KI für Alle 2: Verstehen, Bewerten, Reflektieren

Themenblock Prognosemodelle (Klassifikation und Regression): 02_03Verfahren_kNN

Das k-nearest neighbours Verfahren

Erarbeitet von
Dr. Katja Theune

Lernziele 1
Inhalt 2
Einstieg 2
k-nearest neighbours Verfahren: Idee 2
Distanzmaße: Euklidische Distanz 3
Weitere Distanzmaße 4
Die Wahl von Hyperparameter k 5
Diskussion, Vor- und Nachteile 5
Abschluss 6
Weiterführendes Material 6
Disclaimer 6

Lernziele

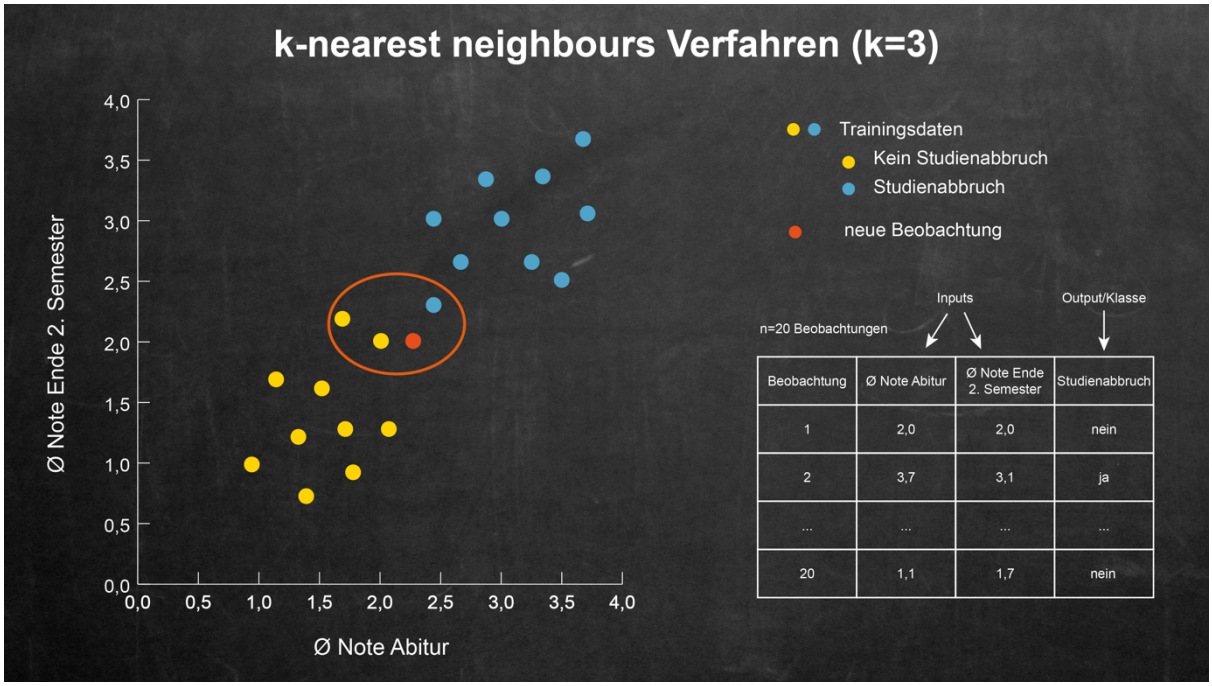
- Du kannst anhand eines einfachen Beispiels mittels der Euklidischen Distanz die nächsten Nachbarn einer neuen Beobachtung bestimmen
- Du kannst anhand eines einfachen Beispiels mittels der Euklidischen Distanz den prognostizierten Output für eine neue Beobachtung bestimmen
- Du kannst die Problematik bei der Wahl von k erläutern
- Du kannst Vor- und Nachteile des k-nearest neighbours Verfahrens erläutern

Inhalt

Einstieg

Das k-nearest neighbours, oder auf Deutsch k-nächste Nachbarn Verfahren, gehört zu den einfachsten Verfahren des maschinellen Lernens und ist immer dort beliebt, wo es um Ähnlichkeiten zwischen Beobachtungen geht. Also z. B. beim Verhalten von Käufer*innen oder Nutzer*innen von Streamingdiensten.

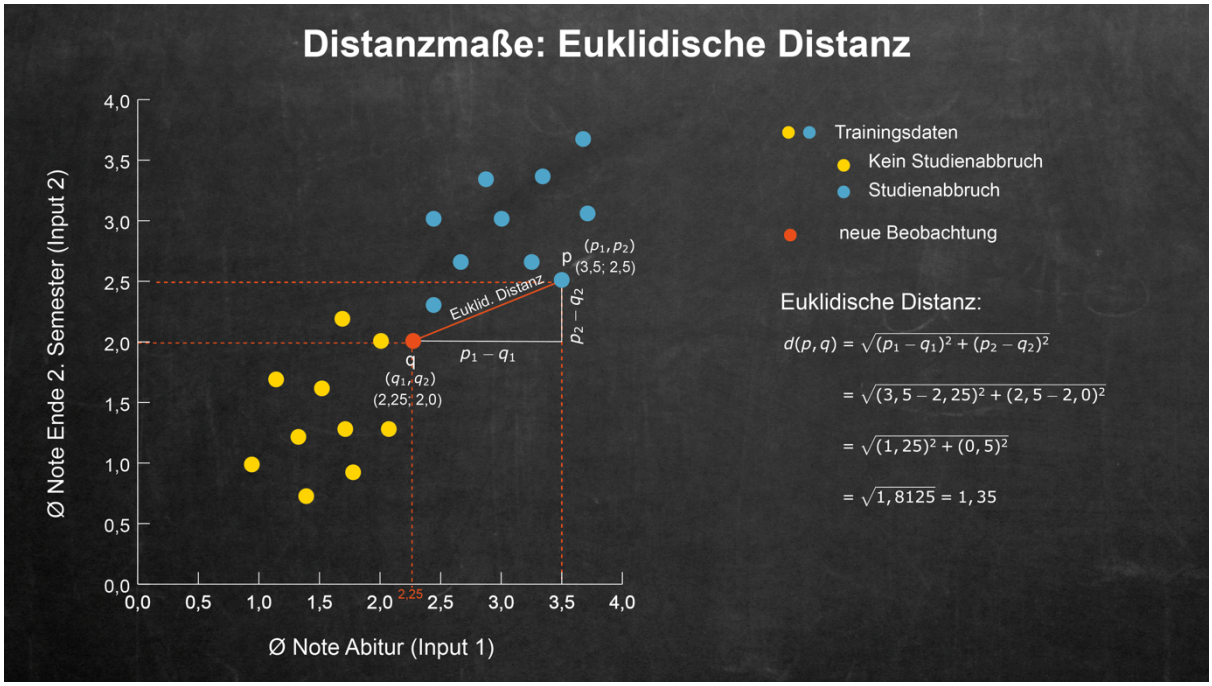
k-nearest neighbours Verfahren: Idee



Das k-nearest neighbours Verfahren nutzt zur Klassifikation das Konzept der Nähe bzw. Ähnlichkeit zwischen Beobachtungen. Eine neue Beobachtung wird auf Basis ihrer k nächsten Nachbarn klassifiziert, und zwar durch einen Mehrheitsentscheid. Schauen wir uns wieder passend zu unserem Anwendungsbeispiel einen fiktiven Beispieldatensatz mit 20 Studierenden an. Als einzige Inputs betrachten wir hier die durchschnittliche Abiturnote und die durchschnittliche Note am Ende des 2. Semesters. Der Output, also unsere Klassen, beschreibt, ob ein Studium abgebrochen wurde oder nicht. Hier würden wir für unsere neue orange Beobachtung bei k = 3 die Klasse „kein Studienabbruch", hier in Gelb, prognostizieren. Diese kommt unter den drei Nachbarn, die orange umkreist sind, am häufigsten vor.

Das k-nearest neighbours Verfahren kann man auch für eine Regression verwenden. Bei einem metrischen Output würde man dann anstatt eines Mehrheitsentscheids z. B. das Mittel der Outputs der k nächsten Nachbarn berechnen und diesen Wert für unsere neue Beobachtung prognostizieren.

Distanzmaße: Euklidische Distanz



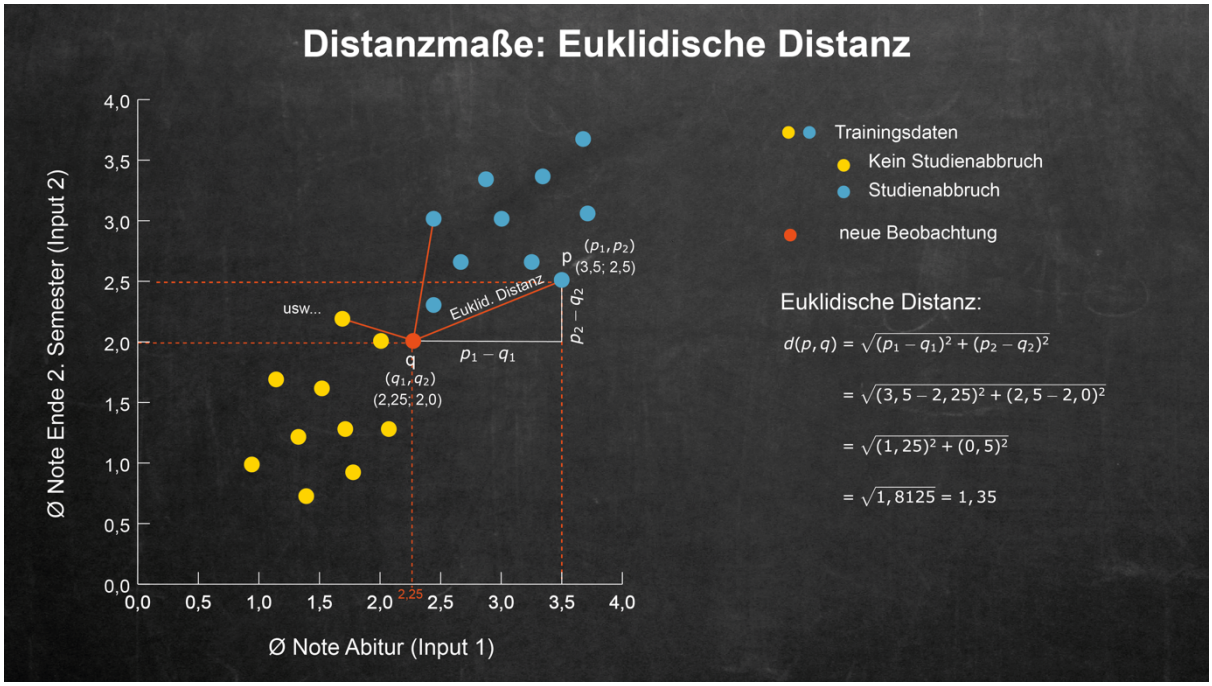
Um die Nähe zwischen Datenpunkten zu messen bzw. zu berechnen, muss ein sogenanntes Distanzmaß gewählt werden. Sehr häufig wird die Euklidische Distanz verwendet. Sie wird auch als Fluglinie bezeichnet und entspricht der Länge der Geraden, die wir zwischen zwei Beobachtungen ziehen können. Anstatt Distanz könnten wir auch Abstand sagen. Aber wie wird sie berechnet?

Schauen wir uns zur besseren Vorstellung einmal unser Beispiel für zwei Inputs an und berechnen die Euklidische Distanz zwischen unserer neuen, orangen Beobachtung q und der blauen Beobachtung p. Unsere neue Beobachtung q hat für Input 1 den Wert $q_1 = 2{,}25$ und für Input 2 den Wert $q_2 = 2{,}0$. Die blaue Beobachtung p hat die Werte $p_1 = 3{,}5$ und $p_2 = 2{,}5$. Damit kann man dann zunächst für beide Inputs die beiden Differenzen bzw. Distanzen $p_1 - q_1$ und $p_2 - q_2$ bestimmen.

Die Euklidische Distanz d berechnet sich dann folgendermaßen (siehe Grafik).

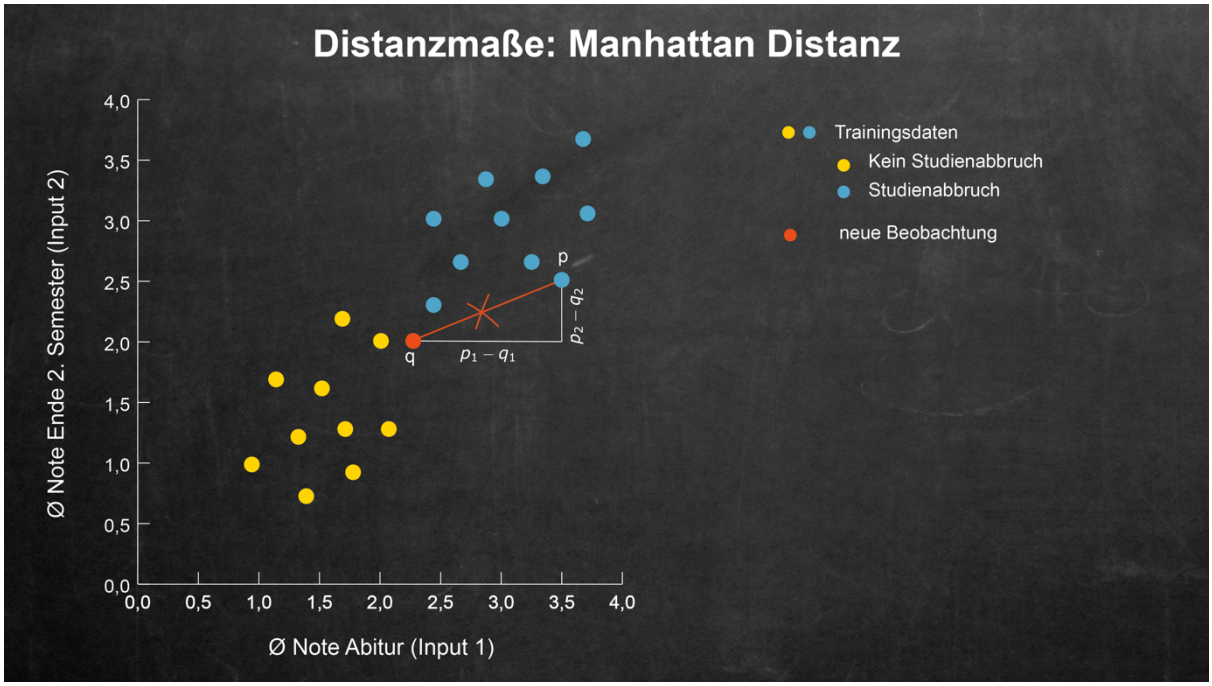
Wir quadrieren diese einzelnen Distanzen, da es nicht auf die Richtung des Abstands ankommt, das Vorzeichen also irrelevant ist. Große Distanzen werden zudem durch die Quadrierung besonders bestraft. Diese einzelnen quadrierten Distanzen werden dann aufsummiert und dann noch die Wurzel gezogen. Wir erhalten eine Distanz von 1,35.

Das können wir jetzt für alle Distanzen zwischen unserer neuen orangen Beobachtung und allen anderen blauen und gelben Beobachtungen machen. Für die Klassifikation wählen wir dann die k Beobachtungen mit den kleinsten Distanzwerten zu unserer neuen Beobachtung.



Weitere Distanzmaße

Ein weiteres Distanzmaß ist die Manhattan oder City-Block Distanz. Sie erinnert daran, wie man mit einem Taxi in Manhattan durch die ganzen Häuserblocks zu seinem Zielort kommen würde. Und so wird diese Distanz auch berechnet.



Diesmal geht es nicht um die Berechnung der Diagonalen, sondern es werden einfach die horizontalen und vertikalen absoluten Distanzen aufsummiert. Absolut bedeutet, dass wir nur den eigentlichen Wert eines Abstands betrachten und nicht sein Vorzeichen.

Um den Abstand bei kategorialen Inputs zu bestimmen, kann man z. B. die Hamming-Distanz verwenden. Diese zählt einfach die Unterschiede in den Ausprägungen der Inputs zwischen zwei Beobachtungen. Es gibt aber natürlich noch einige weitere mögliche Distanzmaße für verschiedene Datentypen.

Die Wahl von Hyperparameter k

Die Anzahl der nächsten Nachbarn k, die wir für die Klassifikation verwenden wollen, ist ein Hyperparameter des Modells, den man festlegen muss bzw. optimieren kann. Die Wahl von k unterliegt einem Verzerrungs-Varianz Trade-off. Ein kleines k führt zu einem sehr flexiblen Modell, dass sich gut an die Trainingsdaten anpasst. Mit kleinem k tendiert das Modell daher zu einer kleinen Verzerrung, aber zu einer hohen Varianz, also zur Überanpassung. Es generalisiert dann nicht gut für andere Daten. Ein Extremfall wäre k = 1. Dann bestimmt nur ein einziger Nachbar die Klassifikation für eine neue Beobachtung. Dagegen führt die Wahl eines sehr großen k zur Unteranpassung des Modells. Wir haben dann meist eine sehr hohe Verzerrung, aber dafür eine kleinere Varianz. Ein Extremfall wäre, wenn k der Anzahl an Beobachtungen entspräche. Dann würde immer die Klasse, die in den Trainingsdaten in der Mehrheit ist, prognostiziert werden. Das beste k liegt also irgendwo dazwischen.

Es gibt verschiedene Möglichkeiten, wie Daumenregeln oder eine cross-validation, um k auszuwählen. Bei einer Klassifikation mit zwei Klassen kann man sich aber schonmal merken, dass ein ungerades k von Vorteil ist, denn dann ist immer eine Klasse in der Mehrheit.

Diskussion, Vor- und Nachteile

Ein bedeutender Vorteil des k-nearest neighbours Verfahrens ist seine intuitive Herangehensweise. Außerdem ist die Trainingsphase, die im eigentlichen Sinne nicht aus Lernen besteht, im Gegensatz zu anderen Verfahren sehr schnell und kann sich dadurch auch schnell auf neue oder veränderte Daten einstellen. Dagegen ist seine dadurch resultierende langsame Klassifizierungsphase ein Nachteil. Man nennt dieses Verfahren daher auch lazy.

Ein Nachteil ist, dass wir die zugrundeliegenden Daten umfangreich aufbereiten müssen. Das k-nearest neighbours Verfahren verwendet z. B. Distanzen zur Klassifikation. Diese sind aber sehr abhängig von der Bandbreite der Werte der einzelnen Inputs. Wir können uns vorstellen, dass Inputs mit großer Bandbreite das Distanzmaß und damit die Klassifikation dominieren würden. Das ist aber nicht erwünscht. Hat man Inputs mit ganz unterschiedlicher Bandbreite, sollten wir diese so aufbereiten, dass sie einen ähnlichen Beitrag zum Distanzmaß liefern, also z. B. normalisieren. Zudem müssen wir jeweils ein passendes Distanzmaß auswählen. Außerdem kann das k-nearst neighbours Verfahren auch nicht mit fehlenden Werten umgehen.

Abschluss

Wir haben das intuitive k-nearest neighbours Verfahren kennengelernt und auch verschiedene Distanzmaße, um die Nachbarn zu bestimmten. Wichtig ist hier die Wahl der Anzahl an zu berücksichtigenden Nachbarn k. Sowohl die Wahl von k als auch die nötige Datenaufbereitung können große Auswirkungen auf unsere Ergebnisse haben.

Weiterführendes Material

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining: Concepts and Techniques* (3. Auflage). Morgan Kaufmann.

James, G., Witten, D., Hastie, T., & Tibshirani, R., & Tylor, J. (2023). *An Introduction to Statistical Learning - with Applications in Python*. Springer.

Lantz, B. (2015). *Machine learning with R* (2. Auflage). Packt Publishing Ltd, Birmingham.

Disclaimer