

-Machine Learning: Boosting-Algorithmen-

Seminararbeit

Student:	David Erdös	67906
Universität:	Hochschule Karlsruhe – Technik und Wirtschaft	
Studiengang:	Informatik Bachelor	
Semester:	Wintersemester 2023	
Dozent:	Prof. Dr. Baier	
Bearbeitet am:	1. Dezember 2023	

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation und Zielsetzung	1
1.2	Struktur der Arbeit	1
2	Grundlagen des Machine Learning (3-4 Seiten)	2
2.1	Modern Approaches in Machine Learning	2
2.2	Role of Boosting Algorithms in ML	2
2.3	Boosting Algorithms in Tabular Data Analysis	2
3	Boosting	3
3.1	Was ist Boosting am Beispiel Wettererkennung	3
3.1.1	Erzeugung eines starken Lernalgorithmus	4
3.2	Gewichtete Abstimmung	4
3.2.1	Grundlegende Annahmen	4
3.2.2	Fehlerberechnung	4
3.2.3	Gewichtung der schwachen Lernalgorithmen	5
3.2.4	Endgültige Vorhersage des starken Lernalgorithmus	5
3.3	Annahme des schwachen Lernalgorithmus	5
4	AdaBoost (5-6 Seiten)	7
4.1	Algorithmus-Struktur und Funktionsweise	7
4.2	Beispielanwendung mit Erläuterung	7
5	Gradient Boosting (5-6 Seiten)	8
5.1	Theoretische Grundlagen	8
5.2	Algorithmus-Struktur und Funktionsweise	8
5.3	Beispielanwendung mit Erläuterung	8
6	Vergleich von AdaBoost und Gradient Boosting (4-5 Seiten)	9
6.1	Gemeinsamkeiten und Unterschiede	9
6.2	Performance-Analyse in Benchmarks	9
6.3	Anwendungsbeispiele und Fallstudien	9
7	Aktuelle Trends und Entwicklungen (2-3 Seiten)	10
7.1	Neueste Forschungsergebnisse	10
7.2	Zukünftige Potenziale von Boosting-Algorithmen	10
8	Fazit und Ausblick (2-3 Seiten)	11
8.1	Zusammenfassung der Erkenntnisse	11
8.2	Reflexion über die Bedeutung für die Praxis	11
8.3	Ausblick auf zukünftige Forschungsthemen	11

Literaturverzeichnis	13
Abbildungsverzeichnis	14
Tabellenverzeichnis	15

1 Einleitung

In der dynamischen Welt des maschinellen Lernens (ML) haben sich Boosting-Algorithmen als revolutionär erwiesen, indem sie beeindruckende Leistungen in einer Vielzahl von Anwendungen erbringen [1, Kapitel 1.1.2]. Besonders bei tabellarischen Datensätzen erzielen sie in fast allen Bereichen die besten Ergebnisse.

In dieser Seminararbeit liegt der Schwerpunkt auf AdaBoost und GradientBoosting, die gemäß Géron [2, S. 192] zu den bekanntesten Vertretern der Boosting-Familie gehören. Die Arbeit wird die beiden Algorithmen detailliert untersuchen und verständlich darstellen.

1.1 Motivation und Zielsetzung

Das Ziel dieser Seminararbeit ist es, ein tiefes Verständnis für die Funktionsweise von AdaBoost und GradientBoosting zu entwickeln und deren Einsatzmöglichkeiten anhand konkreter Beispiele zu demonstrieren. Hierbei wird besonderer Wert darauf gelegt, zunächst die Funktionsweise verständlich zu erläutern.

1.2 Struktur der Arbeit

Die Seminararbeit gliedert sich in mehrere Abschnitte, die schrittweise aufeinander aufbauen. Zunächst wird ein Überblick über die Grundlagen des maschinellen Lernens gegeben, gefolgt von einer detaillierten Betrachtung der Boosting-Algorithmen, insbesondere AdaBoost und GradientBoosting. Anschließend erfolgt ein Vergleich dieser beiden Methoden, wobei deren Stärken und Schwächen in verschiedenen Anwendungsszenarien beleuchtet werden. Abschließend wird ein Ausblick auf zukünftige Entwicklungen und mögliche Forschungsrichtungen im Bereich des Boostings gegeben.

[3, text] [1, text] [2, text] [4, text] [5, text]

2 Grundlagen des Machine Learning (3-4 Seiten)

2.1 Modern Approaches in Machine Learning

Überblick über aktuelle Trends und Innovationen im Machine Learning Vorstellung fortgeschrittener Techniken und Methoden Diskussion über die Bedeutung von Deep Learning und künstlichen neuronalen Netzen

2.2 Role of Boosting Algorithms in ML

Einführung in Boosting-Algorithmen und ihre Relevanz Spezifische Betrachtung von AdaBoost und Gradient Boosting Vergleich von Boosting-Algorithmen mit anderen fortgeschrittenen Methoden

2.3 Boosting Algorithms in Tabular Data Analysis

Bedeutung von tabellenartigen Datensätzen in fortgeschrittenen ML-Anwendungen Analyse, wie AdaBoost und Gradient Boosting bei tabellenartigen Daten effektiv sind Fallstudien und Beispiele aus der Praxis, die den Einsatz dieser Algorithmen zeigen

3 Boosting

Nach meiner eigenen Erfahrung als Student fällt oft auf, dass Gruppenarbeit nicht immer die erwarteten Vorteile bringt. Oft dominiert in Lerngruppen ein einzelner Student, der über fundiertes Wissen verfügt, die Diskussion und die Gruppenleistung spiegelt im Wesentlichen seine individuelle Leistung wider.

In Fällen, wo alle Mitglieder einer Lerngruppe nur begrenztes Wissen zu einem Thema haben, kann die kollektive Leistung sogar hinter dem zurückbleiben, was man durch zufällige Antworten erwarten würde. Die Vorstellung, dass eine Gruppe von Individuen mit begrenztem Wissen gemeinsam Ergebnisse erzielen kann, die sowohl den Durchschnitt als auch jede individuelle Bestleistung übertreffen, widerspricht oft unserer Intuition. Selbst historische Weisheiten, wie sie in der Bibel gefunden werden, betonen die Risiken einer solchen Zusammenarbeit. Dort heißt es metaphorisch: „Wenn aber ein Blinder den andern führt, so fallen sie beide in die Grube“ (Mt 23,16; Mt 23,24; Lk 6,39; Röm 2,19).

Doch im Bereich des maschinellen Lernens offenbart sich ein ganz anderes Szenario. Hier ermöglicht das Boosting-Verfahren, dass die Kombination von schwachen Modellen zu einem leistungsstarken Gesamtsystem führt. Dieser Ansatz, der die aggregierte Intelligenz mehrerer einfacher Modelle nutzt, um komplexe Probleme zu lösen, steht im starken Gegensatz zu den oft enttäuschenden Ergebnissen menschlicher Gruppenarbeit mit begrenztem Wissen [1, S. 3].

3.1 Was ist Boosting am Beispiel Wettererkennung

Boosting (ursprünglich Hypothesis Boosting) bezeichnet eine beliebige Ensemble-Methode, bei der sich mehrere schwache Lerner zu einem starken Lerner kombinieren lassen Géron [2, S. 191].

Die genannte Definition des Boosting lässt sich gut anhand des Beispiels der Wettererkennung veranschaulichen. Betrachten wir folgende einfache Regeln (Schwache Lerner) zur Beurteilung, ob es regnet:

Schwache Lerner	Grenzwert
Nasser Boden	Ja
Wolken am Himmel	Ja
Hohe Luftfeuchtigkeit	> 80%
Personen mit Regenschirm	Ja
Außentemperatur	> 0°C

Tabelle 1: Individuelle Vorhersagen der Schwache Lerner

Beispielsweise ist ein nasser Boden zwar eine Voraussetzung und ein guter erster Filter, allerdings könnte der Boden genauso gut durch einen Rasensprenger nass sein. Die Temperatur ist hingegen ein relativ schlechtes Indiz für die Frage, ob es gerade regnet. Es unterscheidet aber den Fall Regen und Schnee und ist somit trotzdem essentiell für die Klassifikation.

3.1.1 Erzeugung eines starken Lernalers

Der nächste Schritt ist es, die Aussagen der schwachen Lerner in ein nützliches Modell zusammenzufassen, einen sog. ‘starken Lerner’. Im einfachsten Fall lässt man die schwachen Lerner mit gleicher Stimmkraft abstimmen. Aus dem Stimmverhältnis der Vorhersagen der schwachen Lerner lässt sich in unserem Fall die Regenwahrscheinlichkeit ableiten.

Nasser Boden	Wolken	Hohe Luftfeuchte	Regenschirm	Über 0°C	Regenwahrscheinlichkeit (%)
Ja	Ja	Ja	Ja	Ja	100%
Ja	Ja	Nein	Nein	Ja	60%
Nein	Ja	Ja	Ja	Ja	80%
Ja	Nein	Ja	Nein	Ja	60%
Nein	Nein	Nein	Nein	Ja	20%
Ja	Ja	Ja	Nein	Nein	60%
Nein	Ja	Nein	Ja	Ja	80%
...

Tabelle 2: Wahrheitstabelle zur Vorhersage von Regen basierend auf schwachen Lernalern. Die Werte dienen in erster Linie der Veranschaulichung.

3.2 Gewichtete Abstimmung

Aus der Mehrheitsentscheidung, bei der alle schwachen Lerner die gleiche Stimmkraft haben, eine Vorhersage zu treffen, liefert nur selten die besten Ergebnisse. Eine bessere Vorgehensweise, die auch von vielen Boosting-Algorithmen benutzt wird, ist die gewichtete Abstimmung. Diese Methode berechnet die Stimmkraft oder Gewichtung des Lernalers α individuell, wie im Algorithmus 1.1 von Schapire und Freund beschrieben [1, S. 5].

3.2.1 Grundlegende Annahmen

Gegeben:

- Ein Datensatz $X = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$.
- Schwache Lerner t mit Vorhersagefunktion/Hypothese h_t .

Jeder schwache Lerner trifft Vorhersagen basierend auf dem Datensatz X . Der Erfolg dieser Vorhersagen wird durch den Fehler ε_t gemessen.

3.2.2 Fehlerberechnung

Der Fehler ε_t eines schwachen Lernalers ist der Anteil der falschen Vorhersagen:

$$\varepsilon_t = \frac{1}{n} \sum_{i=1}^n \mathbb{I}(h_t(x_i) \neq y_i), \quad (1)$$

wobei \mathbb{I} die Indikatorfunktion ist, die 1 ist, wenn $h_t(x_i) \neq y_i$ (d.h., die Vorhersage ist falsch), und 0 sonst.

3.2.3 Gewichtung der schwachen Lerner

Die Gewichtung α_t jedes schwachen Lerner basiert auf seinem Fehler ε_t :

$$\alpha_t = \ln \left(\frac{1 - \varepsilon_t}{\varepsilon_t} \right). \quad (2)$$

Ein zuverlässiger Lerner t hat eine höhere Gewichtung α_t je nach dem wie groß sein Fehler ε_t ist.

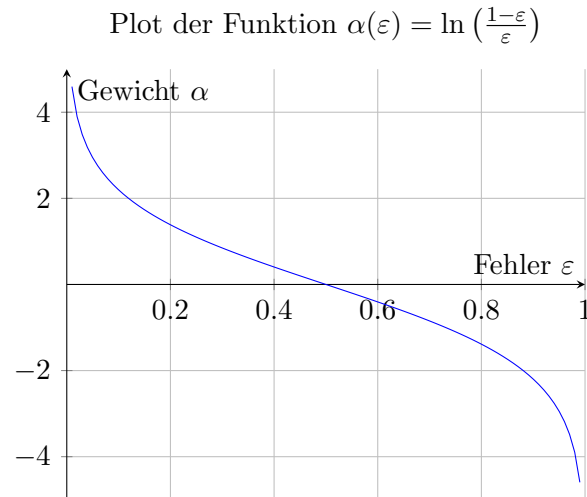


Abbildung 1: Visualisierung der Gewichtungsfunktion α in Abhängigkeit vom Fehler ε

3.2.4 Endgültige Vorhersage des starken Lerner

Die Gesamtvorhersage H des Boosting-Modells ergibt sich aus der gewichteten Abstimmung aller schwachen Lerner.

$$H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x) \right). \quad (3)$$

In anderen Worten wird die Vorhersage eines Lerner auf den Daten $h_t(x)$ mit seiner Gewichtung α multipliziert. Die Summe dieses Produkts für alle Lerner T ergibt die Gesamtvorhersage H . Hier wird lediglich noch die Signum-Funktion angewendet, um den Wert zwischen 0 und 1 zu glätten.

3.3 Annahme des schwachen Lernens

Die Voraussetzung jedes Boosting-Algorithmus ist ein bereits vorhandener schwacher Lernalgorithmus. Das Ziel ist es dann, durch mehrfache Ausführung des Boosting-Algorithmus' die Leistung des Lernalgorithmus zu verbessern, bzw. zu 'boosten'. Laut Schapire und Freund [1, S. S. 4] reicht es aus geringe Ansprüche an die Leistung der schwachen Lerner zu stellen. Es ist völlig hinreichend, wenn der Lernalgorithmus Hypothesen liefert mit etwas besseren Ergebnissen als 50% Fehlerquote, was dem uninformierten Raten gleich käme.

Diese Annahme, dass der Lerner schwache Hypothesen hervorbringt, die mindestens etwas besser

sind als zufälliges Raten, wird die *Annahme des schwachen Lernens* genannt.

4 AdaBoost (5-6 Seiten)

4.1 Algorithmus-Struktur und Funktionsweise

4.2 Beispielanwendung mit Erläuterung

5 Gradient Boosting (5-6 Seiten)

5.1 Theoretische Grundlagen

5.2 Algorithmus-Struktur und Funktionsweise

5.3 Beispielanwendung mit Erläuterung

6 Vergleich von AdaBoost und Gradient Boosting (4-5 Seiten)

6.1 Gemeinsamkeiten und Unterschiede

6.2 Performance-Analyse in Benchmarks

6.3 Anwendungsbeispiele und Fallstudien

!!! REFERENZ ZU 1. EINLEITUNG

7 Aktuelle Trends und Entwicklungen (2-3 Seiten)

7.1 Neueste Forschungsergebnisse

7.2 Zukünftige Potenziale von Boosting-Algorithmen

8 Fazit und Ausblick (2-3 Seiten)

8.1 Zusammenfassung der Erkenntnisse

8.2 Reflexion über die Bedeutung für die Praxis

8.3 Ausblick auf zukünftige Forschungsthemen

Glossar

AdaBoost Ein Machine Learning-Algorithmus, der auf dem Prinzip des Boosting basiert.. 1, 2

Gradient Boosting Eine Methode des maschinellen Lernens, die für Regression und Klassifikation verwendet wird.. 1, 2

Literaturverzeichnis

- [1] R. E. Schapire und Y. Freund, *Boosting : foundations and algorithms* (Adaptive Computation and Machine Learning Series). Cambridge, Mass. [u.a.]: MIT Press, 2012, Includes bibliographical references and index, ISBN: 9780262017183. Adresse: <http://www.gbv.de/dms/ilmenau/toc/672276232.PDF%20;%20https://zbmath.org/?q=an:1278.68021>.
- [2] A. [Géron, *Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow : Konzepte, Tools und Techniken für intelligente Systeme*, 1. Auflage, K. [Rother, Hrsg. Heidelberg: O'Reilly, 2018, Äthorized German translation of the English edition of 'Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems', ISBN 978-1-491-96228-9, (c) 2017 Impressum; ÜS-Bestseller zu Deep Learning Umschlag"Dieses Buch erscheint in Kooperation mit O'Reilly Media, Inc. unter dem Imprint Ö'Reilly". Rückseite der Titelseite. - "Deutschsprachige O'Reilly-Bücher werden vom dpunkt.verlag in Heidelberg publiziert, vermarktet und vertrieben.- Webseite www.oreilly.de; 201712; aa, ISBN: 9783960090618. Adresse: http://www.gbv.de/dms/ilmenau/toc/898831717.PDF%20;%20http://vub.de/cover/data/isbn%3A9783960090618/medium/true/de/vub/cover.jpg%20;%20http://deposit.d-nb.de/cgi-bin/dokserv?id=6ba05f7dfc5242d6bbc6f653e199e877&prov=M&dok_var=1&dok_ext=htm%20;%20https://www.oreilly.de/buecher/13111/9783960090618-praxiseinstieg-machine-learning-mit-scikit-learn-und-tensorflow.html.
- [3] J. [Frochte, *Maschinelles Lernen : Grundlagen und Algorithmen in Python* (Hanser eLibrary), 3., überarbeitete und erweiterte Auflage. München: Hanser, 2020, ISBN: 9783446463554. Adresse: <https://dx.doi.org/10.3139/9783446463554%20;%20https://doi.org/10.3139/9783446463554%20;%20https://www.hanser-elibrary.com/doi/book/10.3139/9783446463554%20;%20http://dx.doi.org/10.3139/9783446463554>.
- [4] G. [James, *An introduction to statistical learning : with applications in Python* (Springer texts in statistics), D. [Witten, T. [Hastie, R. [Tibshirani und J. E. [Taylor, Hrsg. Cham, Switzerland: Springer, 2023, ISBN: 9783031387463.
- [5] T. [Hastie, *The elements of statistical learning : data mining, inference, and prediction* (Springer series in statistics), Second edition, R. [Tibshirani und J. H. [Friedman, Hrsg. New York, NY: Springer, 2009, Hier auch später erschienene, unveränderte Nachdrucke; Literaturverzeichnis: Seite 699-727, ISBN: 9780387848570. Adresse: <http://www.gbv.de/dms/ilmenau/toc/572093853.PDF%20;%20https://zbmath.org/?q=an:1273.62005%20;%20https://swbplus.bsz-bw.de/bsz287727726kap.htm%20;%20https://swbplus.bsz-bw.de/bsz287727726vor.htm%20;%20https://swbplus.bsz-bw.de/bsz287727726inh.htm%20;%20https://swbplus.bsz-bw.de/bsz287727726cov.jpg>.

Abbildungsverzeichnis

1	Visualisierung der Gewichtsfunktion α in Abhängigkeit vom Fehler ε	5
---	---	---

Tabellenverzeichnis

1	Individuelle Vorhersagen der Schwache Lerner	3
2	Wahrheitstabelle zur Vorhersage von Regen basierend auf schwachen Lernern. Die Werte dienen in erster Linie der Veranschaulichung.	4