Bagging

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

Bagging ist eine Ensemble-Methode, die zufällig ausgewählte Teilmengen eines Datensatz als Trainingsdatensätze für ihre Basislerner verwendet. Das Aggregieren der von den Basislerner gelieferten predictions, durch Mehrheitsentscheid bei der Klassifikation oder durch Mittelwertbildung bei der regression, ergibt die prediction der Ensemble-Methode. Ein Random Forest ist die Instanz von Bagging, deren Basislerner Entscheidungsbäume lernen.

Definition

An vierundzwanzig Tagen des Jahres 2020, zwölf aus dem Winter und zwölf aus dem Sommer, zeichnete die Wetterstation Krems an der Donau (Österreich) die Lufttemperatur um 06:00 und um 15:00 Uhr (UTC) auf. Die Vorhersage des 15:00-Uhr-Wertes y eines Tages aus seiner 06:00-Uhr-Temperatur x ist ein regression-Problem, und ein Entscheidungsbaum löst es, indem es die Merkmal-Achse an einem Schwellenwert teilt und den mittleren Label auf jeder Seite vorhersagt.

Abb. 1 zeigt, was passiert, wenn zwei Entscheidungsbäume jeweils aus fünf zufällig ausgewählten Tagen gelernt werden: Sie setzen ihren Schwellenwert bei 8.4 und 9.6 Grad Celsius, und ihre predictions unterscheiden sich um bis zu 22 Grad Celsius. Diese Streuung deutet auf einen Mangel an stability des Basislerner hin: Er ist zu empfindlich gegenüber der spezifischen Wahl des Trainingsdatensatz, was wiederum eine schlechte Generalisierung impliziert.

Bagging ist eine Ensemble-Technik, die diese Instabilität unterdrückt. Bagging führt seine Basislerner parallel aus, im Gegensatz zu Boosting, das seine Basislerner sequenziell ausführt. Der Name ist eine Abkürzung für Bootstrap Aggregating.

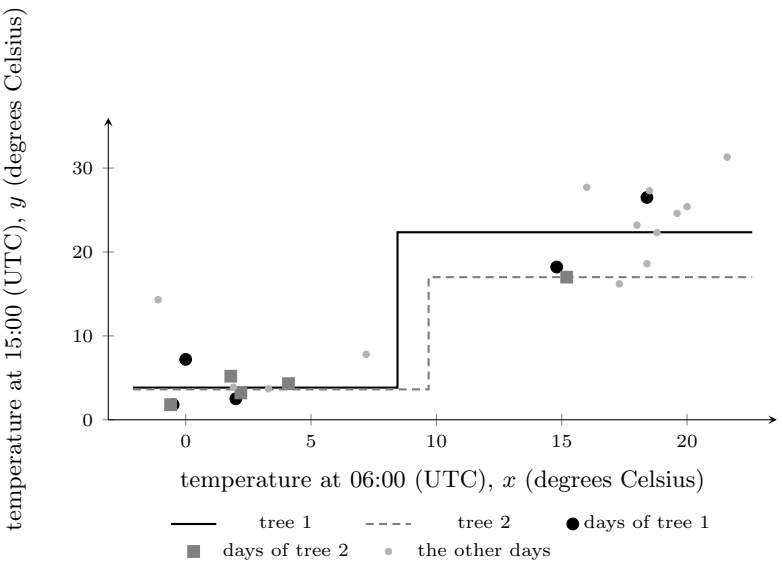


Abbildung 1: Vierundzwanzig Tage des Jahres 2020 in Krems an der Donau, zwölf aus dem Winter und zwölf aus dem Sommer, mit dem 06:00-Uhr-Wert eines Tages als Merkmal und seiner 15:00-Uhr-Temperatur als Label. Zwei Entscheidungsbäume der Tiefe eins, jeweils aus fünf zufällig ausgewählten Tagen gelernt, sind im Stil der Tage gezeichnet, auf denen sie trainiert wurden: die durchgezogene schwarze Stufenkurve mit den gefüllten Kreisen, die gestrichelte graue Stufenkurve mit den gefüllten Quadraten. Die verbleibenden Tage sind die kleinen grauen Punkte. Die beiden Bäume setzen ihren Schwellenwert bei unterschiedlichen Temperaturen und sagen auf beiden Seiten unterschiedliche Niveaus voraus. Daten generiert von `pythondemos/bagging.py`

Bagging verwendet zufällig ausgewählte Teilmengen eines Datensatz als Trainingsdatensätze für seine Basislerner. Gegeben das ursprüngliche Trainingsdatensatz $\mathcal{D}$, erzeugt es gestörte Kopien $\widetilde{\mathcal{D}}^{(1)}, \ldots, \widetilde{\mathcal{D}}^{(M)}$ (typischerweise mittels Bootstrap-Sampling) und trainiert einen Basislerner auf jeder, wodurch Hypothesen $\widehat{h}^{(1)}, \ldots, \widehat{h}^{(M)}$ erzeugt werden. Die finale Hypothese aggregiert diese durch Mehrheitsentscheid für die Klassifikation (d.h. die von den meisten Basislerner vorhergesagte Klasse) oder durch Mittelwertbildung für die regression, eine Aufgabe, deren label space $\mathbb{R}$ ist (siehe Abb. 2). Algorithmus 1 beschreibt die Methode vollständig. Bagging-Methoden unterscheiden sich in ihrer Wahl des Basislerner. Ein Random Forest ist zum Beispiel die Instanz von Bagging, deren Basislerner Entscheidungsbäume lernen.

Algorithm 1 Bagging (Breiman, 1996, Sect. 1)

Input: Trainingsdatensatz $\mathcal{D} = \left\{\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(m)}\right\}$, Basislerner, number M of Basislerner

1: **for** $\ell = 1, \ldots, M$ **do**

2: draw the bootstrap sample $\widetilde{\mathcal{D}}^{(\ell)}$: m Datenpunkte from $\mathcal{D}$, with replacement

3: learn $\widehat{h}^{(\ell)}$ by applying the Basislerner to $\widetilde{\mathcal{D}}^{(\ell)}$

4: **end for**

Output: $\widehat{h}(\mathbf{x}) = \frac{1}{M}\sum_{\ell=1}^{M} \widehat{h}^{(\ell)}(\mathbf{x})$ for regression; the majority vote of $\widehat{h}^{(1)}(\mathbf{x}), \ldots, \widehat{h}^{(M)}(\mathbf{x})$ for Klassifikation

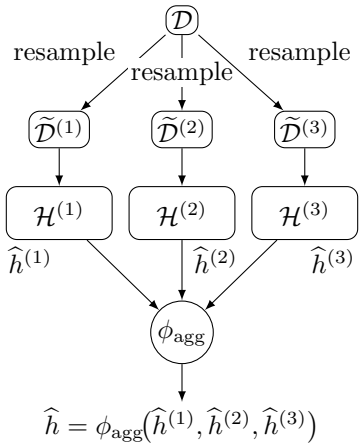


Abbildung 2: Beispiel für Bagging, bei dem drei Basislerner Störungen $\widetilde{\mathcal{D}}^{(1)}, \ldots, \widetilde{\mathcal{D}}^{(3)}$ des ursprünglichen Trainingsdatensatz $\mathcal{D}$ verwenden, um die Hypothesen $\widehat{h}^{(1)}, \ldots, \widehat{h}^{(3)}$ zu lernen. Die finale Hypothese wird durch Aggregation dieser einzelnen Hypothesen mittels einer Aggregationsregel ϕ_{agg} erhalten. Eine Aggregationsregel für regression-Probleme ist der Durchschnitt der einzelnen predictions

Bagging reduziert die Varianz der für die Basislerner verwendeten machine learning (ML)-Methode (Hastie et al., 2009, Sect. 8.7): Jeder Basislerner wird auf einer anderen Resample trainiert, sodass ihre prediction-Fehler nur teilweise korreliert sind. In einem idealisierten Bagging-Szenario mit M Basislerner, die unkorrelierte predictions liefern, jede mit Varianz σ^2, hat die gemittelte prediction eine Varianz von σ^2/M. Im Allgemeinen sind die predictions korreliert. Für M identisch verteilte predictions, jede mit Varianz σ^2 und mit paarweiser Korrelation ρ, hat die gemittelte prediction eine Varianz von

$$\rho\sigma^2 + \frac{1-\rho}{M}\,\sigma^2.$$

Der zweite Term verschwindet, wenn M wächst, während der erste nicht verschwindet: Die Korrelation zwischen den Basislerner setzt eine Untergrenze $\rho\sigma^2$, die keine Anzahl von Basislerner unterschreiten kann (Hastie et al., 2009, Gl. (15.1)). Abb. 3 zeigt, in einem kleinen numerischen Experiment, die geschätzte Varianz des Durchschnitts von M gebaggten Entscheidungsbäume im Vergleich zu dem Wert, den diese Identität aus einem geschätzten σ^2 und ρ vorhersagt.

Die Korrelation ρ stammt aus dem gemeinsamen Trainingsdatensatz. Die predictions der Basislerner wären nur dann unabhängig, wenn das Trainingsdatensatz fest und gegeben wäre, aber es ist selbst als Zufallsvariable (ZV) modelliert, und jeder Basislerner wird auf einer Resample derselben m Datenpunkte trainiert: Eine Resample enthält jeden von ihnen mit einer probability von etwa $1 - 1/e \approx 63.2\%$ (Hastie et al., 2009, Gl. (7.55)), zwei Resamples teilen sich daher etwa $(1 - 1/e)^2 \approx 40\%$, und alle teilen sich dieselbe empirical distribution.

Bagging reduziert nur die Varianz einer ML-Methode, nicht ihren bias. Für die regression-Einstellung haben der Durchschnitt und jedes seiner Mitglieder denselben expectation, sodass die Mittelwertbildung den bias eines Mitglieds unverändert lässt.

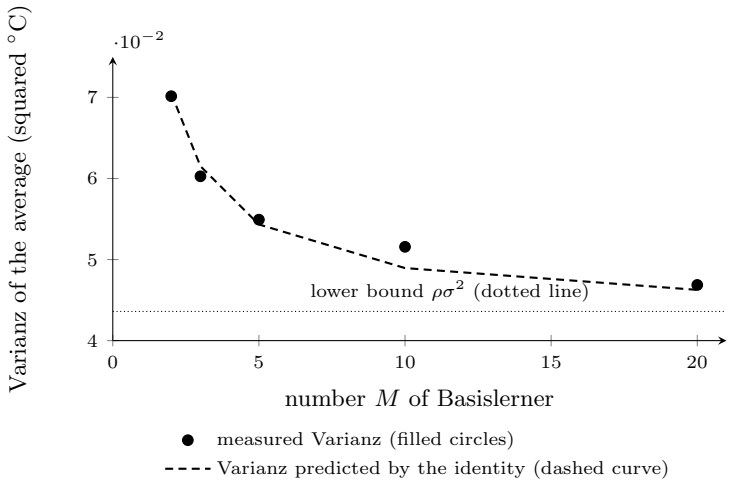


Abbildung 3: Die Varianz des Durchschnitts von M gebaggten Entscheidungsbäume (gefüllte Kreise), im Vergleich zu dem Wert, den die Identität aus einem σ^2 und einem ρ vorhersagt, die auf separaten Trainingsdatensätze geschätzt wurden (gestrichelte Kurve). Die beiden stimmen bei jedem M innerhalb von 5% überein, und beide nähern sich der Untergrenze $\rho\sigma^2 = 0.0436$ (gepunktete Linie) an, anstatt Null. Daten generiert von `pythondemos/bagging.py`

Die Mittelwertbildung führt auch zu stability: Die gelernte Hypothese hängt

nur schwach von einem einzelnen Datenpunkt des Trainingsdatensatz ab. Betrachten Sie eine beliebige ML-Methode mit predictions in $[0, 1]$, deren Basislerner jeweils auf einem Anteil p der m Datenpunkte trainiert werden, die ohne Zurücklegen gezogen werden, und deren predictions über alle solchen Teilstichproben gemittelt werden. Dann ändert das Entfernen eines einzelnen Datenpunkt aus dem Trainingsdatensatz die prediction an einem Testpunkt um mehr als ε für höchstens einen δ-Anteil der entfernten Datenpunkte, für jedes Paar (ε, δ) mit $\delta\varepsilon^2 \geq \frac{1}{4(m-1)} \cdot \frac{p}{1-p}$ (Soloff et al., 2024, Thm. 8). Diese Garantie gilt für jedes Trainingsdatensatz und erfordert nichts von den Basislerner außer den begrenzten predictions — die stability wird durch die Mittelwertbildung selbst erzeugt.

Ein Random Forest wendet Bagging auf Entscheidungsbäume an und lässt zusätzlich jede Aufteilung nur aus einer kleinen zufälligen Teilmenge der Merkmale wählen. Die zufällige Auswahl reduziert die Korrelation zwischen den gelernten Entscheidungsbäume, und eine geringere Korrelation bedeutet eine kleinere Varianz ihres Durchschnitts (Breiman, 2001; Hastie et al., 2009, Gl. (15.1)).

Siehe auch: Ensemble, Bootstrap, Random Forest, Basislerner, Varianz, Boosting, Entscheidungsbaum, Korrelation, stability.

Querverweise

Literatur

1. Breiman (1996). Bagging predictors. Mach. Learn., 24(2), pp. 123–140.
2. Breiman (2001). Random Forests. Machine Learning, 45(1), pp. 5–32.
3. Soloff et al. (2024). Bagging Provides Assumption-free Stability. J. Mach. Learn. Res., 25(131), pp. 1–35.
4. Hastie et al. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer Science+Business Media, New York, NY, USA.