# 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 zur hinreichenden Diagnose von Autismus-Spektrum-Störungen bei Erwachsenen untersucht. Die in dieser Arbeit verwendeten Open Source Daten enthalten dabei die Antworten eines Interviews basierend auf den DSM[1]-5 Diagnosekriterien. Neben diesen Antworten enthält der Datensatz weitere individuelle Informationen zu den befragten Personen. Innerhalb der Arbeit werden dabei mit fast allen Algorithmen sehr gute Resultate für eine hinreichende Diagnose mit einer Genauigkeit von bis zu ca. 99% erreicht.**

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

## I. Einführung

Die Zahl der Diagnosen von Autismus-Spektrum-Störungen (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 von ASS werden statistische Diagnosekriterien, basierend auf dem DSM[1]-IV, DSM[1]-5 und ICD[2]-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. 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, zur Diagnose von ASS verglichen werden. Dabei können die neu gewonnenen Resultate über die Veröffentlichung als Open Source für neue technische Entwicklungen zur Unterstützung von Ärzten in der Diagnostik einer ASS verwendet 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.

[1]Abkürzung für „Diagnostic and Statistical Manual of Mental Disorders“.

[2]Abkürzung für „International Statistical Classification of Diseases and Related Health Problems“.

## 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.

### 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 von ASS mit Hilfe des AQ[3]-10. Eine Beschreibung des Datensatzes liegt der Veröffentlichung bei, enthält jedoch fehlerhafte sowie unzureichende Informationen und Zuordnungen. Hierzu wird im Rahmen der Arbeit der veröffentlichte Datensatz ergänzt und in Tabelle I beschrieben.

| **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 res | Land des Wohnsitzes |
| used app before | Screening-App bereits zuvor benutzt |
| age desc | Gruppierung des Testverfahren 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 Gesamtresultat |
| classifiedASD | Mögliches diagnostiziertes ASS |

Tabelle I
*Der Aufbau des Open Source Datensatzes*

### 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

[3]Test zur Berechnung des Autismus-Sepktrum-Quotienten.

Diagnose durchgeführt. Dieses führt zur in Abbildung 1 dargestellten Verteilung der Daten anhand der Spalten *result* und *classifiedASD*. Dabei ist erkennbar, das die Zuordnung bereits mit Hilfe der Spalte *result* durchführbar ist.

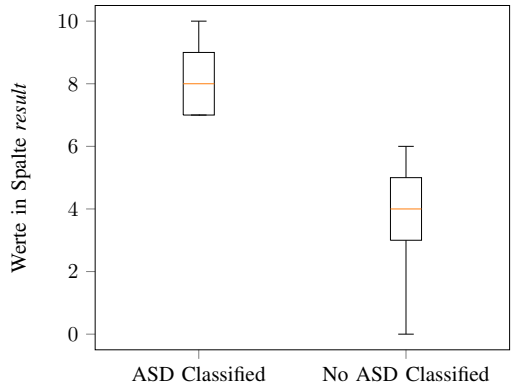


Abbildung 1. *Zuordnung der Datensätze in Abhängigkeit der Werte des Attributes result*

Im Verlauf der Arbeit werden jedoch auch weitere Merkmale verwendet um die Untersuchung der Klassifikation anhand der abgegeben Antworten und Verhaltensweisen der Personen zu entwickeln. Um weitere Merkmale verwenden zu können wird hierzu eine Datenaufbereitung durchgeführt.

Innerhalb der Datenaufbereitung wird in der ersten Analyse eine Filterung von Ausreißern mit fehlerhaften und fehlenden Informationen durchgeführt. 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 nötig 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 innerhalb des Attributes eine neue Spalte erzeugt wird. Dabei wird je Zeile in der dem Wert zugehörigen Spalte der Wert 1 und in allen anderen Spalten des Attributes der Wert 0 eingesetzt. Dies ergibt eine Matrix in der einer Person (repräsentiert durch eine Zeile $i$) ein einziger Wert des Attributes durch eine Markierung der Spalte $j$ mit Hilfe der Funktion $f_{\text{relation}}(i,j) = 1$ zugewiesen wird.

Abschließend werden die numerischen Werte $x$ eines Attributes $X$ („age“ und „result“) normiert. In der Normierung 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 folgende Merkmale extrahiert werden. Dabei wird ein Merkmalsvektor $\vec{v}_M$ ermittelt. Dieser enthält folgende Merkmale:

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

Somit ergibt sich durch die Merkmalsextraktion der Vektor $\vec{m} = (m_1, ..., m_n)^T$ mit $n = 20$ 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 der Merkmalsvektoren $\vec{m}_{\text{with-result}}$, welcher das Merkmal *result* enthält, sowie ein Merkmalsvektor $\vec{m}_{\text{without-result}}$ der das Merkmal nicht enthält. Dies ermöglicht einen Vergleich, ob eine Diagnose 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 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)).

$$\text{ACC} = \frac{\#\text{Richtige Diagnosen}}{\#\text{Durchgeführte Diagnosen}} \tag{1}$$

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.

$$\text{FPR} = \frac{\#\text{Falsche Autismus-Klassifikationen}}{\#\text{Datensätze mit } \textit{classifiedASD} = \text{NO}} \tag{2}$$

$$\text{TPR} = \frac{\#\text{Richtige Autismus-Klassifikationen}}{\#\text{Datensätze mit } \textit{classifiedASD} = \text{YES}} \tag{3}$$

Für jeden Algorithmus werden dabei 30% der aufbereitenden Daten (182 Datensätze) zum Training und 70% der Daten (427 Datensätze) zur Generierung einer aussagekräftigen Statistik zur Evaluation verwendet. Dabei werden jeweils 30% der positiven ASS-Diagnosen 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. Bei der Auswahl der geeigneten Parameter (z.B. die 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.

Bereits in der Datenanalyse ist ersichtlich, dass die Zuordnung der Klassen anhand des Merkmals *result* 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.

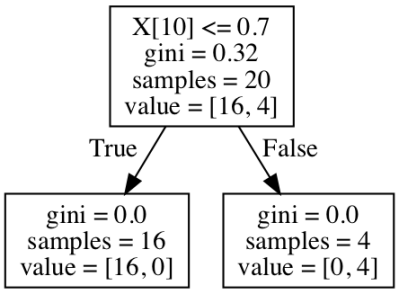


Abbildung 2. *Automatisierter Aufbau des Decision Tree durch sklearn durch Prüfung der Merkmale*

### 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 große 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 bei der Evaluation eine Kreuzvalidierung. Zudem werden bei der Verwendung der *sklearn*-SVM die Parameter $\gamma$ und $C$ variiert um ein Over- und Underfitting zu vermeiden.

Das Resultat zur Wahl der Parameter zeigt hierbei, dass bei einer Wahl des Paramaters $C = 1$ und des zusätzlichen Parameters $\gamma = 0.0977$ für eine SVM mit RBF Kernel, bereits eine Genauigkeit (engl. „Accuracy“ (ACC)) von ca. 99% erreicht werden kann (siehe Abbildung 3 und 4).

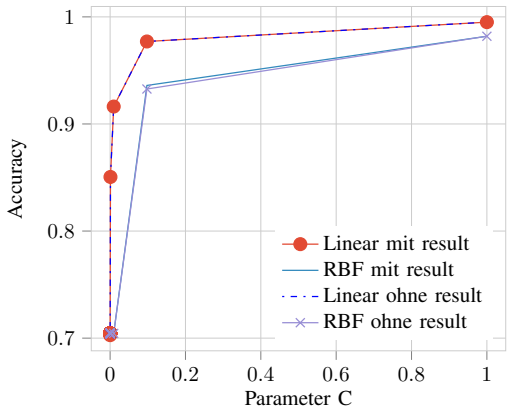


Abbildung 3. *Wahl des Parameter C mit Hilfe einer Kreuzvalidierung für eine SVM mit linearem und RBF Kernel*

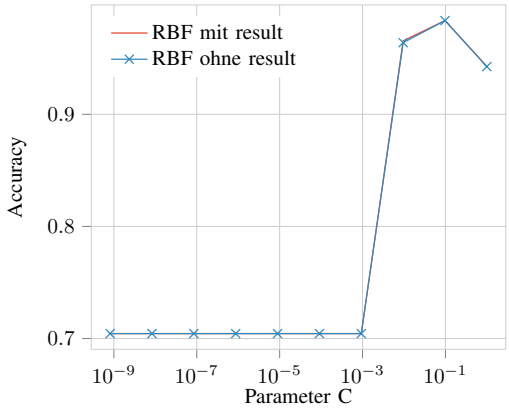


Abbildung 4. *Wahl des Parameter $\gamma$ mit Hilfe einer Kreuzvalidierung für eine SVM mit linearem und RBF Kernel*

### C. K-Nearest-Neighbour

Neben der Support-Vector-Machine dient der Algorithmus K-Nearest-Neighbour ebenso sehr gut zur Unterscheidung der zwei Klassen einer positiven und negativen Autismus Diagnose. Innerhalb 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 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 innerhalb dieser Arbeit zu den Algorithmen des unüberwachten Lernens und wird in der Regel zum Gruppieren (engl. „Clustering“) von Daten verwendet. Dabei ermittelt der

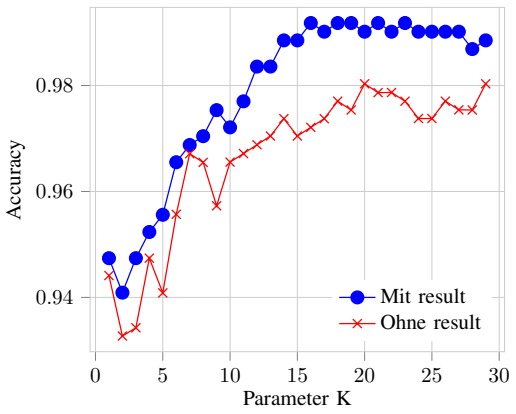


Abbildung 5. *Errechnete Fehlerraten unter der Variation der Anzahl der Nachbarn*

| **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 II

*Die Resultate angewandten Algorithmen ohne das 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% |

Tabelle III

*Die Resultate angewandten Algorithmen inkl. dem Merkmal „result“*

Algorithmus einen Fixpunkt von einer Menge von $k$ Vektoren. Dieser Fixpunkt entspricht dabei dem Mittelwertvektor $\vec{k}_m$ der trainierten Vektoren und definiert das Zentrum eines Clusters. Die Evaluation und Klassifikation geschieht hierbei wie bereits in Abschnitt V-C beschrieben mit Hilfe einer Kreuzvalidierung. Dabei werden die Anzahl der $k$ Merkmalvektoren zur Berechnung des Mittelwertvektors $\vec{k}_m$ variiert. Zur Bewertung der Ergebnisse müssen jedoch die resultierenden Cluster-Zuordnungen des implementierten Algorithmus, unter Verwendung des *sklearn* Frameworks, analysiert werden. Dabei werden innerhalb eines Clusters die Anzahl der positiv und negativ diagnostizieren Datensätze verglichen. Das Cluster wird im Anschluss der Klasse mit der größten Anzahl von Datensätzen innerhalb des Clusters zugeordnet. Das Resultat des Algorithmus zeigt dabei gute Trennschärfe von ca. 90%, welche jedoch innerhalb dieser Arbeit die geringste Genauigkeit bedeutet.

### *E. Evaluation der Algorithmen*

Die Ergebnisse der angewandten Algorithmen zeigen für die Problemstellung stets eine gute Trennschärfe (siehe Tabelle II und III). Der Vergleich der Algorithmen zeigt hierbei, dass anhand des Merkmals *result* stets eine etwas bessere Trennschärfe erreicht wird. In der Datenanalyse in Kapitel III-B konnte hierzu bereits gezeigt werden, dass anhand des Merkmals *result* die Problemstellung eindeutig gelöst werden kann. Dennoch ist zu bemerken, dass ein vermeiden des Merkmals kaum zu einer Verschlechterung der Algorithmen führt. Für die Anwendung in der Diagnose bieten sich somit vor allem der Algorithmus K Nearest Neighbour sowie die Support Vector Machine zur Verhaltensanalyse an. Dieser liefert im Vergleich die bestmögliche Trennschärfe unabhängig von der Verwendung des Merkmals *result*.

## VI. Fazit und Ausblick

Anhand der Ergebnisse aus Kapitel V und V-E lässt sich die Möglichkeit des Einsatzes von maschinellen Lernmethoden zur Diagnose von 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 hohen Falschdiagnosen von bis zu 7%, nur als Hinweis zu bewerten. 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 einbezogen werden. Ebenso enthalten die Datensätze eine Klassifizierung nach dem Vorgaben von NICE [2], welche die Problemstellung bereits vereinfacht durch die Gesamtpunktzahl der Antworten löst (siehe Abschnitt III-B und V-A). Abschließend lässt sich dennoch bestätigen, dass der Einsatz von Methoden des maschinellen Lernens in der Diagnose diese auch positiv 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. New York, NY, USA: 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).