TU Dortmund

Big Data in der Astrophysik

Klassifikation veränderlicher Sterne mit Random Forests

Franny Spätner

Proseminar Big Data Analytics
SoSe 2016

Inhaltsverzeichnis

1 Big Data in der Astrophysik

In der Astrophysik spielt der Begriff Big Data heutzutage eine immer größer werdende Rolle. Dies wird besonders bei der Erkennung von (veränderlichen) Sternen deutlich. Früher konnten die Daten der Teleskope noch manuell ausgewertet werden. Im heutigen Zeitalter jedoch gibt es Teleskope, die mehrere Terabytes pro Stunde aufnehmen. Zur Auswertung solcher Datenmengen werden sehr häufig unterschiedliche Lernverfahren angewandt.

1.1 Feigelson und Jogesh, 2012 [7]

Der Artikel behandelt das Thema Big Data in der Astronomie. Es wird auf die Geschichte der Astronomie eingegangen und auf die heutige Entwicklung, die häufig eine Echtzeitauswertung von extrem großen Datenmengen fordert. Dazu werden zwei aktuelle Problemstellungen vorgestellt.
Die erste ist die Klassifikation veränderlicher Sterne. Anhand von Lichtkurven unterschiedlicher Sterne sollen die zugehörigen Klassen bestimmt werden.

Definition 1: Lichtkurve

Eine Lichtkurve ist eine zweidimensionale Darstellung der Helligkeit über die Zeit.

Dazu werden mehrere Methoden genannt. Häufig wird Crowd Sourcing betrieben. Hierbei wird von Menschen anhand einer Lichtkurve entschieden, um welche Klasse es sich handelt. Als eine weitere Methode wird das Klassifikationsverfahren Random Forests (RF) genannt, welches zur Unterscheidung zwischen Supernovae und anderen astronomischen Ereignissen verwendet wird.
Bei der zweiten Problemstellung handelt es sich um die Entfernungsschätzung (z.B. zu einer Supernova). Hierzu wird häufig das hierarchische Bayes-Modell zur Regression genutzt.
Im Folgenden geht es hauptsächlich um die erste Problemstellung (Klassifikation veränderlicher Sterne) und die Methode Random Forests.

Was: Es sind mehrere Objekte gegeben, deren Klassen bekannt sind. Mithilfe dieser Daten werden neue Objekte klassifiziert. Dabei sollen möglichst wenig Objekte einer falschen Klasse zugeordnet werden.

Wie: Der Algorithmus Random Forest wird zur Klassifikation der Objekte verwendet.

Warum: Das Klassifikationsverfahren Random Forest ist parallelisierbar und arbeitet aus diesem Grund besonder effizient auf großen Datenmengen.

1.2 Huijse, Estevez et al. 2014 [10]

Dieser Artikel befasst sich mit dem Feld der Time Domain Astronomy. In diesem Bereich beschäftigen sich Wissenschaftler mit der zeitlichen Veränderung astronomischer Objekte und Phänomene. Insbesondere gehört auch die Klassifikation veränderlicher Sterne in diesen Bereich.
Es werden unterschiedliche Problemstellungen zu diesem Thema betrachtet (Echtzeiterfassung von Datenstreams, Echtzeitklassifikation von Objekten, Analyse und Evaluation der gewonnen Daten). Wie bereits in [7] wird hier auch auf die Analyse von Lichtkurven und die Klassifizierung veränderlicher Sterne eingegangen.
Dazu wird ebenfalls der Algorithmus Random Forests erwähnt, sowie auch die Support Vector Machines (SVM), die genutzt werden, um bereits bekannte astronomische Objekte in Klassen einzuteilen. Mithilfe des k-Nearest-Neighbors-Algorithmus lassen sich neu erkannte astronomische Objekte klassifizieren (z.B. veränderliche Sterne). Das Gauß'sche Mischverteilungsmodell sowie das Bayesian model averaging werden verwendet, um periodisch veränderliche Sterne anhand der zugehörigen Lichtkurven zu unterscheiden.

1.3 Zusammenfassung

Die Problemstellungen der Artikel [7] und [10] lassen sich wie folgt zusammenfassen:

- Auswertung großer Datenmengen (Aufnahmen immer komplexer werdender Teleskope)
- **Klassifizierung** bereits bekannter astronomischer Objekte(veränderlicher Sterne, Supernovae,...) und neu entdeckter Objekte. Welche Methode zur Klassifikation arbeitet am effizientesten auf sehr großen Datenmengen und liefert die besten Ergebnisse (Korrektheit)?
- Erkennung neuer Objekte (novelty detection)

Definition 2: Klassifizierung

Gegeben:

- Klassen Y, $y \in \{-1, +1\}$
- Objekte X
- eine Trainingsmenge von Beispielen $T = \{(\vec{x_1}, y_1), ..., (\vec{x_n}, y_n)\} \subset X \times Y$

Finde:
eine Funktion $f : X \to Y$, die jedem Objekt $\vec{x}$ eine Klasse y zuordnet.
So dass:

$(f(\vec{x}) - \hat{y})^2$ minimiert wird.
($\hat{y}$ bezeichnet die korrekte Klasse zu dem Objekt $\vec{x}$.)

Die Objekte $\vec{x}$ beinhalten die spezifischen Merkmalsausprägungen. Das heißt, dass die Länge dieser Vektoren der Anzahl an Attributen entspricht, die von dem jeweiligen Anwendungsfall vorgegeben wird. In dem Kontext der veränderlichen Sterne könnten diese Attribute z.B. die Helligkeit oder Periode sein.

2 Veränderliche Sterne

Der Artikel [10] beschreibt unterschiedliche Methoden zur Klassifizierung veränderlicher Sterne. Das hierzu übergeordnete wissenschaftliche Feld ist die Time Domain Astronomy. Veränderliche Sterne haben im Gegensatz zu normalen Sternen keine konstant gleichbleibende Helligkeit sondern variieren diese in periodischen Abständen. Anhand unterschiedlicher Kriterien lassen sich diese Sterne in Klassen einteilen.

2.1 Kriterien zur Einteilung

In [5] werden die Kriterien zur Einordnung unterschiedlicher veränderlicher Sterne genauer beschrieben. Insgesamt sind mehr als 100 Klassen und Unterklassen veränderlicher Sterne bekannt [5, S.2]. Diese lassen sich vereinfacht auf zwei Ebenen einteilen (s. Abb. 1).
Auf der ersten Ebene wird zwischen intrinsischen und extrinsischen Sternen unterschieden. Intrinsische Objekte verändern ihre Leuchtkraft selbst. Extrinsische Objekte haben eine gleichbleibende Leuchtkraft, jedoch verändert sich die von diesen Objekten beobachtbare Helligkeit aufgrund anderer astronomischer Objekte in der Umgebung.
Die zweite Ebene bezieht sich auf die Ursache der Veränderlichkeit. Bei den intrinsischen Objekten wird zwischen pulsierend, eruptiv und kataklysmisch veränderlichen Sternen unterschieden. Ein pulsierender Stern vergrößert bzw. verkleinert seine Oberfläche in periodischen Abständen, woraus eine Änderung seiner Helligkeit resultiert. Eruptiv veränderliche Sterne ändern ihre Helligkeit nicht periodisch sondern abrupt durch Freisetzungen großer Energiemengen. Die Ursache dafür können zum Beispiel thermonukleare Reaktionen sein. Dieses Phänomen kann auch bei den kataklysmisch veränderlichen Sternen auftreten. In diesem Fall ist die Helligkeitsveränderung jedoch wieder periodisch und nicht abrupt. Die Veränderlichkeit der extrinsischen Sterne lässt sich entweder durch eine Rotation des Sterns selbst oder durch eine zeitweise Verdeckung durch ein anderes Objekt begründen. Somit wird hier zwischen rotationsveränderlichen und bedeckungsveränderlichen Sternen unterschieden. [5, S.2]

2.2 Verwendung

Veränderliche Sterne können bei der Analyse der Verteilung und Größe des Universums hilfreich sein. Besonders intrinsisch veränderliche Sterne werden zur

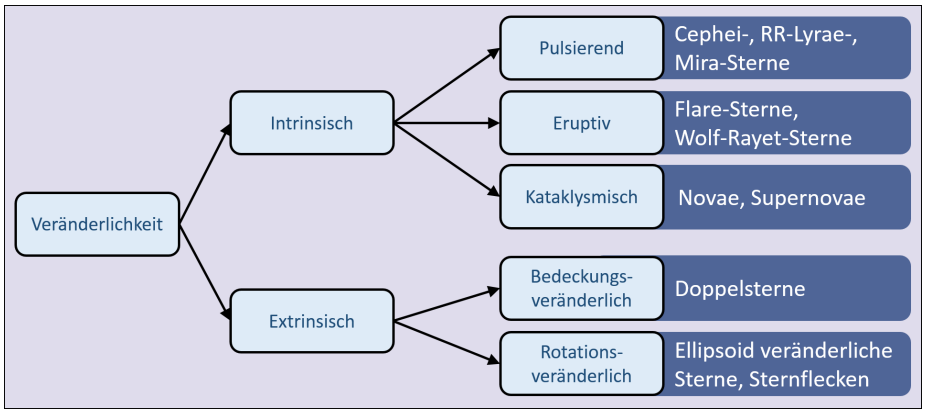

Abbildung 1: Topologische Klassifikation veränderlicher Sterne (nach [10, S.28])

Entfernungsmessung verwendet [10, S.29]. Dagegen lassen sich bei bedeckungsveränderlichen Sternen die Radien sowie die Massen der einzelnen Komponenten besonders gut berechnen[10, S.30].

2.3 Das LSST

Das Large Synoptic Survey Telescope (LSST) ist eins von vielen neuen Teleskopen, die in den nächsten Jahren für die Beobachtung veränderlicher Sterne eingesetzt werden. Es wird in Nordchile aufgebaut und soll ab 2022 die ersten Daten liefern. Innerhalb von 10 Jahren soll so eine 150 Petabyte große Datenbank generiert werden, die Details zu 50 Millionen astronomischer Objekte enthält. [10, S.28]
Das LSST legt den Schwerpunkt auf die Time Domain Astronomy. Es hat einen sehr großen Sichtbereich und wird jede Nacht einen Film des Himmels über der Südhalbkugel der Erde aufnehmen können [10, S.35].
Die extrem großen Datenraten des LSSTs und ähnlicher moderner Teleskope fordern neue Methoden zur Analyse und Auswertung der gesammelten Daten.

2.4 Auswertung der aufgenommenen Daten

Teleskope wie das LSST können Daten in Form von Lichtkurven aufnehmen. Diese müssen jedoch erst weiterverarbeitet werden, bevor die Daten zur Klassifikation genutzt werden können. Aus der Lichtkurve eines Sterns müssen also zunächst die Merkmalsausprägungen extrahiert werden (z.B. die Amplitude, Periode, Helligkeit, Farbe, ...). Hierfür gibt es unterschiedliche Methoden. Zur Bestimmung der Amplitude wurde früher häufig die Pogson's Methode angewandt, ein graphisches Verfahren, das auf der Lichtkurve arbeitet. Heutzutage ist es üblicher mit Hilfe eines Computer ein Polynom niedrigen Grades anhand der Punkte um das Maximum herum zu bestimmen und anschließend des Maximum des Polynoms zu berechnen. [14, S.62f]

Für die Bestimmung der Periode kann das Correntropy Kernelized Periodogram (CKP) oder das Binned Phase Diagram (BPD) genutzt werden. Das CKP benutzt die informationstheoretische Correntropy Funktion, die mit Hilfe der Dichtefunktion Ähnlichkeit über der Zeit misst [10, S.30]. Das BPD berechnet die bedingte Entropie der Lichtkurve und unterschiedlicher Perioden. Es wird die Periode gewählt, bei der die bedingte Entropie am kleinsten ist [10, S.31].
In Abb. 2 sind einige Lichtkurven unterschiedlicher veränderlicher Sterne und ihre Phasendiagramme dargestellt. Die Phase lässt sich mit Hilfe folgender Formel berechnen:

$$\frac{t - t_0}{P}$$

Dabei ist t der Zeitpunkt des aufgenommenen Messpunktes und t_0 der Zeitpunkt eines Maximums oder Minimums (z.B. des zuletzt beobachteten Maximums) und P die Periode des Sterns. Wenn man für jeden Messpunkt die Phase bestimmt und den Graph der Helligkeit über der Phase zeichnet, erhält man das Phasendiagramm des Sterns. [14, S.63]
Wenn die Periode noch nicht bekannt ist, kann das Phasendiagramm auch mit der Fourier Transformation bestimmt werden. Anschließend kann die zugehörige Periode ermittelt werden. Dafür werden mehrere Perioden getestet. Mit diesen wird das Phasendiagramm bestimmt und mit dem durch die Fourier Transformation erhaltenen Diagramm verglichen. Um nicht alle möglichen Perioden testen zu müssen, wird zuvor ein Intervall angegeben. Die Grenzen dieses Intervalls sind abhängig von der Länge und dem Abstand der Datenpunkte und der zu erwartenden Periode. [14, S.64]

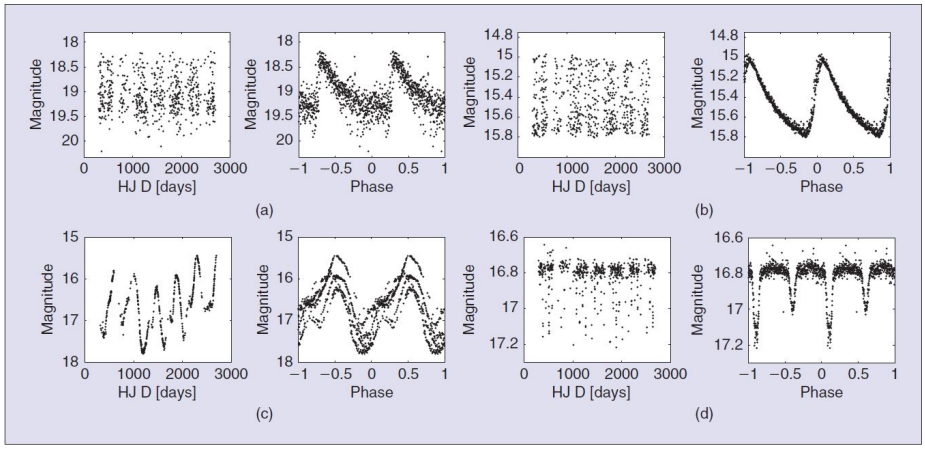

Abbildung 2: Lichtkurve und das dazugehörige Phasendiagramm eines RR-Lyrae-Sterns (a), eines Cephei-Sterns (b), eines Mira-Sterns (c) und eines Doppelsterns (d) (aus [10, S.30]).

Zur Bestimmung der weiteren Parameter können unter anderem Neuronale Netze verwendet werden [10, S.31].

Nach der Parameterbestimmung können auf den gewonnen Daten Methoden aus dem Bereich des maschinellen Lernens angewandt werden.

3 Maschinelles Lernen

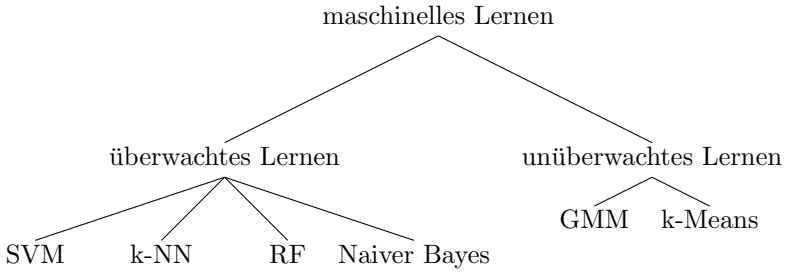

Abbildung 3: Teilgebiete des maschinellen Lernens

Definition 3: Überwachtes Lernen

Unter überwachtes Lernen (supervised Learning) fallen die Methoden, die zur Klassifizierung neuer Objekte eine Trainingsdatenmenge T benutzen, in der mehreren Objekten x eine Klasse y zugeordnet ist. Mithilfe der bekannten Klassen der Objekte in dieser Menge, können diese Algorithmen neue Objekte klassifizieren. [10, S.33]

Definition 4: Unüberwachtes Lernen

Die Methoden aus dem Bereich des unüberwachten Lernens (unsupervised Learning) nutzen ebenfalls eine Trainingsdatenmenge T bestehend aus Beispielobjekten X. Jedoch sind hierbei die Klassen Y der Trainingsdaten noch nicht bekannt und müssen vom Algorithmus selbstständig erkannt werden. Diese Methoden werden vor allem in Fällen eingesetzt, in denen keine oder nicht genug Informationen zur Verfügung stehen, um die Klassen im Voraus festzulegen. Wenn nur ein Teil der Klassen bekannt ist handelt es sich um teil-überwachtes Lernen (semi-supervised Learning).[10, S.34]

4 Entscheidungsbäume

Um den im nächsten Abschnitt aufgeführten Algorithmus für Random Forests zu verstehen, müssen zunächst die Entscheidungsbäume definiert werden.

Definition 5: Entscheidungsbaum

Entscheidungsbäume sind Bäume bestehend aus einer Wurzel, mehreren Knoten und Blättern. In jedem Knoten wird eine Entscheidung getroffen, die bestimmt, mit welchem Knoten man fortfährt. Die Blätter beinhalten die Klasse, welche als Ausgabe zurückgegeben wird.

5 Random Forests

Das Klassifikationsverfahren Random Forest (RF) ist ebenfalls aus dem Bereich des maschinellen Lernens und wurde 2001 von Leo Breiman bekannt gemacht[1]. Innerhalb eines Lernprozesses werden mehrere Entscheidungsbäume durch Randomisierung generiert. Hierfür wird jedes Mal eine zufällige Teilmenge der Trainingsdaten genutzt. Auch die Eigenschaften, die für die Entscheidungen in den einzelnen Knoten genutzt werden, werden aus einer zufällig generierten Teilmenge gewählt.
Der Random Forest-Algorithmus nutzt die Bagging Methode. Der Begriff Bagging setzt sich aus den Worten Bootstrap und Aggregating zusammen. Dies sind die zwei Schritte aus denen der Algorithmus besteht.

Algorithmus 1: Random Forest [9]

Gegeben: Trainingsdatenset L mit n Beispielen und m Attributen
Schritt 1: Für b=1 bis B

1. Generiere eine Bootstrap Lernstichprobe der Größe n.
2. Erstelle einen Entscheidungsbaum zu den gewählten Beispielen durch Wiederholung folgender Schritte:
   (a) Wähle $m' = \sqrt{m}$ Attribute zufällig.
   (b) Finde den „besten Split“(s. 5.1) aus den m' Attributen.
   (c) Erstelle den Knoten mit dem gewählten Attribut.

Schritt 2: Fasse alle so entstandenen Entscheidungsbäume zu einem Wald zusammen (Aggregation).

Im ersten Schritt wird die Generierung einer Bootstrap Lernstichprobe genannt. In der Statistik bezeichnet die Bootstrap Methode das wiederholte Erstellen neuer Stichproben mit Hilfe einer einzigen Stichprobe. Dabei werden n aus n Beispielen zufällig gewählt (hier aus der Trainingsdatenmenge). Somit können in den generierten Stichproben einige Beispiele mehrfach und einige überhaupt nicht auftreten.[8, S.1142]
Nach Abschluss des Algorithmus wird für die Bestimmung der Klasse eines neuen Objektes die Entscheidung jedes einzelnen Baums des Waldes betrachtet. Das

Objekt wird der Klasse zugeordnet, die von den meisten Entscheidungsbäumen gewählt wurde.

5.1 Wahl des besten Splits

Bei dem oben genannten besten Split, handelt es sich um das Verfahren, welches bei der Erstellung eines Entscheidungsbaum genutzt wird, um das beste Attribut für den aktuellen Knoten auszuwählen. Gegeben sind dabei mehrere Attribute und Beispiele. Das Ziel ist es, das Attribut zu finden, welches die gegebenen Beispiele am besten aufteilt, so dass nach möglichst wenig Schritten im Entscheidungsbaum das Ergebnis fest steht. Bei Random Forests können unterschiedliche Methoden zum Finden des besten Splits verwendet werden. Am häufigsten wird der Gini-Index oder die Entropie genutzt. Es ist auch möglich den besten Split zufällig zu wählen[1, S.5].

5.2 Der Gini-Index

Der Gini-Index (auch Gini-Koeffizient) ist ein statistisches Maß, welches häufig zum Finden der besten Split-Variable genutzt wird.

Definition 6: Gini-Index

Sei T eine Menge von Trainingsdaten mit Beispielen aus k Klassen und p_i die relative Häufigkeit der Klasse i in T, dann ist

$$G(T) = 1 - \sum_{i=1}^{k} p_i^2$$

der **Gini-Index der Trainingsdaten**.

Lässt sich die Trainingsdatenmenge T der Größe N mit Hilfe eines Attributs A in zwei (oder mehr) Teilmengen T_1 und T_2 der Größen N_1 und N_2 aufteilen, dann sei

$$Gini_{split}(T, A) = \frac{N_1}{N} G(T_1) + \frac{N_2}{N} G(T_2)$$

der **Gini-Index für diesen Split**.[13, S.131f]

Zur Generierung eines Entscheidungsbaums wird für jeden neuen Knoten $Gini_{split}(T, A)$ für jedes Attribut A berechnet. Es wird das Attribut als bester Split gewählt, für das $Gini_{split}(T, A)$ den **kleinsten** Wert annimmt.[13, S.132]

5.3 Die Entropie

Ähnlich wie der Gini-Index kann auch die Entropie zur Bestimmung des besten Splits verwendet werden.

Definition 7: Entropie

Sei T eine Menge von Trainingsdaten mit Beispielen aus k Klassen und p_i die relative Häufigkeit der Klasse i in T, dann ist

$$E(T) = -\sum_{i=1}^{k} p_i \log_2 p_i$$

die **Entropie der Klassen** in T.

Lässt sich die Trainingsdatenmenge T der Größe N mit Hilfe eines Attributs A in zwei (oder mehr) Teilmengen T_1 und T_2 der Größen N_1 und N_2 aufteilen, dann sei

$$Ent_{split}(T, A) = \frac{T_1}{N} E(T_1) + \frac{T_2}{N} E(T_2)$$

die **Entropie für diesen Split**.[6, S.89]

Wie auch schon beim Gini-Index muss auch hier $Ent_{split}(T, A)$ für jedes Attribut A ausgewertet werden. Als bester Split wird das Attribut gewählt, bei dem $Ent_{split}(T, A)$ den kleinsten Wert annimmt. [6, S.89]

6 Weitere Klassifikationsverfahren aus dem Bereich des maschinellen Lernens

Die am häufigsten genannten Klassifikationverfahren zur Erkennung veränderlicher Sterne sind neben den Random Forests die Support Vector Machine und der k-Nearest-Neighbors-Algorithmus.

6.1 Support Vector Machine (SVM)

Eine Support Vector Machine ist ein Klassifikationsverfahren aus dem Bereich des maschinellen Lernens. Anhand einer Menge an Trainingsobjekten, deren Klassen bekannt sind, wird eine Trennfläche (Hyperebene) bestimmt, welche die gegebenen Objekte in zwei Klassen einteilt. Dabei wird die Entfernung der Objekte, die am nächsten an der Hyperebene liegen, möglichst groß gehalten. Mit Hilfe der bestimmten Hyperebene können dann neue Objekte in die zwei Klassen eingeteilt werden.
Falls die zu analysierenden Daten linear trennbar sind, kann auch die Hyperebene linear bestimmt werden. Dies ist jedoch bei den meisten Daten nicht möglich. In diesen Fällen wird eine Kernel-Funktion genutzt, um die Hyperebene dennoch bestimmen zu können. Mit Hilfe dieser Funktion können die Trainingsdaten in einen höherdimensionalen Raum projiziert werden. Wenn die Kernel-Funktion gut gewählt wurden, lässt sich in diesem Raum eine lineare

Hypereben bestimmen. Anschließend kann die Hyperbene wieder in den Urspungsraum zurückprojiziert werden.
Abbildung 4 verdeutlicht dieses Vorgehen an einem Beispiel. Der erste Graph zeigt die Trainingsdatenmenge, die in einem eindimensionalen Raum liegt und sich nicht durch eine lineare Hyperbene trennen lässt[1]. Wenn man diese Daten in einen zweidimensionalen Raum projiziert, indem man die Werte von „Expression“ quadriert, lässt sich die Hyperebene bestimmen. [12, S.1567]

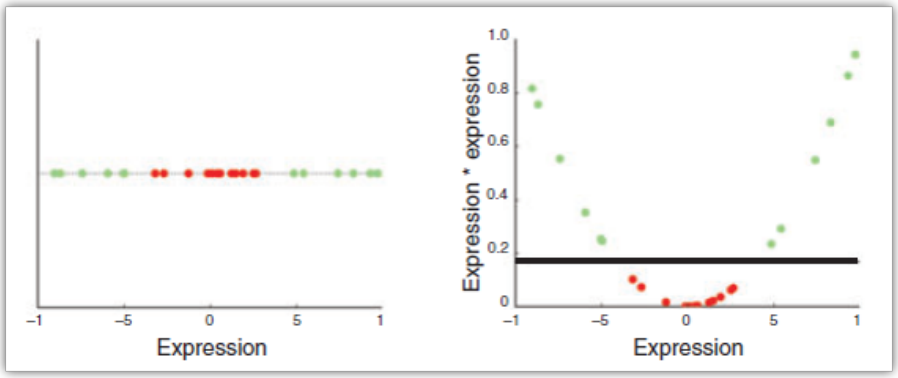

Abbildung 4: Anwendung einer Kernel-Funktion (aus [12, S.1566])

6.2 k-Nearest-Neighbors (k-NN)

Der k-Nearest-Neighbors-Algorithmus ist eine simple, aber sehr häufig eingesetzt Methode zur Klassifizierung. Er benötigt eine Menge an Trainingsobjekten, deren Klassenzuordnung bekannt ist. Für ein zu klassifizierendes Objekt werden anhand eines beliebigen Abstandsmaßes (z.B. Euklidischer Abstand) die k nächsten (ähnlichsten) Beobachtungen zu dem neuen Objekt bestimmt. Das Objekt wird der Klasse zugeordnet, der die meisten von diesen Beobachtungen angehören. [11]

7 Bewertung und Vergleich der Methoden

Jedes der genannten Verfahren (RF, SVM, k-NN) ist geeignet zur Klassifikation veränderlicher Sterne. Jedoch unterschieden sie sich in einigen Punkten und haben ihre eigenen Vor- und Nachteile.

7.1 Vor- und Nachteile der einzelnen Methoden

Das Random Forest Verfahren eignet sich besonders aufgrund der geringen Laufzeit. Da jeder Baum einzeln ausgewertet wird, lässt sich die Evaluierung paral-

[1] In diesem Fall wäre die Hypereben nur ein Punkt.

lelisieren. Dadurch ist dieser Algorithmus sehr effizient auf großen Datenmengen [9, S.587]. Durch die Nutzung randomisierter Teilmengen der Trainingsdaten und Attribute kann in vielen Fällen ein Overfitting[2] vermieden werden [9, S.596]. Für die Verwendung von Random Forest muss jedoch zunächst die optimale Anzahl der zu nutzenden Bäume sowie die Methode zur Wahl des besten Splits bestimmt werden.

Die Support Vector Machine ist eine sehr mächtige und flexible Methode. Da der Kernel frei gewählt werden kann, lassen sich hiermit viele unterschiedliche Formen von Daten verabeiten (z.B. diskrete Daten, Graphen)[2]. Die Bestimmung der Hyperbene benötigt in den meisten Fällen nur wenige Support Vektoren. Daher ist die Anwendung schnell möglich. Zusätzlich ermöglicht sie das Arbeiten in hohen Dimensionen. Jedoch muss für den dazu verwendeten Kernel-Trick ein passender Kernel gewählt werden. Ein Nachteil bei der Verwendung von SVMs ist die hohe Laufzeit. Je höher die Dimension ist, in der man arbeitet, umso mehr Kandidaten für die Hypereben gibt es. Dies wirkt sich enorm auf die Laufzeit aus. Mit $\mathcal{O}(n^3)$ ist diese zu groß, um zum Beispiel effizient auf Datenströmen arbeiten zu können. [2, S.56]

K-NN ist von den genannten Algorithmen der simpelste und daher am einfachsten zu implementieren. Die Laufzeit hängt lediglich von der Größe der Trainingsmenge, des gewählten k und dem Suchalgorithmus zur Bestimmung der k nächsten Nachbarn ab. Somit ist bei geeigneter Wahl dieser Parameter die Laufzeit vergleichsweise kurz und die Genauigkeit laut [11, S.916] ziemlich hoch. Ein Nachteil des k-NN-Algorithmus ist jedoch, dass es keine Vorgabe zur Gewichtung der einzelnen Attribute gibt. Es kann somit passieren, dass die Distanz durch irrelevante Attribute beeinflusst wird und die wichtigeren Attribute zu wenig Einfluss auf das Ergebnis nehmen können[11, S.921]. Diese Problematik wird bei den Random Forests durch das Finden des besten Splits gelöst.

7.2 Vergleich der Methoden in der Anwendung

Eine Kombination aus k-NN und SVM wurde in dem OGLE-Projekt zur Bestimmung veränderlicher Sterne angewandt. Als Attribute wurden die Periode, die durchschnittliche Helligkeit und die Farbe verwendet. Die zu bestimmenden Klassen waren Cephei-, RR-Lyrae- und Doppelsterne. Außerdem wurde auch das Phasendiagramm der Lichtkurve bei der Auswertung von der SVM betrachtet. Mit dieser Methode wurden fast 99% der Daten korrekt klassifiziert. [10, S.32]

In der Hipparcos Untersuchung wurde als Klassifikationsverfahren Random Forests gewählt. Das Trainingsdatenset bestand aus 2000 veränderlichen Sternen

[2] Als Overfitting bezeichnet man ein Phänomen beim Lernen mit einer Trainingsdatenmenge. Hierbei kann es vorkommen, dass das gelernte Modell sehr gut auf die Trainingsdatenmenge abgestimmt ist und diese Beispiele immer korrekt klassifiziert. Neue Objekte werden jedoch häufig falsch klassifiziert. Die Ursache hierfür ist, dass das gelernte Modell zu stark spezialisiert ist und sich zu sehr auf einzelne Details fokussiert.[3]

und 26 Klassen. Es wurde dabei herausgefunden, dass die wichtigsten Attribute die Periode, die Amplitude, die Farbe und die Lichtkurve sind [10, S.32]. Insgesamt wurden weniger als 10% der Sterne falsch klassifiziert [4, S.15]. Außerdem wurde festgestellt, dass der Hauptgrund einer Falschklassifizierung mit der Bestimmung der Periode zusammenhängt. Wenn diese schon aus der Lichtkurve falsch bestimmt wurde, wird die endgültige Klassifizierung in den meisten Fällen auch verfälscht sein [10, S.32].

8 Fazit

Die modernen Teleskope in der Astrophysik generieren derart große Datenmengen, dass sich diese nicht mehr von Hand auswerten lassen. Zur Klassifikation von veränderlichen Sternen werden somit immer häufiger Algorithmen aus dem Bereich des maschinellen Lernens eingesetzt. Die drei vorgestellten Methoden RF, SVM und KNN sind alle für diese Problematik geeignet. Ein genauer Vergleich dieser Methoden gestaltet sich schwer, da sie alle von unterschiedlichen Parametern abhängig sind (u.a. Wahl des besten Splits, des Kernels, der Variable k), welche große Einwirkungen auf die Laufzeit sowie die Korrektheit der Ergebnisse haben. Bei geeigneter Wahl der Parameter weist jeder dieser Algorithmen sehr gute Klassifikationsraten auf (Fehlerrate kleiner als 10%). Der Algorithmus Random Forests sticht besonders heraus, da die Möglichkeit zur Parallelisierung große Auswirkungen auf die Laufzeit hat.

Literatur

[1] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, October 2001.

[2] Colin Campbell and Yiming Ying. Learning with support vector machines. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 5(1):1–95, February 2011.

[3] Tom Dietterich. Overfitting and undercomputing in machine learning. *ACM Comput. Surv.*, 27(3):326–327, September 1995.

[4] P. Dubath, L. Rimoldini, M. Süveges, J. Blomme, M. López, L. M. Sarro, J. De Ridder, J. Cuypers, L. Guy, I. Lecoeur, K. Nienartowicz, A. Jan, M. Beck, N. Mowlavi, P. De Cat, T. Lebzelter, and L. Eyer. Random forest automated supervised classification of Hipparcos periodic variable stars. *Monthly Notices of the Royal Astronomical Society*, 414(3):2602–2617, July 2011.

[5] Laurent Eyer and Nami Mowlavi. Variable stars across the observational hr diagram. *Journal of Physics: Conference Series*, 118(1):012010, August 2008.

[6] Usama M. Fayyad and Keki B. Irani. On the handling of continuous-valued attributes in decision tree generation. *Machine Learning*, 8(1):87–102, January 1992.

[7] Eric D. Feigelson and G. Jogesh Babu. Big data in astronomy. *Significance*, 9(4):22–25, August 2012.

[8] Gary L. Grunkemeier and YingXing Wu. Bootstrap resampling methods: something for nothing? *The Annals of Thoracic Surgery*, 77(4):1142–1144, July 2004.

[9] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction*. Springer, New York, 2 edition, 2009.

[10] P. Huijse, P. A. Estevez, P. Protopapas, J. C. Principe, and P. Zegers. Computational intelligence challenges and applications on large-scale astronomical time series databases. *IEEE Computational Intelligence Magazine*, 9(3):27–39, August 2014.

[11] LiLi Li, YanXia Zhang, and YongHeng Zhao. k-nearest neighbors for automated classification of celestial objects. *Science in China Series G: Physics, Mechanics and Astronomy*, 51(7):916–922, July 2008.

[12] William S. Noble. What is a support vector machine?. *Nature Biotechnology*, 24(12):1565–1567, December 2006.

[13] Leo Odongo and Eunice Muchai. Comparison of crisp and fuzzy classification trees using gini index impurity measure on simulated data. *European Scientific Journal*, 10(18):130–134, June 2014.

[14] John R. Percy. *Understanding Variable Stars*. Cambridge University Press, Cambridge UK, 2007.