Vorwort

Einleitung

Maschinelles Lernen bedeutet, mathematische Modelle an Daten anzupassen, um daraus Erkenntnisse oder Vorhersagen zu gewinnen. Diese Modelle erwarten als Eingabe sogenannte Merkmale. Ein *Merkmal* ist eine numerische Darstellung eines bestimmten Aspekts von Rohdaten. In der Machine-Learning-Pipeline vermitteln Merkmale zwischen Daten und Modellen. *Merkmalskonstruktion* (engl. Feature Engineering) wird der Vorgang genannt, Merkmale aus Rohdaten zu gewinnen und in eine Form zu bringen, die sich für das Machine-Learning-Modell eignet. Sie ist ein entscheidender Schritt in der Machine-Learning-Pipeline, denn die richtigen Merkmale können den schwierigen Vorgang des Modellierens erleichtern und so eine bessere Qualität der Ergebnisse ermöglichen, die die Pipeline ausgibt. Anwender aus der Praxis wissen, dass beim Aufbau einer Machine-Learning-Pipeline die Merkmalskonstruktion und die Datenbereinigung die meiste Zeit benötigen. Trotz seiner Bedeutung wird das Thema jedoch selten eigenständig behandelt. Das mag daran liegen, dass die richtigen Merkmale nur im Kontext sowohl des Modells als auch der Daten definiert werden können; da es so unterschiedliche Daten und Modelle gibt, ist es also schwer, die Merkmalskonstruktion projektübergreifend zu verallgemeinern.

Dennoch geschieht Merkmalskonstruktion nicht einfach aus dem Stegreif. Es gibt zugrunde liegende Prinzipien, die sich am besten an Beispielen zeigen lassen. Jedes Kapitel dieses Buchs widmet sich einer Aufgabe aus der Datenanalyse: der Darstellung von Text- oder Bilddaten, der Dimensionsreduktion automatisch erzeugter Merkmale, der Anwendung von Normierung usw. Stellen Sie sich das Buch als eine Sammlung miteinander verwobener Kurzgeschichten und nicht als einen einzigen langen Roman vor. Jedes Kapitel bietet einen Einblick in die Vielzahl der bekannten Verfahren zur Merkmalskonstruktion. Zusammen veranschaulichen sie die übergreifenden Prinzipien.

Ein Fachgebiet zu beherrschen, bedeutet nicht nur, die Definitionen zu kennen und die Formeln herleiten zu können. Es genügt nicht, zu wissen, wie ein Mechanismus funktioniert und was er vermag – man muss auch verstehen, woher sein Aufbau rührt, wie er sich zu anderen Techniken verhält und welche Vor- und

Nachteile jede Herangehensweise hat. Meisterschaft bedeutet, genau zu wissen, wie etwas gemacht wird, ein Gespür für die Grundprinzipien zu haben und diese in das eigene schon bestehende Wissensgerüst einordnen zu können. Man wird kein Meister, indem man einfach ein Buch liest, obwohl ein gutes Buch neue Türen öffnen kann. Es braucht praktische Erfahrung – die Ideen müssen angewandt werden, was wiederum ein iterativer Vorgang ist. Mit jeder Iteration lernen wir die Ideen besser kennen und werden immer geschickter und kreativer bei ihrer Anwendung. Das Ziel dieses Buchs ist es, die Anwendung der darin vorgestellten Ideen zu erleichtern.

Dieses Buch versucht, immer zuerst die Grundgedanken und danach die Mathematik zu lehren. Statt zu diskutieren, *wie* etwas gemacht wird, wollen wir erklären, *warum* es gemacht wird. Unser Ziel ist es, die *Intuition* hinter den Ideen zu vermitteln, sodass Sie als Leser ein Verständnis dafür bekommen, wie und wann sie anzuwenden sind. Es gibt aber auch Unmengen von Beschreibungen und Bildern für diejenigen, die eher auf andere Weise lernen. Dazu liefern wir die mathematischen Formeln, um der Intuition Genauigkeit zu verleihen und eine Brücke zwischen diesem Buch und anderen Angeboten zu schlagen.

Das Buch setzt Kenntnisse der Grundbegriffe des maschinellen Lernens voraus, etwa was ein Modell oder ein Vektor ist, aber es enthält auch einen Auffrischungskurs, damit wir alle auf demselben Stand sind. Erfahrung mit linearer Algebra, Wahrscheinlichkeitsverteilungen und Optimierung sind hilfreich, aber nicht notwendig.

Python-Bibliotheken

Die Codebeispiele in diesem Buch sind in Python geschrieben und verwenden eine Reihe freier und quelloffener Pakete. Die Bibliothek NumPy (*http://www.numpy.org/*) implementiert numerische Vektor- und Matrixoperationen. Pandas (*http://pandas.pydata.org/*) stellt den DataFrame als Grundbaustein der Datenanalyse in Python zur Verfügung. scikit-learn (*http://scikit-Learn.org/stable/*) ist ein allgemeines Machine-Learning-Paket mit einer umfassenden Sammlung von Modellen und Merkmalstransformationen. Matplotlib (*https://matplotlib.org/*) und die Stilbibliothek Seaborn (*https://seaborn.pydata.org/*) bieten Unterstützung für Diagramme und Visualisierung. Die Beispiele sind als Jupyter-Notebooks in unserem GitHub-Repository (*https://github.com/alicezheng/feature-engineering-book*) zu finden.

Wegweiser durch dieses Buch

Die ersten Kapitel bieten einen gemächlichen Einstieg für Neulinge auf dem Gebiet der Datenanalyse und des maschinellen Lernens. Kapitel 1 stellt die Grundkonzepte in der Machine-Learning-Pipeline vor: Daten, Modelle, Merkmale usw. In Kapitel 2 schauen wir uns die Anfänge der Merkmalskonstruktion für numerische Daten an: Filtern, Klassifikation, Skalierung, logarithmische und Potenztransformationen

sowie Kreuzmerkmale. Kapitel 3 taucht ein in die Merkmalskonstruktion für natürlichen Text und untersucht Techniken wie Bag-of-Words, *n*-Gramme und Phrasenerkennung. Kapitel 4 untersucht den Algorithmus TF-IDF (Begriffshäufigkeit – inverse Dokumentenhäufigkeit, engl. *Term Frequency – Inverse Document Frequency*) als ein Beispiel der Merkmalsskalierung und diskutiert, warum er funktioniert. Das Buch nimmt bei Kapitel 5 Fahrt auf, wenn wir über effiziente Kodierungsverfahren für kategoriale Variablen sprechen, darunter Merkmals-Hashing und Klassenzählung. Spätestens wenn wir in Kapitel 6 bei der Hauptkomponentenzerlegung (PCA, engl. *Principal Component Analysis*) angelangt sind, befinden wir uns tief im Land des maschinellen Lernens. Kapitel 7 betrachtet den *k*-Means-Algorithmus als Verfahren zur Merkmalsgewinnung, wodurch das nützliche Konzept der Stapelung von Modellen aufgezeigt wird. Kapitel 8 beschäftigt sich ganz mit Bildern, die im Hinblick auf Merkmalsgewinnung eine viel größere Herausforderung darstellen als Textdaten. Wir schauen uns zwei Verfahren der Merkmalsgewinnung per Hand an, SIFT und HOG, bevor wir anschließend das Deep Learning als neueste Technik zur Merkmalsgewinnung für Bilder erläutern. Zum Abschluss zeigen wir in Kapitel 9 anhand ausführlicher Beispiele ein paar unterschiedliche Verfahren, indem wir einen Empfehlungsalgorithmus für einen Datensatz von akademischen Aufsätzen erstellen.

Merkmalskonstruktion ist ein weites Feld, und jeden Tag werden neue Verfahren entwickelt, insbesondere auf dem Gebiet des automatisierten Erlernens von Merkmalen. Um das Buch auf einen handlichen Umfang zu beschränken, mussten wir einiges auslassen. So behandeln wir nicht die Fourier-Analyse für Audiodaten, obwohl das ein wunderschönes Thema und nah verwandt mit der Eigenfunktionszerlegung in der linearen Algebra ist, die wir in den Kapiteln 4 und 6 streifen. Auch überspringen wir die Diskussion zufälliger Merkmale, die ebenso eng mit der Fourier-Analyse verknüpft sind. Wir bieten zwar eine Einführung ins Erlernen von Merkmalen für Bilddaten durch Deep Learning, gehen aber nicht näher auf die zahllosen in der Weiterentwicklung befindlichen Deep-Learning-Modelle ein. Weiterführende Forschungsfelder, etwa Zufallsprojektionen, komplexe Modelle zur Merkmalsgewinnung aus Text wie word2vec und Brown-Clustering sowie Latent-Raum-Modelle wie Latente Dirichlet-Allokation und Matrixfaktorisierung, lassen wir ebenfalls aus. Wenn Ihnen diese Begriffe nichts sagen, haben Sie Glück. Sollten Sie sich jedoch für die allerneueste Forschung bei der Merkmalskonstruktion interessieren, dann ist dies vermutlich nicht das richtige Buch für Sie.

In diesem Buch verwendete Konventionen

Die folgenden typografischen Konventionen werden in diesem Buch verwendet:

Kursiv
: Kennzeichnet neue Begriffe, URLs, E-Mail-Adressen, Dateinamen und Dateiendungen.

`Nichtproportionalschrift`
: Wird für Programmlistings sowie für Programmelemente in Textabschnitten wie Namen von Variablen und Funktionen, Datenbanken, Datentypen, Umgebungsvariablen, Anweisungen und Schlüsselwörtern verwendet.

`Nichtproportionalschrift fett`
: Kennzeichnet Befehle oder anderen Text, den der Nutzer wörtlich eingeben soll.

`Nichtproportionalschrift kursiv`
: Kennzeichnet Text, den der Nutzer durch eigene oder zum Kontext passende Werte ersetzen soll.

Das Buch enthält außerdem zahlreiche Gleichungen der linearen Algebra. Wir folgen bezüglich der Notation diesen Konventionen: Skalare werden mit kursiven Kleinbuchstaben geschrieben (z. B. a), Vektoren mit fetten Kleinbuchstaben (z. B. $\mathbf{v}$) und Matrizen mit fetten kursiven Großbuchstaben (z. B. $\boldsymbol{U}$).

Dieses Symbol kennzeichnet einen Tipp oder Vorschlag.

Dieses Symbol kennzeichnet eine allgemeine Bemerkung.

Dieses Symbol kennzeichnet eine Warnung oder einen Rat zur Vorsicht.

Verwendung von Codebeispielen

Zusatzmaterial wie Codebeispiele oder Übungen können Sie von *https://github.com/alicezheng/feature-engineering-book* herunterladen.

Dieses Buch soll Ihnen helfen, Ihre Arbeit zu erledigen. Im Allgemeinen dürfen Sie die Codebeispiele aus diesem Buch in Ihren eigenen Programmen und der dazugehörigen Dokumentation verwenden. Sie müssen uns dazu nicht um Erlaubnis fragen, solange Sie nicht einen wirklich signifikanten Teil des Codes reproduzieren. Beispielsweise benötigen Sie keine Erlaubnis, um ein Programm zu schreiben, in dem mehrere Codefragmente aus diesem Buch vorkommen. Wollen Sie dagegen eine DVD mit Beispielen aus Büchern von O'Reilly verkaufen oder verteilen, benötigen Sie eine Erlaubnis. Eine Frage zu beantworten, indem Sie aus diesem Buch zitieren und ein Codebeispiel wiedergeben, benötigt keine Erlaubnis. Eine beträcht-

liche Menge Beispielcode aus diesem Buch in die Dokumentation Ihres Produkts aufzunehmen, bedarf hingegen einer Erlaubnis.

Wir freuen uns darüber, zitiert zu werden, verlangen es aber nicht. Ein Zitat enthält Titel, Autor, Verlag und ISBN. Ein Beispiel: »*Merkmalskonstruktion für Machine Learning* von Alice Zheng und Amanda Casari (O'Reilly). Copyright 2018 Alice Zheng und Amanda Casari, 978-3-96009-093-9.«

Wenn Sie glauben, dass Sie die Codebeispiele über eine angemessene Nutzung oder die oben gewährte Nutzungserlaubnis hinaus verwenden, dann kontaktieren Sie uns bitte unter *kommentar@oreilly.de*.

Danksagung

Zuallererst möchten wir unseren Lektoren Shannon Cutt und Jeff Bleiel dafür danken, dass sie uns bei unserem ersten Buch durch den uns beiden bis dato unbekannten Marathon einer Buchveröffentlichung geleitet haben. Ohne die enge Zusammenarbeit mit euch hätte dieses Buch nie das Licht der Welt erblickt. Ebenso vielen Dank an Ben Lorica, O'Reilly-Mastermind, dessen Ermutigungen und Bestätigungen das Buch von einer verrückten Idee zu einem tatsächlichen Produkt brachten. Danke an Kristen Brown und das Produktionsteam von O'Reilly für ihre überragende Sorgfalt bei Details und ihre extreme Geduld bei unseren Rückmeldungen.

Wenn es stimmt, dass es ein ganzes Dorf braucht, um ein Kind aufzuziehen, dann braucht es ein ganzes Parlament von Datenanalytikern, um ein Buch zu veröffentlichen. Wir wissen jeden Vorschlag und jeden Hinweis auf mögliche Verbesserungen und alle Nachfragen zu Unklarheiten zu schätzen. Andreas Müller, Sethu Raman und Antoine Atallah nahmen sich kostbare Zeit für technische Überprüfungen. Antoine tat das nicht nur blitzschnell, sondern stellte dabei auch seine dicken Maschinen für Experimente zur Verfügung. Ted Dunnings gewandte Beherrschung von Statistik und seine Meisterschaft im maschinellen Lernen sind legendär. Er ist zudem unglaublich großzügig mit seiner Zeit und seinen Ideen, und von ihm stammen buchstäblich die Methode und das Beispiel, die im Kapitel über den *k*-Means-Algorithmus beschrieben werden. Owen Zhang gewährte einen Blick in seine Kaggle-Schatzkiste zur Verwendung von Antwortraten-Merkmalen, die wir der von Misha Bilenko gesammelten Machine-Learning-Folklore über die Klassenzählung zugesellten. Ein weiteres Dankeschön geht an Alex Ott, Francisco Martin und David Garrison für zusätzliches Feedback.

Besonderer Dank von Alice

Ich möchte der Familie GraphLab/Dato/Turi für ihre großzügige Unterstützung in der ersten Phase dieses Projekts danken. Die Idee erwuchs aus dem Umgang mit unseren Anwendern. Beim Bau einer ganz neuen Machine-Learning-Plattform für

Datenanalytiker war uns aufgefallen, dass die Welt ein systematischeres Verständnis von Merkmalskonstruktion braucht. Danke an Carlos Guestrin für die Freistellung vom geschäftigen Start-up-Leben, damit ich mich aufs Schreiben konzentrieren konnte.

Danke an Amanda, die als technische Gutachterin begann und später einsprang, um diesem Buch zum Leben zu verhelfen. Du bringst Dinge über die Ziellinie! Nachdem dieses Buch nun fertig ist, müssen wir ein neues Projekt finden, und sei es nur, um unsere Arbeitssitzungen bei Tee und Kaffee und Sandwiches und leckerem Essen fortzusetzen.

Ein besonderes Dankeschön an meine Freundin und Heilerin Daisy Thompson für ihre beständige Unterstützung während aller Phasen dieses Projekts. Ohne deine Hilfe hätte ich viel länger gebraucht, um mich darauf einzulassen, und mich über den Marathon geärgert. Du hast, wie immer mit deiner Arbeit, Licht und Leichtigkeit in dieses Projekt gebracht.

Besonderer Dank von Amanda

Da dies ein Buch und keine Auszeichnung fürs Lebenswerk ist, will ich versuchen, meinen Dank auf das vorliegende Projekt zu beschränken.

Vielen, vielen Dank an Alice dafür, dass sie mich als technische Lektorin und dann Mitautorin eingebracht hat. Ich lerne ständig so viel von dir, unter anderem wie man bessere mathematische Scherze macht und komplexe Konzepte verständlich erklärt.

Nur der Reihenfolge nach zuletzt geht ein ganz besonderer Dank an meinen Mann Matthew für das nahezu unmögliche Kunststück, mir Halt zu geben, mich auf dem Weg zu meinem nächsten Ziel zu bestärken und nie zuzulassen, dass ein Konzept vage abgetan wird. Du bist der beste Partner und mein Lieblingskomplize. In den größten wie den kleinsten sonnigen Momenten spornst du mich an, dich stolz zu machen.