Implemetierung eines Random Forest-Ensembles zur Identifikation der Klassifizierungsunsicherheit auf CIFAR-10 Bilddaten

Lennart Fuhrig (Matrikelnummer: 201726546)
Schriftliche Ausarbeitung im Kurs Machine Learning I *im Wintersemester 2022 / 23*
(Dated: February 3, 2023)

Die vorliegende Arbeit resümiert die Implementierung eines probabilistischen Estimators [1], der Vorhersageunsicherheiten evaluiert. Hierfür wird der Bagging-Ansatz mit einem Random Forest Basis Klassifizierer auf den CIFAR-10 Datensatz angewendet. Die Unsicherheit des Ensembles wird dabei über die relative Häufigkeit der gewählten Klasse ermittelt. Zusammenfassend werden die experimentellen Ergebnisse graphisch aufbereitet, präsentiert und abschließend diskutiert.

MOTIVATION UND EINLEITUNG

Heutige Maschine Learning (ML) Algorithmen sind in der Lage Objekte in Bilddateien mit hoher Genauigkeit zu klassifizieren. In der Regel wird dabei der sogenannte *precision score* für die Bewertung und den Vergleich der Klassifizierungsmodelle herangezogen. Ob eine gemachte Vorhersage zuverlässig ist oder nicht, wird bei dieser Betrachtung jedoch gänzlich vernachlässigt. Beim Einzug der Künstlichen Intelligenz (KI) in cyber-physische Systeme kann dieses schwerwiegende Folgen hervorrufen, da getroffene Entscheidungen oftmals sicherheitskritische Anwendungen beeinflussen und nur bei ausreichender Vorhersagesicherheit umgesetzt werden sollten. Die probabilistische Klassifikation verfolgt diesen Ansatz, indem über die klassische Vorhersage hinaus auch eine Wahrscheinlichkeitsverteilung angegeben wird. Diese kennzeichnet, ob eine Entscheidung mit Unsicherheiten behaftet ist, wodurch eine Aufteilung der Verantwortung zwischen Klassifizierer und Entscheidungsträger ermöglicht wird. Die bei *multiclass classification* häufig verwendete *one-hot-Codierung*, welche lediglich den maximalen Ausgangswert des Modells als Gesamtvorhersage bestimmt, eignet sich nicht um Rückschlüsse auf die Sicherheit der *prediction* zu ziehen [2].

Die vorliegende Arbeit beschreibt aus diesem Grund die Entwicklung eines Ensembles aus mehreren Klassifizieren, welches bei gleichbleibender Erkennungsrate eine zuverlässige Aussage über die Unsicherheit der getätigten Klassenzuweisung zu generieren vermag.

FRAGESTELLUNG

Aus der einleitenden Motivation und den gegebenen Rahmenbedingungen [2] lässt sich folgende Forschungsfrage ableiten, die in den weiteren Abschnitten beantwortet werden soll: Wie kann ein probabilistischer *Random Forest* (RF) Klassifizierer als Ensemble implementiert werden und eignet sich die relative Häufigkeit der getätigten Klassenzuweisungen, um die Vorhersageunsicherheit des Ensembles zu bestimmen?

STAND DER TECHNIK

CIFAR-10 ist ein von Entwicklern des *Canadian Institutes for Advanced Research* erstellter Datensatz, der im Bereich der Bilderkennung weit verbreitet und für die Optimierung und das Testen von maschinellen Lernalgorithmen vorgesehen ist. Er besteht aus 60.000 32x32 Pixel großen RGB Bildern, die in zehn verschiedene Klassen zu jeweils 6.000 Bilder eingeteilt und mit entsprechenden Klassenlabels versehen sind [3, 4]. In *Comparison of ML classifiers for Image Data* wurden verschiedene, auf diesen Datensatz trainierte, Modelle gegenübergestellt. Mit etwa 65% erzielte das Convolutional Neural Network (CNN) die höchste Genauigkeit auf die Testbilder [5]. Moderne ML Modelle, in der Regel CNNs, erreichen Erkennungsraten von über 95% [6].

Bagging oder *Bootstrap Aggregation* ist eine Ensemble-Lernmethode, die darauf abzielt, Genauigkeiten durch die Implementierung einer Reihe von homogenen maschinellen Lernalgorithmen zu maximieren. Die Verwendung mehrerer Basis-Klassifizierer, welche separat mit einer Stichprobe aus dem Datensatz trainiert werden, kann durch einen Abstimmungs- oder Durchschnittsansatz eine stabilere und genauere Vorhersage generieren [7]. Für die Implementierung eines *Bagging Predictors* [8] bietet *scikit-learn* ein fertiges Modul [9]. Scikit-learn ist eine der am weitesten verbreiteten Bibliotheken für maschinelles Lernen in *Python* und bietet eine Vielzahl von Algorithmen und Funktionen für die Datenvorbereitung, Modellauswahl und -bewertung.

Random Forests sind algorithmische Modelle, die von sich aus ein Ensemble darstellen. Sie wurden 2001 von Leo Breiman in seinem gleichnamigen Paper [10] vorgestellt und können zur Verbesserung der Vorhersagegenauigkeit von Entscheidungsbäumen ([11]) eingesetzt werden, indem sie mehrere dieser kombinieren. Random Forests sind ein wichtiger Bestandteil von *ML-Pipelines* und bilden oft die Basis für komplexere Modelle [12].

Das Paper *A review of probabilistic forecasting and prediction with machine learning* [13] bietet eine umfassende Übersicht über die Anwendung der probabilistischen Vorhersagen. Vorteile dieser Methoden werden in *Ensembles for Uncertainty Estimation: Benefits of*

Prior Functions and Bootstrapping [14] analysiert und vorgestellt. Letzteres belegt, dass die Kombination verschiedener Techniken die Schätzung von Unsicherheiten verbessern kann und aktueller Forschungsstand ist.

Die probabilistische Objekterkennung für das autonome Fahren stellt ein spezifisches Anwendungsfeld dar, welches in den letzten Jahren erhebliche Fortschritte gemacht hat. In der Publikation *Probabilistic Object Detection: Definition and Evaluation* [15] wird die Bedeutung dieser untersucht. Ein Vergleich verschiedener Ansätze wird in *A review and comparative study on probabilistic object detection in autonomous driving* [16] vorgenommen. Beide Arbeiten stellen die Wichtigkeit der Forschungsfrage im Bezug auf Sicherheit und Zuverlässigkeit für cyber-physische Systeme dar.

MODEL UND LÖSUNGSANSATZ

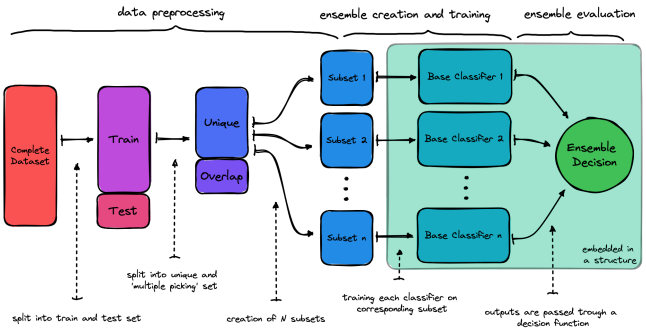

FIG. 1. Visualisierung der Erstellung des Bagging Ensembles Quelle: In Anlehnung an [7]

Die Pipeline vom rohen Datensatz bis hin zur Ensemble Vorhersage, wie auch die Definition der notwendigen Arbeitspakete ist in Figure 1 dargestellt. Obwohl der erste Teil nicht explizit in der Aufgabenstellung gefordert ist, stellt dieser eine unerlässliche Voraussetzung für jedes ML Projekt dar: die Vorbereitung des Datensatzes auf den späteren Anwendungsfall.

Das *Data Preprocessing* beschreibt neben der Aufteilung des Datensatzes in benötigte *Subsets* auch die Extraktion von essentiellen *Features* aus den Daten. Die gesamte Modellentwicklung und -untersuchung wird anhand der CIFAR-10 Bilder durchgeführt. Eine Reduzierung der Datenmenge, um die Entwicklungszeit, vor allem die Trainings- und Evaluierungsdauer, zu beschleunigen erfolgt über eine Konvertierung in einkanalige Graustufenbilder. Sowohl eine Skalierung der Pixelwerte von $0-255$ auf $0-1$ als auch eine Dimensionsreduzierung von $32x32x1$ auf 1024 wird durchgeführt, um die Bilder für die Übergabe an das Modell vorzubereiten. Räumlichen Merkmale gehen bei der beschriebenen Vorbereitung nicht verloren [5]. Abschließend werden die aufbereiteten Trainingsdaten, für das im folgenden beschriebene Ensemble, weiter unterteilt. Das *unique* Subset wird aus dem Großteil dieser erstellt und steht im weiteren Verlauf für die Erstellung einzigartiger Teilmengen für die Einzelklassifizierer, die kein anderer zum Training bekommt, zur Verfügung. Verbleibende Trainingsbilder werden für das Training aller Ensemblemitglieder genutzt und als *overlap* Subset zusammengefasst. Die genaue Einteilung der Subsets erfolgt beim Erstellen des Ensembles [2].

Der *Random Forest Base Classifier*, als grundlegendes Klassifizierungsmodell im Ensemble, wurde mithilfe der Hyperparameteroptimierung *GridSearchCV* [17] ausgelgt. Dabei wurden die Parameter *min_samples_leaf*, *max_features* und *n_estimators* variiert, da diese den größten Einfluss auf die Vorhersagegenauigkeit des RF haben [18]. Für jeden Hyperparameter wurden Bereiche rund um den *default* Wert [19] festgelegt. Anschließend wurde das Modell für jede Kombination trainiert und bewertet, indem die Genauigkeit auf den Testdatensatz ermittelt wurde. Die beste Parametervariation dient im Folgenden für die Definition des Basisklassifizierers.

Zunächst wird ein *Ensemble Classifier* aus mehreren dieser erstellt. Um die Aufteilung der Trainingsdaten auf die einzelnen Mitglieder gemäß Aufgabenstellung zu realisieren, wird eine Ensemble Klasse implementiert. Dafür wird die in Figure 1 angedeutete Struktur aus N Klassifizierern erstellt, die anschließend, je nach Ensemblegröße, die einzigartigen und überlappenden Daten auf die Teilnehmer aufteilt (Table I). Aus dem *unique* Subset werden Bilder sequentiell herausgezogen, während dem *overlap* Set zufällig ausgewählte Bilder entnommen und auf die homogenen Klassifzierer aufgeteilt werden. Anschließend wird jeder Random Forest auf den ihm zugewiesenen Datensatz trainiert.

Trainset		10	20	40
unique	40.000	4.000	2.000	1.000
overlap	10.000	6.000	5.000	3.000
$\sum$	50.000	10.000	7.000	4.000

TABLE I. Aufteilung der Trainingsbilder auf die verschieden großen Ensembles (10, 20 und 40 Mitglieder) je Klassifizierer

Abschließend erfolgt die *Ensemble Evaluation*. Durch die Kombination der Vorhersagen der einzelnen Mitglieder über den Mehrheits-Score lässt sich die Gesamtvorhersage bestimmen. Für ein Ensemble mit N Einzelklassifizierern, berechnet sich der Output o_i für Klasse i zu: $o_i = {}^{n_i}/_N$; mit n_i = Anzahl der Klassifizierer, die das Bild zur entsprechenden Klasse zählen. Für jedes Bild erhält man für alle Klassen eine relative Häufigkeit, wovon der Maximalwert o_{max} die Vorhersage des Ensembles angibt (Figure 2). Die Definition einer Grenze p mit $0 < p < 1$ erlaubt dementsprechend folgende Frage: Wie viele Bilder mit Maximalwert $o_{max} < p$ gibt es und wie viele davon sind richtig bzw. falsch klassifiziert?

```
# is prediction_i subject to uncertainty?
if o_max_i < p:
    M += 1

    # is it right or wrong?
    if pred_i == label_i: R += 1
    else: F += 1
```

Unter M werden alle Bilder zusammengefasst, die von Ensemble mit einer Vorhersagesicherheit unterhalb des Grenzwertes klassifiziert wurden – sie sind mit einer gewissen Unsicherheit behaftet. R und F geben die Anzahl der richtig bzw. falsch klassifizierten Bilder an und sind auf die Anzahl der "unsicheren Vorhersagen" relativiert. Alle drei Kenngrößen werden auf die Anzahl der evaluierten Bilder bezogen und anschließend über den Grenzwerten p_{range} von 0.0 bis 1.0 aufgetragen.

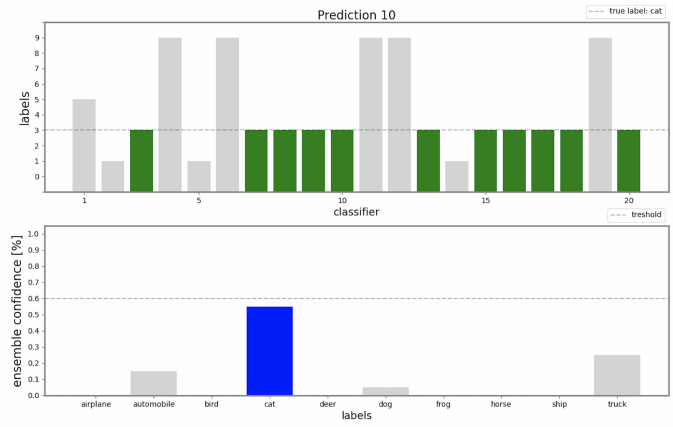

FIG. 2. Exemplarische Visualisierung des Vorhersageprozesses eines Ensemble mit 20 Mitgliedern für ein korrekt erkanntes Bild bei einer Vorhersagesicherheit $o_{max} < 0.6$ Quelle: Eigene Darstellung

ERGEBNISSE

Figure 3 dient der exemplarischen Darstellung der Ergebnisse. Für $p <= 0.3$ liegen die Kurven von M, R und F bei ≈ 0, woraus sich schließen lässt, dass jedes der Bilder mit $o_{min} >= 0.3$ klassifiziert wurde. Im weiteren Verlauf, mit steigendem Grenzwert p, werden immer mehr Bilder mit einer Unsicherheit ausgewertet. Da R und F zusammen M ergeben, steigen diese dementsprechend dabei mit an. Für $p_{max} = 1.0$ werden hier 87.2% als unsicher eingestuft, die restlichen 12.8% werden entsprechen mit vollständiger Einigkeit vom Ensemble Estimator, wie in Figure 4 (Anhang) zu sehen ist, klassifiziert. Wie hoch der Anteil an tatsächlich richtig ausgewerteten Bildern dabei ist lässt sich aus den dargestellten Kurven nicht ablesen. Jedoch ist zu erkennen, dass bei sinkender Schwelle das Verhältnis von R/F ebenfalls abfällt (Anhang Figure 6). Bei einem kleinen Wert für p ist demnach der Anteil der falsch klassifizierten Bilder höher als bei einer hohen Grenze. Vice versa ist der Anteil der korrekt klassifizierten Bilder höher, wenn auch der Grenzwert für eine "sichere Vorhersage" hoch ist. Diese Tendenz ist bei allen Ensemblegrößen sowohl für die Test- als auch Trainingsdaten zu erkennen (siehe Anhang Figure 7 - Figure 9).

Auf die absolute Vorhersagegenauigkeit der Bagging-Modelle hat eine steigende Anzahl an Mitgliedern einen tendenziell verschlechternden Einfluss. Bei der Auswertung der Trainingsdaten ist dieser besonders stark, was auf eine ungünstige Auslegung der Subsetgrößen für *unique* und *overlap* hindeutet. Dieser Einfluss kann jedoch auch durch die verschiedenen Anzahlen an Trainingsiterationen der verschiedenen Ensembles hervorgerufen werden. So wird das 10er Ensemble über $10 * 10.000 = 100.000$ Iterationen trainiert, während das große über insgesamt 160.000 Bilditerationen trainiert wird. Weiter ist aus den verschiedenen Verläufen zu erkennen, dass eine Erhöhung der Teilnehmerzahl die Kurven leicht nach oben verschiebt. Bei gleicher Schwelle werden demnach bei den größeren Ensembles mehr Bilder mit Unsicherheiten ausgewertet.

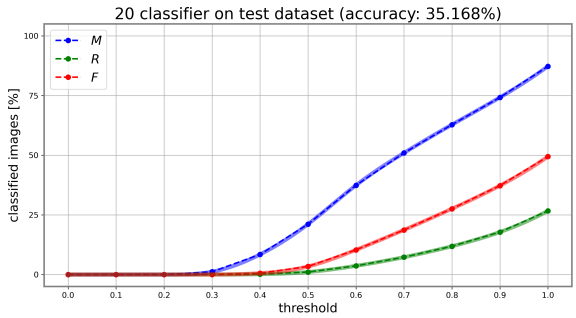

FIG. 3. Darstellung der M-, R-, und F-Kurven gegen Schwellgrenzwert p für ein Ensemble mit 20 Klassifizierern auf die Testdaten Quelle: Eigene Darstellung

ZUSAMMENFASSUNG

In der vorliegenden schriftlichen Ausarbeitung wurde untersucht, ob die relative Häufigkeit der Klassenzuweisungen eines Ensembles Aussagen über die Vorhersageunsicherheit des Modells zulässt. Hierfür wurde eine eigene Klasse in Python implementiert, die es erlaubt verschieden große Ensembles mit dem CIFAR-10 Bilddaten zu trainieren und evaluieren. Die Ergebnisse legen nahe, dass eine Korrelation zwischen der relativen Häufigkeit der Vorhersagen und den Unsicherheiten besteht, da bei geringeren Schwellwerten ein erhöhter Anteil an falsch

klassifizierten Bildern zu beobachten ist ($^{R}/_{F}$). Eine gleichbleibende Vorhersagegenauigkeit konnte dabei aber nicht eingestellt werden, wodurch weitere Untersuchungsfragen aufgeworfen werden. Weiterführend muss zum einen untersucht werden, welchen Einfluss die Ensemblegröße auf die Wahrscheinlichkeitsvorhersage hat und vor allem wie diese mit einer geeigneten Metrik konkret ausgewertet werden kann. Zum anderen ist zu ermitteln, welche Auswirkungen das Verhältnis der Subsets (*unique* und *overlap*) hat. Als weiterer Untersuchungsansatz bleibt offen, ob Modelle, die besser für die Auswertung von Bilddaten geeignet sind als RF, andere Ergebnisse in Bezug auf die probabilistische Klassifikaton erzielen.

[1] Lennart Fuhrig, *GitHub Repository Machine Learning 1*
[2] Peter Nalbach, *Themen für Schriftliche Ausarbeitung in Machine Learning 1 WiSe 2022 / 23*
[3] Alex Krizhevsky et al., *The CIFAR-10 dataset*
[4] S. Luber, N. Litzel, *Was ist CIFAR-10?*, März 2020
[5] S. Dahiya et. al., *Comparison of ML classifiers for Image Data*, 2020
[6] Will Cukierski, *CIFAR-10 - Object Recognition in Images*, 2013
[7] Fernando López, *Ensemble Learning: Bagging & Boosting*, Januar 2021
[8] Leo Breiman, *Bagging predictors*, 1996
[9] scikit-learn, *sklearn.ensemble.BaggingClassifier*
[10] Leo Breiman, *Random Forests*, Oktober 2001
[11] IBM, *What is a Decision Tree?*
[12] IBM, *What is a Random Forest?*
[13] H. Tyralis, G. Papacharalampous, *A review of probabilistic forecasting and prediction with machine learning*, 2022
[14] V. Dwaracherla et. at., *Ensembles for Uncertainty Estimation: Benefits of Prior Functions and Bootstrapping*, 2022
[15] D. Hall et.al., *Probabilistic object detection: Definition and evaluation*, 2020
[16] D. Feng et.al., *A review and comparative study on probabilistic object detection in autonomous driving*, 2021
[17] scikit-learn, *sklearn.model_selection.GridSearchCV*
[18] Coelho, Aimee, *Narrowing the Search: Which Hyperparameters Really Matter?*, Juni 2020
[19] scikit-learn: *sklearn.ensemble.RandomForestClassifier*

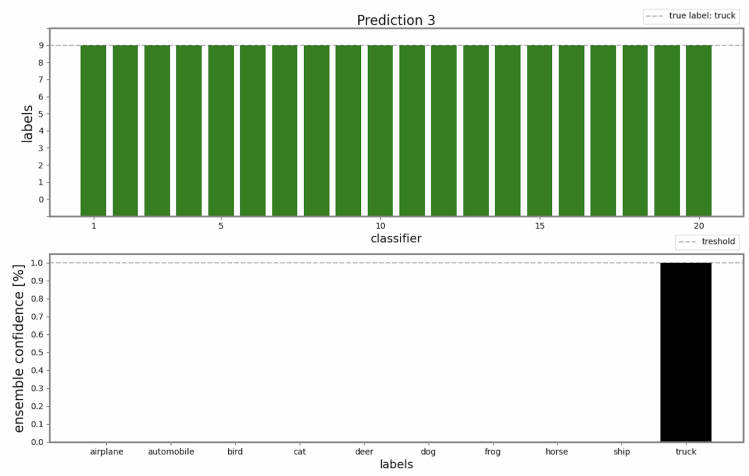

FIG. 4. Einheitliche, korrekte Klassifizierung eines *Trucks* mit resultierendem o_{max} von 1.0

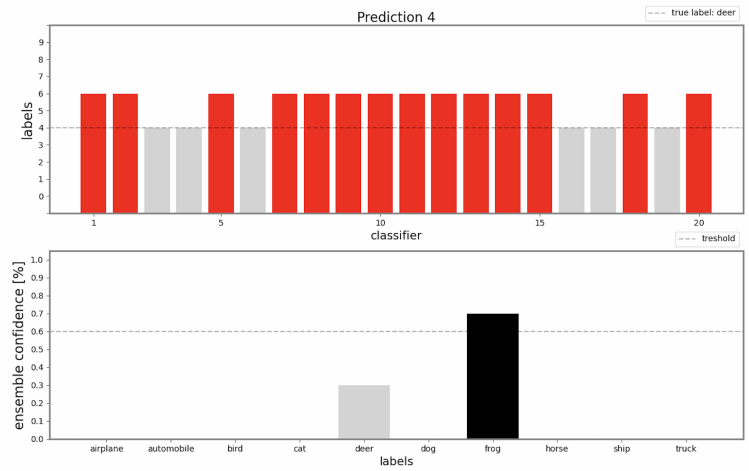

FIG. 5. Hohes o_{max} bei falscher Klassifizierung (weitere Bespiele als animiertes Bild in GitHub [1] unter results)

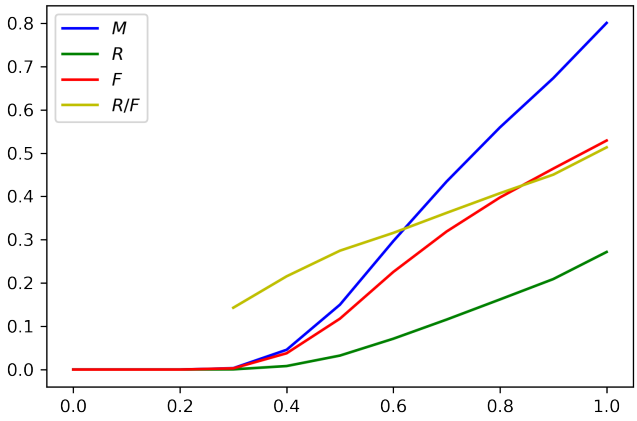

FIG. 6. Ergänzender $^{R}/_{F}$-Verlauf zu Figure 3 (qulitativer Verlauf für alle Ensemblegrößen identisch)

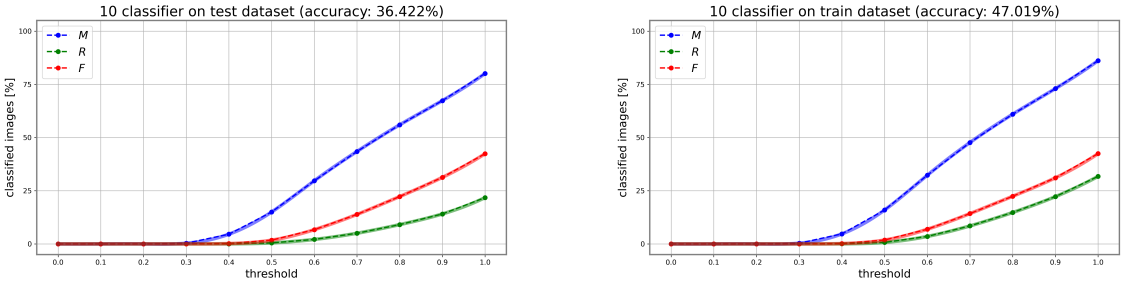

FIG. 7. Graphische Darstellung der M-, R-, und F-Kurven gegen den Grenzwert p für ein Ensemble mit 10 Klassifizierern für die Test- (links) und Trainingsdaten (rechts)

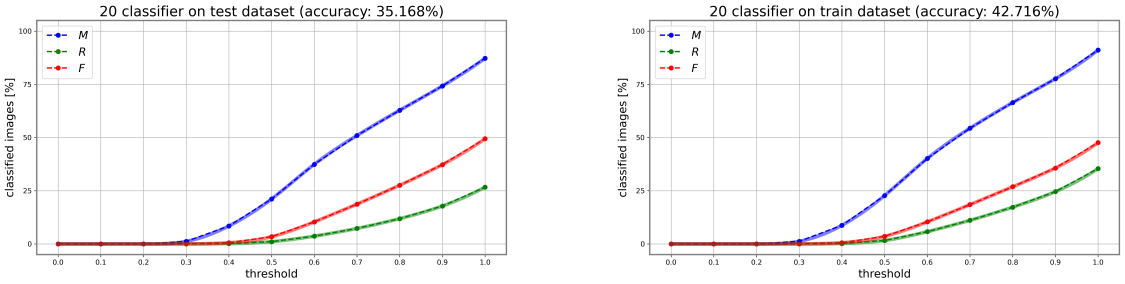

FIG. 8. Ensemble mit 20 Klassifizierern

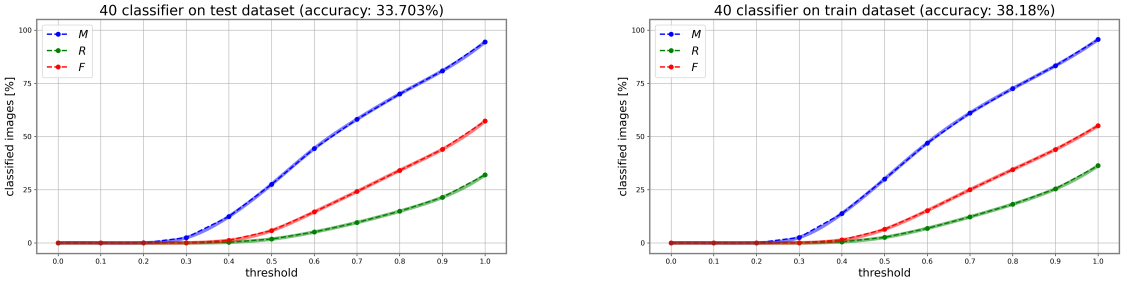

FIG. 9. Ensemble mit 40 Klassifizierern

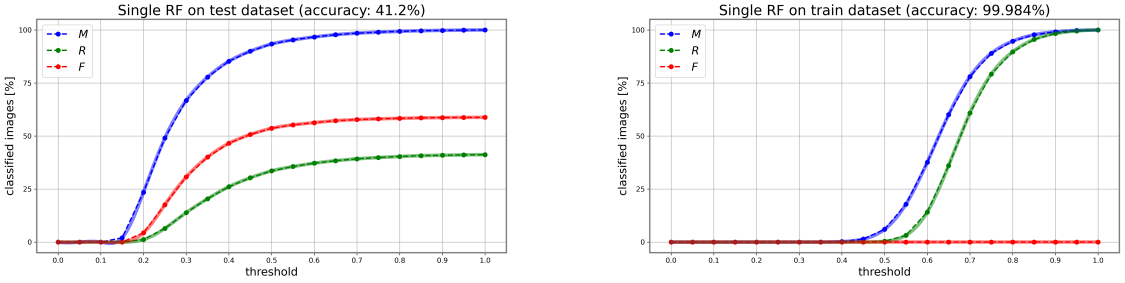

FIG. 10. Random Forest als einzelner Klassifizierer. Ermittlung der Verläufe über die `predict_proba`-Methode, die ähnlich der vorgestellten Evaluierung die Ausgaben der einzelnen Entscheidungsbäume innerhalb des RF als relative Häufigkeit auswertet. Verläufe sind nur bedingt mit den anderen Ensembles vergleichbar, da der einzelne RF mit dem gesamten Trainingsdaten ohne weitere Aufteilung trainiert wurde (Einfluss vor allem im rechten Plot zu sehen)