Maschinelles Lernen

Hans-Friedrich Witschel

Dieses Kapitel gibt einen grundlegenden Überblick über die Einsatzbereiche und die Funktionsweise von maschinellem Lernen. Das Hauptziel dabei ist, die Potentiale dieser Technologie für den Einsatz in verschiedenen Geschäftsbereichen beurteilen zu können sowie ihre Grenzen zu kennen.

Menschliches Lernen

Wir beginnen damit, uns die Prinzipien des Lernens im Allgemeinen zu vergegenwärtigen. Wie lernen Menschen? Betrachten wir dies am Beispiel der Unterscheidung von Gegenständen, zum Beispiel von Tassen und Gläsern. Wie ist es zum Beispiel möglich, dass ein Mensch eine Tasse als solche erkennt, auch wenn er oder sie diese spezielle Tasse noch nie im Leben gesehen hat? Es lässt sich dadurch erklären, dass der Mensch eine Art mentales **Modell** einer Tasse abgespeichert hat. Dieses mentale Modell besteht aus gewissen **Merkmalen** sowie den für eine Tasse typischen Ausprägungen dieser Merkmale. Und woher haben wir Menschen wiederum dieses Modell? Es entsteht durch Lernen. Das Lernen wiederum vollzieht sich anhand einer Vielzahl von Beispielen. Nicht nur sehen wir im Kindesalter sehr viele Gegenstände, zum Beispiel Tassen oder Gläser, wir erhalten dazu auch oft explizit oder implizit die Information, um welchen Gegenstand es sich handelt, beispielsweise, indem unsere Eltern zu uns sagen: «Schau mal, dort ist deine Tasse! » Bei solch einer Kombination von Gegenstand und vorgegebener Klassifizierung desselben sprechen wir von einem **Trainingsbeispiel**.

Die Vielzahl der Trainingsbeispiele, die uns in unserem Leben begegnen, führen dazu, dass wir z. B. wissen, welche Merkmale typischerweise eine Tasse auszeichnen bzw. welche Merkmale besonders gut geeignet sind, eine Tasse von anderen Gegenständen zu unterscheiden. Wir lernen auch, welche Abweichungen von der «Normtasse» möglich sind, so dass man immer noch von einer Tasse spricht. Man spricht dabei auch von **Mustern** – die Kombination verschiedener Muster, d. h. zum Beispiel verschiedener möglicher Spielarten von Tassen, ergibt ein Modell.

Abbildung 1 veranschaulicht einige solche Merkmale: Zum Beispiel beobachten wir anhand vieler Trainingsbeispiele, dass die meisten Tassen einen Henkel haben, während dies bei Gläsern nicht der Fall ist. Wenn es wie im Fall der Eltern eine Instanz gibt, welche die zu einem Gegenstand gehörige Klasse vorgibt, so sprechen wir von **Klassifikation**, einer Form des sogenannten **überwachten Lernens**. Dies soll im Folgenden unser Fokus sein.

Maschinelles Lernen

Genau wie Menschen können auch Maschinen aus Trainingsbeispielen lernen. Das Prinzip ist dabei genau gleich: Zu unterscheidende Objekte werden durch Merkmale beschrieben und die Maschine erhält eine Vielzahl von Trainingsbeispielen, welche jeweils aus dieser Beschreibung sowie einer vorgegebenen Klassifikation bestehen.

Abbildung 1: Typische Ausprägungen von Tasse und Glas sowie einige zur Beschreibung geeignete Merkmale (eigene Illustration)

Die Maschine berechnet daraus ein Modell, welches genau wie beim Menschen eine Abstraktion darstellt, d. h. die für eine Objektklasse typischen Kombinationen von Merkmalsausprägungen (Muster). Der einzige Unterschied zwischen Mensch und Maschine ist, dass Menschen die Identifikation der zur Unterscheidung von Objekten potenziell geeigneten Merkmale selbständig vornehmen, während dies den Maschinen (meist) vorgegeben werden muss.

Die Unterscheidung verschiedener Arten von Objekten anhand charakteristischer Merkmale ist eine Fähigkeit, die in vielen Bereichen des Geschäftslebens nützlich ist. Ein typisches Beispiel ist das zielgerichtete Marketing: Hierbei geht es um die Unterteilung von Kund:innen in solche, die an einem Angebot Interesse haben könnten, und jenen, bei denen dies nicht der Fall ist. Die Interessierten entsprechen dabei sozusagen den Tassen und die Uninteressierten den Gläsern. Auch bei dieser Unterscheidung kann man aus vielen Beispielen lernen, was interessierte Kundschaft auszeichnet – dabei können **demografische Merkmale** wie Alter oder Geschlecht eine Rolle spielen, oder aber **Verhaltensweisen** wie z. B. Reaktionen auf vorherige Angebote. Die nachfolgende Tabelle zeigt einige weitere Beispiele für die Anwendung von Klassifikation im Geschäftsleben, wobei jeweils erläutert wird, was den Tassen und Gläsern aus unserem einführenden Beispiel entspricht.

Eine Gemeinsamkeit all dieser Anwendungsfälle besteht darin, dass es dabei um operative Entscheidungen geht, die in grosser Zahl getroffen werden müssen, um Risiken zu vermeiden (Betrug, Kreditvergabe, Churn) oder Chancen zu nutzen (Marketing). Ziel ist dabei meist eine Automatisierung zwecks Steigerung der Effizienz. Es geht aber oft auch um eine erhöhte Qualität der **Entscheidungen**: Einerseits werden Entscheidungen konsistenter, wenn sie von einem von Maschinen erlernten Modell getroffen werden, andererseits werden Maschinen typischerweise mit mehr Trainingsbeispielen gefüttert als jeder einzelne Mensch – z. B. mit allen Schadensmeldungen einer Versicherung, nicht nur mit denen, die einzelnen Versicherungsangestellten im Lauf ihrer Tätigkeit begegnen. Dadurch sind die Entscheidungen der Maschine in gewisser Weise breiter abgestützt als die eines Menschen. In Fällen, bei denen sich die zu treffenden Unterscheidungen aus Ereignissen ableiten (z. B. Kreditausfall) und nicht aus den Entscheidungen von Menschen, kann man auch davon sprechen, dass maschinell erlernte Modelle weniger der Gefahr von Voreingenommenheit oder anderen **Verzerrungen** («Biases») ausgesetzt sind, die in der menschlichen Natur liegen.

Aufgabe	**Objekte**	**«Tassen»**	**«Gläser»**
Betrugserkennung (z. B. in einer Versicherung)	Schadensfälle	Berechtigte Schadensmeldungen	Betrügerische Schadensmeldungen
Spamfilter	Emails	Spam bzw. irrelevante Emails	Relevante Emails
Kreditvergabe	Kreditanträge	Anträge, bei denen die Rückzahlung der verliehenen Summe erfolgt	Anträge, bei denen es zu einem Kreditausfall kommt
Kundenabwanderung (Churn)	Kund:innen	Unzufriedene / abwanderungsbereite Kund:innen	Loyale Kund:innen

Tabelle 1: Beispielanwendungen von maschinellem Lernen (Klassifikation)

Formen des maschinellen Lernens

Neben der Klassifikation gibt es eine weitere wichtige Form des überwachten maschinellen Lernens, die **Regression**. Hierbei soll anstatt einer Klasse ein numerischer Wert vorhergesagt werden. Zu den wichtigsten Beispielen für Regressionsaufgaben zählen die Vorhersage von Nachfrage (z. B. voraussichtliche Anzahl verkaufter Produkte, voraussichtliche Anzahl Besucher:innen etc.), die Vorhersage von zu erzielenden Preisen (z. B. beim Verkauf einer Immobilie) oder die Vorhersage von anfallenden Kosten. Die folgende Tabelle fasst dies zusammen.

Auch aus diesen Vorhersagen lassen sich konkrete operative Entscheidungen ableiten, beispielsweise die Einkaufsmenge (Nachfragevorhersage) oder das Festlegen eines monatlichen Beitrags für Antragstellende (Krankenversicherung).

Neben dem überwachten Lernen gibt es auch das sogenannte **unüberwachte Lernen**, bei dem Maschinen mit Beispielen trainiert werden, ohne dass dazu eine vorgegebene Klassifikation oder ein vorgegebener numerischer Wert existiert. Die Aufgabe besteht dann beispielsweise darin, Muster in den Beispielen zu entdecken, anhand derer sich Objekte gruppieren lassen (**Clustering**). Auch Menschen können dies intuitiv: Wenn man Kindern Bilder von Tassen und Gläsern vorlegt, ohne dazu zu sagen, wie diese Objekte benannt werden, werden die meisten Kinder diese nach Objektklasse sortieren, d. h. sie werden die zugrundeliegenden Muster erkennen und die Objekte clustern. Eine der wichtigsten Anwendungen des Clusterings im Geschäftsleben besteht in der Segmentierung von Kund:innen für Marketingzwecke.

Manche Aufgaben lassen sich auch mit Kombinationen von überwachtem und unüberwachtem Lernen lösen bzw. können mit beiden Herangehensweisen angegangen werden, z. B. die Erkennung von **Anomalien** (welche beispielsweise auch zur Betrugsaufdeckung nützlich sein kann) oder das Ableiten von **Assoziationsregeln**, welche in Empfehlungssystemen («Kunden, die X kauften, interessierten sich auch für Y») eingesetzt werden. Insgesamt sind all diese Formen des maschinellen Lernens Teil des grossen Gebiets des Data Mining (Tan et al., 2005) und tragen somit dazu bei, relevante Muster in Daten zu finden.

Schliesslich soll noch das sogenannte verstärkende Lernen (**Reinforcement Learning**) erwähnt werden, bei dem eine Maschine lernt, Aktionen auszuführen, welche zu einer mehr oder weniger grossen Belohnung (je nach Konsequenz der Aktion und deren Erwünschtheit) führen.

Aufgabe	Objekte (Beispiel)	Vorherzusagender Wert
Vorhersage der Nachfrage	Zeitraum, z. B. Woche	Anzahl verkaufte Produkte
Vorhersage erzielter Preis	Immobilien	Verkaufspreis
Vorhersage Kosten	Kostenstelle + Zeitraum Antragstellende bei einer Krankenversicherung	Auf Kostenstelle gebuchte Ausgaben Durch die Person verursachte Kosten

Tabelle 2: Beispielanwendungen von maschinellem Lernen (Regression)

Eine Standard-Vorgehensweise
Die Entwicklung von Modellen des maschinellen Lernens sowie des sogenannten Data Mining folgt immer in etwa den gleichen Schritten. Abbildung 2 zeigt diese Schritte, welche in der sogenannten «Cross-Industry Standard Procedure for Data Mining» zusammengefasst werden (Wirth & Hipp, 2000). Dieses Kapitel wird im Folgenden diese Schritte nacheinander näher beleuchten und sich dabei jeweils auf überwachtes Lernen (Klassifikation und Regression) konzentrieren.

Business & Data Understanding: Formalisierung
Das «Business Understanding» besteht in einem gründlichen Verständnis der Geschäftsziele des Data-Mining-Vorhabens, während das «Data Understanding» darauf abzielt, ein Verständnis von z. B. verfügbaren Merkmalen (Attributen) zu erlangen sowie davon, aus welchen Informationssystemen sie zusammengestellt werden können.

Im Endergebnis geht es dabei vor allem darum, ein beliebiges Vorhersageproblem zu formalisieren, d. h. in eine immer gleiche Form zu bringen, welche von den etablierten Algorithmen als Eingabe «verstanden» bzw. erwartet wird. Wir diskutieren hier zunächst die Formalisierung von Klassifikationsaufgaben. Eine Klassifizierungsaufgabe zu **formalisieren** bedeutet, dass man von einer informellen Problembeschreibung (wie sie vom Unternehmen bereitgestellt wird) zu einer Beschreibung kommt, welche die (Daten-)Objekte, ihre Merkmale (Attribute) und das Klassenattribut definiert. Das Klassenattribut beschreibt den Wert, der für jedes Objekt vorhergesagt werden soll, z. B. ob es sich um eine Tasse oder ein Glas handelt.

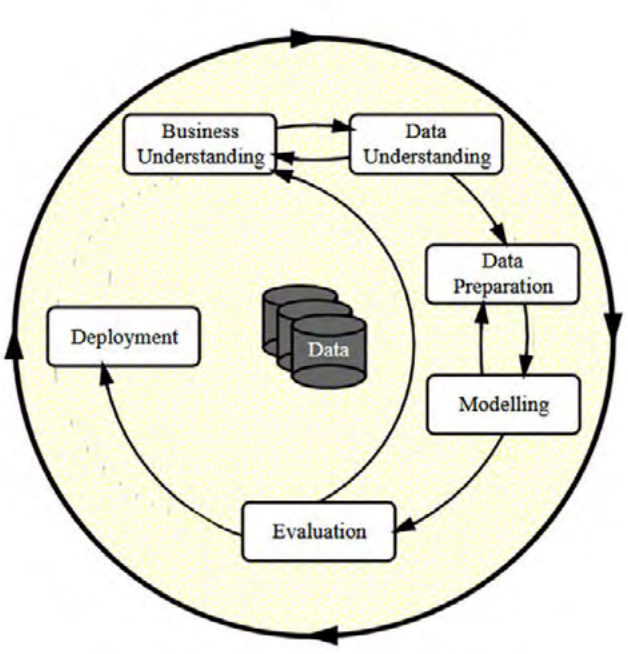

Abbildung 2: Der CRISP-DM-Zyklus ("Cross-Industry Standard Procedure for Data Mining"), aus (Wirth & Hipp, 2000)

Als immer wiederkehrendes Beispiel betrachten wir die «Wintercheck»-Aktion der Firma Swiss Bikes: Swiss Bikes betreibt eine Kette von Fahrradgeschäften mit eigener Fahrradmarke im oberen Preissegment. Die Firma bietet ihren bestehenden Kund:innen jedes Jahr im Herbst einen speziellen Service an: «Machen Sie Ihr Fahrrad fit für den Winter!» Dieser Service beinhaltet die Überprüfung von Bremsen, Reifen etc. In den vergangenen Jahren, immer im November, hat Swiss Bikes ein Mailing zu diesem Angebot an alle bestehenden Kund:innen verschickt, die ihr Fahrrad mindestens einmal zur Reparatur gebracht hatten. Das Mailing enthält einen Gutschein, mit dem man 10 % Rabatt auf den auf den Wintercheck erhält. Aus den Vorjahren weiss Swiss Bikes, welche Kunden ihre Gutscheine eingelöst haben («Trainingsbeispiele»). Dieses Jahr möchten sie das Interesse der Neukunden an diesem Angebot vorhersagen können, um gezielter entscheiden zu können, ob es sich lohnt, diesen einen Brief zu schicken.

Bei Swiss Bikes könnte die Marketingabteilung die Aufgabe so formuliert haben: «Können wir unsere Wintercheck-Mailings zielgerichteter gestalten und die Briefe nur an die wirklich interessierten Kund:innen versenden?» Aus dieser informellen Beschreibung muss man nun ableiten, dass a) die (Daten-)Objekte Kund:innen sind und dass b) das Klassenattribut die Reaktion auf das Angebot zum Wintercheck ist (mit Werten «ja» oder «nein»). Dann müssen Attribute identifiziert werden, von denen angenommen werden kann, dass sie helfen, das Kundeninteresse herauszufinden, und die in den Informationssystemen von Swiss Bikes verfügbar sind. Das Ergebnis der Formalisierung ist eine erste klassifizierte Stichprobe von Daten, wie Abbildung 3 dargestellt.

Kunden-ID	Datum letzte Reparatur	Anzahl Reparaturen (letzte 3 Jahre)	Region	Letztgekauftes Fahrrad (Kategorie)	Response
213232	May 23	1	city	Trekking	No
123244	Feb 24	5	city	Racing	Yes
546657	Nov 22	0	city	Single Speed	No
764566	Oct 20	0	rural	Single Speed	No
453232	Dec 21	0	rural	Mountain	No
785423	Apr 23	3	rural	Trekking	Yes
132567	Apr 23	2	city	Mountain	Yes
456467	Apr 22	1	rural	Mountain	No
342256	Nov 20	2	city	Single Speed	No
798888	Nov 23	4	rural	Trekking	Yes

Abbildung 3: Eine klassifizierte Stichprobe von Daten als Ergebnis einer Formalisierung Merkmale (eigene Illustration)

In einer solchen Stichprobe gehen Data-Mining-Algorithmen stets davon aus, dass jede Zeile ein (Daten-)Objekt repräsentiert (hier: eine Kundin oder einen Kunden) und dass eines der Attribute das Klassenattribut ist. Die anderen Attribute können zur Vorhersage, d. h. zur Ableitung von Mustern, genutzt werden.

Diese Stichprobe ist ein Beispiel von **strukturierten** oder auch tabellenförmigen Daten, welche in Geschäftsanwendungen sehr typisch sind. Es gibt allerdings auch Fälle, in denen die zu klassifizierenden Objekte sich nicht in Tabellenform erfassen lassen, beispielsweise, wenn es sich um Bilder oder Texte handelt (man spricht dann von **unstrukturierten** Daten). Diese Anwendungsfälle werden in den nachfolgenden Kapiteln näher behandelt.

Data Preparation

Sehr oft müssen Daten bereinigt und vorverarbeitet werden, bevor man sie den Algorithmen des maschinellen Lernens übergeben kann. Neben der primären Aufgabe, die Ausgangsdaten in eine Form wie in Abbildung 3 zu bringen, sind oft weitere Anpassungen nötig, die im Folgenden kurz angesprochen werden. Auch wenn dieser Prozess oft 80 % oder mehr eines entsprechenden Projekts einnimmt, soll hier nur kurz darauf eingegangen werden, da es um das Grundverständnis geht, um Potenziale der Technologien einschätzen zu können, nicht um die «schmutzigen Details».

Einige wichtige Schritte bei der Vorverarbeitung sind:

- Umgang mit **fehlenden Werten**: Fehlende Werte sind ein häufiges Problem in realen Daten, die teils durch mangelhafte Datenerfassung, nachträgliche Änderungen etc. verursacht werden. Viele Algorithmen können mit ihnen nicht umgehen, sodass man geeignete Strategien finden muss, um ggf. Werte zu ergänzen oder zu vermeiden. Dies reicht von einfachen Strategien wie dem Löschen von Objekten, die fehlende Werte aufweisen, bis zu statistisch komplexeren Vorgehensweisen zum Abschätzen («Imputation») von Werten
- **Feature Selection**: Oft kann es sich lohnen, Merkmale auszusortieren, die keinen nennenswerten Beitrag zur Vorhersage zu leisten versprechen. Offensichtliche Kandidaten für die Aussortierung sind dabei Attribute, welche für alle Objekte den gleichen Wert haben, oder solche, bei denen es gar keine Überschneidung der Werte gibt (z. B. IDs) und aus denen man daher keine Muster ableiten kann.
- **Feature Construction**: Oft ist es nötig, aus vorhandenen Informationen neue Merkmale abzuleiten bzw. zu konstruieren. Ein typisches Beispiel sind Datumsangaben: Da ein Datum nie wiederkehrt, taugt es nicht als Basis für Vorhersagen bzw. zur Ableitung von Mustern. Andererseits lassen sich aus Datumsangaben vorhersagekräftige Attribute ableiten wie z. B. Wochentag, Jahreszeit etc. Dazu kann auch zählen, konkrete Informationen durch abstraktere zu ersetzen, z. B. einen Produktnamen durch den Namen der Produktkategorie. Dies kann hilfreich sein, wenn sich bestimmte Muster eher auf der Ebene der Produktkategorien als auf der Ebene einzelner Produkte zeigen.
- **Typkonversionen**: In manchen Fällen ist es nötig, dass beispielsweise alle Attribute numerisch sind. In solchen Fällen kann man kategoriale Attribute (also solche mit nichtnumerischen diskreten Werten) mit gewissen Tricks

transformieren, so dass rein numerische Attribute entstehen. Manchmal ist es auch notwendig, Attribute mit scheinbar numerischen Werten (z. B. Monate, die mit 1 bis 12 nummeriert sind) nicht als numerisch zu behandeln.

Modellbildung

Um nützliche Klassifizierungsmodelle für Vorhersagen in einem Geschäftsszenario zu konstruieren, ist ein tiefes Verständnis der zahlreichen **Algorithmen**, die von Data-Mining-Tools angeboten werden, nicht zwingend notwendig.

Es gibt jedoch zwei Aufgaben, die ein gewisses Verständnis erfordern: Erstens muss man den richtigen Klassifikator für eine bestimmte Aufgabe auswählen, und zweitens muss man in der Lage sein, die **Parameter** eines Algorithmus auf sinnvolle Weise einzustellen. Bei der zweiten Aufgabe ist es aufgrund der Vielzahl an Algorithmen und ihrer jeweils individuellen Arten von Parametern schwierig, allgemeine Handlungsempfehlungen zu geben. Allerdings gibt es dabei ein wichtiges Prinzip: Die **Komplexität** eines erlernten Modells muss zur Komplexität der in den Daten enthaltenen Muster passen – es kann passieren, dass ein Modell zu einfach ist, um die Komplexität abzubilden, es kann aber auch zu komplex sein und Muster enthalten, die in den Trainingsdaten an manchen Stellen beobachtbar waren, die sich aber nicht verallgemeinern lassen. Die Komplexität eines Algorithmus lässt sich im Allgemeinen über bestimmte Parameter regulieren.

Beginnen wir jedoch mit der Auswahl eines passenden Algorithmus: Die folgende Tabelle gibt einen Überblick über Kriterien, die für die Auswahl eines Algorithmus relevant sein können (hier mit Fokus auf Klassifikationsalgorithmen, basierend auf [Kotsiantis et al., 2007]):

Kriterium	Erklärung	Passende Algorithmen
Genauigkeit der Vorhersagen (im Allgemeinen)	Wie oft stimmen die Vorhersagen der mit diesem Algorithmus erlernten Modelle? (Hier sollte man aber jeweils individuell prüfen, ob bei den konkreten Daten, mit denen man arbeitet, evtl. andere Algorithmen genauso gut oder besser abschneiden!)	Neuronale Netzwerke, Gradient Boosting, Support Vector Machines
Schnelligkeit der Modellbildung	Manche Algorithmen benötigen sehr lange Trainingszeit, die mit der Datenmenge evtl. schnell ansteigt. Dies kann Entwicklungsarbeiten unnötig verzögern	Naive Bayes, Entscheidungsbäume, k-nearest Neighbour
Schnelligkeit der Vorhersagen	Manchmal müssen Vorhersagen in Echtzeit erfolgen, d. h. mit nur sehr kurzer Verzögerung. Dies ist nicht mit allen Modellen möglich	Viele, ausser beispielsweise k-nearest neighbour
Umgang mit fehlenden Werten	Manche Algorithmen können Modelle auch bei fehlenden Attributwerten erlernen, andere nicht	z. B. Naive Bayes, logistische Regression
Umgang mit redundanten oder irrelevanten Attributen	Wenn Fachleute ein Brainstorming zu potenziell relevanten Merkmalen durchführen, kann es hilfreich sein, wenn ein Algorithmus selbst in der Lage ist, aus allen identifizierten Attributen die wirklich hilfreichen herauszufiltern. Andernfalls muss man dies in einem vorgelagerten Schritt selbst tun	Insbesondere Support Vector Machines
Interpretierbarkeit	Die Interpretierbarkeit von Modellen stellt einen sehr grossen Wert dar: einerseits lassen sich gefundene Muster plausibilisieren und es lässt sich dadurch Vertrauen in das Modell gewinnen, andererseits können aus den Mustern wichtige Erkenntnisse gewonnen werden, aus denen z. B. Optimierungspotenziale abgeleitet werden können. Schliesslich können erklärbare Vorhersagen besser von Menschen weiterverwendet werden.	Entscheidungsbäume, logistische Regression, Regellerner
Abhängigkeit von Parameterwerten	Je weniger Parameter ein Algorithmus hat und je weniger sensibel er auf Änderungen der Parameterwerte reagiert, desto einfacher ist er zu optimieren	Naive Bayes, logistische Regression

Bei der näheren Betrachtung der Tabelle fällt auf, dass unterschiedliche Algorithmen unterschiedliche Stärken und Schwächen haben. Manchmal muss man daher Kompromisse eingehen. Beispielsweise sind bei einigen Anwendungsfällen die Modelle mit der höchsten Vorhersagegenauigkeit nicht interpretierbar. Da **Interpretierbarkeit** aber oft sehr nützlich ist, sollte man dies im Einzelfall genau prüfen – ggf. kann ein interpretierbares Modell mit einer leicht schlechteren Genauigkeit zu bevorzugen sein, insbesondere auch, da die Vorhersagen des Modells oft in menschliche Entscheidungsprozesse eingebunden sind und die Interpretierbarkeit die Menschen hierbei oft besser unterstützt (Rudin, 2019).

Welche Parameter sind bei der Optimierung von Modellen hilfreich? Wie oben beschrieben verfügen die meisten Algorithmen über die Möglichkeit, die Komplexität der erlernten Modelle zu beeinflussen. Wir betrachten dazu das Beispiel von Entscheidungsbäumen. Abbildung 4 zeigt zwei Versionen eines Entscheidungsbaums, welche beide aus den Daten des Swiss Bikes Wintercheck (siehe Abbildung 3) abgeleitet wurden.

Man kann hier einerseits die Interpretierbarkeit von Entscheidungsbäumen beobachten. So lässt sich beispielsweise der Baum auf der linken Seite wie folgt schreiben: Kund:innen, welche in den letzten drei Jahren keine oder nur eine Reparatur an ihrem Fahrrad haben vornehmen lassen, werden auf das Angebot des Winterchecks nicht reagieren. Kund:innen, die ihr Fahrrad für zwei oder mehr Reparaturen gebracht haben, werden hingegen reagieren, aber nur, wenn sie kein Single-Speed-Fahrrad haben.

Andererseits beobachten wir, dass die zum Trainieren des Modells verwendete Stichprobe sehr klein ist (nur zehn Kund:innen). Darüber hinaus gibt es nur vier Kund:innen, welche mehr als eine Reparatur an ihrem Fahrrad hatten. Der untere rechte Teil des Entscheidungsbaums, d. h. die Unterteilung dieser vier Kund:innen, basiert auf einer sehr geringen Evidenz, insbesondere, da es nur eine Person gibt, welche mehr als eine Reparatur und ein Single-Speed-Fahrrad hat (vorletzte Zeile in Abbildung 3) – der Entscheidungsbaum sagt nun jedoch auf Basis dieser einen Beobachtung für alle Kund:innen mit mehr als zwei Reparaturen und einem Single-Speed-Fahrrad voraus, dass kein Interesse besteht. Aufgrund dieser zu geringen Evidenz kann es sinnvoller sein, ein weniger komplexes, dafür aber robusteres Modell zu wählen wie das auf der rechten Seite von Abbildung 4. Ein solches Modell kann man erhalten, indem man die Tiefe eines Entscheidungsbaum von vornherein begrenzt bzw. im Nachhinein verringert (man spricht dann von «**Pruning**»). Die erlaubte maximale Tiefe eines Entscheidungsbaums ist somit ein möglicher Parameter, mit dem man die Komplexität des erlernten Modells steuern kann.

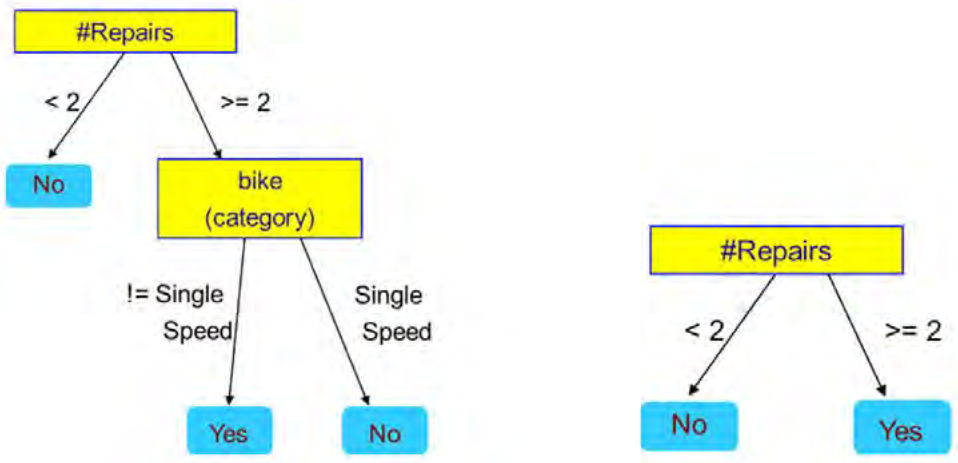

Abbildung 4: Zwei Versionen eines Entscheidungsbaums, gelernt jeweils aus den Daten von Abbildung 3 Merkmale (eigene Illustration)

Schliesslich soll noch erwähnt werden, dass die Auswahl eines Algorithmus mit möglichst hoher Vorhersagegenauigkeit und die Optimierung von dessen Parameterwerten nicht manuell erfolgen muss: Mit Hilfe von sogenannten **AutoML**-Werkzeugen lässt sich das Testen verschiedener Konfigurationen automatisieren. Die Werkzeuge testen dabei systematisch verschiedene Algorithmen mit diversen Parametereinstellungen und präsentieren schliesslich ein «Gewinnermodell», welches im Bezug auf eine zuvor gewählte Erfolgsmetrik am besten abschneidet. Relevante Erfolgsmetriken werden im folgenden Abschnitt näher behandelt.

Evaluation

Bevor man ein Modell für Vorhersagen nutzt, sollte man die erwartete Qualität dieser Vorhersagen und somit den erwarteten wirtschaftlichen Nutzen quantifizieren. Dies sollte einerseits dazu dienen, zu entscheiden, ob ein gegebenes Modell gut genug ist, um in der Praxis eingesetzt zu werden (siehe Deployment-Phase unten). Andererseits lassen sich die hier diskutierten Evaluationsverfahren und -metriken auch einsetzen, um die Modellentwicklung iterativ zu verfeinern – so, wie dies n Abbildung 2 durch den Pfeil angedeutet ist, welcher von «Evaluation» zurück zum «Business Understanding» deutet. Dabei muss man nicht zwingend wieder ganz vorne anfangen: Typischerweise spielen Evaluationsergebnisse bei der Modellbildung eine grosse Rolle, d. h. man optimiert anhand der unten eingeführten Verfahren und Metriken die Wahl des Algorithmus und der zugehörigen Parameterwerte. Es ist aber auch oft zu beobachten, dass Evaluationsergebnisse einen Einfluss auf die Phase der Datenaufbereitung («Data Preparation») haben, insbesondere, was die Wahl neuer Merkmale angeht.

Um die Güte eines Klassifizierungsmodells zu beurteilen, muss man es auf Daten anwenden – und um eine faire Beurteilung zu ermöglichen, sollte es sich dabei nicht um Daten handeln, mit denen das Modell trainiert wurde.

Zu diesem Zweck nimmt man eine klassifizierte Stichprobe (so wie z. B. die in Abbildung 3) und hält einen Teil davon für die Evaluation zurück. Das heisst, man verwendet einen Teil der Stichprobe, um ein Modell zu trainieren (**Trainingsmenge**). Auf dem anderen Teil der Daten (**Testmenge**) lässt man das neu gelernte Modell Vorhersagen machen und vergleicht sie mit der «wahren» Klasse der Testinstanzen (die bekannt ist, da wir mit einer klassifizierten Stichprobe arbeiten). Dieser Ansatz wird «**Holdout**»-Evaluierung oder Percentage Split genannt und ist im oberen Teil von Abbildung 5 dargestellt. Die Abbildung zeigt eine recht typische Aufteilung, bei der zwei Drittel der Daten für das Training verwendet werden, das übrige Drittel als Testmenge.

Im unteren Teil von Abbildung 5 ist eine alternative Vorgehensweise dargestellt, die sogenannte **Kreuzvalidierung**. Sie wird gern verwendet, wenn die verfügbare Menge an Daten eher klein ist. Man teilt dabei die gesamte klassifizierte Stichprobe in z. B. zehn Teile (bei einer zehnfachen Kreuzvalidierung) und führt dann zenhn Durchläufe durch, bei denen jeweils ein anderer der zehn Teile die Rolle der Testmenge spielt, während mit den restlichen neun Teilen ein Modell trainiert wird. Für jede Testmenge wird die Qualität der Vorhersagen gemessen und die zehn so erhaltenen Resultate werden gemittelt.

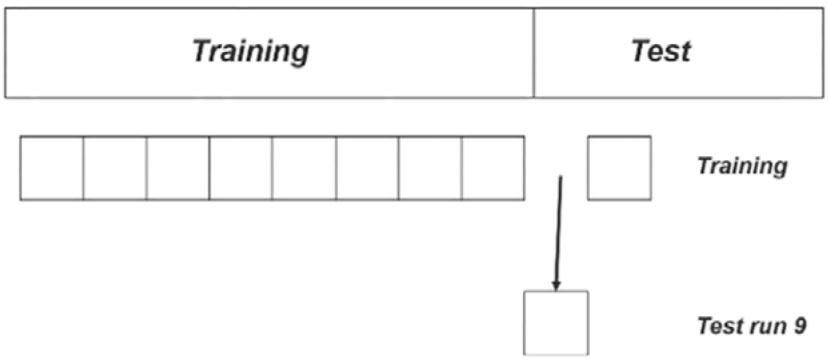

Abbildung 5: Aufteilung von Daten in Trainings- und Testmenge mittels Holdout (oben) bzw. Kreuzvalidierung (unten)
Merkmale (eigene Illustration)

Oben haben wir bereits den Begriff der **Vorhersagegenauigkeit** (engl. Accuracy) verwendet. Er bezieht sich auf den Prozentsatz der korrekten Vorhersagen in der Testmenge. Umgekehrt wird die Güte eines Modells manchmal als Fehlerrate ausgedrückt, d. h. als Prozentsatz der falschen Vorhersagen (100 %-Accuracy).

Es ist oft nützlich, nicht nur zu wissen, wie viele Fehler ein Klassifikator gemacht hat, sondern auch, um welche Art von Fehlern es sich handelt. Bei einem Klassifizierungsproblem, bei dem das Klassenattribut nur zwei Werte annehmen kann, z. B. «ja» und «nein», unterscheidet man zum Beispiel zwischen

- **Falsch-Positive**: Dies sind Instanzen der Testmenge mit einem wahren Wert von «nein», für die das Klassifizierungsmodell einen «ja»-Wert vorausgesagt hat. Im Beispiel des Swiss-Bikes-Winterchecks sind dies Personen, die nicht interessiert sind, für die der Klassifikator aber Interesse vorhersagt.
- **Falsch-Negative**: Dies sind Instanzen mit einem wahren «ja»-, aber einem vorhergesagten «nein»-Wert. Für den Wintercheck sind dies interessierte Kunden, die vom Klassifikator nicht erkannt wurden.

Wahr-positive und wahr-negative Ergebnisse werden analog dazu definiert.

Um diese verschiedenen Fehlertypen aufzuzeigen, geben Data-Mining-Tools in der Regel so genannte **Konfusionsmatrizen** aus. Eine Konfusionsmatrix zeigt die wahren/falschen positiven/negativen Ergebnisse wie in Abbildung 6.

		Vorhersage	
		Ja	Nein
Echtes Interesse	Ja	44	62
	Nein	53	841

Abbildung 6: Konfusionsmatrix für eine Testmenge von 1000 Swiss Bikes-Kunden Merkmale (eigene Illustration)

Betrachten wir nun das Beispiel des Winterchecks von Swiss Bikes und treffen wir folgende Annahmen:

- Jeder Wintercheck bringt einen Umsatz von CHF 100 und einen Gewinn von CHF 40 für Swiss Bikes
- Der Versand eines Werbebriefes kostet CHF 1
- Die typische Rücklaufquote (anders als in der Stichprobe aus der Abbildung 3) liegt bei rund 10 %.

In einer solchen Situation sollten falsch-Positive und falsch-Negative nicht gleich behandelt werden: Ein falsch-negative Vorhersage bedeutet – wenn wir davon ausgehen, dass die Marketingabteilung der Empfehlung des Modells jeweils folgt – einen entgangenen Gewinn von CHF 40, während eine falsch-positive Vorhersage zunächst nur bedeutet, dass ein Brief für CHF 1 vergeblich verschickt wurde.

Betrachten wir nun zwei Klassifikatormodelle:

- Eine «dumme» Variante, die immer «nein» vorhersagt. Aufgrund unserer Antwortrate von 10 % (siehe Annahmen oben) hat dieser Klassifikator eine Vorhersagegenauigkeit von 90 %.
- Ein Klassifikator mit einer Konfusionsmatrix wie in Abbildung 6 dargestellt. Die Genauigkeit dieses Klassifizierers beträgt nur 88,5 %.

Trotz der (geringfügig) besseren Genauigkeit des «dummen» Baseline-Klassifikators würden wir aus wirtschaftlicher Sicht den zweiten Klassifikator bei weitem vorziehen: Im Gegensatz zur Baseline, die zu einem Gewinn von CHF 0 führt (kein Kunde wird je kontaktiert), hilft er uns, 44 interessierte Kunden zu finden, d. h. einen Gewinn von 44 * 40 = CHF 1 760 zu erzielen, abzüglich CHF 44 für Briefe. Natürlich haben wir in diesem Fall auch 53 falsch-positive Briefe, was zu einer Verschwendung von CHF 53 Porto führt. Aber es bleibt immer noch ein Nettogewinn von CHF 1 663.

An diesem Beispiel kann man gut sehen, dass die Verwendung von Vorhersagegenauigkeit (Accuracy) als Evaluationsmetrik oft nicht zielführend ist. Dies ist insbesondere dann der Fall, wenn einerseits eine Klasse sehr viel seltener ist als die andere (im vorliegenden Fall das Interesse der Kund:innen) und andererseits die mit falsch-positiven oder falsch-negativen Vorhersagen verbundenen Kosten oder Gewinne stark unterschiedlich sind. Dies passiert in der Praxis sehr häufig, beispielweise bei fast allen in Tabelle 1 aufgelisteten Anwendungsfällen – Betrugsfälle bei der Versicherung, nicht zurückgezahlte Kredite oder abwanderungswillige Kund:innen sind (hoffentlich) eher die Ausnahme. In allen diesen Fällen sind aber die Kosten für falsch-negative Vorhersagen (also das Nichtaufdecken von Betrug, Kreditausfällen oder Churn) deutlich kostspieliger als ein falscher Alarm, also eine falsch-positive Vorhersage. In all diesen Fällen wollen wir eher Klassifikationsmodelle «belohnen» (also gut bewerten), welche gegenüber der kleineren Klasse eine hohe Sensitivität aufweisen, auch wenn dies meist zu einer höheren Quote von falschen Alarmen führt.

Eine Evaluationsmetrik, welche robust gegenüber stark unterschiedlich grossen Klassen ist, ist die sogenannte «Area under the (ROC) curve» (**AUC**). Die Definition von AUC ist relativ komplex und soll hier nicht im Detail diskutiert werden.

Ein weiteres oft geeignetes Verfahren ist eine sogenannte kostensensitive Evaluation. Dabei verwendet man eine Kostenmatrix, um die unterschiedlichen Kosten auszudrücken. Eine Kostenmatrix hat das gleiche Layout wie eine Konfusionsmatrix – aber hier enthalten die Einträge einen Preis, mit dem die Anzahl der Instanzen, die in die Zelle fallen, multipliziert wird, um Teilkosten zu erhalten. Das Aufsummieren dieser Teilkosten ergibt die Gesamtkosten, die unsere neue Evaluationsmetrik darstellen.

		Vorhersage	
		Ja	Nein
Echtes Interesse	Ja	-39	0
	Nein	1	0

Abbildung 7: Kostenmatrix für den Swiss-Bikes-Wintercheck Merkmale (eigene Illustration)

Abbildung 7 zeigt die Kostenmatrix für den SB-Wintercheck, welche auf den oben bereits beschriebenen Annahmen beruht: Wenn der Klassifikator «nein» vorhersagt, wird kein Brief verschickt, es entstehen keine Kosten. Wenn der Klassifikator «ja» richtig vorhersagt, verdient Swiss Bikes CHF 39 (CHF 40 Gewinn minus CHF 1 Porto). Ist die Vorhersage «ja» falsch, so entstehen Kosten von CHF 1.

Mit diesen Überlegungen und basierend auf der Konfusionsmatrix in Abbildung 6 erhalten wir Kosten von CHF 0 für den Baseline-Klassifikator (schickt nie einen Brief) und Kosten von CHF –1 663 (d. h. einen Gewinn von CHF 1 663) für unseren anderen Klassifikator, siehe auch die obenstehenden Berechnungen. Dieses Evaluationsergebnis drückt ziemlich genau und leicht verständlich aus, welchen wirtschaftlichen Nutzen der zweite Klassifikator verspricht.

Schliesslich soll noch darauf verwiesen werden, dass oft auch qualitative Evaluationen sinnvoll sind: wenn Modelle nach einiger Optimierung von Parametern quantitativ, also hinsichtlich o. g. Metriken, nicht wie gewünscht abschneiden, so kann es sich lohnen, sich einige falsch positive oder falsch negative Vorhersagen im Detail zu anzusehen und festzustellen, was mögliche Gründe für die Fehlklassifikation sein könnten. Oft stösst man dabei beispielsweise auf Ideen für weitere Merkmale (Features) – beispielsweise könnte es sein, dass man bei der Analyse des Swiss-Bikes-Winterchecks feststellt, dass unter den falsch negativ klassifizierten Kund:innen viele sind, die zwar nur wenige Reparaturen haben durchführen lassen, aber über eine Kundenkarte verfügen. Dies könnte die Verwendung eines Merkmals «Hat-Kundenkarte» nahelegen. Damit lassen sich dann evtl. stark verbesserte Vorhersageergebnisse erzielen.

Deployment

Wenn die Evaluation ein geeignetes Modell identifiziert hat, für das ein ausreichend hoher wirtschaftlicher Nutzen erwartet wird, dann wird es Zeit, sich über den konkreten praktischen Einsatz des Modells Gedanken zu machen.

Dabei spielen zunächst technische Erwägungen eine Rolle, wie die Einbettung des Modells in die **digitalen Umsysteme** des betreffenden Unternehmens. Im Beispiel Swiss Bikes könnte z. B. ein Customer-Relationship-Management-System existieren, aus welchem die Daten für das Training des Modells stammen, in dem aber ggf. auch die Ausgabe des Modells erfolgen soll, sodass die vom Modell als interessiert vorhergesagten Kund:innen direkt für eine Kampagne selektiert werden können. Die entsprechende technische Integration muss vorbereitet und gründlich getestet werden, bevor sie den Mitarbeitenden zur Verfügung gestellt wird.

Neben der technischen Ebene spielt auch die Einbettung in die **Arbeitsprozesse** eine grosse Rolle. Diese betriff die im letzten Absatz bereits angedeutete digitale Unterstützung der Prozesse. Oft ist aber auch zu bedenken, dass die Vorhersagen eines Modells nicht sofort und 1:1 in Handlungen umgesetzt werden sollen. Dies mag beim Swiss-Bikes-Wintercheck unbedenklich sein (d. h. die selektierten Kund:innen können ohne menschliche Prüfung direkt kontaktiert werden), ist aber in vielen anderen Fällen problematisch. Wenn z. B. ein Modell Versicherungsbetrug vorhersagt, wird die Versicherung meistens nicht automatisch eine Nachricht an die Kundschaft generieren lassen, um mitzuteilen, dass der Schaden nicht übernommen wird. Vielmehr wird man in diesen Fällen eine menschliche Prüfung veranlassen. Hier kommt wieder die oben erwähnte Interpretierbarkeit von Modellen ins Spiel: Je transparenter die Gründe für die Vorhersage eines Modells sind, desto einfacher ist es für die menschlichen Akteure, diese zu prüfen und weiterzuverwenden. Dies ist auch für die Akzeptanz des Systems durch die Mitarbeitenden, deren Arbeitsprozesse es unterstützen soll, von grosser Bedeutung.

Schliesslich ist es wichtig, die Gültigkeit des Modells angesichts sich ändernder Bedingungen fortlaufend zu prüfen. Man sollte dabei insbesondere beobachten, ob sich die Verteilung von Werten bei den Merkmalen, mit denen das Modell trainiert wurde, verändert. Im Beispiel von Swiss Bikes kann man beispielsweise sehen, dass in der verwendeten Stichprobe (siehe Abbildung 3) keine Kund:innen mit E-Bikes vertreten sind, was möglicherweise daran liegt, dass die Daten schon vor längerer Zeit gesammelt wurden und die heutzutage grössere Verbreitung von E-Bikes nicht ausreichend widerspiegeln. Bei solchen Verschiebungen (engl. «Feature Drift») ist es oft ratsam, ein Modell mit neuen Daten zu trainieren. Wenn diese nicht in ausreichender Menge vorliegen, muss entschieden werden, ob das alte Modell angesichts des beobachteten Feature Drifts noch vertrauenswürdig ist.

Fazit

Wie im einleitenden Teil dieses Kapitels beschrieben, haben Klassifikations- und Regressionsmodelle das Potenzial, häufig wiederkehrende, wissensintensive operative Entscheidungen in Unternehmen nicht nur (teilweise) zu automatisieren, sondern auch qualitativ besser zu machen.

Um dieses Potenzial auszuschöpfen, ist es wichtig, Entscheidungen als Klassifikations- oder Regressionsaufgaben formalisieren zu können (bzw. erkennen zu können, für welche Aufgaben dies möglich und sinnvoll ist), vorhandene Unternehmensdaten so aufbereiten zu können, wie es als Eingabe von den entsprechenden Data-Mining-Werkzeugen erwartet wird, die richtigen Algorithmen und deren Parameter zu finden, die Güte und den wirtschaftlichen Nutzen eines Modells zu beurteilen und die Einbettung in die Arbeitsprozesse eines Unternehmens sinnstiftend gestalten zu können. Für die meisten dieser Schritte ist kein vertieftes Verständnis der Algorithmen (und der zugrundeliegenden Mathematik) nötig, welche die eigentliche Modellbildung vornehmen. Dies und der Umstand, dass die Suche nach den besten ML-Konfigurationen heute schon gut durch Auto-ML-Werkzeuge unterstützt wird, führen dazu, dass maschinelles Lernen immer zugänglicher wird und das oben erwähnte Potenzial mit relativ geringem Aufwand realisiert werden kann.

Dabei steckt wie oben erwähnt oft die meiste Arbeit in der Aufbereitung der Daten, welche zum Trainieren der Modelle verwendet werden. Die hierbei ggf. aufgedeckten Qualitätsprobleme sollten dabei am besten an der Quelle behoben werden: Wenn z. B. gewisse Daten systematisch fehlen, sollte man nicht (nur) im Rahmen eines ML-Projekts eine schnelle Lösung zum Auffüllen der fehlenden Werte wählen, sondern auch darüber nachdenken, wie das Fehlen in Zukunft vermieden werden kann. Dies setzt oft eine Zusammenarbeit von Data Scientists (welche meist für die Aufbereitung der Daten zuständig sind) mit diversen anderen Akteur:innen innerhalb des Unternehmens voraus – das maschinelle Lernen kann als eine Aufgabe begriffen werden, bei der alle im Unternehmen mitarbeiten (müssen), um gewisse Entscheidungen datengetriebener treffen zu können.

Referenzen

Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence, 1(5), 206–215. doi.org/10.1038/s42256-019-0048-x

Tan, P.-N., Steinbach, M., & Kumar, V. (2005). Introduction to Data Mining. Addison Wesley. www-users.cs.umn.edu/~kumar/dmbook/index.php

Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a standard process model for data mining. Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining, 1, 29–39.