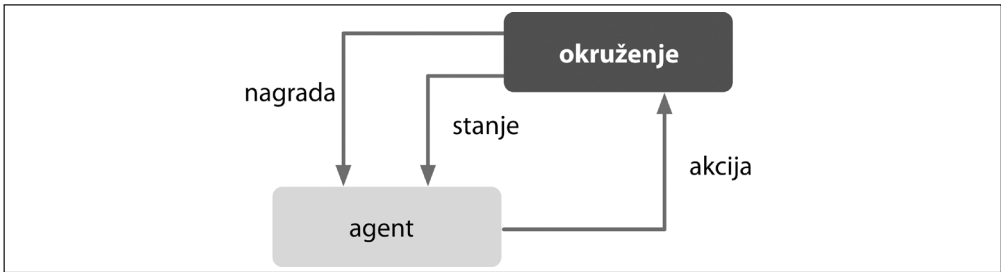
Na sledećoj slici ilustrovan je koncept linearne regresije. Dati su atribut x i ciljna promenljiva y . Na osnovu njih ćemo uklopiti pravu liniju za ove podatke, koja minimizira odstupanje (obično je to prosečan kvadrat odstupanja) između tačaka podataka i uklopljene linije. Sada možemo da upotrebimo presek i nagib za predviđanje ciljne promenljive novih podataka:



Rešavanje interaktivnih problema pomoću učenja uslovljavanjem

Još jedan tip mašinskog učenja je **učenje uslovljavanjem**, u kojem je cilj razviti sistem (**agent**) koji poboljšava svoju performansu na osnovu interakcija sa okruženjem. Pošto informacije o aktuelnom stanju okruženja obično uključuju takozvani signal nagrade, možemo da zamislimo učenje uslovljavanjem kao oblast koja se odnosi na **nadgledano učenje**. Međutim, u učenju uslovljavanjem ova povratna informacija nije tačna oznaka ili vrednost istine, već je to mera koliko dobro je funkcija nagrade izmerila akciju. Kroz interakciju sa okruženjem agent može da upotrebi učenje uslovljavanjem za obučavanje serije akcija koje maksimiziraju ovu nagradu pomoću istraživačkog pristupa pokušaja i greške ili promišljenim planiranjem.

Popularan primer učenja uslovljavanjem je šah. Ovde agent odlučuje o serijama poteza, u zavisnosti od stanja na tabli (okruženje), a nagrada će biti definisana kao pobjeda ili poraz na kraju igre:



Postoje mnogi različiti podtipovi učenja uslovljavanjem. Međutim, osnovna šema je da agent u učenju uslovljavanjem pokuša da maksimizuje nagradu kroz serije interakcija sa okruženjem. Svako stanje može da bude povezano sa pozitivnom ili negativnom nagradom, a nagrada može da bude definisana kao postizanje uopštenog cilja, kao što su pobjeda ili poraz u igri šaha. Na primer, u šahu ishod svakog poteza može da se zamisli kao različito stanje okruženja.

Da bismo dalje istražili primer šaha, razmislimo o određenim konfiguracijama na šahovskoj tabli koje su povezane sa stanjima koja će najverovatnije dovesti do pobjede - na primer, uklanjanje šahovske figure protivnika sa table ili pretnja kraljici. Međutim, druge pozicije su povezane sa stanjima koja će najverovatnije dovesti do poraza u igri, kao što je gubitak šahovske figure u korist protivnika u sledećem potezu. Sada nagrada (pozitivna za pobjedu, ili negativna za gubitak partije) neće biti data do kraja igre. Osim toga, finalna nagrada će takođe zavisiti od toga kako protivnik igra. Na primer, protivnik može da žrtvuje kraljicu, ali na kraju da pobjedi u igri.

Učenje uslovljavanjem se bavi učenjem odabira serije akcija koje maksimiziraju ukupnu nagradu, koja može da se dobije ili odmah nakon izvršavanja akcije ili pomoću odložene *povratne* informacije.

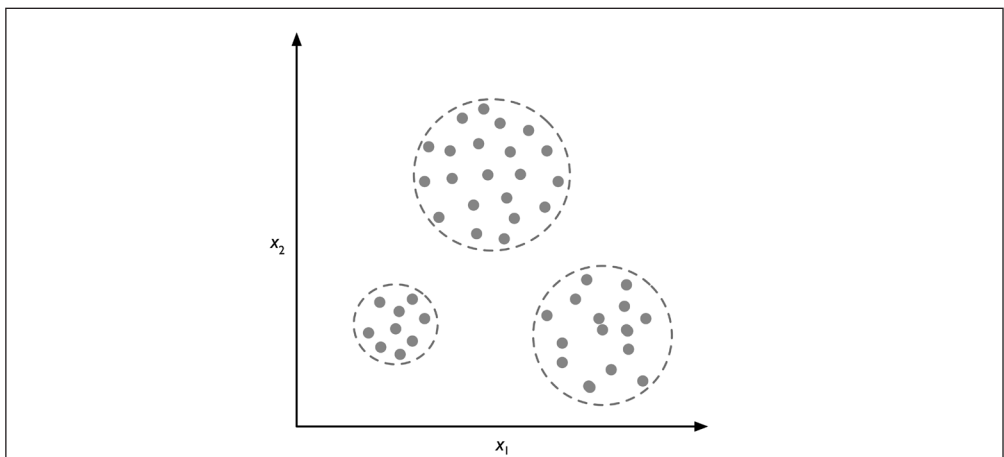
Otkrivanje skrivenih struktura pomoću nenadgledanog učenja

U nadgledanom učenju unapred znamo tačan odgovor kada obučavamo model, a u učenju uslovljavanjem definišemo meru nagrade za određene akcije koje izvršava agent. Međutim, u nenadgledanom učenju koristimo neoznačene podatke ili podatke nepoznate strukture. Upotrebom tehnika nenadgledanog učenja možemo da istražimo strukturu podataka za izdvajanje značajnih informacija, bez smernica poznate promenljive ishoda ili funkcije nagrade.

Pronalaženje podgrupa pomoću klasterovanja

Klasterovanje je istraživačka tehnika analize podataka koja omogućava da organizujemo grupu informacija u značajne podgrupe (**klaster**), bez potrebe da imamo neko predznanje o članovima grupe. Svaki klaster koji se javlja u toku analize definiše grupu objekata koji dele određeni stepen sličnosti, ali su više različiti od objekata u drugim klasterima; zbog toga se klasterovanje ponekad naziva i **nenadgledana klasifikacija**. Klasterovanje je odlična tehnika za strukturiranje informacija i za izvođenje značajnih odnosa algoritmom iz podataka. Na primer, omogućava trgovcima da otkriju grupe kupaca na osnovu njihovog interesovanja da bi razvili različite marketinške programe.

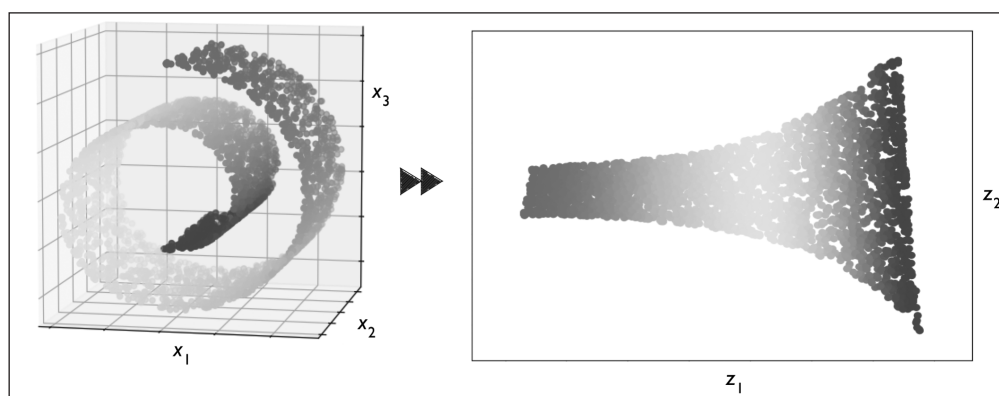
Na sledećoj slici prikazano je kako klasterovanje može da se primeni za organizovanje neoznačenih podataka u tri različite grupe na osnovu sličnosti njihovih atributa x_1 i x_2 :



Redukcija dimenzionalnosti za kompresovanje podataka

Još jedna podoblast nenadgledanog učenja je **redukcija dimenzionalnosti**. Često koristimo podatke visoke dimenzionalnosti (svako opažanje uključuje visok broj merenja), što može predstavljati izazov za ograničavanje prostora za skladištenje i performanse izračunavanja za algoritme mašinskog učenja. Nenadgledana redukcija dimenzionalnosti je pristup koji se uobičajeno koristi u pretprocesiranju atributa za uklanjanje šuma iz podataka, koji takođe može da degradira prediktivnu performansu određenih algoritama, i za kompresovanje podataka u manje dimenzionalne podoblasti dok se zadržavaju najrelevantnije informacije.

Redukcija dimenzionalnosti može ponekad da bude korisna i za vizuelizaciju podataka; na primer visokodimenzionalni skup atributa može da bude projektovan u jednodimenzionalnim, dvodimenzionalnim ili trodimenzionalnim prostorima atributa da bi se vizuelizovali pomoću 2D ili 3D scatterplotova ili histograma. Na sledećoj slici prikazan je primer gde je primenjena nelinearna redukcija dimenzionalnosti za kompresovanje 3D Swiss Rolla u novi 2D potprostor atributa:



UVOD U OSNOVNU TERMINOLOGIJU I NOTACIJE

Sada, kada smo opisali tri široke kategorije mašinskog učenja - nadgledano, nanadgledano učenje i učenje uslovljavanjem, predstavice osnovnu terminologiju koju ćemo koristiti u ovoj knjizi. U sledećim odeljcima opisaćemo uobičajene termine koje ćemo koristiti kada govorimo o različitim aspektima skupa podataka i matematičke notacije koje će nam pomoći da preciznije i efikasnije komuniciramo.

Pošto je mašinsko učenje ogromna oblast i veoma je interdisciplinarno, pre ili kasnije sigurno ćete se susresti sa mnogo različitih termina koji se odnose na iste koncepte. Najčešće upotrebljavane termine koji se mogu pronaći u literaturi mašinskog učenja opisaćemo u drugom pododeljku, koji može biti koristan kao referentni odeljak kada čitate raznovrsnu literaturu o mašinskom učenju.

Notacije i konvencije upotrebljene u ovoj knjizi

U sledećoj tabeli prikazan je deo skupa podataka Iris, koji je klasičan primer u oblasti mašinskog učenja. Iris skup podataka sadrži mere 150 cvetova Iris iz tri različite vrste - Setosa, Versicolor i Virginica. Ovde svaki primer cveta predstavlja jedan red u skupu podataka, a mere cveta u santimetrima su sačuvane kao kolone, koje nazivamo i **atributi** skupa podataka:

uzorci (instance, opažanja)	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

latica

čaišni listić

oznake klase (ciljevi)

atributi (atributi, mere, dimenzije)

Da bi notacija i implementacija ostale jednostavne, a ipak efikasne, upotrebićemo neke osnove linearne algebre. U sledećim poglavljima ćemo upotrebiti matricu i vektor za referencu podataka. Pratićemo uobičajenu konvenciju za predstavljanje svakog primera kao posebnog reda u matrici atributa X , gde je svaki atribut sačuvan u posebnoj koloni.

Iris skup podataka se sastoji od 150 primera i četiri atributa i može da bude napisan kao 150×4 matrica, $\mathbf{X} \in \mathbb{R}^{150 \times 4}$:

$$\begin{bmatrix} x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & x_4^{(1)} \\ x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & x_4^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{(150)} & x_2^{(150)} & x_3^{(150)} & x_4^{(150)} \end{bmatrix}$$

Konvencije notacije

U ostatku ove knjige, osim ako je naznačeno drugačije, upotrebićemo superskript i da bismo ukazali na i-ti trening primer i indeks j da bismo ukazali na j-tu dimenziju skupa podataka za trening.

Upotrebićemo mala podebljana slova da bismo ukazali na vektore ($\mathbf{x} \in \mathbb{R}^{n \times 1}$), velika podebljana slova da bismo ukazali na matrice ($\mathbf{X} \in \mathbb{R}^{n \times m}$) i iskošena slova ($x^{(n)}$ ili $x_m^{(n)}$) da bismo ukazali na pojedinačne elemente u vektoru ili matrici.

Na primer, $x_1^{(150)}$ se odnosi na prvu dimenziju primera cveta 150, odnosno sepal length. Prema tome, matrica atributa predstavlja jednu instancu cveta i može da bude napisana kao četvorodimenzionalni neobrađeni vektor $\mathbf{x}^{(i)} \in \mathbb{R}^{1 \times 4}$:

$$\mathbf{x}^{(i)} = [x_1^{(i)} \quad x_2^{(i)} \quad x_3^{(i)} \quad x_4^{(i)}]$$



I svaka dimenzija atributa je 150-dimenzionalni vektor kolone $\mathbf{x}^{(i)} \in \mathbb{R}^{150 \times 1}$ - na primer:

$$\mathbf{x}_j = \begin{bmatrix} x_j^{(1)} \\ x_j^{(2)} \\ \dots \\ x_j^{(150)} \end{bmatrix}$$

Slično tome, sačuvaćemo ciljne promenljive (oznake klasa) kao 150-dimenzionalni vektor kolone:

$$\mathbf{y} = \begin{bmatrix} y^{(1)} \\ \dots \\ y^{(150)} \end{bmatrix} \quad (y \in \{\text{Setosa, Versicolor, Virginica}\})$$

Terminologija mašinskog učenja

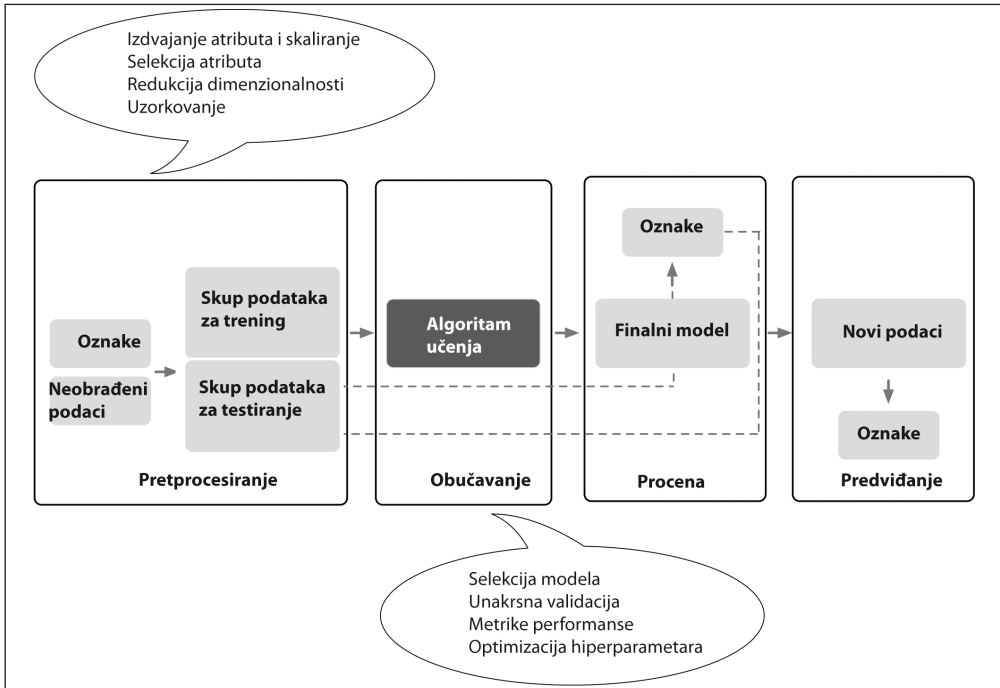
Mašinsko učenje je obimna oblast i veoma je interdisciplinarna, jer spaja mnoge naučnike iz drugih oblasti istraživanja. Kao što se često dešava, mnogi termini i koncepti su ponovo otkriveni ili redefinisani i možda su vam već biti poznati, ali se javljaju pod različitim nazivima. U sledećoj listi možete pronaći selekciju uobičajeno upotrebljenih termina i njihovih sinonima, koji će vam biti korisni dok čitate ovu knjigu i drugu literaturu o mašinskom učenju:

- trening primer - red u tabeli koji predstavlja skup podataka i sinonim je za opažanje, zapis, instancu ili uzorak (u većini konteksta, uzorak se odnosi na kolekciju trening primera)
- trening - uklapanje modela za parametarske modele slično proceni parametara
- atribut, skraćeno x - kolona u tabeli podataka ili matrici podataka (dizajn); sinonim je za prediktor, promenljivu, ulaz, atribut ili kovarijanse
- cilj, skraćeno y - sinonim za ishod, izlaz, promenljivu odgovora, zavisnu promenljivu, oznaku (klase) i ground truth
- funkcija greške - Često se koristi kao sinonim za cost funkciju. Ponekad se funkcija greške naziva i error funkcija. U nekoj literaturi termin greška se odnosi na grešku merenu iz jedne tačke podataka, a cost je mera koja izračunava grešku (prosečno ili zbirno) u celom skupu podataka.

MAPA ZA IZGRADNJU SISTEMA MAŠINSKOG UČENJA

U prethodnim odeljcima smo predstavili osnovne koncepte mašinskog učenja i tri različita tipa učenja. U ovom odeljku ćemo govoriti o drugim važnim delovima sistema mašinskog učenja, zajedno sa algoritmima učenja.

U sledećem dijagramu prikazan je tipičan tok rada za učenje mašinskog učenja u prediktivnom modelovanju, o čemu ćemo govoriti u sledećim pododeljcima:



Pretprocesiranje - oblikovanje podataka

Prvo ćemo opisati mapu za izgradnju sistema mašinskog učenja. Neobrađeni podaci retko dolaze u obliku koji je potreban za optimalnu performansu algoritma mašinskog učenja. Prema tome, pretprocesiranje podataka je jedan od najvažnijih koraka u svakoj primeni mašinskog učenja.

Ako, kao primer, upotrebimo skup podataka Iris cveta iz prethodnog odeljka, možemo da zamislamo neobrađene podatke kao seriju slika cveća, iz kojih možemo da izdvojimo značajne attribute. Korisni attribute mogu da budu boja, nijansa i intenzitet cvetova ili visina, dužina i širina cvetova.

Mnogi algoritmi mašinskog učenja takođe zahtevaju da su izabrani attribute na istoj skali za optimalnu performansu, što se često postiže transformisanjem attribute u rasponu $[0, 1]$ ili pomoću standardne normalne raspodele srednje vrednosti i jedinične varijanse, kao što ćete videti u sledećim poglavljima.

Neki od izabраних аtribute mogu da budu povezani i, prema tome, сувишни do određenog stepena. U tim slučajevima tehnike redukcije dimenzionalnosti su korisne za kompresovanje atributa u niže dimenzione potprostore. Redukcija dimenzionalnosti prostora atributa ima prednosti, jer je potrebno manje prostora za skladištenje, a algoritam obučavanja može da se pokreće mnogo brže. U određenim slučajevima redukcija dimenzionalnosti takođe može da poboljša prediktivnu performansu modela ako skup podataka sadrži велики broj nerelevantnih atributa (ili šum), odnosno ako skup podataka ima slab odnos signala i šuma.

Da bismo odredili da li se algoritam mašinskog učenja izvršava dobro u trening skupu podataka i da li se dobro generalizuje u novim podacima, takođe ćemo da nasumično razdvojimo skup podataka u poseban skup podataka za trening i skup podataka za testiranje. Upotrebicemo trening skup podataka za obučavanje i optimizaciju modela mašinskog učenja, dok ćemo zadržati skup podataka za testiranje do samog kraja za procenu finalnog modela.

Trening i selektovanje prediktivnog modela

Kao što ćete videti u narednim poglavljima, mnogo različitih algoritama mašinskog učenja je razvijeno za rešavanje različitih problema. Važna tačka koja se može rezimirati iz poznatog članka Davida Wolperta „No free lunch theorems“ je da ne možemo da učimo „besplatno“ („The Lack of A Priori Distinctions Between Learning Algorithms“, D. H. Wolpert, 1996; „No free lunch theorems for optimization“, D. H. Wolpert i W. G. Macready, 1997). Ovaj koncept možemo povezati sa popularnom izrekom Abrahama Maslowa „Ako vam je jedini alat čekić, pretpostavljam da je primamljivo da sve tretirate kao ekser“. Na primer, svaki algoritam klasifikacije ima svoje nerazdvojive pomake i ni jedan model klasifikacije nije superioran ako ne upotrebimo algoritam. Prema tome, u praksi je važno uporediti bar nekoliko različitih algoritama da bismo obučili i selektovali model sa najboljom performansom. Međutim, pre nego što uporedimo različite modele, prvo treba da donesemo odluku o metrici za merenje performanse. Jedna od uobičajeno upotrebljenih metrika je tačnost klasifikacije, koja je definisana kao udeo tačno klasifikovanih instanci.

Kako možemo da znamo koji se model dobro izvršava u finalnom skupu podataka za testiranje i u podacima iz realnog sveta ako ne koristimo ovaj skup podataka za testiranje za selekciju modela, već ga zadržavamo za procenu finalnog modela? Da bismo rešili taj problem, možemo upotrebiti različite tehnike koje se nazivaju unakrsna validacija. U unakrsnoj validaciji mi dalje razdvajamo skup podataka u podskupove za trening i validaciju da bismo procenili performansu generalizacije modela. Na kraju, takođe ne možemo da očekujemo da će podrazumevani parametri različitih algoritama za učenje koje obezbeđuju biblioteke softvera biti optimalni za specifičan

problem. Prema tome, često ćemo koristiti tehnike optimizacije hiperparametara koje nam pomažu da fino podesimo performansu modela u narednim poglavljima.

Možemo čak da zamislimo te hiperparametre kao parametre koji nisu naučeni iz podataka, već predstavljaju ručice modela koje možemo da okrenemo da bismo poboljšali performansu konkretnog modela. Sve ovo će vam postati mnogo jasnije u narednim poglavljima kada budete videli stvarne primere.

Procena modela i predviđanje neviđenih instanci podataka

Nakon što smo selektovali model koji je usklađen u trening skupu podataka, možemo da upotrebimo skup podataka za testiranje da bismo procenili koliko se model dobro izvršava u ovim neviđenim podacima da bismo procenili takozvanu generalizacionu grešku. Ako smo zadovoljni performansom, možemo da upotrebimo ovaj model za predviđanje novih, budućih podataka. Važno je da naglasimo da se parametri za prethodno pomenute procedure, kao što su skaliranje atributa i redukcija dimenzionalnosti, dobijaju samo iz trening skupa podataka, a isti parametri su kasnije ponovo primenjeni za transformisanje skupa podataka za testiranje, kao i bilo koje instance novih podataka - performansa merena u skupu podataka za testiranje može, u suprotnom, biti preterano optimistična.

UPOTREBA PYTHONA ZA MAŠINSKO UČENJE

Python je jedan od najpopularnijih programskih jezika za istraživanje podataka. Zahvaljujući veoma aktivnim programerima i zajednici otvorenog koda, razvijen je veliki broj korisnih biblioteka za naučna izračunavanja i mašinsko učenje.

Iako su performanse interpretiranih jezika, kao što je Python, za računski intenzivne zadatke inferiorne u odnosu na programske jezike nižeg nivoa, proširene biblioteke, kao što su NumPy i SciPy, razvijene su za nadgradnju Fortran i C implementacija nižeg nivoa za brze vektorizovane operacije u višedimenzionalnim nizovima.

Za zadatke programiranja mašinskog učenja uglavnom ćemo koristiti scikit-learn, jednu od najpopularnijih biblioteka mašinskog učenja otvorenog koda. U narednim poglavljima, kada se budemo fokusirali na duboko učenje koje se zove podoblast mašinskog učenja, upotrebićemo najnoviju verziju TensorFlow biblioteke, koja je specijalizovana za veoma efikasno obučavanje takozvanih modela duboke neuronske mreže upotrebom grafičkih kartica.

Instaliranje Pythona i paketa iz Python Package Indexa

Python je dostupan za sva tri glavna operativna sistema - Microsoft Windows, macOS i Linux, a instalacioni fajl i dokumentacija mogu da se preuzmu sa zvaničnog Python veb sajta <https://www.python.org>.

Ova knjiga je napisana za Python verziju 3.7 ili noviju; preporučujemo da upotrebite najnoviju verziju Pythona 3, koja je trenutno dostupna. Neki kod može biti kompatibilan sa verzijom Python 2.7, ali pošto je zvanična podrška za Python 2.7 okončana 2019. godine, a većina biblioteka otvorenog koda je već prestala da podržava Python 2.7 (<https://python3statement.org>), preporučujemo da upotrebite Python 3.7 ili noviju verziju.

Dodatni paketi koje ćemo koristiti u ovoj knjizi mogu da budu instalirani pomoću pip instalacionog programa, koji je deo Python Standard Libraryja od verzije Python 3.3. Više informacija o pip programu možete pronaći na adresi <https://docs.python.org/3/installing/index.html>.

Nakon što uspešno instalirate Python, možete da izvršite pip iz terminala da biste instalirali dodatne Python pakete:

```
pip install SomePackage
```

Već instalirani paketi mogu da budu ažurirani pomoću oznake `--upgrade` flag:

```
pip install SomePackage --upgrade
```

Upotreba Anaconda Python distribucije i upravljača paketima

Preporučena alternativna Python distribucija za naučna izračunavanja je Anaconda, koju je razvila kompanija „Continuum Analytics“. Anaconda je besplatna (uključujući i komercijalnu upotrebu) poslovna Python distribucija koja obuhvata sve važne Python pakete za istraživanje podataka, matematiku i inženjerstvo u jednu jednostavnu međuplatformsku distribuciju. Anaconda instalacioni program možete da preuzmete sa adrese <https://docs.anaconda.com/anaconda/install/>, a vodič za upotrebu Anaconda distribucije dostupan je na adresi <https://docs.anaconda.com/anaconda/user-guide/getting-started/>.

Nakon što uspešno instalirate Anacondu, možete da instalirate nove Python pakete upotrebom sledeće komande:

```
conda install SomePackage
```

Postojeći paketi mogu da se ažuriraju upotrebom sledeće komande:

```
conda update SomePackage
```

Paketi za naučna izračunavanja, istraživanje podataka i mašinsko učenje

U ovoj knjizi mi ćemo uglavnom koristiti višedimenzionalne nizove biblioteke NumPy za skladištenje podataka i manipulisanje njima. Povremeno ćemo upotrebiti pandas biblioteku, koja je izgrađena na osnovi NumPy biblioteke, a obezbeđuje dodatne alatke za manipulaciju podacima višeg nivoa i olakšava upotrebu tabelarnih podataka. Da biste poboljšali iskustvo u učenju i vizuelizovali kvantitativne podatke, što je često veoma korisno da biste ih bolje razumeli, upotrebite veoma prilagodljivu biblioteku Matplotlib.

Brojevi verzija glavnih Python paketa koji su upotrebljeni za pisanje ove knjige prikazani su u sledećoj listi. Uverite se da su brojevi verzija vaših instaliranih paketa isti ili veći od ovih da biste bili sigurni da se primeri koda pravilno pokreću:

- NumPy 1.17.4
- SciPy 1.3.1
- scikit-learn 0.22.0
- Matplotlib 3.1.0
- .pandas 0.25.3

REZIME

U ovom poglavlju smo istražili mašinsko učenje na veoma visokom nivou i upoznali ste „veću sliku“ i glavne koncepte koje ćemo istražiti detaljnije u sledećim poglavljima. Naučili ste da je nadgledano učenje sastavljeno od dve važne podoblasti: od klasifikacije i regresije. Dok nam modeli klasifikacije omogućavaju da kategorizujemo objekte u poznate klase, regresionu analizu možemo da upotrebimo za predviđanje kontinualnih ishoda ciljnih promenljivih. Nenadgledano učenje ne obezbeđuje samo korisne tehnike za otkrivanje struktura u neoznačenim podacima, već može biti korisno i za kompresovanje podataka u koracima pretprocesiranja atributa.

Ukratko smo pregledali tipičnu mapu za primenu mašinskog učenja za rešavanje problema, što ćemo upotrebiti kao osnovu za detaljnije razmatranje i praktične vežbe u narednim poglavljima. Na kraju smo podesili Python okruženje i instalirali i ažurirali potrebne pakete da bismo se pripremili za pregled mašinskog učenja u akciji.

Kasnije u ovoj knjizi, osim samog mašinskog učenja, predstavimo i različite tehnike za pretprocesiranje skupa podataka, što će vam pomoći da dobijete najbolju

performansu iz različitih algoritama mašinskog učenja. Detaljno ćemo opisati algoritme klasifikacije, a takođe ćemo istražiti različite tehnike za regresionu analizu i klasterizaciju.

Pred vama je veoma uzbudljiv „put“, jer ćemo opisati mnoge moćne tehnike u ogromnoj oblasti mašinskog učenja. Međutim, pristupi ćemo mašinskom učenju korak po korak, postepeno nadgrađujući znanje. U sledećem poglavlju ćemo započeti ovo „putovanje“ implementiranjem jednog od najranijih algoritama mašinskog učenja za klasifikaciju, što će nas pripremiti za Poglavlje 3, „Predstavljanje klasifikatora mašinskog učenja pomoću scikit-learn“, u kome ćemo opisati naprednije algoritme mašinskog učenja upotrebom biblioteke mašinskog učenja scikit-learn otvorenog koda.

