# Exposé
# Bachelor Thesis
# Open-World Classification

Sebastian Kühn
Betreuer: Dr. Patrick Schäfer
Institution: Humboldt Universität zu Berlin am Institut für Informatik

Juli 2022

# 1 Forschungsthema

Klassifikation ist ein Teilgebiet des Maschinellen Lernens, bei dem ein Modell anhand von gelabelten Trainingsdaten lernt auf neuen ungesehenen Daten Labels vorherzusagen. Beispiele sind das Erkennen von Liedern oder das Klassifizieren von DNA-Sequenzen. Trainings- und Testdaten sind z.B. eine Folge von Basenpaaren und Klassen sind verschiedene Organismen wie zum Beispiel eine Bakterienart. Traditionelle Modelle machen eine "Closed World Assumption", d.h. die Labels der Trainingsdaten und Testdaten sind identisch und es müssen keine Vorhersagen für Objekte mit bislang unbekannten Labels getroffen werden.
Ein Problem der Closed World Assumption entsteht, wenn ein zu klassifizierendes Objekt nicht zu den gelernten Labels passt, solche werden im Folgenden als unbekannte Objekte bezeichnet. Ein Klassifikationsverfahren weist nun fälschlicherweise dem unbekannten Objekt das am besten passende Label aus den Trainingsdaten zu. Bezüglich der vorherigen Beispiele können keine neuen Lieder und keine neuen Organismen erkannt werden, da sie bestehenden Trainingslabels zugeordnet werden. Das Problem, unbekannte Objekte möglichst vor der Klassifikation zu erkennen, wird "Open-World Classification" oder "Open-Set Recognition" [Aka+22] genannt.

# 2 Zielsetzung und Erkenntnisinteresse

Die zu klassifizierenden Daten können beispielsweise Bilder, Text oder Zeitreihen sein. Der Fokus dieser Arbeit liegt auf univariaten Zeitreihen. Eine Zeitreihe ist eine zeitlich geordnete Folge von Werten. Ein Beispiel ist in Figure 1 zu sehen. Die x-Achse ist die Zeitachse und in Kalendertagen unterteilt. Die y-Achse ist der jeweilige dazugehörige Kraftstoffpreis von Super E5 an öffentlichen Tankstellen in Deutschland. Es handelt sich um eine univariate Zeitreihe, da die Zeitreihe lediglich ein Feature besitzt. Besäße die Zeitreihe mehrere Features, zum Beispiel zusätzlich die Kraftstoffpreise von Super E10 und/oder Diesel, wäre es eine multivariate Zeitreihe. Ziel meiner Bachelorarbeit ist die Implementierung und Evaluation verschiedener Ansätze zur Open-World Classification in Bezug auf Zeitreihen.
In der Arbeit werden insbesondere die Python Bibliotheken sklearn und sktime verwendet. Sklearn und sktime sind Bibliotheken für Maschinelles Lernen, wobei sktime ausschließlich Funktionen für Zeitreihen bietet. Meine Arbeit wird in der Programmiersprache Python verfasst.

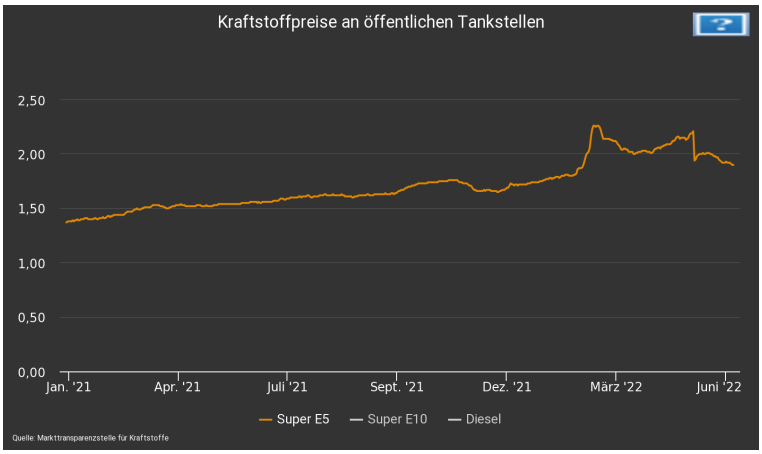


Figure 1: Beispiel einer Zeitreihe
https://www.dashboard-deutschland.de/indicator/tile_1648135639982
letzter Zugriff: 13.07.2022

# 3 Hintergrund und Stand der Technik

## 3.1 Open-World Classification

Bisherige Ansätze in der Zeitreihen-Klassifikation basieren auf der Closed World Assumption. [Bag+17] gibt einen Überblick zum aktuellen Stand der Technik. Open-World Classification ist Gegenstand aktueller Forschung. Spezielle Ansätze für Zeitreihen gibt es aber bis jetzt nur wenige. *Open Set InceptionTime*, der erste Open-World Classification Ansatz für Zeitreihen, wurde erst dieses Jahr in [Aka+22] veröffentlicht. Unbekannte Objekte werden über einen Schwellwert bezüglich der DTW Distanz und der Cross-Correlation zwischen den einzelnen klassenspezifischen barycenters und der zu klassifizierenden Zeitreihe bestimmt. Ein barycenter repräsentiert eine Klasse von Zeitreihen und kann beispielsweise aus dem Durchschnitt von Datenpunkten, welche zu der gleichen Klassen gehören und gleichen Zeitstempel besitzen, bestimmt werden. Die DTW Distanz wird mithilfe einer Matrix M bestimmt, deren Elemente die quadratische Distanz aller Datenpunkte zweier Zeitreihen der Länge n repräsentieren. Gesucht wird der Pfad der Länge n durch M mit minimaler Summe der jeweiligen paarweisen Distanzen. Cross-Correlation berechnet die Korrelation aller Werte zweier Zeitreihen, als Funktion der Verschiebung einer Zeitreihe, auch Lag genannt.
In [Ren19] wird ein allgemeiner Ansatz zur Open-World-Classification beschrieben, welcher durch Mutation von DNA Sequenzen inspiriert wurde. Die Klasse von unbekannten Objekten wird anhand augmentierter Trainingsdaten erkannt. Diese Idee wird in der Arbeit übernommen und ist im Lösungsansatz *Augmentierung* (siehe Abschnitt 4) enthalten.

Die Lösungsansätze zur Open-World-Classification nutzen gängige Klassifikations-Algorithmen und teilweise Anomaly Detektoren, welche nun im Folgenden beschrieben werden.

## 3.2 Klassifikations-Algorithmen

Die nachfolgende Taxonomie wurde von [Bag+17] übernommen:
**Whole-series**-basierende Algorithmen vergleichen Zeitreihen als Vektor in Kombination mit klassischen Verfahren zur Klassifikation. *Rocket* transformiert jede Zeitreihe in 2k viele Features. Eine Zeitreihe wird mit k zufälligen Kernels multipliziert und anschließend wird pro Kernel das Maximum und das Verhältnis der positiven Werte bestimmt, die neuen Features der Zeitreihe. Anschließend erfolgt eine lineare Klassifikation mit den neuen Features, d.h. die Trainingsdaten werden genutzt um eine lineare Entscheidungsgrenze zu trainieren.
**Interval**-basierende Algorithmen vergleichen zufällig gewählte feste Intervalle (Teilsequenzen) zweier Zeitreihen, anstatt die gesamten Zeitreihen, miteinander. *Time series forest* unterteilt die zwei Zeitreihen in zufällige Intervalle und berechnet jeweils Durchschnitt, Standardabweichung und durchschnittliche Steigung im Intervall. Der Vektor der statistischen Werte wird als Features für einen Random Forest genutzt. Ein Random Forest besteht aus mehreren Entscheidungsbäumen. Jeder der Entscheidungsbäumen wird mithilfe einer zufälligen Teilmenge der Trainingsdaten trainiert, bei welcher Datenpunkte mehrfach vertreten sein dürfen. Ziel ist es pro Knoten eine Entscheidungsgrenze bezüglich eines Features zu wählen, so dass jeder der Datenpunkte im Blatt einer möglichst eindeutigen Klasse angehört. Für eine Vorhersage werden alle Entscheidungsbäume von Wurzel bis Blatt durchlaufen und die Mehrheit der Klassen bestimmt die Klassifikation. Der jeweilige Pfad ist eindeutig anhand der trainierten Entscheidungsgrenzen bestimmt.
**Shapelet**-basierende Algorithmen klassifizieren anhand des Vorkommens von Teilsequenzen innerhalb der Zeitreihe. Shapelet transform nutzt dafür ein Sliding Window. Dabei wird das Shapelet über die Zeitreihe geschoben. Es wird die Distanz des Shapelets zu jeder Stelle der Zeitreihe minimiert. Shapelets werden oft mit k-NN klassifiziert.
**Dictionary** basierende Algorithmen klassifizieren anhand der Häufigkeit von Mustern in Zeitreihen. *Bag-of-patterns* transformiert die Teilsequenzen in eine Folge von Wörtern. Die Häufigkeit der Wörter wird mittels eines Histograms gezählt. Das Histogram kann für eine k-NN Klassifikation genutzt werden.

In der Arbeit werden zwei verschiedene Verfahren zur Klassifikation benutzt. Es werden *Time series forest* und *Rocket* verwendet.

## 3.3 Anomalie Detektoren

Der **Isolation Forests** bestimmt Anomalien mittels Entscheidungsbäumen. Die einzelnen Entscheidungsbäume werden Isolation Trees genannt. Ein Isolation Tree enthält in jedem Blatt nur einen einzelnen Datenpunkt und der

Datenpunkt wird als isoliert bezeichnet. Datenpunkte welche in weniger Schritten isoliert werden können, also eine geringe Pfadlänge besitzen, sind Anomalien. Der Anomalie Score ergibt sich aus dem Verhältnis der durchschnittlichen Tiefe in den Isolation Trees und der durchschnittlichen Tiefe aller Blätter in den Isolation Trees.
Die **One-Class SVM** ist ein modifiziertes Verfahren der Support Vector Machine. SVM ist ein Large Margin Classifier. Klassengrenzen werden dahingegend optimiert, um einen möglichst großen Bereich um die Entscheidungsgrenze frei von Objekten zu halten. One-Class SVM erzeugt eine solche Entscheidungsgrenze um den Ursprung der Daten. Punkte nahe am Ursprung sind Anomalien.
**Local Outlier Factor** ist ein dichte-basiertes Verfahren. Die lokale Dichte eines Punktes wird anhand der Distanz zu den k-nächsten-Nachbarn berechnet und im Verhältnis zu derer lokalen Dichte gesetzt. Anhand des daraus resultierenden Verhältnisses werden Anomalien erkannt.

# 4 Lösungsansätze

Im folgenden werden drei Ansätze zur Open-World-Classification beschrieben, die implementiert und evaluiert werden sollen.
(1) **Anomalie-Erkennung**: Wir können das Problem zweistufig mittels vorgelagertern Anomalieerkennungsproblem umformulieren. Der Anomalie Detektor wird auf den Trainingsdaten trainiert. Unbekannte Objekten können somit erkannt werden, bevor die Klassifikation stattfindet. Es werden drei Anomalie Detektoren evaluiert: *Isolation Forests*s, *One-Class SVM* und *Local Outlier Factor*. Diese Verfahren sind in der Bibliothek sklearn implementiert.

(2) **Augmentierung**: In den Trainingsdaten wird eine neue Klasse für unbekannte Objekte mit neuen Daten $D_{train}$ eingefügt und der Klassifikator wird auf diesen Daten trainiert. Werden Testdaten dieser Klasse zugeordnet, handelt es sich um ein unbekanntes Objekt. Es werden zwei verschiedene Ansätze verwendet um $D_{train}$ zu erzeugen.
a) $D_{train}$ besteht aus zufällig generierten Daten.
b) $D_{train}$ wird aus den ursprünglichen Trainingsdaten gewonnen. Es wird eine Teilmenge der ursprünglichen Trainingsdaten zufällig ausgewählt und an n vielen Punkten werden die Features zufällig verändert. Die Idee mutierte Trainingsdaten zu nutzen stammt aus [Ren19]. Dort wird $D_{train}$ für ein zweites Modell genutzt, deren Klassifikation anhand einer Likelihood Ratio berechnet wurde und aus der Likelihood Ratio vom originalen Modell rausgenommen wurde.

(3) **One-vs-Rest Threshold**: Klassifikation lässt sich in binäre und mehrklassen Klassifikation unterteilen. Ein binäres Klassifikationsmodell unterscheidet zwischen zwei verschiedenen Klassen. Ein Ansatz Mehrklassen-Klassifikation anhand von binärer Klassifikation umzusetzen ist One-vs-Rest. Das mehrklassen Klassifikationsproblem wird in k binäre Klassifikationsprobleme übersetzt. Besitzt der Trainingsdatensatz k verschiedene Labels, werden k Modelle trainiert.

Ein Modell wird pro Label trainiert und der Trainingsdatensatz wird jeweils angepasst. Er beinhaltet zwei Klassen, eine Klasse für das zu vorherzusagene Label und eine Klasse für die restlichen Labels. Gesamt Vorhersage ist das Label mit der maximalen Wahrscheinlichkeit, also das jeweilige Modell mit der maximalen Wahrscheinlichkeit. Um One-vs-Rest für Open-World-Classification zu nutzen, lernen wir einen Schwellwert. Es handelt sich um ein unbekanntes Objekt, wenn die maximale Wahrscheinlichkeit unter diesem Schlwellwert liegt.

# 5 Daten/Evaluation

Der UCR Times Series[1] Datensatz wird für die Trainings-, Evaluierungs- und Testdaten genutzt. Für jeden der drei Open-World Classification Ansätze wird *Time series forest* und *Rocket* evaluiert. Es wird jeweils eine Klasse aus den Trainingsdaten entfernt und die Accurancy bezüglich dieser Klasse berechnet, es als unbekanntes Objekt erkannt zu haben. Dies wird für jedes Label wiederholt und die mittlere Accurancy gemessen.

# 6 Vorläufige Gliederung

Zu Beginn möchte ich *Time series forest* und *Rocket* evaluieren. Anschließend werden die einzelnen Verfahren jeweils mit den Lösungsansätzen zur Open-World Classification kombiniert und evaluiert. Evaluiert wird mittels Accuracy. Es folgt ein Vergleich um einen möglichen Trade-off vom Klassifizieren von unbekannten Objekten und dem Klassifizieren von bekannten Objekten festzustellen, wie er bei Open Set InceptionTime festgestellt wurde, siehe [Aka+22].

[1] https://www.cs.ucr.edu/∼eamonn/time_series_data/

# Quellenverzeichnis


[LTZ09] Fei Tony Liu, Kai Ting, and Zhi-Hua Zhou. "Isolation Forest". In: Jan. 2009, pp. 413–422. DOI: 10.1109/ICDM.2008.17.

[Bag+17] Anthony Bagnall et al. "The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances". In: *Data Mining and Knowledge Discovery* 31.3 (2017), pp. 606–660. ISSN: 1573-756X. DOI: 10.1007/s10618-016-0483-9. URL: https://doi.org/10.1007/s10618-016-0483-9.

[Lön+19] Markus Löning et al. *sktime: A Unified Interface for Machine Learning with Time Series*. 2019. DOI: 10.48550/ARXIV.1909.07872. URL: https://arxiv.org/abs/1909.07872.

[Ren19] Jie Ren. "Improving Out-of-Distribution Detection in Machine Learning Models". In: *Google AI Blog* (Dec. 17, 2019).

[Ala20] Mahbubul Alam. "Isolation Forest: A Tree-based Algorithm for Anomaly Detection". In: *Towards Data Science* (Oct. 28, 2020).

[Ban20] Amey Band. "Multi-class Classification — One-vs-All One-vs-One". In: *Towards Data Science* (May 9, 2020).

[DPW20] Angus Dempster, François Petitjean, and Geoffrey I. Webb. "ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels". In: *Data Mining and Knowledge Discovery* 34.5 (July 2020), pp. 1454–1495. DOI: 10.1007/s10618-020-00701-z. URL: https://doi.org/10.1007%2Fs10618-020-00701-z.

[Jay20] Vaibhav Jayaswal. "Local Outlier Factor (LOF) — Algorithm for outlier identification". In: *Towards Data Science* (Aug. 31, 2020).

[Shr20] Soumya Shrivastava. "Cross Validation in Time Series". In: *Medium* (Jan. 14, 2020).

[zai20] $z_a i$. "Random Forest Explained". In: *Towards Data Science* (Sept. 16, 2020).

[Aka+22] Tolga Akar et al. "Open Set Recognition for Time Series Classification". In: *Advances in Knowledge Discovery and Data Mining*. Ed. by João Gama et al. Cham: Springer International Publishing, 2022, pp. 354–366. ISBN: 978-3-031-05936-0.