KI für Alle 2: Verstehen, Bewerten, Reflektieren

Themenblock Prognosemodelle (Klassifikation und Regression): 02_03Verfahren_ BäumeWälder_02

Der random forest

Erarbeitet von
Dr. Katja Theune

Lernziele

- Du kannst die Konzepte bagging und random feature subspacing erläutern
- Du kannst anhand eines einfachen Beispiels den prognostizierten Output für eine neue Beobachtung bestimmen
- Du kannst die Problematik bei der Wahl der Hyperparameter erläutern
- Du kannst Vor- und Nachteile eines random forests erläutern

Inhalt

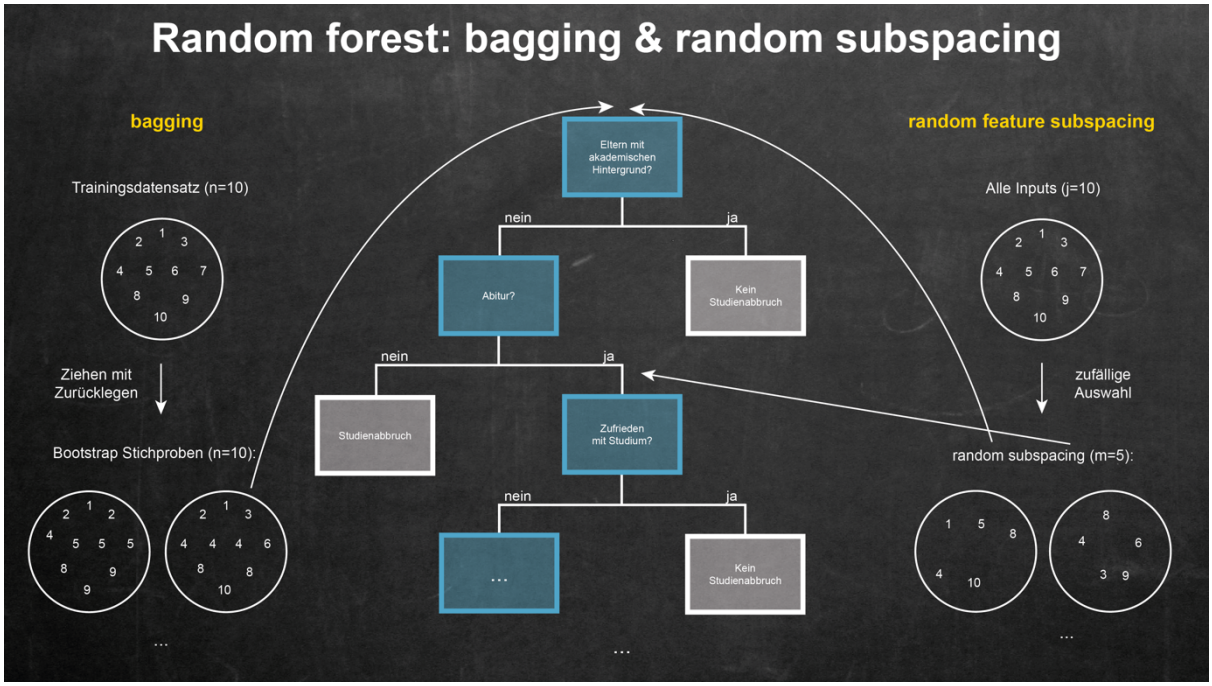
Einstieg

Das Verfahren eines einzelnen Baumes ist sehr einfach nachzuvollziehen, hat aber einige Nachteile. Daher ist ein beliebtes Verfahren der Entscheidungswald. Wie der Name schon suggeriert, besteht dieser aus mehreren einzelnen Bäumen, welche sozusagen zu einem Wald kombiniert werden.

Random forests: Grundlegende Konzepte

Es gibt wie immer verschiedene Herangehensweisen, mehrere Bäume zu einem Wald zu kombinieren. Stellvertretend möchte ich den sogenannten random forest erläutern.
Die einzelnen Bäume bleiben hier ungestutzt, sind also sehr tief und haben viele Knoten.

Quelle [1,2]

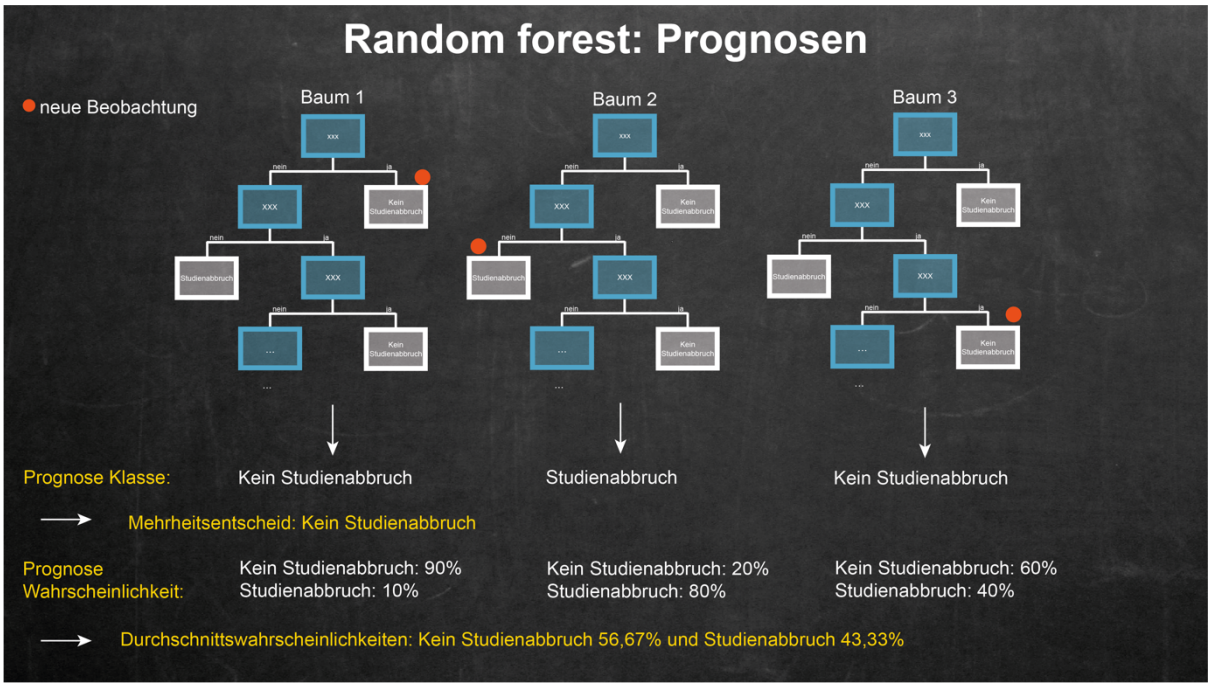


Die einzelnen Bäume werden mit der Methode des bagging und des random feature subspacing gebildet. Bagging steht für bootstrap aggregation und bedeutet, dass jeder Baum auf einem eigenen speziellen Trainingsdatensatz, den sogenannten bootstrap Stichproben, basiert. Diese verschiedenen bootstrap Stichproben werden durch Ziehen mit Zurücklegen von Beobachtungen aus dem gesamten originalen Trainingsdatensatz gewonnen. D. h. jede Beobachtung kann mehrmals in der bootstrap Stichprobe auftauchen. Alle bootstrap Stichproben beinhalten genauso viele Beobachtungen wie der originale Trainingsdatensatz.

Quelle [3]

Random feature subspacing bedeutet, dass an jedem Knoten nicht alle im originalen Trainingsdatensatz beinhalteten Inputs verwendet und auf ihre Eignung zur Aufteilung überprüft werden, sondern nur eine zufällige Auswahl m von ihnen. Das führt dann zu sehr unterschiedlichen Bäumen und einer noch geringeren Varianz.

Random forests: Prognose



Jeder Baum eines random forests, hier sind es jetzt drei, gibt eine eigene Klassenprognose für die neue Beobachtung ab. Um zu einer Gesamtprognose für diese Beobachtung zu kommen, können wir für eine konkrete Klassenzugehörigkeit einen Mehrheitsentscheid heranziehen. Hier würden wir uns dann z. B. für die Klasse „kein Studienabbruch" entscheiden, da sie am häufigsten vorkommt.

Jeder Baum kann aber auch eine Prognose der Klassenwahrscheinlichkeiten abgeben. Für eine Gesamtprognose bilden wir darüber einfach den Durchschnitt. Hier würden wir für unsere neue Beobachtung eine Wahrscheinlichkeit zur Klasse „kein Studienabbruch" zu gehören von 56,67 % und zur Klasse „Studienabbruch" zu gehören von 43,33 % prognostizieren.

Bei einer Regression würde man das Mittel über alle von den Bäumen prognostizierten Werte bilden.

Diese Idee, mehrere einfache Klassifikatoren zu aggregieren, nennt man Ensemble-Verfahren und kann auch auf andere Verfahren des maschinellen Lernens übertragen werden, um ihre Prognosegenauigkeit zu verbessern.

Bedeutung der Inputs für die Prognose

In vielen Anwendungsfällen wollen wir neben der Prognosegenauigkeit wissen, welche unserer Inputs besonders wichtig für unsere Prognose waren. Denn nicht alle Inputs üben den gleichen Einfluss auf unser Endergebnis aus. Random forests können uns diese Wichtigkeit der einzelnen Inputs liefern. Dafür können wir z. B. einfach schauen, inwieweit ein Input bei den einzelnen Bäumen zur Reduktion der Verunreinigung in den Knoten geführt hat, also z. B. den Gini Index reduziert hat. Über alle Bäume kann man dann diese Werte mitteln. Hohe Werte der Reduktion deuten dann auf besonders bedeutende Inputs für unsere Prognose hin.

Die Wahl von Hyperparametern

Wie bei allen Verfahren des maschinellen Lernens ist auch bei Bäumen und Wäldern die Wahl der Hyperparameter wichtig, vor allem wenn wir uns auch an den Trade-off von Verzerrung und Varianz erinnern. Zu den Hyperparametern gehört u. a. die Baumtiefe, also die Anzahl an Knoten. Ein Baum mit vielen Knoten ist sehr auf die Trainingsdaten angepasst und führt zu einer hohen Varianz. Ein Baum mit wenigen Knoten führt dagegen häufig zu einer höheren Verzerrung.
Kombiniert man wie beim random forest aber mehrere ungestutzte, tiefe Bäume, reduziert sich dadurch die Varianz, die geringe Verzerrung bleibt aber bestehen.

Aber auch die Anzahl an Bäumen, die wir in einem Wald berücksichtigen wollen oder die Anzahl an Inputs, die wir beim random feature subspacing an jedem Knoten verwenden, sind Hyperparameter. Gerade letzteres hat einen großen Einfluss. Denn je weniger Inputs wir verwenden, desto verschiedener sind die Bäume und desto kleiner ist die Varianz. Das geht aber häufig auf Kosten einer höheren Verzerrung.

Diskussion, Vor- und Nachteile

Die Nachteile einzelner Bäume, wie z. B. die geringere Prognosegenauigkeit, können durch die Kombination vieler ungestutzter Bäume zu einem random forest deutlich verbessert werden.
Das im Vergleich zu anderen Verfahren einfache Datenhandling bei Bäumen und die Möglichkeit, komplexe Zusammenhänge zwischen den Inputs und Outputs abzubilden, bleibt auch bei den random forests bestehen.

Ein Nachteil ist natürlich, dass die Verwendung vieler Bäume die Interpretierbarkeit im Vergleich zu einem einzelnen Baum verringert und auch viel Rechenpower benötigt.

Abschluss

Wir haben nun den random forest als eine Variante, Bäume zu einem Entscheidungswald zu kombinieren, kennengelernt. Neben der Prognosegenauigkeit ist ein wichtiger Vorteil, dass der random forest uns Informationen über die Bedeutung von Inputs für die Prognose

liefern kann und trotz vieler Bäume meist besser nachzuvollziehen ist als andere Verfahren des maschinellen Lernens.

Quellen

Quelle [1] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5-32.

Quelle [2] https://www.stat.berkeley.edu/~breiman/RandomForests/

Quelle [3] Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2), 123-140.

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