Sigmoidfunktion

Autor und Herausgeber

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland (ORCID 0000-0001-7538-0990)
Editors: Konstantina Olioumtsevits (ORCID 0000-0003-2697-8659), Ekkehard Schnoor (ORCID 0000-0001-8252-1499), Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Zusammenfassung

Die Sigmoidfunktion bildet die reelle Zahlengerade auf das offene Intervall $(0, 1)$ ab. Sie übersetzt den reellwertigen Score eines binären Klassifikators in eine Schätzung für die Wahrscheinlichkeit eines bestimmten Labelwerts. Sie ist glatt und streng monoton wachsend, mit einer Ableitung, die sich mit wachsendem Betrag des Arguments der Null nähert. Das maschinelles Lernen (ML) nutzt sie in zwei Rollen: als Aktivierungsfunktion in einem Neuron und als Teil der Log-Likelihood-Funktion zum Lernen der Modellparameter eines Wahrscheinlichkeitsmodells.

Definition

Eine Frostwarnung für Krems an der Donau (Österreich) muss angeben, wie wahrscheinlich Frost ist, und nicht nur, ob er erwartet wird. Abb. 1 zeigt die an dieser Station um 06:00 (UTC) gemessene Lufttemperatur. Ein Klassifikator, der vor Frost warnt, liest den 06:00-Messwert eines Tages, seinen Merkmalsvektor $\mathbf{x}$ mit diesem einzelnen Merkmal, ein und liefert einen reellwertigen Score $h(\mathbf{x})$. Die Klassifikation erfolgt über einen Schwellenwert: Eine Frostwarnung für den nächsten Tag wird genau dann ausgegeben, wenn $h(\mathbf{x}) > 0$, und der Betrag $|h(\mathbf{x})|$ misst die Konfidenz in diese Warnung. In der Praxis ist es oft nützlicher, eine Wahrscheinlichkeit anzugeben. Die Sigmoidfunktion übersetzt den Score $h(\mathbf{x})$ in eine Zahl zwischen 0 und 1, die als Wahrscheinlichkeit für Frost gelesen werden kann, wie Abb. 1 für den letzten Tag zeigt.

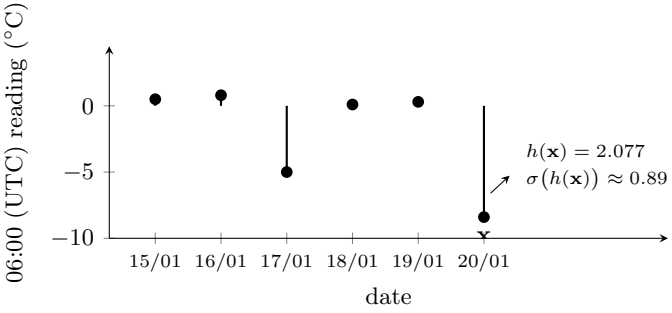

Abbildung 1: Sechs aufeinanderfolgende Tage in Krems an der Donau; jeder Balken zeigt die an diesem Tag um 06:00 (UTC) gemessene Lufttemperatur. Der Messwert des letzten Tages ist der Merkmalsvektor $\mathbf{x}$, den ein Klassifikator einliest und aus dem er den Score $h(\mathbf{x}) = 2.077$ berechnet. Dieser Score ist positiv, sodass eine Frostwarnung ausgegeben wird, und die Sigmoidfunktion wandelt ihn in $\sigma(2.077) \approx 0.89$ um, was einer Wahrscheinlichkeit von 89 Prozent für Frost am 21.01. entspricht. Daten erzeugt mit `pythondemos/sigmoid.py`

Die Sigmoidfunktion bildet die reelle Zahlengerade auf das offene Intervall $(0, 1)$ ab. Sie ist die Funktion $\sigma : \mathbb{R} \to (0, 1)$, die gegeben ist durch

$$\sigma(z) := \frac{1}{1 + e^{-z}}.$$

Sie steigt von $\sigma(z) \to 0$ für $z \to -\infty$ auf $\sigma(z) \to 1$ für $z \to +\infty$ an und verläuft durch $\sigma(0) = 1/2$ (siehe Abb. 2). Sie wird auch als logistische Funktion bezeichnet. Für zwei Klassen mit den Scores z und 0 liefert die Softmax-Funktion das Paar $\big(\sigma(z), 1 - \sigma(z)\big)$, sodass die Sigmoidfunktion die Softmax-Funktion für zwei Klassen ist: Sie liefert die Wahrscheinlichkeit der einen Klasse, und die andere Klasse hat die GegenWahrscheinlichkeit.

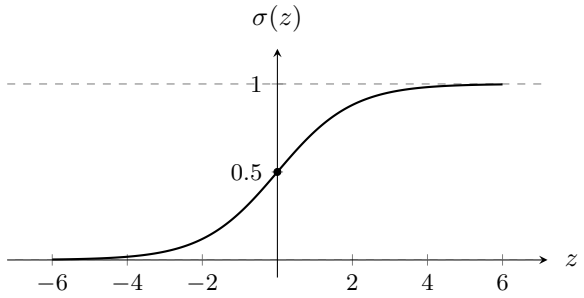

Abbildung 2: Die Sigmoidfunktion $\sigma(z) = 1/(1 + e^{-z})$ bildet die reelle Zahlengerade auf das offene Intervall $(0, 1)$ ab, mit $\sigma(0) = 1/2$

Die Sigmoidfunktion ist glatt und streng monoton wachsend, und sie erfüllt die Symmetrie $\sigma(-z) = 1-\sigma(z)$ (Bishop, 2006, Abschn. 4.2). Ihre Ableitung liegt

in geschlossener Form vor und lässt sich durch die Funktion selbst ausdrücken (Bishop, 2006, Abschn. 4.3.2, Gl. (4.88)),

$$\sigma'(z) = \sigma(z)\big(1 - \sigma(z)\big).$$

Diese Eigenschaften machen die Sigmoidfunktion zu einer beliebten Wahl für die Aktivierungsfunktion eines künstliches neuronales Netz (KNN)s. Ein Nachteil dieser Wahl besteht darin, dass die Ableitung schnell abnimmt, wenn das Argument dem Betrag nach wächst, was den Fortschritt Gradientenbasierter Lernverfahren verlangsamt – das Problem verschwindender Gradienten (Vanishing-Gradient-Problem). Das Problem verschwindender Gradienten ist der Grund, warum sigmoidale Aktivierungsfunktionen in den verdeckten Schichten vorwärtsgerichteter tiefe Netze selten verwendet werden; dort werden andere Aktivierungsfunktionen wie die ReLU (ReLU) bevorzugt (Goodfellow et al., 2016, Abschn. 6.3.2).

Ein Wahrscheinlichkeitsmodell für ein binäres Label verwendet die Sigmoidfunktion, um einen Score in eine Wahrscheinlichkeit umzuwandeln. Sei das Label y entweder 1 oder -1 und der Score $\mathbf{w}^\top \mathbf{x}$ mit den Modellparameter $\mathbf{w}$. Das Modell lautet

$$\mathbb{P}\,(y = 1 \mid \mathbf{x}) = \sigma\big(\mathbf{w}^\top \mathbf{x}\big), \qquad \mathbb{P}\,(y = -1 \mid \mathbf{x}) = \sigma\big(-\mathbf{w}^\top \mathbf{x}\big),$$

wobei die zweite Gleichung aus der Symmetrie $\sigma(-z) = 1 - \sigma(z)$ folgt. Beide Fälle lassen sich als der einzelne Ausdruck $\sigma(y\,\mathbf{w}^\top \mathbf{x})$ schreiben, wodurch die Sigmoidfunktion in die Log-Likelihood eingeht. Für ein Trainingsdatensatz $\mathcal{D}^{(\text{train})} = \big\{\,\big(\mathbf{x}^{(r)}, y^{(r)}\big)\,\big\}_{r=1}^{m}$ aus gelabelte Datenpunkte lautet die Log-Likelihood der Modellparameter

$$\sum_{r=1}^{m} \ln \sigma\big(y^{(r)} \mathbf{w}^\top \mathbf{x}^{(r)}\big).$$

Deren Maximierung über $\mathbf{w}$ ist eine maximale Likelihood-Schätzung und entspricht der logistische Regression (Bishop and Bishop, 2024, Abschn. 5.4.3; Shalev-Shwartz and Ben-David, 2014, Abschn. 9.3). Jeder Summand ist das Negative eines logistische Verlustfunktiones, sodass die Maximierung der Log-Likelihood und die Minimierung des durchschnittlichen logistische Verlustfunktiones dieselbe Berechnung darstellen (Bishop, 2006, Abschn. 4.3.2). Betrachtet sei beispielsweise die Klassifikatorabbildung h, die mittels logistische Regression aus den Tagen des Jahres 2024 in Krems gelernt wurde. Wo ihr Score null ist, liefert die Sigmoidfunktion den Wert 1/2, sodass die Warnung genau dort ausgegeben wird, wo die Wahrscheinlichkeit ein Halb überschreitet. Ein Score von 0.907 wird zu einer Wahrscheinlichkeit von 0.71 für Frost am nächsten Tag, und ein Score von -2.534 wird zu 0.07.

Synonyme: logistische Funktion.

Siehe auch: Aktivierungsfunktion, ReLU, Softmax-Funktion, logistische Regression, lineares Modell, Klassifikator, Wahrscheinlichkeit, glatt.

Querverweise

Literatur

1. Bishop (2006). Pattern Recognition and Machine Learning. Springer Science+Business Media, New York, NY, USA.

2. Bishop and Bishop (2024). Deep Learning: Foundations and Concepts. Springer Nature, Cham, Switzerland.

3. Goodfellow et al. (2016). Deep Learning. MIT Press, Cambridge, MA, USA.

4. Shalev-Shwartz and Ben-David (2014). Understanding Machine Learning: From Theory to Algorithms. Cambridge Univ. Press, New York, NY, USA.