

# Maschinelles Lernen in der Diagnostik einer Autismus-Spektrum-Störung bei Erwachsenen

Andreas Zinkl  
OTH Regensburg  
Regensburg, Deutschland  
andreas.zinkl@st.oth-regensburg.de

**Zusammenfassung**—In dieser Arbeit wird der Einsatz von Algorithmen aus dem Bereich des maschinellen Lernens als Hinweis für eine Diagnose einer Autismus-Spektrum-Störung bei Erwachsenen untersucht. Die in dieser Arbeit verwendeten Open Source Daten enthalten dabei Resultate aus einer Bewertung des Verhaltens, basierend auf den DSM<sup>1</sup>-5 Diagnosekriterien. Neben diesen Antworten enthält der Datensatz weitere individuelle Informationen zu den befragten Personen. Im Rahmen der Arbeit werden dabei mit fast allen Algorithmen sehr gute Resultate mit einer Genauigkeit von bis zu ca. 99% erreicht.

**Schlüsselwörter**—Maschinelles Lernen, Autismus, Autismus-Spektrum-Störung, Diagnose, DSM-5

## I. EINFÜHRUNG

Die Anzahl diagnostizierter Autismus-Spektrum-Störung (ASS) steigt nach WEINTRAUB [1] jährlich stetig an. Diese Diagnosen basieren dabei auf Diagnosekriterien aus einem Katalog von Verhaltens- und Interessensmustern [1, 5, 6, 3]. Bei der Diagnose einer ASS werden statistische Diagnosekriterien, basierend auf dem DSM<sup>1</sup>-IV, DSM<sup>1</sup>-5 und ICD<sup>2</sup>-10, zur Detektion verwendet [5, 3]. Die Bewertung dieser Diagnosekriterien unterliegt dabei dem behandelnden Facharzt und hängt somit auch von dessen persönlicher Einschätzung und Erfahrung ab [3]. Der Ablauf einer Diagnose entspricht dabei einer, aus dem Bereich des maschinellen Lernens und der Biometrie bekannten Vorgehensweise zur Klassifikation von Verhaltensmustern und bietet somit eine Möglichkeit zur Anwendung von Algorithmen aus diesem Bereich an.

## II. PROBLEMBESCHREIBUNG

Der für dieses Projekt vorliegende Datensatz wurde bereits von THABTAH [5, 6] zur Erstellung eines Konzepts für maschinelle Lernverfahren für die Autismus-Diagnostik verwendet. In diesem Projekt sollen nun konkrete Algorithmen, anhand der Erkenntnisse von THABTAH, als Hinweis zur Diagnose einer ASS verglichen werden. Der Vergleich der Algorithmen erfolgt dabei über die aus der Biometrie und des maschinellen Lernens bekannten Verfahren zur Analyse von Verhaltensmustern anhand statistischer Verhaltensanalysen.

## III. DATENVORVERARBEITUNG

Zu Beginn der Vorverarbeitung ist es zunächst notwendig den Informationsgehalt des Open Source Datensatzes zu analysieren. Im Anschluss daran können in der Datenaufbereitung fehlende Werte interpoliert und Ausreißer gefiltert werden.

<sup>1</sup>Abkürzung für „Diagnostic and Statistical Manual of Mental Disorders“.

<sup>2</sup>Abkürzung für „International Statistical Classification of Diseases and Related Health Problems“.

## A. Beschreibung des Datensatzes

Der in dieser Arbeit verwendete Datensatz wurde von THABTAH [5, 7], im Zuge seiner Arbeit zur Erstellung eines Konzepts für die Diagnose einer ASS, im Umfang von 704 Testresultaten gesammelt und veröffentlicht. Die Datensammlung basiert dabei auf den von der Organisation NICE [2] beschriebenen Richtlinien zur Diagnose einer ASS mit Hilfe des AQ<sup>3</sup>-10. Eine Beschreibung des Datensatzes liegt der Veröffentlichung bei, enthält jedoch fehlerhafte sowie unzureichende Informationen und Zuordnungen. Hierzu wird im Rahmen dieser Arbeit der veröffentlichte Datensatz ergänzt und in Tabelle I beschrieben.

Tabelle I  
DER AUFBAU DES OPEN SOURCE DATENSATZES

Attribut-Name	Beschreibung
age	Alter in Jahren
gender	Geschlecht männlich / weiblich
ethnicity	Ethnische Herkunft der Person
jaundice	Mit Gelbsucht geboren
autism	Autismus-Diagnose innerhalb der Familie
relation	Person, die das Testverfahren durchführt
country of residence	Land des Wohnsitzes
used app before	Screening-App bereits zuvor benutzt
age desc	Gruppierung des Testverfahrens anhand des Alters
A1	Antwort zur Frage 1 (Trifft zu = 1, sonst = 0)
A2	Antwort zur Frage 2 (Trifft zu = 0, sonst = 1)
A3	Antwort zur Frage 3 (Trifft zu = 0, sonst = 1)
A4	Antwort zur Frage 4 (Trifft zu = 0, sonst = 1)
A5	Antwort zur Frage 5 (Trifft zu = 1, sonst = 0)
A6	Antwort zur Frage 6 (Trifft zu = 0, sonst = 1)
A7	Antwort zur Frage 7 (Trifft zu = 1, sonst = 0)
A8	Antwort zur Frage 8 (Trifft zu = 0, sonst = 1)
A9	Antwort zur Frage 9 (Trifft zu = 0, sonst = 1)
A10	Antwort zur Frage 10 (Trifft zu = 1, sonst = 0)
result	Anhand der Antworten errechnetes Gesamtergebnis
classifiedASD	Mögliche diagnostizierte ASS

## B. Datenanalyse und -aufbereitung

Das Verfahren zur Zuordnung der einzelnen Datensätze zu den Werten in *classifiedASD* wurde nach THABTAH [5] dabei anhand der von NICE [2] beschriebenen Richtlinien zur ASS-Diagnose durchgeführt. Dies führt zur in Abbildung 1 dargestellten Verteilung der Daten anhand der Spalten *result* und *classifiedASD*. Dabei ist erkennbar, dass eine eindeutige Zuordnung bereits mit Hilfe der Spalte *result* durchführbar ist.

Im Verlauf der Arbeit werden jedoch auch weitere Merkmale verwendet, um die Klassifikation anhand der abgegebenen Antworten und Verhaltensweisen der Personen zu untersuchen.

<sup>3</sup>Test zur Berechnung des Autismus-Spektrum-Quotienten.

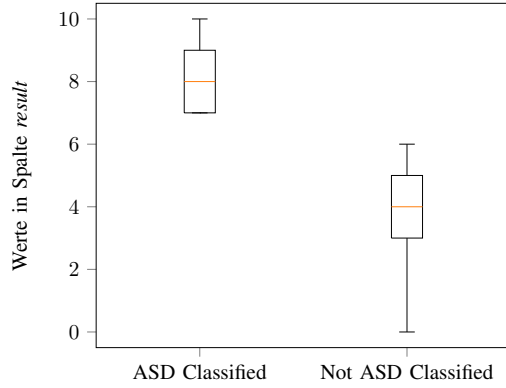


Abbildung 1. Zuordnung der Datensätze in Abhängigkeit der Werte des Attributes „result“

Tabelle II  
INDEX BESCHREIBUNG DES MERKMALSVEKTORS

Index	Attribut
0-9	A1 - A10 (Antworten von Frage 1 bis 10)
10	result
11	age
12	gender
13	jaundice
14	autism
15	relation self
16	relation parent
17	relation healthcare
18	relation relative
19	relation others

Um weitere Merkmale verwenden zu können, wird hierzu eine Datenaufbereitung durchgeführt.

Innerhalb der Datenaufbereitung werden in der ersten Analyse Ausreißer mit fehlerhaften oder fehlenden Informationen herausgefiltert. Dabei werden aufgrund von fehlenden Informationen bei den Attributen *ethnicity*, *relation* und *age*, 95 Datensätze für die weitere Verarbeitung entfernt.

Für eine eindeutige Klassifikation ist es außerdem notwendig, die in den Attributen *jaundice* und *autism* enthaltenen nominalen Werte entsprechend zu normalisieren. Hierbei werden die bool'schen Werte „yes“ und „no“ zu den numerischen Werten 1 für „yes“ und 0 für „no“ abgeändert.

Die ordinalen Werte im Attribut *relation* werden mit Hilfe der „1-aus-n“ (engl. „one-hot“) Kodierung aufbereitet. Dies bedeutet, dass für jeden Wert des Attributes eine neue Spalte erzeugt wird. Dabei wird der Wert in der zugehörigen Spalte auf 1 und in allen anderen Spalten des Attributes auf 0 gesetzt.

Abschließend werden die numerischen Werte  $x$  eines Attributes  $X$  normiert. In der Normierung der Attribute *age* und *result* wird dabei der maximale Wert des Attributes errechnet und der Anteil des aktuellen Wertes an dem maximalen Wert als normierter Wert ermittelt ( $f_{\text{Normierung}}(x) = \frac{x}{\max(X)}$ ). Somit ergibt sich eine Normierung der Werte im Intervall  $[0, 1]$ .

#### IV. MERKMALSEXTRAKTION

Basierend auf der Beschreibung des Open Source Datensatzes und der erfolgten Datenanalyse können zunächst die in Tabelle II dargestellten Merkmale extrahiert werden. Dabei wird in dieser Arbeit ein Merkmalsvektor  $\vec{v}_M$  ermittelt.

Somit ergibt sich aus der Tabelle II ein Merkmalsvektor  $\vec{m} = (m_0, \dots, m_n)^T$  mit  $n = 19$  und insgesamt 20 Merkmalen. Es ist jedoch zu beachten, dass aufgrund der Datenanalyse, das Merkmal *result* die Problemstellung für den Datensatz bereits löst. Aus diesem Grund werden in der Merkmalsextraktion zwei Vektoren generiert. Dies ist der Merkmalsvektor  $\vec{m}_{\text{with-result}}$ , welcher das Merkmal *result* enthält, sowie ein Merkmalsvektor  $\vec{m}_{\text{without-result}}$  der das Merkmal nicht enthält. Dies ermöglicht in der Evaluation einen Vergleich, ob eine Klassifikation ohne das Merkmal *result* mit gleicher Qualität durchgeführt werden kann.

#### V. AUSWAHL DER ALGORITHMEN

Die Auswahl geeigneter Algorithmen fordert zunächst eine Einordnung der Problemstellung. Die Diagnose einer ASS kann dabei nach MÜLLER und GUIDO [4, S. 94] zur Vorhersage einer Klassifikation zugeordnet werden. In dieser Arbeit werden nun unterschiedliche Algorithmen mit Hilfe von überwachtem und unüberwachtem Lernen [4, S. 93] gegenübergestellt. Die in dieser Arbeit gegenübergestellten Algorithmen sind der Entscheidungsbaum (engl. „Decision Tree“), die Support-Vector-Machine (SVM), der K-Nearest-Neighbour sowie der K-Means. Zur Evaluation eines jeden Algorithmus dient die Berechnung der Genauigkeit bzw. Trennschärfe (engl. „Accuracy“ (ACC)).

$$ACC = \frac{\# \text{Richtige Klassifikationen}}{\# \text{Durchgeführte Klassifikationen}} \quad (1)$$

Zudem wird zum näheren Vergleich der Genauigkeit in der Diagnose, die Rate der richtig diagnostizierten Datensätze (engl. „True Positive Rate“ (TPR)) und die Rate der fälschlicherweise diagnostizierten Datensätze (engl. „False Positive Rate“ (FPR)) verwendet.

$$FPR = \frac{\# \text{Autismus-Klassifizierung von } \textit{classifiedASD} = \text{NO}}{\# \text{Datensätze mit } \textit{classifiedASD} = \text{NO}} \quad (2)$$

$$TPR = \frac{\# \text{Autismus-Klassifikationen von } \textit{classifiedASD} = \text{YES}}{\# \text{Datensätze mit } \textit{classifiedASD} = \text{YES}} \quad (3)$$

Für jeden Algorithmus werden dabei 30% der aufbereiteten Daten (182 Datensätze) zum Training und 70% der Daten (427 Datensätze) zur Generierung einer aussagekräftigen Statistik zur Evaluation verwendet. Dabei werden je 30% der positiven und negativen ASS-Diagnosen für das Training verwendet, um eine gleichmäßige Verteilung innerhalb der Trainingsdaten zu erhalten. Die Aufteilung der Trainings- und Testdaten erfolgt dabei zufällig zur Laufzeit des Algorithmus. Die Auswahl der geeigneten Parameter (z.B. der Wahl des Parameter C im Algorithmus der SVM) wird in dieser Arbeit stets mit Hilfe einer Kreuzvalidierung (engl. „Cross-Validation“) durchgeführt.

##### A. Decision Tree

Von der Problemstellung und der anschließenden Datenanalyse ausgehend, ist der Algorithmus des Entscheidungsbaumes (engl. „Decision Tree“) sehr gut geeignet. Der Grund diesbezüglich liegt im Vorgang der Diagnostik. Hierbei werden die Fragen schrittweise abgearbeitet. Die Resultate der Fragen

werden dabei in zwei möglichen Formen „Trifft zu“ und „Trifft nicht zu“ bewertet. Diese Art des Vorgehens ähnelt dabei der Funktionsweise des Entscheidungsbaum-Algorithmus. In dieser Arbeit wird hierzu nun der von *sklearn* implementierte Algorithmus verwendet. Um die Trennschärfe noch weiter zu verbessern wird der Parameter der maximalen Tiefe  $d$  des Entscheidungsbaumes  $T$  definiert. Die maximale Tiefe  $d(T)$  wird dabei für  $n$  Merkmale auf  $d(T) = n - 1$  gesetzt. Dies ermöglicht es, dass für jedes Merkmal maximal eine Entscheidung getroffen wird und führt zur gewünschten Verbesserung der Trennschärfe.

Bereits aus der Datenanalyse (siehe Kapitel III-B) ist ersichtlich, dass die Zuordnung der Klassen anhand des Merkmals *result* eindeutig durchführbar ist. Dies bestätigt der in Abbildung 2 dargestellte Aufbau des Entscheidungsbaumes. Dieser wurde dabei mit Hilfe von 20 Trainingsdaten durch das Framework *sklearn* automatisch generiert. Dabei wählt der Algorithmus ebenso das Merkmal *result* als Entscheidungsmerkmal zur Klassifizierung.

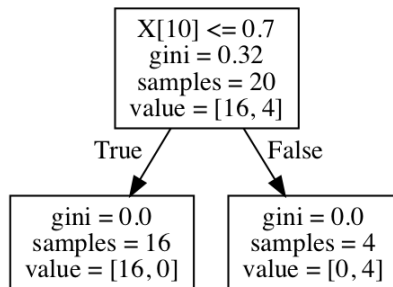


Abbildung 2. Automatisierter Aufbau des Decision Tree durch „sklearn“ durch Prüfung der Merkmale

## B. Support-Vector-Machine

Da es sich bei der Problemstellung um eine Klassifikation von zwei statischen Klassen handelt, ist der Einsatz einer Support-Vector-Machine (SVM) denkbar. Besonders die Dimension der Merkmalsvektoren ist hierbei ein Grund zur Wahl einer SVM. In dieser Arbeit werden dabei der „Radial Basic Function“ (RBF) Kernel sowie ein linearer Kernel verwendet. Die in dieser Arbeit implementierte SVM wird mit Hilfe des Framework *sklearn* umgesetzt. Um eine für die SVM möglichst aussagekräftige Statistik zu erhalten und die bestmöglichen Parameter zu ermitteln, erfolgt die Evaluation mit Hilfe einer Kreuzvalidierung. Zudem werden bei der Verwendung der *sklearn*-SVM die Parameter  $\gamma$  und  $C$  variiert um ein Over- und Underfitting zu vermeiden und die bestmögliche Wahl der Parameter zu identifizieren.

Das Resultat der Wahl der Parameter zeigt hierbei, dass bei einer Wahl des Parameters  $C = 1$  und des zusätzlichen Parameters  $\gamma = 0.0977$  für eine SVM mit RBF Kernel, bereits eine Trennschärfe (engl. „Accuracy“ (ACC)) von ca. 99% erreicht werden kann (siehe Abbildung 3 und 4). Die Wahl des Parameters  $C$  für den linearen Kernel entspricht der Wahl des RBF Kernel und führt zu ähnlich guten Resultaten mit einer Accuracy von 98%.

## C. K-Nearest-Neighbour

Neben der Support-Vector-Machine dient der Algorithmus K-Nearest-Neighbour ebenso gut zur Unterscheidung der zwei

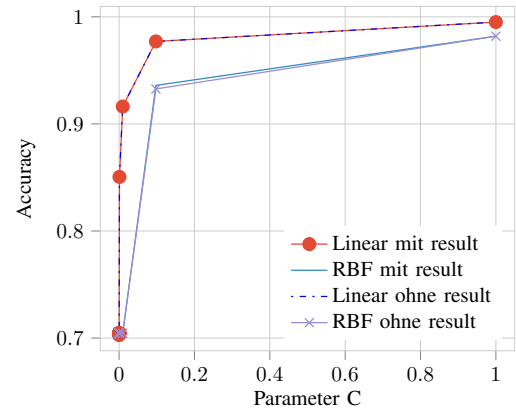


Abbildung 3. Wahl des Parameters  $C$  mit Hilfe einer Kreuzvalidierung für eine SVM mit linearem und RBF Kernel

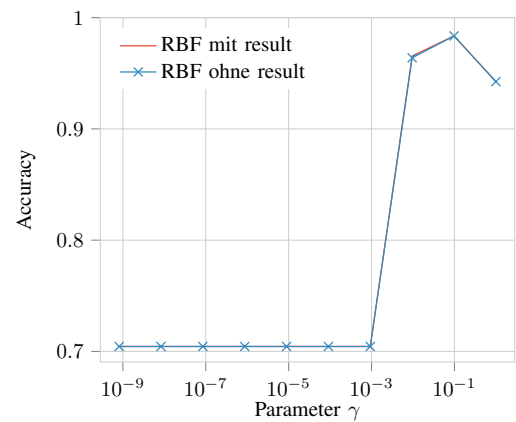


Abbildung 4. Wahl des Parameters  $\gamma$  mit Hilfe einer Kreuzvalidierung für eine SVM mit RBF Kernel

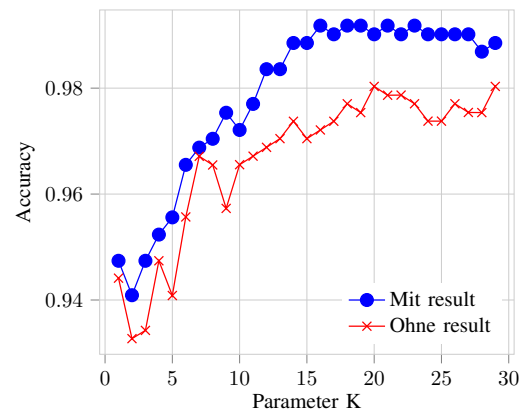


Abbildung 5. Errechnete Fehlerraten unter der Variation der Anzahl der Nachbarn

Klassen einer positiven und negativen Autismus-Diagnose. Im Rahmen der Auswertung des Algorithmus werden hierzu die Anzahl der  $k$  nächsten Nachbarn variiert und eine Kreuzvalidierung der Daten durchgeführt.

Das Ergebnis des Algorithmus zeigt dabei nach Abbildung 5, dass bei einer Anzahl von  $k = 15$  Nachbarn für den

Datensatz mit dem Merkmal *result* eine Trennschärfe von bis zu 99% erreicht werden kann. Im Vergleich dazu erreicht der Algorithmus ohne das Merkmal *result* mit der Wahl  $k = 20$  eine Trennschärfe von 96%. Die bessere Trennschärfe bei der Verwendung des zusätzlichen Merkmals ist dabei auf das Ergebnis aus der Datenanalyse zur Einteilung der Klassifikation anhand der Spalte *result* zurückzuführen.

#### D. K-Means

Der Algorithmus K-Means zählt, im Gegensatz zu den anderen Algorithmen, im Rahmen dieser Arbeit zu den Algorithmen des unüberwachten Lernens und wird in der Regel zum Gruppieren (engl. „Clustering“) von Daten verwendet. Dabei ermittelt der Algorithmus einen Fixpunkt von einer Menge von  $k$  Vektoren. Dieser Fixpunkt entspricht dabei dem Mittelwertvektor  $\bar{k}_m$  der trainierten Vektoren und definiert das Zentrum eines Clusters. Die Evaluation und Klassifikation geschieht hierbei, wie bereits in Kapitel V-C durchgeführt, mit Hilfe einer Kreuzvalidierung. Hierzu wird ebenso die Anzahl der  $k$  Merkmalsvektoren zur Berechnung des Mittelwertvektors  $\bar{k}_m$  variiert. Zur Bewertung der Ergebnisse müssen jedoch die resultierenden Cluster-Zuordnungen des implementierten Algorithmus, unter Verwendung des *sklearn* Frameworks, analysiert werden. Dabei wird innerhalb eines Clusters die Anzahl der positiv und negativ diagnostizierten Datensätze verglichen. Das Cluster wird im Anschluss der Klasse mit der größten Menge von Datensätzen einer Klasse zugeordnet. Das Resultat des Algorithmus zeigt dabei eine gute Trennschärfe von ca. 90%, welche jedoch im Rahmen dieser Arbeit der geringsten Genauigkeit entspricht.

#### E. Evaluation der Algorithmen

Die Ergebnisse der angewandten Algorithmen zeigen für die Problemstellung stets eine gute Trennschärfe (siehe Tabelle III und IV).

Tabelle III

DIE RESULTATE DER ANGEWANDTEN ALGORITHMEN OHNE DAS MERKMAL „RESULT“

Algorithmus	ACC	TPR	FPR
Decision Tree	92.2%	85.0%	14.9%
lineare SVM	95.0%	95.1%	4.8%
RBF SVM	97.8%	95.8%	4.1%
K Nearest Neighbour	96.9%	93.0%	6.9%
K Means	92.9%	78.2%	21.8%

Tabelle IV

DIE RESULTATE DER ANGEWANDTEN ALGORITHMEN INKL. DEM MERKMAL „RESULT“

Algorithmus	ACC	TPR	FPR
Decision Tree	100%	100%	0%
lineare SVM	98.5%	98.2%	1.7%
RBF SVM	98.1%	100%	0%
K Nearest Neighbour	99.1%	96.7%	3.0%
K Means	90.8%	77.1%	22.9%

Ein Vergleich der Algorithmen zeigt hierbei, dass anhand des Merkmals *result* stets eine bessere Trennschärfe erreicht wird. In der Datenanalyse in Kapitel III-B konnte diesbezüglich bereits gezeigt werden, dass anhand des Merkmals *result* die Problemstellung eindeutig gelöst werden kann. Dennoch ist zu

bemerken, dass ein Weglassen des Merkmals kaum zu einer Verschlechterung der Algorithmen führt. Für die Anwendung in der Diagnostik bieten sich vor allem der Algorithmus K-Nearest-Neighbour, sowie die Support Vector Machine zur Verhaltensanalyse an. Diese liefern im Vergleich die beste Trennschärfe unabhängig von der Verwendung des Merkmals *result*. Ist die Klassifikation jedoch wie durch NICE [2], mit einer Verwendung des Attributes *result*, definiert und durchzuführen, so ist zu einer Verwendung des Decision Tree zu raten, da dieser durch die vereinfachte Klassifikation die höchste Trennschärfe bietet.

#### VI. FAZIT UND AUSBLICK

Anhand der Ergebnisse aus Kapitel V-E lässt sich die Möglichkeit des Einsatzes von maschinellen Lernmethoden in der Diagnose einer ASS bestätigen. Es können dabei durch die Anwendung eines geeigneten Algorithmus sehr gute Ergebnisse zur Tendenz einer Diagnose ermittelt werden. Diese können dabei dem behandelnden Arzt eine objektive Sichtweise der Bewertung des Verhaltens ermöglichen. Jedoch ist bei einem Einsatz im Bereich der Diagnose ein Ergebnis der Methoden, aufgrund der teilweise hohen Falschdiagnosen von bis zu 7%, nur als Hinweis zu betrachten. Der verwendete Datensatz von THABTAH [5, 7, 6] enthält zudem bereits normierte Antworten. Somit kann die Gewichtung der Antwort zur Analyse des Verhaltens nicht mit einfließen. Ebenso enthalten die Datensätze eine Klassifizierung nach den Vorgaben von NICE [2], welche die Problemstellung bereits vereinfacht durch die Gesamtpunktzahl der Antworten löst (siehe Kapitel III-B und V-A). Abschließend lässt sich dennoch feststellen, dass der Einsatz von Methoden des maschinellen Lernens mit sehr guten Hinweisen bei der Diagnose einer ASS unterstützen kann.

#### LITERATUR

- [1] Karen Weintraub. „The prevalence puzzle: Autism counts“. In: *Nature* 479.7371 (Nov. 2011), S. 22–24. DOI: 10.1038/479022a.
- [2] National Institute for Health and Care Excellence (NICE). *Autism Spectrum Quotient (AQ-10)*. 2012. (Besucht am 26.04.2018).
- [3] Ludger Tebartz van Elst, Monica Biscaldi-Schäfer und Andreas Riedel. „Autismus-Spektrum-Störungen im DSM-5“. In: *InFo Neurologie & Psychiatrie* 16.4 (2014), S. 50–59. DOI: 10.1007/s15005-014-0005-5.
- [4] Andreas C. Müller und Sarah Guido. *Einführung in Machine Learning mit Python*. 1. Aufl. O'Reilly Media, dpunkt.verlag GmbH, 2017, S. 1000. ISBN: 978-3-96010-112-3.
- [5] Fadi Thabtah. „Autism Spectrum Disorder Screening: Machine Learning Adaptation and DSM-5 Fulfillment“. In: *Proceedings of the 1st International Conference on Medical and Health Informatics 2017*. ICMHI '17. ACM, 2017, S. 1–6. ISBN: 978-1-4503-5224-6.
- [6] Fadi Thabtah. „Machine learning in autistic spectrum disorder behavioral research: A review and ways forward“. In: *Informatics for Health and Social Care* (Feb. 2018), S. 1–20. DOI: 10.1080/17538157.2017.1399132.
- [7] Fadi Thabtah. *ASDTests - A mobile app for ASD screening*. URL: <http://www.asdtests.com/> (besucht am 25.04.2018).