Hyperparameter

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

Ein Hyperparameter ist jede Größe, welche die Komponenten einer Methode des maschinellen Lernens festlegt. Beispiele hierfür sind der maximale Grad bei der polynomialen Regression, die Stärke der Regularisierung bei der Ridge-Regression und die Anzahl der Datenpunkte, die zur Konstruktion eines Trainingsdatensatzes verwendet werden. Hyperparameter können auch das Verhalten des Optimierungsverfahrens bestimmen, das zur Lösung der empirischen Risikominimierung eingesetzt wird, also um ein Modell zu trainieren; ein Beispiel ist die Lernrate beim Gradientenabstieg. Ein Hyperparameter wählt eine Methode des maschinellen Lernens aus mehreren Alternativen aus und kann daher durch Techniken der Modellauswahl bestimmt werden, etwa solche, die auf dem Validierungsfehler basieren.

Definition

Die Vorhersage der Lufttemperatur um 15:00 (UTC) eines Tages in Krems an der Donau (Österreich) aus dem Messwert von 06:00 (UTC) erfordert Entscheidungen, bevor überhaupt ein Training stattfindet: welche Abbildungen die Methode liefern darf, wie viele Datenpunkte zu erheben sind und wie diese Abbildungen durchsucht werden sollen. Abb. 1 zeigt zwölf Tage als Datenpunkte, jeweils mit dem Messwert von 06:00 als Merkmal x und dem Messwert von 15:00 als Label y. Drei Geraden stellen Hypothesen des zugrunde liegenden Hypothesenraums bzw. Modells der Methode des maschinellen Lernens dar. Eine Verlustfunktion dient dazu, den auf einem Datenpunkt auftretenden Fehler der Vorhersage zu messen.

Ein Hyperparameter ist jede Größe, die eine dieser Komponenten einer Methode des maschinellen Lernens festlegt. Der maximale Grad bei der polynomialen Regression legt das Modell fest, die Stärke der Regularisierung bei der Ridge-Regression beeinflusst die Verlustfunktion (Bishop, 2006, Abschn. 1.3), und die Anzahl der erhobenen Datenpunkte bestimmt den Trainingsdatensatz.

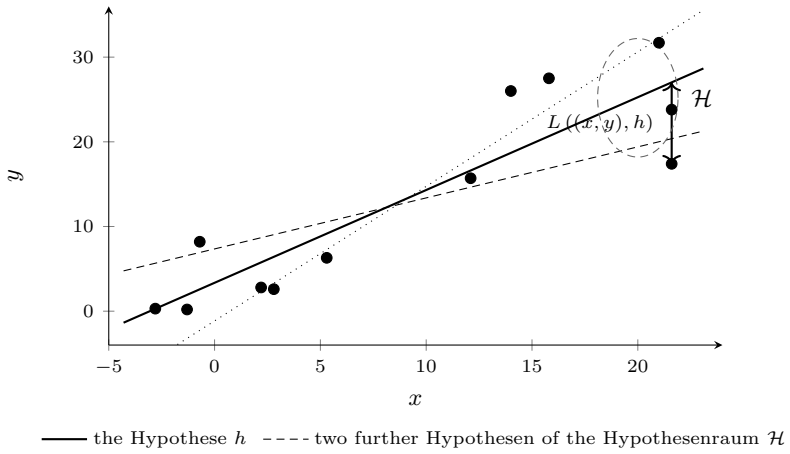

Abbildung 1: Zwölf Tage des Jahres 2020 in Krems an der Donau (gefüllte Kreise), mit dem Messwert eines Tages um 06:00 (UTC) als Merkmal x und dem Messwert um 15:00 (UTC) als Label y. Die drei Geraden stellen Hypothesen des Modells dar: Die durchgezogene Linie ist eine Hypothese h, die gestrichelte und die gepunktete Linie sind zwei weitere; die sie umschließende gestrichelte Schleife markiert den Hypothesenraum $\mathcal{H}$. Der vertikale Pfeil markiert den Fehler der Vorhersage eines einzelnen Datenpunkts unter h, gemessen durch die Verlustfunktion $L((x,y),h)$. Daten erzeugt mit `pythondemos/hyperparameter.py`

Die eingangs genannte dritte Entscheidung, wie das Modell durchsucht werden soll, wird durch Hyperparameter des Optimierungsverfahrens festgelegt, das die empirische Risikominimierung löst. Die Lernrate des Gradientenabstiegs begrenzt beispielsweise den Abstand zwischen der den Trainingsfehler minimierenden Iterierten des Gradientenabstiegs und ihrem Startpunkt nach einer festen Anzahl von Gradientenschritten, und eine größere Lernrate erhöht diese Schranke. Damit wächst die Menge der erreichbaren Modellparameter und mit ihnen der effektive Hypothesenraum. Abb. 2 veranschaulicht dies für fünf Gradientenschritte auf einem Trainingsdatensatz, ausgehend von der Nullinitialisierung $\mathbf{w}^{(0)} = (0,0)^\top$, bei drei Werten $\eta_1 < \eta_2 < \eta_3$ der Lernrate. Die effektiven Hypothesenräume $\mathcal{H}^{(\eta_1)} \subset \mathcal{H}^{(\eta_2)} \subset \mathcal{H}^{(\eta_3)}$ sind ineinander geschachtelt, sodass die Lernrate eine nach Größe geordnete Familie von Modellen indiziert, genauso wie es der maximale Grad bei der polynomialen Regression tut.

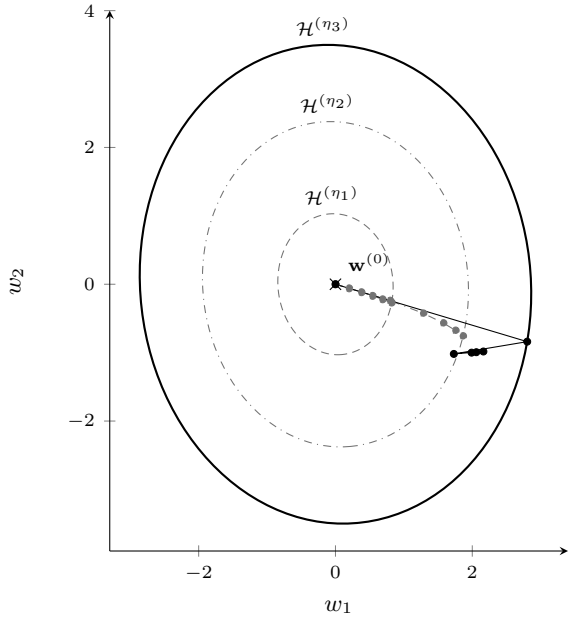

Abbildung 2: Gradientenabstieg auf einem Trainingsdatensatz aus 30 Datenpunkten mit zwei Merkmalen, ausgehend von der Nullinitialisierung $\mathbf{w}^{(0)} = (0, 0)^\top$ (Kreuz) und gestoppt nach fünf Gradientenschritten, für eine kleine, eine mittlere und eine große Lernrate, $\eta_1 = 0.1/L$, $\eta_2 = 0.4/L$ und $\eta_3 = 1.4/L$, wobei L die größte Krümmung des Trainingsfehlers bezeichnet. Die Punkte sind die Iterierten, gezeichnet im Stil ihrer Lernrate: gestrichelt für η_1, strichpunktiert für η_2, durchgezogen für η_3. Die Ellipse $\mathcal{H}^{(\eta_k)}$ verläuft durch die am weitesten von $\mathbf{w}^{(0)}$ entfernte Iterierte der Lernrate η_k und hat die Form der Niveaumengen des Trainingsfehlers; sie ist der effektive Hypothesenraum, den diese Lernrate zulässt, und enthält jeden damit ausgeführten Gradientenschritt. Die drei Ellipsen sind ineinander geschachtelt, und jede enthält sämtliche Iterierten jeder kleineren Lernrate. Daten erzeugt mit `pythondemos/hyperparameter.py`

Welche Komponente er auch immer festlegt: Ein Hyperparameter wählt eine Methode des maschinellen Lernens aus einer Familie von Alternativen aus, und genau das ermöglicht eine einheitliche Regel zu seiner Auswahl.

Die Eignung eines bestimmten Hyperparameterwerts lässt sich mithilfe von Techniken der Modellauswahl wie der Validierung beurteilen (Hastie et al., 2009, Abschn. 7.10). Wie beim Lernen der Modellparameter mittels empirischer Risikominimierung auf einem Trainingsdatensatz wird ein Hyperparameter durch Minimierung des Validierungsfehlers gewählt, wobei der Validierungsfehler an die Stelle des Trainingsfehlers tritt (Bishop, 2006, Abschn. 1.3). Die Suche erfolgt häufig über ein Gitter oder eine zufällig gezogene Menge von Kandidatenwerten, wobei derjenige mit dem kleinsten Validierungsfehler auf dem Validierungsdatensatz beibehalten wird.

Abb. 3 zeigt diese Auswahl für den Grad eines polynomialen Modells, das mittels polynomialer Regression aus 20 Datenpunkten gelernt wurde. Der Trainingsfehler sinkt mit jeder Erhöhung des Grades und würde den größten auswählen. Der Validierungsfehler auf 100 weiteren Datenpunkten ist bei Grad 3 am kleinsten und steigt darüber hinaus steil an, sodass die Validierung den Grad 3 auswählt.

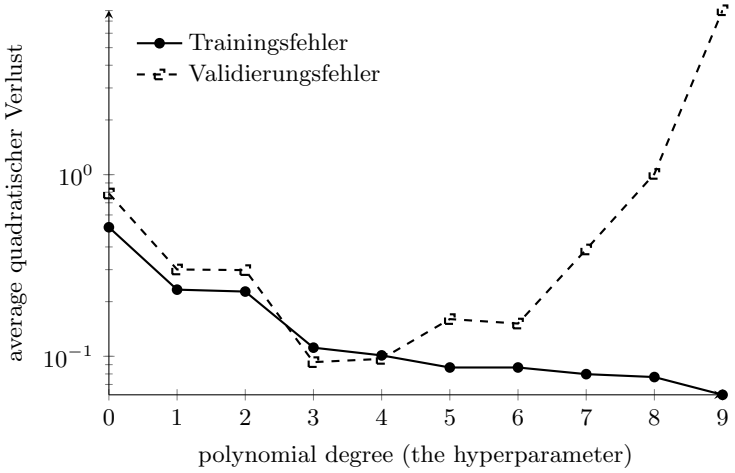

Abbildung 3: Trainingsfehler (durchgezogene Linie, gefüllte Kreise) und Validierungsfehler (gestrichelte Linie, offene Quadrate) eines polynomialen Modells in Abhängigkeit von seinem Grad, dem Hyperparameter. Der Trainingsfehler nimmt mit dem Grad niemals zu und würde Grad 9 auswählen; der Validierungsfehler ist bei Grad 3 am kleinsten, dem Wert, den die Validierung auswählt. Daten erzeugt mit `pythondemos/hyperparameter.py`

Siehe auch: Modell, Validierung, Modellparameter, Validierungsfehler, Trainingsfehler, Modellauswahl, Lernrate, Gradientenabstieg, polynomiale Regression.

Querverweise

Literatur

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

2. Hastie et al. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer Science+Business Media, New York, NY, USA.