Woche 06: Wie war das nochmal? – Überblick und Ausblick

Skript

Erarbeitet von
Dr. Maike Mayer

Lernziele

- Erinnern wichtiger Inhalte zu Datentypen und dem Training von KI-basierten Systemen
- Erinnern der Teile des Maschinellen Lernens
- Nachvollziehen der Zusammenhänge zwischen den Inhalten

Inhalt

Einstieg

In der dritten Woche dieses Kurses hast du gelernt, dass sich Maschinelles Lernen in zwei Teile oder Felder einteilen lässt. Und natürlich in drei Kategorien – methodisch betrachtet –, aber die kannst du mittlerweile bestimmt schon runterbeten. Zurück zu den Teilen: Es gibt

Projekt „KI für alle"
HeiCAD – Heine Center for Artificial Intelligence and Data Science

ein neueres Feld bzw. einen neueren Teil des Maschinellen Lernens, nämlich das Deep Learning, das tiefe Lernen aus Daten, mit seinen Modellen, den neuronalen Netzen. Und es gibt einen eher traditionellen Teil, der sich mit statistischen Lernmethoden wie Entscheidungsbäumen, linearer Regression oder Clustering beschäftigt. Der Vorteil: Diese Lernmethoden funktionieren – im Gegensatz zu Deep Learning – auch mit kleineren Datenmengen schon gut.

Einblendung Schlagwörter/Illustrationen

Aber warum erzähle ich dir das? Ich erzähle dir das, weil wir uns in dieser Woche näher mit einer dieser statistischen Lernmethoden beschäftigen wollen. Also ehrlich gesagt (Achtung: Spoiler!) werden dir alle gerade genannten Lernmethoden in den nächsten Wochen begegnen, aber irgendwo müssen wir ja anfangen. Und diese Woche steht die Regression im Rampenlicht.

Einblendung Schlagwörter/Illustrationen

Vorhang auf: Die Regression

Bei der Regression handelt es sich um ein Verfahren des Supervised Learnings, mit dem wir uns in der letzten Woche näher beschäftigt haben. Mit Hilfe einer Regression kann man Zusammenhänge zwischen einer Zielgröße oder einem Zielmerkmal (wie der Wohnungsmiete) und verschiedenen anderen Features bzw. anderen Merkmalen (wie beispielsweise der Wohnungsgröße) bestimmen. Exemplarisch lernst du in dieser Woche die lineare und die logistische Regression kennen. Eine lineare Regression wird eingesetzt, wenn sich der Zusammenhang zwischen dem Zielmerkmal und einem Feature optisch durch eine gerade Linie – also durch eine Gerade – beschreiben lässt. Eine logistische Regression wird verwendet, wenn wir den Zusammenhang zwischen einem Feature und der Wahrscheinlichkeit, in eine von zwei Kategorien des Zielmerkmals zu fallen, messen wollen. Dieser Zusammenhang lässt sich mit einer S-Kurve beschreiben.

Einblendung Schlagwörter/Illustrationen

Für Regressionen werden in der Regel metrische Zielmerkmale verwendet. Metrisch? Da klingelt was, oder? Zu Beginn dieses Kurses hast du schonmal gehört, was metrische Daten sind. Sie werden auch als numerische Daten bezeichnet und umfassen – einfach gesagt – Zahlen. Also natürliche Zahlen, ganze Zahlen und Kommazahlen.

Moment! Aber eben war doch auch von Kategorien die Rede … Gut aufgepasst. Logistische Regressionen können für kategoriale Merkmale mit zwei Ausprägungen verwendet werden. Auch hier kurz zur Erinnerung: Kategoriale Daten können Kategorien wie Studienfächer oder Wohnorte sein oder auch Skalen, bei denen nicht die Werte selbst entscheidend sind, sondern ihre Reihenfolge. Als Zielmerkmal werden bei einer logistischen Regression jedoch

nicht die Kategorien selbst verwendet, sondern die Wahrscheinlichkeit, zu einer Kategorie zu gehören. Und diese Wahrscheinlichkeit ist wieder metrisch.

Einblendung Schlagwörter/Illustrationen

Unabhängig von metrischen und kategorialen Daten und von der Art der Regression ist das Ziel einer Regression jedoch das Gleiche: Mit Hilfe der Geraden oder Kurve können wir Zusammenhänge zwischen dem Zielmerkmal und dem Feature beschreiben und so Vorhersagen für neue Werte eines Features in Hinblick auf das Zielmerkmal machen. Also beispielsweise für eine neue Wohnung mit einer bestimmten Größe die voraussichtliche Miete vorhersagen.

So viel zur Theorie. Wie du vielleicht schon geahnt hast, zeigen wir dir diese Woche dann auch, wie Regressionen bzw. die dazugehörigen Regressionsmodelle in Python umgesetzt werden können. Dafür benötigst du sowohl die Datenstruktur Numpy Array, die du bereits kennst, als auch ein neues Modul, nämlich Scikit-learn. Mit Hilfe dieses Moduls zeigen wir dir unter anderem, wie du ein Regressionsmodell trainieren kannst. Und wie so ein Training funktioniert, weißt du schon aus der letzten Woche.

Einblendung Schlagwörter/Illustrationen

Zeit fürs Training

Wird ein KI-basiertes System trainiert, dann ist das Ziel dieses Trainings, dass das Modell für neue, unbekannte Input-Daten eine gute Vorhersage treffen kann. Um ein System zu trainieren, teilen wir unseren Datensatz zunächst in eine Trainings- und eine Testmenge auf, die sich nicht überschneiden dürfen. Mit der Trainingsmenge, die in der Regel größer als die Testmenge ist, lernt das System Zusammenhänge zwischen den Daten, also beispielsweise zwischen dem Zielmerkmal und einem Feature. Mit der Testmenge wird dann überprüft, ob das System, basierend auf dem, was es mit den Trainingsdaten gelernt hat, auch für die neuen Daten der Testmenge gute Vorhersagen trifft. Wenn man bereits während des Trainings testen will, kann man auch noch eine sogenannte Validierungsmenge erstellen.

Einblendung Schlagwörter/Illustrationen

Bei Trainings-, Validierungs- und Testmenge müssen wir jedoch sicherstellen, dass sie repräsentativ für unsere gesamten Daten sind, wir also beispielsweise gleich viele Hunde- und Katzenbilder in der jeweiligen Menge haben. Wenn die Trainingsmenge beispielsweise viel mehr Hunde- als Katzenbilder enthält, kann das System zwar Hunde gut erkennen, wird aber bei Katzenbildern eher schlecht aufgestellt sein.

Einblendung Illustrationen

Aber nicht nur die Repräsentativität der Daten kann einen Einfluss auf die Vorhersagequalität des Systems haben. In dieser Woche wollen wir einen näheren Blick darauf werfen, was passiert, wenn man die Komplexität des Zusammenhangs zwischen Zielmerkmal und Feature über- bzw. unterschätzt. Dies bezeichnet man dann als Overfitting – also Überanpassung – bzw. als Underfitting – also Unteranpassung. Und auch über Outlier bzw. Ausreißer wollen wir diese Woche sprechen sowie über ihre Auswirkungen auf das Training eines Systems.

Einblendung Schlagwörter

Auf geht's!

In dieser Woche steht also die Regression im Fokus. Zunächst lernst du den theoretischen Hintergrund der linearen und logistischen Regression kennen. Im Anschluss zeigen wir dir dann selbstverständlich auch, wie du ein Regressionsmodell in Python erstellen und trainieren kannst. Wir werden in dieser Woche aber auch nochmal einen näheren Blick auf das Training von Systemen werfen und dir Überanpassung, Unteranpassung und Ausreißer vorstellen. Abschließend haben wir dann noch ein paar Anwendungsbeispiele für Regressionen für dich zusammengestellt – beispielsweise bei der Frage, ob mehr Geld glücklicher macht. Mit Hilfe von Störchen und Babys wollen wir dann aber auch noch auf einen im Alltag weit verbreiteten Fehler hinweisen: Ein Zusammenhang zwischen Merkmalen allein sagt nichts über Ursache und Wirkung aus.

Einblendung der Videotitel

Quelle [1]

Also Augen auf bei Scheinzusammenhängen und viel Spaß in Woche 6!

Quellen

Quelle [1] Matthews, R. (2000). Storks deliver babies (p= 0.008). *Teaching Statistics, 22*(2), 36-38. https://doi.org/10.1111/1467-9639.00013

Disclaimer

Transkript zu dem Video „Woche 06: Wie war das nochmal? – Überblick und Ausblick“, Dr. Maike Mayer.