



Datengenerator für Daten mit Bias als Grundlage für Data Science Projekte

Studienarbeit

für die Prüfung zum

Bachelor of Science

des Studiengangs Informatik

an der Dualen Hochschule Baden-Württemberg Stuttgart

von

Simon Jess, Timo Zaoral

Juni 2022

Bearbeitungszeitraum

04.10.2021 - 10.06.2022

Matrikelnummer, Kurs

8268544, 6146532, INF19C

Betreuer

Prof. Dr. Monika Kochanowski

Erklärung

Wir versicherern hiermit, dass wir die vorliegende Studienarbeit mit dem Thema: *Datengenerator für Daten mit Bias als Grundlage für Data Science Projekte* selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben. Wir versichern zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

Stuttgart, Juni 2022

Simon Jess

Timo Zaoral

Abstract

Fasst die Aufgabenstellung und Ergebnisse kompakt und übersichtlich in wenigen Zeilen zusammen (4-7 Zeilen).

Inhaltsverzeichnis

Abkürzungsverzeichnis	V
Abbildungsverzeichnis	VII
Tabellenverzeichnis	VIII
Listings	IX
1 Einleitung	1
1.1 Motivation	2
1.2 Zielsetzung	3
1.3 Aufbau der Arbeit	4
2 Stand der Technik	5
2.1 Daten als wertschöpfende Ressource	5
2.1.1 Daten	5
2.1.2 Datenqualität	7
2.2 Künstliche Intelligenz (KI)	9
2.2.1 Künstliche Intelligenz allgemein	9
2.2.2 Teilgebiet maschinelles Lernen	10
2.2.3 Ethik in der künstlichen Intelligenz	16
2.3 Vorurteile im Zusammenhang mit KI	22
2.3.1 Bias	22
2.3.2 Diskriminierung durch Vorurteile in Daten	26
2.3.3 Gegenmaßnahmen	28
3 Praktischer Teil	31
3.1 Szenarien	31

3.1.1	Szenario zur Bewertung von Bewährungsanträge	31
3.1.2	Szenario zur Vorhersage eines sozialen Punktesystems	32
3.1.3	Vergleich der Szenarien	33
3.2	Konzeption	34
3.2.1	Grobkonzept	34
3.2.2	Feinkonzept	35
3.3	Umsetzung	39
3.3.1	Umsetzung des Szenario zur Bewertung von Bewährungsanträge . .	39
3.3.2	Umsetzung des Szenario zur Vorhersage von einem sozialen Punktesystem	48
3.4	Datenauswertung	59
3.5	Evaluation der Ergebnisse	69
4	Schluss	73
4.1	Zusammenfassung	73
4.2	Ausblick	76
A	Datengenerator für das Szenario zur Bewertung von Bewährungsanträge	81
B	Datengenerator für das Szenario zur Vorhersage eines sozialen Punktesystems	86
C	Tableau für das Szenario zur Bewertung von Bewährungsanträge	94
D	Tableau für das Szenario zur Vorhersage eines sozialen Punktesystems	104

Abkürzungsverzeichnis

KI Künstliche Intelligenz

ML Machine Learning

bzw. beziehungsweise

M Männlich

W Weiblich

dt. deutsch

Abbildungsverzeichnis

2.1	Weltweit jährlich anfallende Datenmenge [7]	5
2.2	Big Data 3 V rule [8]	6
2.3	Data Quality Dimensions [8]	7
2.4	Exemplarisch dargestellte Disziplinen von KI[1]	9
2.5	Arten des maschinellen Lernen[1]	11
2.6	Der Prozess von überwachtem Lernen [19]	12
2.7	Veränderung der Performance durch die Datenmengen [20]	13
2.8	Overfitting und Underfitting [1]	14
2.9	Scheinkorrelationen in der Bilderkennung [21]	14
2.10	Gartner Hype Cycle für KI [22]	16
2.11	Ethische und rechtliche Grundlagen für eine KI [3]	18
2.12	Übersicht ethischer Leitlinien und die Abgedeckten Aspekte [27]	20
2.13	Exemplarische Arten von Bias in den drei Teilbereichen Daten, Algorithmen und Mensch [34]	22
2.14	Erkennung von Schwarzen als Gorillas [37]	26
2.15	Microsoft Chat Bot Tay Twitter Kommentare [39]	27
3.1	Programmablaufplan der fünf Hauptschritte zur Generierung der Daten	35
3.2	Verbindungen zwischen den Attributen eines Bewährungsantrages	36
3.3	Verbindungen zwischen den Attributen des zweiten Szenario	37
3.4	Programmablaufplan zur Generierung der Regeln vom Szenario für Bewährungsanträge	44
3.5	Auswertung der Hautfarbe im Zusammenhang mit der Bewertung nach Prozent von der gesamt Anzahl an Daten	60
3.6	Auswertung der Hautfarbe im Zusammenhang mit der Härte der Strafe und der Bewertung nach der gesamt Anzahl an Daten	61

3.7 Auswertung der Hautfarbe im Zusammenhang mit der Laufenden Strafe nach Prozent von der gesamt Anzahl an Daten	63
3.8 Überblick der sozialen Punkteverteilung über den gesamten Datensatz	64
3.9 Auswertung der Anzahl von Personen und durchschnittlicher Punktestand pro Alter	65
3.10 Auswertung der Anzahl von Personen eingegliedert in die politische Orientierung pro Punktestand	66
3.11 Auswertung der Anzahl von Personen eingegliedert in die politische Orientierung und Bildungsabschluss pro Punktestand	68

Tabellenverzeichnis

2.1	Data Quality Dimensions Merkmale	8
3.1	Tabelle für die Auswirkung der Attributen des ersten Szenario	32
3.2	Tabelle der Attribute und Auswirkungen vom Szenario für das soziale Punktesystem	33
3.3	Tabelle für den Vergleich beider Szenarien	33
3.4	Tabelle zur Bestimmung der Wahrscheinlichkeiten für das Geschlecht . . .	40
3.5	Tabelle der Wahrscheinlichkeiten für die Härte der Strafe nach Geschlecht .	41
3.6	Tabelle zur Bestimmung der Wahrscheinlichkeiten für die Hautfarbe unter Berücksichtigung des Geschlechts	42
3.7	Tabelle zur Bestimmung der Wahrscheinlichkeiten für die Altersgruppen . .	49
3.8	Tabelle der Wahrscheinlichkeiten für die Politische Orientierung nach Alter	50
3.9	Tabelle der Wahrscheinlichkeiten für den Bildungsabschluss nach Alter . .	51
3.10	Tabelle der Wahrscheinlichkeiten für das Soziale Engagement nach Politischer Orientierung	51
3.11	Tabelle der Wahrscheinlichkeiten für die Wohnlage	52
3.12	Tabelle zur Zusammenfassung der Evaluierung der Anforderungen	72

Listings

3.1	Codezeile zur Bestimmung des Geschlechts einer Person nach angegebenen Wahrscheinlichkeiten	40
3.2	Codezeilen zum Erstellen eines Dictionary mit den zur Bewertung relevanten Attributen	42
3.3	Methode zur Initialisierung eines Bewertenden	44
3.4	Methode eines Bewertenden zum Bewerten von Anträgen	45
3.5	Letzte Zelle des Szenario der Bewährungsantrag für die Interaktion des Benutzenden	47
3.6	Codezeile zur Auswahl der Ausprägung der Wohnlage basierend auf angegebenen Wahrscheinlichkeiten	52
3.7	Codeausschnitt zum Hinzufügen eines Dateneintrags zum gesamt Datenset	53
3.8	Codeausschnitt zum Erstellen des Regel Dictionary	54
3.9	Codeausschnitt der Funktion zum Bewerten von Personen	55
3.10	Codeausschnitt für das Hinzufügen einer Verzerrung beim Bewerten der Personen	56
3.11	Letzte Zelle des Szenario des sozialen Punktesystems für die Interaktion des Benutzenden	57

1 | Einleitung

Die fortschreitende Digitalisierung ist kaum noch aus unserem Alltag wegzudenken. Durch immer mehr Programme, die den Alltag erleichtern sollen, nutzen wir die Errungenschaften der Digitalisierung täglich. Häufig ist hier die Rede von künstlicher Intelligenz. Dabei ist uns meist nicht einmal bewusst, dass im Hintergrund mit künstlicher Intelligenz gearbeitet wird. Egal ob als intelligenten Routenplaner oder Sprachsteuerung, hinter all diesen Anwendungen steckt heute nicht mehr nur ein Optimierungsalgorithmus sondern KI.[1] Mit der Digitalisierung hat man begonnen große Datenmengen zu sammeln. Durch den technischen Fortschritt im Bereich von Big Data, werden diese Datenmengen heutzutage unvorstellbar groß. Mit dem Erfassen und Speichern von Daten ist man in der Lage seine Produkte stetig zu verbessern und zudem neue Geschäftsmodelle zu schaffen. Zu diesen neuen Geschäftsmodellen gehört die nicht mehr aus unserem Alltag wegzudenkende KI. Sie ist in der Lage Entscheidungen und Vorhersagen auf Basis von Daten zu treffen, die durch einen Menschen nur mit großem Aufwand getätigt werden können. Egal ob eine Entscheidung oder eine Vorhersage von einer KI getroffen wird, sie basiert auf Daten der Vergangenheit. Aus diesem Grund sind Daten, sobald sie verarbeitet und genutzt werden, eine so wertvolle Ressource.[2]

Für eine KI werden Daten zum Lernen genutzt. Entscheidend für die Qualität der KI ist somit die Datengrundlage auf der die KI basiert. Lernen bedeutet, dass Zusammenhänge und die dadurch abgebildeten Verhaltensweisen in den Daten von der KI erkannt und gelernt werden. Durch diese Art des Lernens, wie auch wir Menschen lernen, ergeben sich jedoch nicht nur Potentiale, sondern auch Risiken. Abhängig von der Datenqualität und Richtigkeit bzw. Zuverlässigkeit der Daten werden zukünftige Entscheidungen und Vorhersagen getroffen. Eine KI betrachtet dabei die Daten vollkommen neutral ohne Hintergrundwissen und ethische Wertvorstellungen. Für manche Entscheidungen gibt es jedoch nicht zwingend richtig oder falsch. Häufig ist es ein schmaler Grat dazwischen. In diesen Fällen wird das menschliche Handeln durch Ethik gesteuert. Eine KI besitzt jedoch keine Ethik und so können Entscheidungen einer KI durch unterschiedliche Ursachen benachteiligend oder gar diskriminierenden sein.[3]

Durch KI öffnen sich viele neue Möglichkeiten und Geschäftsmodelle. Sie wird in immer mehr Bereichen eingesetzt. Doch wenn eine KI vor moralischen Entscheidungen steht sollte man bedenken, dass eine Maschine keine Ethik besitzt. Dies kann zu fatalen Fehlentscheidungen führen und „the dark side of KI“ zum Vorschein bringen.

1.1 Motivation

Mit den Vorteilen der KI kommen immer auch Nachteile. Um die Schattenseite einer KI verstehen zu können, muss man das Thema KI etwas genauer betrachten. Eine KI ist meist ein Instrument zur Vorhersage oder Erkennung. Die Entscheidungen werden durch maschinelles Lernen getroffen. Beim maschinellen Lernen werden, vereinfacht gesagt, Verhaltensweisen und Zusammenhänge in Daten analysiert und diese für zukünftige Entscheidungen als Vorlage genutzt. Die besondere Eigenschaft hierbei ist, dass die Daten, auch Trainingsdaten genannt, Daten aus der Vergangenheit sind. Das Lernen funktioniert ähnlich wie bei uns Menschen, die KI bekommt Trainingsdaten, die zeigen, wie Sie zu entscheiden hat und übernimmt diese Verhaltensweise. Da eine KI auf diese Art und Weise lernt und Entscheidungen trifft, ist naheliegend, dass es wie beim Menschen durch diese Form des Lernens auch ungewünschte Effekte gibt. Bei uns Menschen lernen wir in der Regel von den Eltern. Bei einer KI sind die Eltern die Daten, die Verhaltensweisen beibringen.[4]

Bei der KI und speziell dem Machine Learning (ML) ergeben sich mehrere zu berücksichtigende Probleme. Das häufigste Problem des ML ist das Under- und Overfitting. Dabei wird entweder zu wenig aus den Trainingsdaten gelernt und deshalb willkürlich entschieden oder die Trainingsdaten werden „auswendig“ gelernt und deshalb bei neuen Daten willkürlich entschieden.[1]

Ein unbekanntes Problem von KI und ML ist die Verzerrung in den Trainingsdaten. Wenn Trainingsdaten aufgrund unterschiedlichster Ursachen unerwünschte Zusammenhänge beinhalten, wird von Bias gesprochen. So können zum Beispiel Entscheidungen aufgrund eines unbekannten Zusammenhangs in den Trainingsdaten, häufig auf diskriminierenden Verhaltensmustern oder allgemein Vorurteilen, basieren. Die Problematik liegt darin, dass dem Endnutzer*in in der Regel nicht bekannt ist, dass es einen Bias in den Daten geben kann. In den meisten Fällen ist eine solche Verzerrung verborgen und wird erst im produktiven Betrieb der KI festgestellt.[1]

Diese Verzerrungen führen meist zu Skandalen in der Medienwelt. Es wurde bereits diverse Male in der Presse darüber berichtet, dass beispielsweise in Unternehmen Bewerbungen durch ein KI vorsortiert wurden und dabei Frauen aus nicht nachvollziehbaren Gründen aussortiert wurden. Ein solches diskriminierendes Verhaltensmuster wurde daraufhin in den Trainingsdaten erkannt.[5]

Diese Diskriminierungen sind jedoch nicht zu vergessen immer auf Trainingsdaten und so in der Regel auf reale Daten aus der Vergangenheit zurückzuführen. Das Problem des Bias in Daten ist daher, durch menschliches Verschulden, eine Schattenseite der KI.

1.2 Zielsetzung

KI ist in allen Lebensbereichen vorhanden und auch nicht mehr wegzudenken. Jedoch ist die Schattenseite der KI, den meisten Menschen unbekannt. Dabei spielt die Ethik eine besondere Rolle, denn im Gegensatz zu uns Menschen, verfügt eine KI nicht über ethische Werte und Moral. Häufig spielt die Ethik jedoch in der Entscheidungsfindung eine nicht zu vernachlässigende Rolle. Die Folge aus der fehlenden Ethik bei einer KI kann zu Fehlentscheidungen und fatalen Folgen führen.

Aus diesem Grund soll mehr Bewusstsein für Bias in Daten geschaffen werden. Insbesondere die Entwickler von KI Lösungen müssen für die Thematik mehr sensibilisiert werden, sodass mögliche Benachteiligungen nicht erst in der Praxis festgestellt werden. Dafür soll ein Datengenerator, welcher Daten mit Bias erzeugt entwickelt werden. Um diese Daten in der Lehre einzusetzen zu können soll zusätzlich eine Auswertung entwickelt werden, welche den Bias als Visualisierung veranschaulicht.

Die Umsetzung liegt den folgenden Anforderungen zugrunde:

- Konzeption zweier Szenarien, die realitätsnah sind
- Erstellung eines Datengenerators für zufallsgenerierte Daten
 - Python Script zum generieren eines großen Datensets
 - Flexibilität in der Generierung von Datensätzen
 - Erzeugung von flexibel wählbaren Vorurteilen in den Datensätzen
 - Bewertete und unbewertete Daten zur weiteren Nutzung bereitstellen
- Erstellung einer Auswertung zur Veranschaulichung des Bias
 - Visuelle Auswertung in Tableau
 - Analyse zur Nachvollziehbarkeit des Bias

Ziel ist es, einen Datengenerator für Daten mit Bias zu entwickeln und zusätzlich eine visualisierte Auswertung, die den Bias veranschaulicht. Dieser soll in der Lehre zum Einsatz kommen und für die Thematik von Bias in Daten sensibilisieren.

1.3 Aufbau der Arbeit

Nachdem nun die Motivation und das Ziel der Arbeit gesteckt sind, muss zuallererst zum Thema hingeführt und die Hintergründe erklärt werden. Dies geschieht durch einen Aufbau in drei Passagen, welche alle auf Basis der Literatur erarbeitet wurden. In der ersten Passage wird das Thema Daten als neue wertschöpfende Ressource aufgearbeitet. Daraufhin wird zum Thema KI übergeleitet und wie diese aus den wertvollen Daten Ergebnisse erzielt. Dabei wird zum einen tief auf das Teilgebiet maschinelles Lernen eingegangen und zum anderen auf die Ethik in der KI. Im Teil des maschinellen Lernens, werden die unterschiedlichen Arten und wie diese mit Daten zusammenarbeiten vorgestellt. Daraufhin wird aus der KI als Instanz welche Entscheidungen trifft auf das Thema der Ethik übergeleitet. Dabei wird vor allem behandelt, wie sich die Rolle der Ethik beim Einsatz von KI entwickelt. In der letzten Passage werden die zuvor gesammelten Informationen kombiniert und das Thema behandelt, wie sich Vorurteile und Verzerrungen in den Daten im Zusammenhang mit der KI auswirken. Dies setzt sich daher mit Bias in KI Trainingsdaten auseinander. Dabei liegt der Fokus auf der möglicherweise entstehenden Diskriminierung. Im Gegensatz werden zusätzlich Ansätze und Konzepte von Gegenmaßnahmen betrachtet. Danach ist die dreier Passage abgeschlossen und die Grundlagen zum Thema KI und Daten mit Bias durch die Literatur geklärt.

Im nächsten großen Abschnitt wird die in der Zielsetzung durch Anforderungen definierte praktische Umsetzung des Datengenerators näher betrachtet. Zu Beginn werden dafür die beiden Szenarien erarbeitet und eingeführt. Dabei wird deren Hintergrund aus der Realität und deren Aufbau genauer beschrieben. Zum Abschluss werden noch beide Szenarien in einem kurzen Teilkapitel auf deren wichtigsten Informationen verglichen. Daraufhin werden im nächsten großen Unterkapitel die Anforderungen an die Datengeneratoren in Form eines Konzepts aufgestellt. Dieses unterscheidet sich in ein Grob sowie für jedes Szenario ein Feinkonzept. Anschließend folgt die Beschreibung der Umsetzung des geplanten Grob und Feinkonzepts aufgeteilt in jedes Szenario. Die letzte geforderte Anforderung aus der Zielsetzung wird im nächsten Unterkapitel der Datenauswertung erarbeitet. In diesem werden die erstellten Auswertungen aus Tableau herangezogen und beschrieben. Zum Schluss dieses Abschnitts wird das Ergebnis der Datengeneratoren und der Auswertungen vorgestellt und gegen die Anforderungen evaluiert.

Abschließend folgt im letzten Abschnitt der Arbeit eine Zusammenfassung der gesamten ersten drei Abschnitte und passend dazu eine kritische Reflexion derer. Beendet wird die Arbeit mit einem Ausblick darüber, wie das Umsetzungsergebnis verwendet und erweitert werden kann und welche Relevanz Bias in der KI zukünftig haben könnte. Dabei wird zusätzlich noch ein Literatur Hinweis als kleinen interessanten Ausblick gegeben.

2 | Stand der Technik

In diesem Kapitel wird der Stand der Technik näher beleuchtet. Der Fokus liegt dabei auf den Themen Daten, KI und Bias. Zu Beginn wird auf Basis der Literatur erläutert, was Daten sind, was Datenqualität bedeutet und worum es sich bei einem Bias handelt. Daraufhin wird näher auf KI, das Teilgebiet ML und die Ethik in der KI eingegangen. Zuletzt werden die Themen in einen gemeinsamen Kontext gebracht und der Einfluss eines Bias auf eine KI betrachtet sowie Gegenmaßnahmen untersucht.

2.1 Daten als wertschöpfende Ressource

2.1.1 Daten

Dass Daten eine wertvolle Ressource seien, meinte bereits 2006 der britische Mathematiker Clive Humby mit dem berühmten Zitat: „Data is the new oil“.[6] Hiermit ist gemeint, dass Daten in ihre Rohform nicht sonderlich wertvoll sind, diese jedoch an Wert gewinnen, sobald man beginnt sie zu verarbeiten. Denn lange Zeit waren Daten nur ein Nebenprodukt der Digitalisierung. Daten wurde gesammelt und gespeichert, aber nicht weiterverwendet. Mit dem technologischen Fortschritt im Bereich von Datenanalysen und mit aufkommen der KI wurden Daten von Zeit zu Zeit immer wertvoller. So wurden neue datengetriebene Geschäftsfeld ermöglicht, die einen Mehrwert aus Rohdaten schaffen können. Insbesondere das rasante Aufkommen des Internet of Things hat diese Entwicklung stark vorangetrieben. Seither steigt die Menge der jährlich gesammelten Daten exponentiell an.[2]

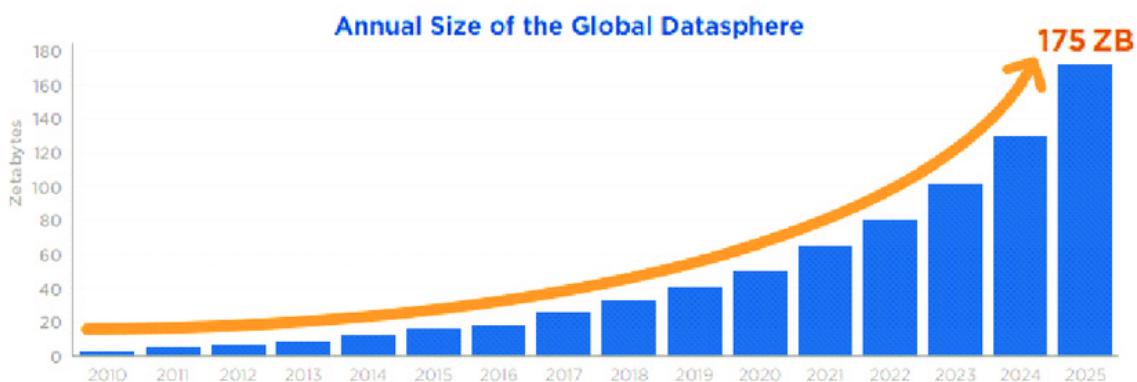


Abbildung 2.1: Weltweit jährlich anfallende Datenmenge [7]

Abbildung 2.1 stammt aus dem Jahr 2018 und verdeutlicht, dass bereits damals erwartet wurde, dass bis im Jahr 2025 rund 175 Zetabyte (10^{21} Byte) Daten jährlich gesammelt werden. Im Vergleich dazu waren es 2018 gerade einmal 33 Zetabyte weltweit.[7][8]

Der Begriff Daten selbst wird im ISO2382-1 Standard wie folgt definiert: „reinterpretierbare Darstellung von Informationen in einer formalisierten Weise, die für die Kommunikation, Interpretation oder Verarbeitung geeignet ist“. [9] Daraus lässt sich ableiten, dass Daten Informationen der Vergangenheit repräsentieren und für zukünftige Verwendung die Informationen aus der Vergangenheit in einer einheitlichen Form repräsentieren. Damit ist jedoch nicht die einheitliche Form der Daten selbst gemeint.

Daten gibt es in unterschiedlichen Formen. Es wird zwischen strukturierten und unstrukturierten Daten unterschieden. Strukturierte Daten sind Datensätze bestehend aus einzelnen Variablen die eindeutige Größen darstellen. Beispiel hierfür sind Sensordaten oder Unternehmenszahlen aus einem ERP System. Sie werden tabellarisch gespeichert und können einfach weiterverarbeitet werden. Oftmals sind diese Daten heterogen, was bedeutet, dass sich die Variablen unterscheiden und beispielsweise Spalte 1 vollkommen andere Daten beinhaltet als Spalte 2. Ein Beispiel hierfür wären Sensoren für Luftfeuchtigkeit und Helligkeit in einem Büro. Als unstrukturierte Daten bezeichnet man Daten, die nicht in sinnvolle einheitliche Variablen unterteilt werden können. Zu dieser Art von Daten zählt man Bilder, Videos, Audio und Textdaten. Sie sind meist homogen, denn die Pixel in einem Bild nehmen zwar unterschiedliche RGB Werte an, jedoch repräsentieren sie alle einen Pixel. Dabei ist es egal ob es sich um Pixel 1 oder Pixel 42 handelt.[1] Bei dieser unvorstellbar großen Datenmenge die jährlich generiert wird, wird davon ausgegangen, dass rund 80% als unstrukturierte Daten vorliegen.[2]

Häufig wird in dem Zusammenhang mit Daten auch von Big Data gesprochen. Eine einheitliche Definition für den Begriff Big Data existiert jedoch nicht. Denn der Begriff Big Data umfasst die gesamte Wertschöpfungskette. Diese beinhaltet die Datenerzeugung, das Sammeln und Speichern der Daten bis hin zur Verarbeitung und Nutzung für Analysen oder Visualisierungen.[8][10] Es handelt sich dabei um Informationen mit hohem Volumen (high-volume), hoher Geschwindigkeit (high-velocity) und hoher Vielfalt (high-variety). Diese drei charakteristischen Eigenschaften werden in der Literatur auch als die „3 V rule“ bezeichnet und ist in den meisten Definitionen wiederfinden.[8][11]

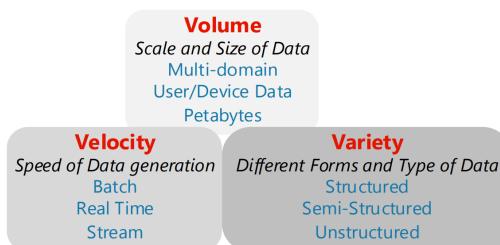


Abbildung 2.2: Big Data 3 V rule [8]

In der Abbildung 2.2 werden diese Eigenschaften, die Big Data mit sich bringt näher beschrieben. Unter „Volume“ wird die große Menge an Daten, also der damit verbundene benötigte Speicher, und ihre Skalierbarkeit betrachtet. Bei der „Velocity“ liegt der Fokus auf der Geschwindigkeit in der die Daten erzeugt werden. Dies hat wiederum je nach Geschwindigkeit der Datenerzeugung Einfluss auf das Volumen. Abschließend in dem Dreieck gibt es die „Variety“. Big Data ist von Natur aus eine Vielfalt an strukturierten und unstrukturierten Daten. All diese drei Eigenschaften beeinflussen sich gegenseitig und bilden die grundlegenden Eigenschaften von Big Data.

In Rohform sind diese Daten, wie eingangs erwähnt, jedoch nicht sonderlich von Wert. Egal ob Big Data oder nicht, einen Mehrwert und Informationen liefern sie erst, sobald man sie nutzt. Dabei ist es egal ob für Simulationen, Monitoring oder KI. In der Vergangenheit sind Daten ein Nebenprodukt der Digitalisierung gewesen. Heute sind sie ein eigenes Geschäftsfeld und „Enabler“ für viele bisher nicht möglich gewesenen Anwendungen.[2][12]

2.1.2 Datenqualität

Im Zusammenhang mit Daten fällt immer häufiger auch der Begriff Datenqualität. Hier trifft Quantität auf Qualität. Wie bereits erwähnt, ist die Menge an Daten die bereits zur Verfügung steht, riesig. Quantität ist daher nicht das Problem. Die Qualität der Daten hat hier jedoch sehr großen Einfluss. Nicht selten können Daten nicht verwendet werden, da die Qualität nicht ausreichend ist. Gerade für Analysen, Auswertungen und Vorhersagen, wie sie durch die KI getroffen werden sollen, wird hohe Datenqualität benötigt.[13] Datenqualität selbst lässt sich auf unterschiedliche Arten und Weisen verstehen.[11] Eine gängige Definition für Datenqualität in der Literatur ist: „fitness for use“.[10] Es bedeutet, dass die Datenqualität von Nutzungskontext und Anwendungsfall abhängt und in erster Linie nicht allgemeine Qualitätsanforderungen erfüllt werden, sondern die für den Use Case benötigten Qualitätsanforderungen.[10][11]

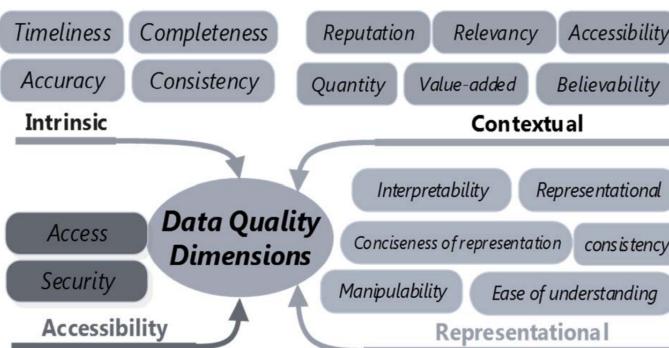


Abbildung 2.3: Data Quality Dimensions [8]

Für die Validierung der Datenqualität gibt es die sogenannten Data Quality Dimensions. In Abbildung 2.3 sind die vier unterschiedlichen Dimensionen: Intrinsic (deutsch (dt.): Intrinsisch), Contextual (dt.: Kontextbezogen), Accessibility (dt.: Zugänglichkeit) und Representational (dt.: Repräsentativ) dargestellt. Jede Dimension besitzt eigene Merkmale, an denen die Datenqualität gemessen werden kann. Näher erläutert werden die Dimensionen in der Tabelle 2.1.

Intrinsic	Zur intrinsischen Datenqualität gehören, wie in Abbildung 2.3 zu sehen, die Merkmale: Zeitlos, Vollständigkeit, Genauigkeit, Konsistenz. Die intrinsische Dimension bildet die Qualitätsmerkmale der Daten selbst ab. und gibt Aufschluss über Objektivität und Glaubwürdigkeit der Daten.[11]
Contextual	Bei der kontextbezogenen Datenqualität wird betrachtet in welchem Ausmaß die Daten für den Nutzenden einsetzbar sind. Darin werden Eigenschaften wie: Datenmenge (Quantität), Zugänglichkeit, Aktualität und Mehrwert der Daten betrachtet um die Nutzbarkeit zu messen.[2]
Representational	Repräsentativ bedeutet im Kontext der Datenqualität, dass die Daten Interpretierbarkeit und dabei primär einfach zu Verstehen sind, Konsistent sind und Manipulierbarkeit, also Veränderbar sind.[13]
Accessibility	Die vierte Dimension, die Zugänglichkeit, bezieht sich insbesondere auf das Speichersystem der Daten. Diese Dimension wird an der möglichst einfachen Zugänglichkeit und der entgegenstehenden Sicherheit der Daten gemessen.[13]

Tabelle 2.1: Data Quality Dimensions Merkmale

Datenqualität lässt sich in unterschiedlichsten Dimensionen und anhand unterschiedlicher Qualitätsmerkmale messen. Die in Abbildung 2.3 dargestellten und in Tabelle 2.1 erläuterten Dimensionen sind dabei nur eines der gängigen Modelle aus der Literatur.

Trotz des Bewusstseins für Datenqualität, besteht rund 80 Prozent der Arbeit eines Data Scientist daraus, Daten aufgrund von Qualitätsanforderungen vorzuverarbeiten.[1] Angefangen beim einfachen Umwandeln bis hin zu komplexeren Bereinigungen und Decodierungen der Daten. Daten werden in den meisten Fällen nicht die gewünschten Qualitätsanforderungen erfüllen können. Datenqualität ist keine verallgemeinerbare Formel, sondern immer abhängig vom Anwendungsfall und dem Kontext, in dem die Daten genutzt werden sollen.

Dass Datenqualität eine wichtige Rolle spielt, unabhängig von dem Verwendungszweck, ist keine Frage. In der Wissenschaft sowie Wirtschaft wird sich immer intensiver mit der Thematik der Datenqualität auseinandersetzt, da heutzutage die Quantität kaum noch ein Problem darstellt, sondern viel mehr die Qualität der Daten.[14] Denn Quantität ist nicht gleich Qualität!

2.2 KI

2.2.1 Künstliche Intelligenz allgemein

KI als Begriff wird vielseitig und unterschiedlich verwendet. Die Vision von KI ist es, die intellektuellen und menschlichen Fähigkeiten als ein KI-System nachzubilden.[14] Letztendlich beschreibt KI ein Forschungsbereich aus der Informatik, der aus einer Vielzahl aus Technologien besteht.[15]

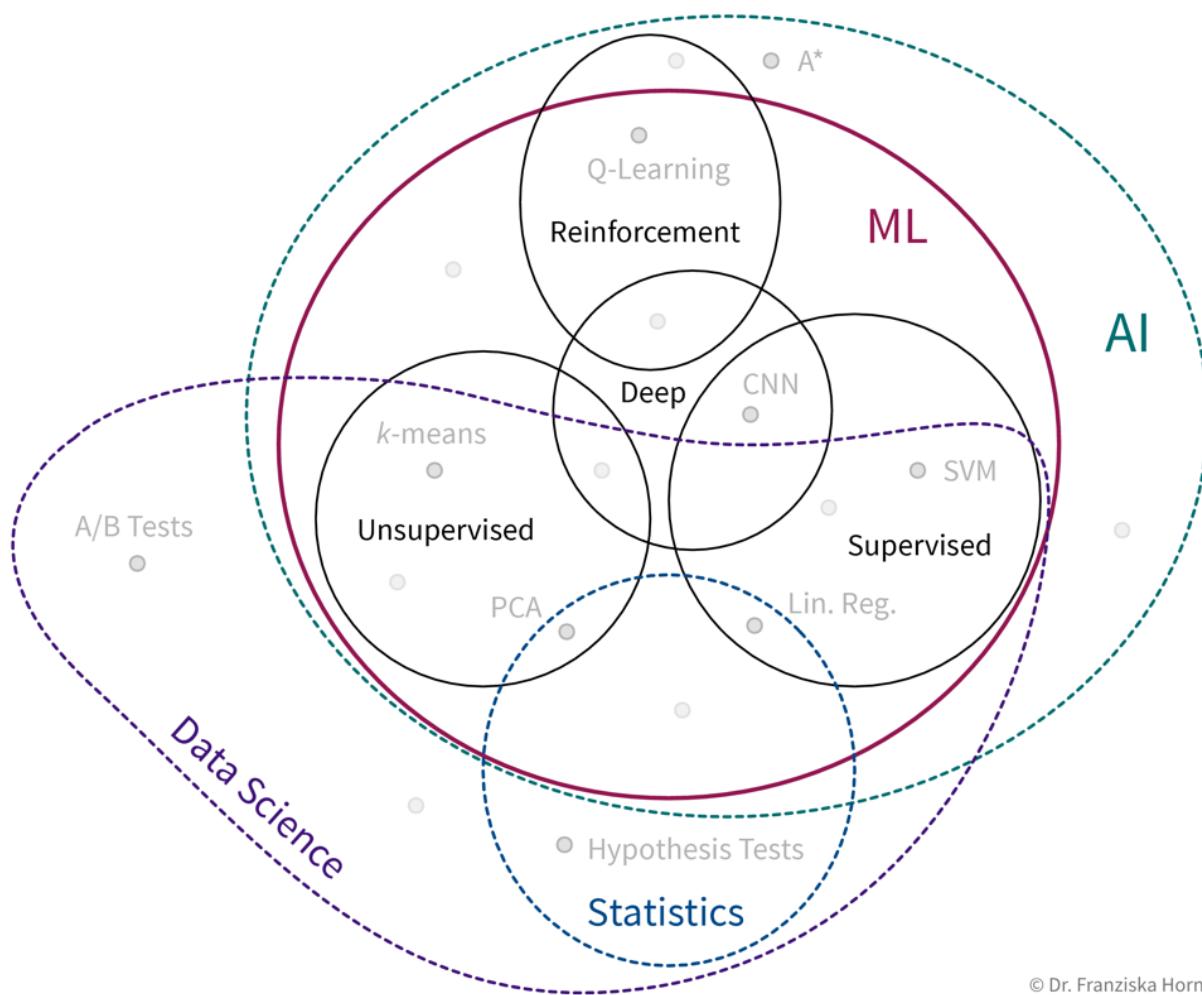


Abbildung 2.4: Exemplarisch dargestellte Disziplinen von KI[1]

Die Abbildung 2.4 zeigt exemplarisch, welche Position KI in der Informatik einnimmt. Zur KI gehören bekannte Teilgebiete wie das ML, Optimierungsalgorithmen, aber auch in der Robotik kommt KI zum Einsatz. Selbst wird die Disziplin aber auch von anderen übergeordneten Disziplinen genutzt.[15] Ein Beispiel hierfür ist Data Science. Die Abbildung 2.4 veranschaulicht, dass einige Aspekte aus der KI im Bereich Data Science aufgegriffen werden, jedoch deutlich mehr als nur KI hinter Data Science steckt.

In der Literatur wurde in der Vergangenheit oft zwischen „schwacher“ und „starker“ KI unterschieden. Eine schwache KI ist dabei für ein spezifisches Anwendungsproblem konzipiert. Der KI wird eine Aufgabe gegeben und diese versucht mittels Algorithmen und mathematischen Funktionen die Aufgabe zu lösen. Diese Art der KI versucht Aufgaben zu lösen und Ergebnisse zu liefern, wie sie durch menschliche Intelligenz entstehen können.[14] Ein Beispiel hierfür ist die Bilderkennung, bei der entschieden wird, ob auf dem Bild eine Erdbeere zu sehen ist, oder nicht.

Als starke KI wird ein System verstanden, welches in der Lage ist ein breites Spektrum an Aufgaben zu erledigen.[16] Sie verfügt über weitere Methoden zur Datenverarbeitung. In dieser Form der KI wird mehr versucht die menschlichen Fähigkeiten und Intelligenz abzubilden. Es wird nicht nach einer spezifischen Vorgehensweise vorgegangen, sondern Aufgaben werden mit individuellen Lösungswegen erledigt.[14]

Mit dem Begriff KI wird demzufolge nur ein grundlegendes Prinzip von KI-Systemen definiert. Die Aufgabe dieser KI-Systeme ist es, Daten jeglicher Form zu interpretieren, Schlussfolgerungen daraus zu ziehen und eine Aussage über das zu erreichende Ziel zu liefern. Unabhängig davon, ob es sich um einen optimalen Lösungsweg, eine Vorhersage oder eine Schlussfolgerung handelt.[15]

2.2.2 Teilgebiet maschinelles Lernen

Maschinelles Lernen ist das Teilgebiet der KI, welches häufig als KI bezeichnet wird. Unter ML wird verstanden, dass ein ML Modell von Lernalgorithmen entwickelt wird, welches in der Lage ist, das Wissen aus Erfahrungen anzuwenden. Dabei soll dem Modell ein unbekannter Input geliefert, dieses verarbeitet die Eingabewerte mit dem Wissen, welches es besitzt und liefert ein Ergebnis als Output zurück. Der Output ist dabei abhängig von der zu lösenden Aufgabe. Klassische Anwendungsfälle von ML sind daher Vorhersagen, Schlussfolgerungen, Optimierungen, Entscheidungen, Sprach- und Bilderkennung beziehungsweise (bzw.) Verarbeitung und weitere ähnliche Aufgaben.[17]

Um das Teilgebiet ML genauer zu betrachten, unterscheidet man in der Literatur zwischen „supervised learning“ (dt.: überwachtes Lernen), „unsupervised learning“ (dt.: unüberwachtes Lernen) und dem „reinforcement learning“ (dt.: bestärktes Lernen).[16] Diese Unterscheidung wird anhand der Art des Lernens getroffen.[1] In Abbildung 2.5, werden diese unterschiedlichen Ansätze des Lernens vereinfacht dargestellt.

Das „unsupervised learning“ (dt.: unüberwachtes Lernen), in der Abbildung 2.5 links dargestellt, basiert auf dem Prinzip, dass dem Modell ein Input ohne zusätzliche Informationen gegeben wird und das Modell selbstständig Zusammenhänge erlernt. Lediglich die zu suchende Struktur ist dem Modell zu liefern, damit ist gemeint, ob nach bspw. Aus-

KAPITEL 2. STAND DER TECHNIK

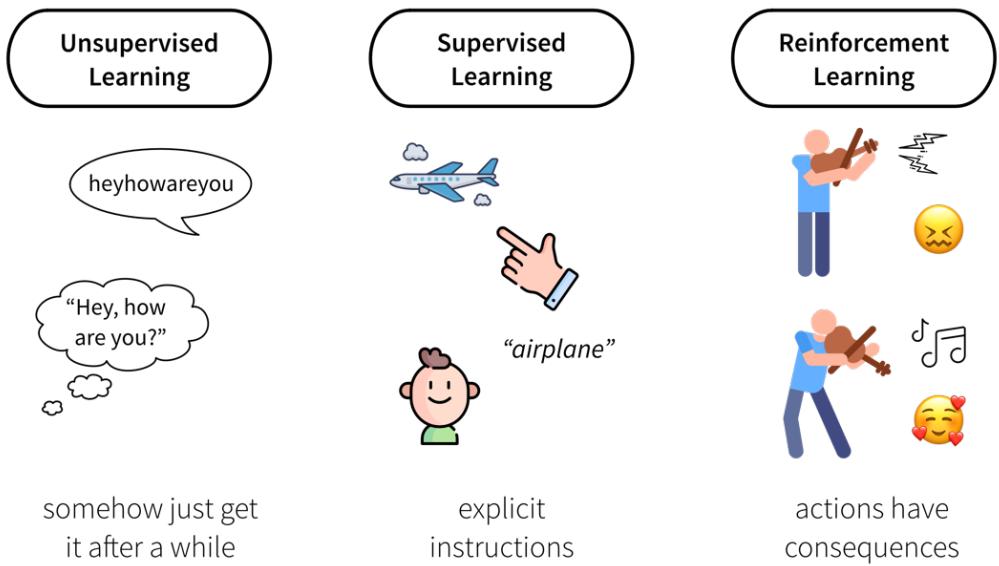


Abbildung 2.5: Arten des maschinellen Lernen[1]

reißen gesucht werden soll. In der Abbildung 2.5, wird dies an dem Beispiel eines Texts veranschaulicht. Dem Modell wird lediglich die Buchstabenkette gegeben und dieses erkennt darin die einzelnen Wörter und, dass es sich um eine Frage handelt. Ein häufiger Anwendungsfall für das unüberwachte Lernen ist das Clustering, da das Modell in den Daten Zusammenhänge erkennt, die ein Mensch nicht direkt sieht.[4][16]

Bei dem „supervised learning“ (dt.: überwachtes Lernen), wird ein sehr Menschen ähnliches Lernverfahren genutzt. Vereinfacht gesagt wird bei dieser Art des Lernens etwas vorgemacht/erklärt und das Modell merkt sich diesen Zusammenhang. Auf diese Art und Weise lernt das Modell Strukturen und kann dies auf ihm unbekannte Daten anwenden. Das Beispiel aus Abbildung 2.5 veranschaulicht diesen Lernverfahren. Hier wird auf ein Flugzeug gezeigt und dem Kind gesagt, dass es sich um ein Flugzeug handelt. Das Kind merkt sich diesen Zusammenhang und ist beim nächsten Mal selbst in der Lage das Flugzeug zu erkennen und es so zu benennen. Die gängigen Anwendungsfelder des überwachten Lernens sind Vorhersagen und Entscheidungen.[4]

Die dritte und bisher am wenigsten verbreitete Art des Lernens ist das „reinforcement learning“ (dt.: bestärktes Lernen). Es handelt sich dabei um Feedback basiertes Lernen. Es bedeutet, dass das Modell eine Auswahl an Handlungen hat, aus denen das Modell frei entscheiden darf, welche ausgeführt wird. Die Entscheidungen werden erst im Nachgang bewertet und in Form einer meist numerischen Bewertung an das Modell zurückgegeben. So lernt das Modell, wenn es etwas gut gemacht hat oder auch falsche Handlungen ausgeführt hat. In Abbildung 2.5, wird dies anhand einer musizierenden Person dargestellt. Die Person spielt etwas auf ihrem Instrument und bekommt nach dem Spielen eines Tons das Feedback der Zuhörenden, ob es sich gut anhört oder nicht. So lernt die Person wann

sich etwas gut anhört und verbessert durch das Feedback ihr können. Häufig findet das bestärkende Lernen deshalb Einsatz in der Robotik, in der ein Roboter anhand von Feedback Bewegungsabläufe erlernt. Bekannt ist hier ein Roboter, der eine menschliche Hand nachbildet und erlernt einhändig einen Rubik's Cube zu lösen.[1][4][18]

Im Rahmen dieser Arbeit wird der Fokus speziell auf das supervised Learning gelegt. Denn überwachtes Lernen kommt meist dann zum Einsatz, wenn Vorhersagen oder Entscheidungen getroffen werden. Dieser Aspekt wird im Verlauf der Arbeit immer wieder eine Rolle spielen. Einzigartig am überwachten Lernen ist, wie bereits erwähnt, dass zum Lernen von Zusammenhängen immer ein „Lehrer“ benötigt wird. Im Falle eines ML Modells handelt es sich dabei nicht um eine Lehrer, sondern um Daten, genauer gesagt Trainingsdaten. Das Prinzip nach dem das überwachte Lernen funktioniert, ist in Abbildung 2.6 exemplarisch anhand eines Beispiels dargestellt.

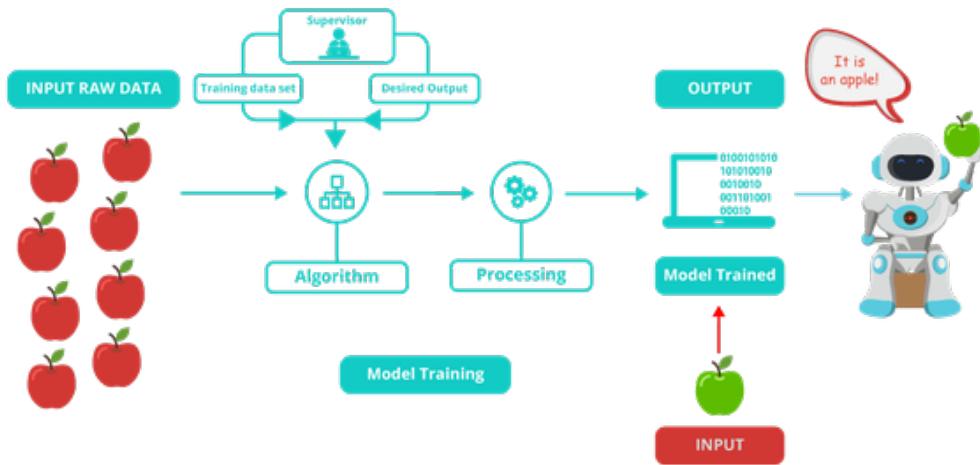


Abbildung 2.6: Der Prozess von überwachtem Lernen [19]

Zu Beginn wird das zu lösende Problem definiert, dabei geht es darum die Lernaufgabe des Modells festzulegen. Beim überwachten Lernen wird zwischen Regressionen und Klassifikation unterschieden. Regressionen kommen meist bei Schlussfolgerungen oder Vorhersagen zum Einsatz und Klassifikationen bei Entscheidungen. Anhand des Problems wird dann ein Algorithmus als Grundlage für das Modell ausgewählt. Um dem Modell nun Wissen beizubringen, werden sogenannte Trainingsdaten verwendet. Dabei handelt es sich um Daten in dem Format, wie sie auch später im Betrieb als Eingaben X geliefert werden. Diese werden jedoch davor mit Labels versehen. Diese Labels werden durch Experten für die Trainingsdaten erstellt. Im nächsten Schritt wird dann das Modell auf Basis der Trainingsdaten trainiert. Diese Phase ist die entscheidende Lernphase des Modells. Im Prinzip ist das Modell nach dem Training bereit für den Einsatz. Meist wird jedoch noch mit einem, dem Modell bisher unbekannten und nicht gelabelten, Datensatz

überprüft, ob das Modell richtig funktioniert und die Qualität anhand unterschiedlicher Metriken, wie der Genauigkeit bewertet. Danach ist das Modell Einsatzbereit und kann mit Eingaben X gespeist werden und schließt dann auf Basis des erlernten Wissens auf eine Ausgabe Y.[1][4] Auf diese Art und Weise werden beim überwachten Lernen die input Output Probleme gelöst.[20]

Es gibt aber auch einige negative Eigenschaften die bei ML Modelle zu berücksichtigen sind und zu möglichen Problemen führen können. Die meisten dieser Risiken lassen sich jedoch inzwischen durch Gegenmaßnahmen verhindern. Zu den häufigsten Problemen gehören zu geringe Datenmengen, die Passgenauigkeit, Scheinkorrelationen und Verzerrungen.

1. Die Datenmenge ist ein entscheidender Faktor im überwachten Lernen. Viele Modelle scheitern daran, dass sie auf Basis von zu wenigen Trainingsdaten trainiert wurden und so nicht in der Lage sind treffende Schlüsse zu ziehen.[16][20]

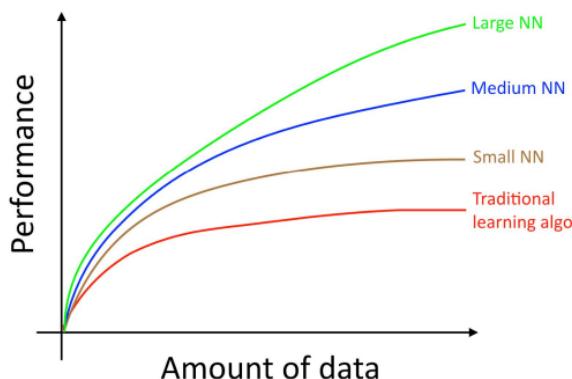


Abbildung 2.7: Veränderung der Performance durch die Datenmengen [20]

In Abbildung 2.7 ist der Zusammenhang zwischen der Datenmenge und der Performance des Modells dargestellt. Hier wird zwar der spezielle Fall von unterschiedlich großen neuronalen Netzen betrachtet, jedoch lässt sich dies auch auf allgemein ML übertragen. Der Verlauf ähnelt dem eines beschränkten Wachstums. Das bedeutet für das Modell, dass es zu Beginn mit einer ausreichenden Datenmenge möglich ist eine gute Performance zu erzielen, jedoch um später wenige Prozente an Verbesserung zu erzielen werden sehr große Datenmengen benötigt.[20] In einigen Szenarien existieren aber von Beginn an zu wenig Daten für eine gute Performance. Als Gegenmaßnahme können lediglich mehr Daten generiert und gesammelt werden.

2. Bei der Passgenauigkeit wird ein Problem bei dem Modell selbst adressiert, nämlich das overfitting (dt.: „überangepasst“) und underfitting (dt.: „unterangepasst“). Vereinfacht gesagt, wird beim overfitting der Trainingsdatensatz durch das Modell zu genau repräsentiert. Erkennbar wird overfitting, wenn man die Performance von

dem Modell bei einem Trainings- und eine Testdatensatz betrachtet. Ist die Performance bei den Trainingsdaten akzeptabel und bei den Testdaten deutlich schlechter, ist es nahezu sicher, dass das Modell zu sehr auf die Trainingsdaten angepasst ist. Beim underfitting hingegen, ist die Performance bei beiden Datensätzen nicht akzeptabel. Das liegt daran, dass beim underfitting zu wenig Zusammenhänge erlernt wurden und so keine realistischen Ergebnisse reproduziert werden können. Diese

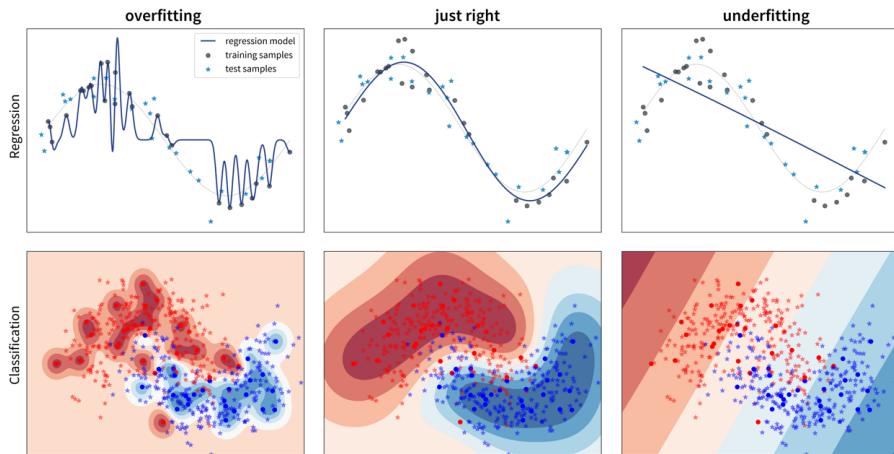


Abbildung 2.8: Overfitting und Underfitting [1]

Problematik wird in Abbildung 2.8 veranschaulicht. Auf der linken Seite ist ein über angepasstes Modell, welches den Trainingsdatensatz „auswendig gelernt“ hat und nicht auf neue Datenpunkte verallgemeinern kann. Rechts ist das unter angepasste Modell zu sehen, welches den Zusammenhang von Input und Output nicht abbilden kann und zu ungenau ist, da es „zu wenig gelernt“ hat.[1]

3. Nicht nur overfitting und underfitting sorgen in Modellen für Ungenauigkeiten auch Scheinkorrelationen sind häufig eine Fehlerursache. Bei Scheinkorrelationen, werden von dem Modell Zusammenhänge erlernt, die keinerlei Zusammenhang mit der tatsächlichen Entscheidung haben.

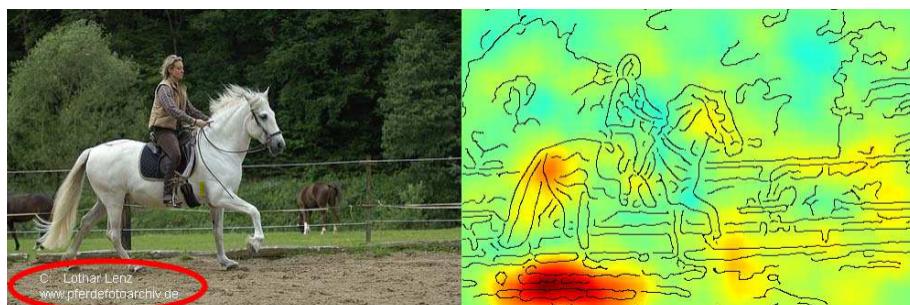


Abbildung 2.9: Scheinkorrelationen in der Bilderkennung [21]

Ein Beispiel dafür ist in Abbildung 2.9 zu sehen. In der ein Pferdebild nicht anhand

der Kontur des Pferdes, sondern anhand des Wasserzeichens in dem Bild erkannt wurde. Grund dafür war, dass einige der Trainingsdaten dieses Wasserzeichen beinhaltet haben und das Modell daraus eine Scheinkorrelation erlernt hat.[1][21][3]

4. Ein letztes und schwerwiegendes Problem ist die Verzerrung von Trainingsdaten, auch Bias genannt. Dieser Aspekt wird später in Kapitel 2.3.1 genauer betrachtet. Allgemein kann man sagen, existieren Verzerrungen in Trainingsdaten, so wird das Modell diese Verzerrungen reproduzieren.[3]

Zusammenfassend wird bei ML, speziell beim überwachten Lernen auf Basis von Daten, Trainingsdaten genannt, Wissen aus der Vergangenheit erlernt. Dazu werden Zusammenhänge in den Daten genutzt und die dazugehörigen Bewertungen in Form von Labels. So entsteht eine Abhängigkeit des Modells von den Trainingsdaten. Im Falle von zu wenig Trainingsdaten, wird das Modell die Realität unterrepräsentieren und daher nicht einsetzbar. Die Daten selbst können aber ebenfalls zu Scheinkorrelationen und Verzerrungen führen, die das Ergebnis des Modells mitunter stark beeinflussen.[3]

2.2.3 Ethik in der künstlichen Intelligenz

Ein Aspekt, der bisher nicht beleuchtet wurde, ist die Ethik. Die Ethik spielt in der KI eine immer größer werdende Rolle. In einer Gesellschaft, die für Inklusion steht, werden häufig Entscheidungen hinterfragt und kritisiert. Dies gilt insbesondere für Entscheidungen und Vorhersagen, die durch KI getroffen werden und so ohne direkten menschlichen Einfluss getroffen werden.

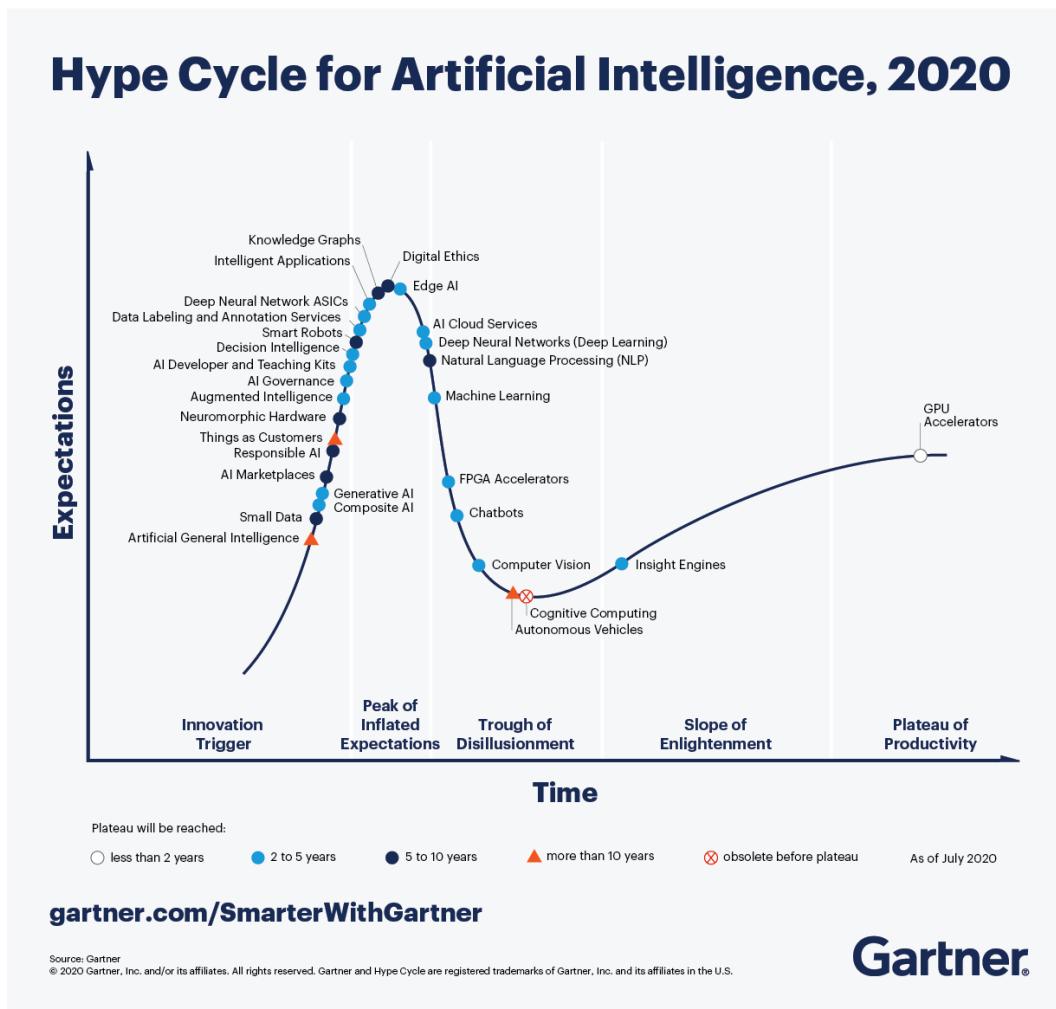


Abbildung 2.10: Gartner Hype Cycle für KI [22]

Im Gartner Hype Cycle für KI aus dem Jahr 2020, dargestellt in Abbildung 2.10, wurde das Thema Digitale Ethik auf dem Höhepunkt der Kurve platziert. Der Forschungsbereich rund um die digitale Ethik befindet sich jedoch erst im Anfangsstadium. Trotzdem sind die Erwartungen bereits sehr hoch, denn auch der gesellschaftliche Druck steigt an. Ursache dafür ist unter anderem der Skandal um den Chatbot Tay von Microsoft der innerhalb von nur einem einzigen Tag rassistische Äußerungen von den Nutzern lernte und daraufhin offline genommen werden musste.[3] Um einen einheitlichen Rahmen für die Ethik in der

KI zu schaffen wurde 2018 von der EU eine Kommission von hochrangigen KI Experten (kurz. HEG KI) gegründet, aber auch Forschungsinstitute setzten sich vermehrt mit der Thematik auseinander. 2021 waren es allein in Deutschland über 40 Projekte.[23]

Grundlegend ist die Aufgabe der Ethik in der KI, dass ein Rahmen geschaffen wird, in dem eine KI agiert und dabei keine ethischen Grundprinzipien verletzt.[3] Ethik ist jedoch nicht einfach in Form von Code programmierbar und kann daher auch nicht von Grund auf in eine KI integriert werden. In den meisten Fällen ist Ethik und ethische Entscheidungen stark von dem Rahmen und Kontext abhängig. Ziel der Ethik ist es sich so zu verhalten und zu handeln, dass keine beteiligte Person in ihren Rechten beschnitten wird oder einen Schaden erleidet.[24] Dabei bezieht sich die Ethik für die Vorgabe von Leitlinien meist auf die Grundrechte, die in Deutschland durch das Grundgesetz definiert werden. In Zeiten der Digitalisierung reichen diese ethischen Grundsätze jedoch oft nicht aus. Mit immer schneller aufkommenden neuen Technologien kommen auch neue Probleme und nicht selten auch ethische Probleme auf. Da die Ethik aber auf historischen Erfahrungen basiert, fehlen diese, um einen klaren Weg vorzugeben. So kommt es, dass die Ethik mehr und mehr eine zentrale Rolle in der Gesellschaft gewinnt, wenn es um den Umgang mit neuen digitalen Technologien geht.[3]

Die KI hat die Frage nach Ethik dabei insbesondere damit geprägt, dass Entscheidungen von einem maschinellen Modell getroffen werden, welche in der Vergangenheit oft nicht transparent und nachvollziehbar waren. Für Viele fehlt daher das Vertrauen in eine Entscheidung, die durch einen Computer getroffen wurde. Um diese Bedenken zu minimieren und ein Vertrauen zu schaffen benötigt man Leitlinien für Ethik in der KI.[24]

Die Ethik wird dabei meist mit dem Konzept von vertrauenswürdiger KI in Verbindung gebracht. In diesem Konzept werden die folgenden drei Anforderungen an eine KI gestellt:

1. Rechtmäßigkeit:

Eine KI soll das geltenden Recht und die gesetzlichen Bestimmungen einhalten.

2. Ethik:

Eine KI soll ethische Grundsätze und Werte einhalten.

3. Robustheit:

Eine KI soll technisch Robust gegen Manipulation gestaltet sein.

Erfüllt eine KI alle drei Anforderungen gilt sie als eine vertrauenswürdige KI. Ziel des Konzepts ist es, die Vorteile von KI zu maximieren aber gleichzeitig die Risiken, egal ob rechtliche oder ethische, möglichst gering zu halten.[17] Dafür muss eine Grundlage auf Basis der drei Kategorien Recht, Ethik und Technologie geschaffen werden, die nur im Zusammenspiel zu einer umsetzbaren vertrauenswürdigen KI führen.

Wenn man sich nun mit dem Konzept einer vertrauenswürdigen KI genauer beschäftigt, müssen Fragestellungen, wie sie in Abbildung 2.11 zu sehen sind, gestellt werden.



Abbildung 2.11: Ethische und rechtliche Grundlagen für eine KI [3]

Die Abbildung 2.11 stellt eine Auswahl konkreter Aspekte, der oben beschriebenen Grundsätze, dar. Es sind die einzelnen Aspekte die für eine vertrauenswürdige KI notwendig sind. Begonnen bei ethischen Grundsätzen wie der Autonomie und Fairness bis hin zu klar rechtlich zuzuordnenden Grundsätzen wie der Sicherheit und dem Datenschutz.[3]

1. Autonomie und Kontrolle

Unter Autonomie und Kontrolle versteht man in Bezug auf die KI die Selbstbestimmtheit. Grundlage der Selbstbestimmtheit ist, dass jedes Individuum Entscheidungen frei und selbstbestimmt treffen kann. Der Konflikt mit der KI entsteht durch die Beeinflussung von Entscheidungen durch Entscheidungsvorschläge oder Vorher sagen. Durch das Vertrauen auf die KI kann es zu einer Beeinträchtigung im Entscheidungsprozess kommen. Die Autonomie fordert deshalb die Selbstbestimmtheit der Nutzenden sowie die Transparenz über Risiken und mögliche Beeinträchtigungen in der Autonomie.[3][24]

2. Fairness

Fairness ist ein grundlegendes Prinzip in der Gesellschaft. Dabei geht es um den Gleichbehandlungsgrundsatz, der besagt, dass es keine ungerechtfertigte Ungleichbehandlung geben darf. Dieses Prinzip muss auch von einer KI erfüllt werden, damit eine vertrauenswürdige Basis einer KI geschaffen werden kann. Es ist darauf zu achten, dass es keine zu Unrecht existierenden Vorurteile oder Diskriminierungen gibt. In diesem Kontext spricht man bei KI von Bias „dt.: Vorurteil“. Ein Bias entsteht durch die Trainingsdaten im ML und sorgt für eine unzulässige Ungleichbehandlung

durch die KI. Das Thema Bias und Diskriminierung durch Bias wird in den Kapiteln 2.3.1 und 2.3.2 nochmals genauer betrachtet.[3][25]

3. Transparenz

Mit der Transparenz ist in erster Linie die Transparenz der KI gemeint. Der Fokus liegt darauf, dass es für ein vertrauenswürdigen Umgang mit KI notwendig ist, nachzuvollziehen wie Entscheidungen getroffen werden. Der Bedarf von Transparenz ist besonders hoch, da Entscheidungen Unfair erscheinen können und eine Entscheidungsgrundlage Klarheit liefern kann. Technologisch handelt es sich jedoch bei den meisten Modellen um sogenannte Black-Box Modelle, welche in ihrer Entscheidung nicht nachvollziehbar sind. Trotzdem wird von einer transparenten KI die Interpretierbarkeit, die Nachverfolgbarkeit sowie die Reproduzierbarkeit des Ergebnisses erwartet.[3][26][24]

4. Verlässlichkeit

Die Verlässlichkeit ist die ein Aspekt der sich mit der tatsächlichen Funktion der KI befasst. Sie umfasst die Korrektheit der Ausgabe aber auch die technologische Robustheit. Für die Korrektheit ist die Implementierung des Modells sowie die verwendeten Trainingsdaten ausschlaggebend. Bei der technologischen Robustheit wird die Anfälligkeit für Fehler und Manipulationen adressiert. Insbesondere Adversarial oder Poison attacks spielen bei der Gewährleistung von einer verlässlichen Funktion ein Risiko dar, da sie das Verhalten einer KI manipulieren können. Die Korrektheit durch Fehler oder Manipulationen gefährdet werden und so auch die Verlässlichkeit der KI. Dabei ist die Verlässlichkeit die Grundlage für den Einsatz von KI und daher sowohl während der Entwicklung aber auch dem produktiven Einsatz zu jedem Zeitpunkt zu gewährleisten.[3][26]

5. Sicherheit

Bei der Sicherheit ist nicht die der KI selbst gemeint, denn diese ist Bestandteil der Verlässlichkeit. Die Sicherheit bezieht sich auf das gesamte System in dem sich die KI und deren Komponenten befinden. Dabei sollen mögliche Schwachstellen für den Missbrauch der KI geschützt werden aber auch allgemein die Sicherheitsanforderungen der gängigen Zertifizierungen wie der ISO Norm 27001 für die allgemeine IT-Sicherheit.[3][27]

6. Datenschutz

Um die Privatsphäre und so das Recht auf informationelle Selbstbestimmung gewährleisten zu können wird Datenschutz benötigt. Auch eine KI muss mit Datenschutzrichtlinien konfrontiert werden. Häufig wird mit sensiblen Daten, egal ob personenbezogene Daten oder Geschäftsgeheimnisse, gearbeitet die es sowohl rechtlich

KAPITEL 2. STAND DER TECHNIK

als auch ethisch betrachtet zu schützen gilt. Die Daten ermöglichen es nämlich selbst in teils anonymisierter Form Rückschlüsse zu ziehen. Eine KI ist daher verpflichtet die rechtlichen Bestimmungen der Datenschutz-Grundverordnung (DSGVO) und des Bundesdatenschutzgesetzes (BDSG) einzuhalten.[3]

Mit den sechs Aspekten die in Abbildung 2.11 aufgeführt sind, werden jedoch nur die grundlegendsten Anforderungen an KI Systeme näher betrachtet. Neben diesen Anforderungen gibt es noch eine Vielzahl weiterer Aspekte die in einem bestimmten Kontext zwingend erforderlich sind.

authors	[Peikka et al. 2018]	[Holdren et al. 2016]	[Beijing Academy of Artificial Intelligence 2019]	[Organisation for Economic Co-operation and Development 2019]	[Brundage et al. 2018]	[Fiorini et al. 2018]	[Future of Life Institute 2017]	[Crawford et al. 2016]	[Campolo et al. 2017]	[Whittaker et al. 2018]	[Crawford et al. 2019]	[Dakopoulos et al.]	[Abrassart et al. 2018]	[OpenAI 2018]	[The IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems 2016]	[The IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems 2019]	[Microsoft Corporation 2019]	[DeepMind 2018]	[Google 2018]	[Cutler et al. 2018]	[Partnership on AI 2018]
key issue	AI principles of the EU	AI principles of the US	AI principles of China	AI principles of the OECD	analysis of abuse scenarios of AI	meta-analysis about principles of beneficial use of AI	large collection of different principles	statements on social implications of AI	principles of the FAT ML community	code of ethics released by the Université de Montréal	several short principles for the ethical use of AI	detailed description of ethical aspects in the context of AI	brief guideline about basic ethical principles for the ethical use of AI	several short principles for the ethical use of AI	IBM's short list of keywords for the ethical use of AI	principles of an association between several industry leaders					
privacy protection																					
fairness, non-discrimination, justice																			18		
accountability																			18		
transparency, openness																			17		
safety, cybersecurity																			16		
common good, sustainability, well-being																			16		
human oversight, control, auditing																			12		
solidarity, inclusion, social cohesion																			11		
explainability, interpretability																			10		
science-policy link																			10		
legislative framework, legal status of AI systems																			9		
future of employment/worker rights																			8		
responsible/intended research funding																			8		
public awareness, education about AI and its risks																			8		
dual-use problem, military, AI arms race																			8		
field-specific deliberations (health, military, mobility etc.)																			7		
human autonomy																			7		
diversity in the field of AI																			4		
certification for AI products																			3		
protection of whistleblowers																			2		
cultural differences in the ethically aligned design of AI systems																			2		
hidden costs (labeling, clickware, content moderation, energy, resources)																			1		
notes on technical implementations	yes, but very few	none	none	none	yes	none	none	none	none	none	none	none	none	none	none	none	none	none	none		
proportion of women among authors (f/m)	(8/10)	(2/3)	ns	ns	(5/21)	(5/8)	ns	(4/2)	(3/1)	(6/4)	(12/4)	(1/12)	(8/10)	ns	varies in each chapter	varies in each chapter	ns	ns	ns		
length (number of words)	16546	22787	76	3249	34017	8609	646	11530	18273	25759	38970	1359	4754	441	40915	108.092	2272	75	417		
affiliation (government, industry, science)	government	government	government	science	science	science	science	science	science	science	science	non-profit	industry	industry	industry	industry	industry	industry	1481		
number of ethical aspects	9	12	13	8	14	12	13	9	12	13	5	11	4	14	19	5	6	6	6		

Abbildung 2.12: Übersicht ethischer Leitlinien und die Abgedeckten Aspekte [27]

In der Abbildung 2.12 werden unterschiedliche Leitlinien für vertrauenswürdige KI verglichen. Für den Vergleich werden die unterschiedlichen betrachteten Aspekte der einzelnen Leitlinien aufgeführt. Die daraus entstehende Tabelle veranschaulicht dann, welche Leitlinie welche Aspekte beinhaltet. Zudem wurden die Aspekte der Häufigkeit nach absteigend sortiert. Das bedeutet, die obenstehenden Aspekte werden am häufigsten in Leitlinien betrachtet. Dabei stellt sich heraus, dass einige bereits erwähnte Aspekte wie die Fairness,

KAPITEL 2. STAND DER TECHNIK

Sicherheit und Verlässlichkeit in nahezu allen Leitlinien ein Bestandteil sind. Sie sind daher auch in 2.12 bei den obersten Punkten dabei. Es wird aber ebenso der Konflikt mit der Erklärbarkeit und der Selbstbestimmtheit bzw. Autonomie veranschaulicht. Denn die Erklärbarkeit wird lediglich in 10 Leitlinien berücksichtigt und die Autonomie in nur 7 von gesamt 22 Leitlinien. Des Weiteren werden neu aufkommende Problematiken wie die Diversität im Umfeld von KI bisher nahezu gar nicht betrachtet.[27][28]

Die Ethik spielt bei dem Einsatz von KI eine immer größer werdende Rolle. Ethik allein reicht jedoch nicht aus um eine vertrauenswürdige KI zu schaffen. Es ist ein Zusammenspiel von ethischen, rechtlichen und technologischen Herausforderungen, die es zu lösen gilt. Um ein einheitliches Verständnis von vertrauenswürdiger KI zu schaffen, werden Zertifizierungen benötigt. Nur so ist es möglich den Einsatz von KI auf einer vertrauenswürdigen Basis zu schaffen. Dabei müssen die grundlegenden Anforderungen wie Fairness, Autonomie, Sicherheit, Verlässlichkeit, Transparenz und Datenschutz immer erfüllt sein. Es kann zusätzlich noch weitere Anforderungen geben, die sich aufgrund der Rahmenbedingungen, dem Kontext oder den verarbeiteten Daten ergeben können.[3][27]

2.3 Vorurteile im Zusammenhang mit KI

2.3.1 Bias

Als Bias versteht man in der Literatur allgemein eine Verzerrung eines Wertes oder die Abweichung von einem Standard.[29] Der Begriff stammt ursprünglich aus dem Englischen und ist auf Deutsch übersetzt gleichbedeutend mit den Worten: Vorurteil, Voreingenommenheit, Neigung und Tendenz. In der KI versteht man unter Bias ebenfalls eine Verzerrung. Dabei bezieht sich die Verzerrung jedoch meist auf die Ausgabe einer KI, also der Ausgabe von ML Algorithmen.[30] Der Bias selbst ist dabei unabhängig von einer positiven oder negativen Verzerrung.[31][32] Diese Verzerrungen durch einen Bias werden in das Konzept der vertrauenswürdigen KI, aus Kapitel 2.2.3, in den Aspekt der Fairness eingeordnet. Fairness selbst, ist ein elementarer Bestandteil der Ethik in der Gesellschaft. Etwas gilt als „fair“, wenn keine unzulässige Ungleichbehandlung stattfindet. Im Umkehrschluss gilt etwas als „unfair“, wenn es eine Gruppe oder Individuen aufgrund von Eigenschaften ungerechtfertigt systematisch Diskriminiert.[31]

Für die Arbeit wird speziell der Bias im ML betrachtet. Schwerpunkt wird dabei auf den negativen Einfluss von Verzerrungen im ML und die Auswirkungen eines Bias auf die Ausgabe gelegt. Ein Bias im ML kann aus unterschiedlichen Gründen zustande kommen. Dabei wird in der Literatur oft eine Unterscheidung in die drei Bereiche Bias durch Daten, Bias durch Algorithmen und Bias durch Menschen gemacht.[33][34]

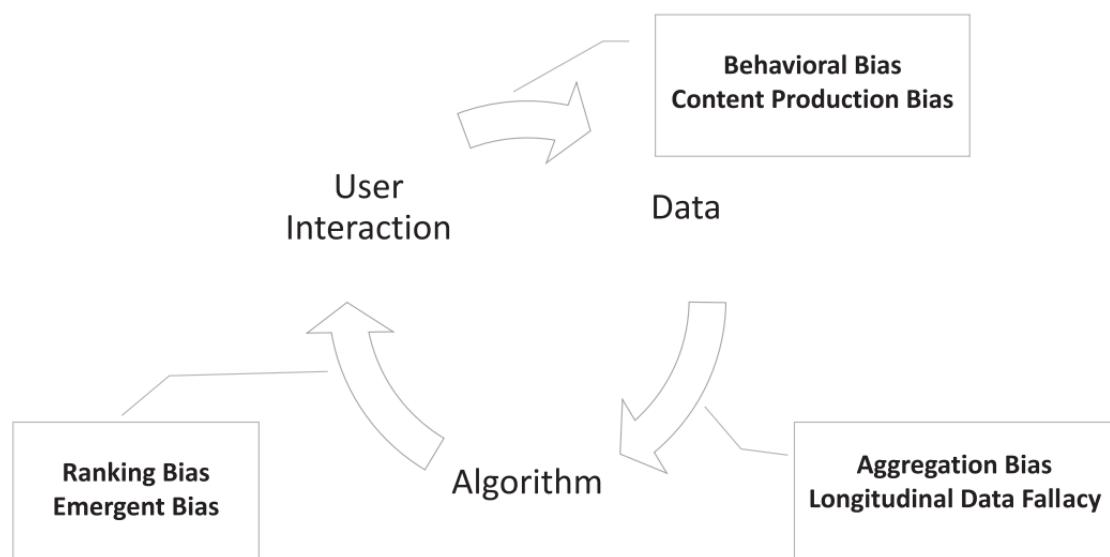


Abbildung 2.13: Exemplarische Arten von Bias in den drei Teilbereichen Daten, Algorithmen und Mensch [34]

Eine eindeutige Unterteilung fin diese drei Bereiche ist jedoch schwer möglich. Meist beeinflussen sich die Bereiche gegenseitig, wie es in der Abbildung 2.13 dargestellt ist. Zum Beispiel eine Verzerrung die durch einen Menschen zustande kommt, wird Einfluss auf die Daten haben, die bei der Entwicklung der KI eine elementare Rolle spielen. Aber Ebenso ist es unter Umständen möglich, dass die KI zuvor, durch seine Ausgabe, einen Einfluss auf das Handeln eines Menschen gehabt hat.[34] Ein Bias ist demnach also entweder in den Daten, einem Algorithmus oder einem Menschen zu verorten. Seine Auswirkungen bzw. seine Verzerrung wird aber erst in dem danach folgenden Bereich sichtbar, wie es in Abbildung 2.13 zu sehen ist. Der Bias uns seine Verzerrung in der KI bilden so eine Kreislauf der in allen Bereichen seinen Einfluss hat.[34] Bias ist aber nur der Überbegriff. In den drei benannten Kategorien gibt es eine Vielzahl an unterschiedlichen Ausprägungen von Bias. Deshalb werden im Folgenden die Kategorien in denen Bias vorkommen und die dazugehörigen spezifische Ausprägungen von Bias genauer untersucht.

Bias durch Daten:

Der Bias durch Daten hat seinen Ursprung in der Funktionsweise von ML. Wie bereits in Kapitel 2.2.2 genauer beschrieben, wird beim ML mit Trainingsdaten gearbeitet. Speziell beim überwachten ML werden historische Daten genutzt, um Zusammenhänge und Strukturen aus Daten zu erlernen. So wird eine Verhaltensweise, meist die eines Menschen, durch ein Modell nachgebildet. Entscheidend für die Funktionalität sind daher die Trainingsdaten. Sie sind auch der Ort an dem sehr häufig ein Bias auftritt.[31][35]

Die wichtigsten Ausprägungen von Bias in Daten sind: Measurement Bias, Sampling Bias und Label Bias[33][34]

Measurement Bias:

Der „measurement bias“ (dt.:Verzerrung der Messung) entsteht durch die Art in der Merkmale ausgewählt, verwendet und ermittelt werden. Ein Beispiel für diese Art der Verzerrung ist, dass für das Kriminalitätsrisiko Bewertung die Verhaftungsrate im Freundes-/Familienkreis herangezogen wurden. So entstand der Trugschluss, dass Minderheitengruppen aufgrund einer höheren Verhaftungsrate gefährlicher seien. Zurückzuführen ist der Bias auf die häufigere Kontrolle von Minderheitengruppen und eine dadurch verursachte höhere Verhaftungsrate.[31][33][34]

Sampling Bias:

Ein „sampling bias“ (dt.: Stichprobenverzerrung) ist eine der häufigsten Arten von Verzerrungen in Datensätzen. Hier entsteht die Verzerrung dadurch, dass eine bestimmte Gruppe wie beispielsweise dunkelhäutigen Personen im Vergleich zu hellhäutigen Personen unter repräsentiert wird oder vollständig fehlt.

Wenn Gruppen im unterrepräsentiert sind oder gar ganz in den Trainingsdaten fehlen, kann dies dazu führen, dass der Algorithmus schlechter generalisiert und so schlechter gegenüber dunkelhäutigen Personen funktioniert. Die Stichprobe in Form der Trainingsdaten ist, wenn ein Sampling Bias enthalten ist aus diesem Grund nicht repräsentativ für die reale Welt und somit auch nicht als Trainingsdatensatz für ML geeignet.[33][34]

Label Bias:

„Label bias“ (dt.: Verzerrung der Beschriftung) bedeutet, dass es eine Verzerrung in der Bewertung der Trainingsdaten gibt. Zurückzuführen ist dieser Bias auf seine Entstehung, genauer auf die bewertenden Personen. Die Verzerrung wird durch die individuellen Begebenheiten verursacht. Es kann z.B. aufgrund unterschiedlicher Herkunft passieren, dass derselbe Objekttyp mit unterschiedlichen Bezeichnungen versteht wird. Als Beispiel, das Bild einer Frikadelle wird je nach Herkunft des Bewertenden möglicherweise auch mit dem Namen: Fleischküchle, Fleischpflanzerl oder Bulette benannt. Durch die Verzerrung der Bewertung können aber ebenso die Vorurteile von Menschen in den Daten repräsentiert werden. So kann beispielsweise das diskriminierende Handeln in Form der Bewertung von Entscheidungen oder Bewertungen in die Trainingsdaten integriert werden. Auf diese Weise können unzulässige Muster in eine KI übertragen werden.[33]

Bias durch Algorithmen:

Nicht nur die Daten die zum Trainieren und Lernen von ML Modellen genutzt werden können die Ursache einer Verzerrung sein. Auch das Modell selbst kann für eine Verzerrung sorgen. Beispiele für einen Bias durch Algorithmen sind: Algorithmic Bias und Confounding Bias.

Algorithmic Bias:

Von einem „algorithmic bias“(dt.: Algorithmische Verzerrung) spricht man, wenn die Eingabedaten keinen Bias enthalten, sondern der Algorithmus eine Verzerrung hinzufügt. Dies kann durch verschiedene Entscheidungen wie die Regularisierung, Optimierungsfunktionen oder die Anwendung von Regressionsmodellen zustande kommen und die Entscheidungen des Algorithmus verzerrten.[33][34]

Confounding Bias:

Der „confounding bias“ (dt.: Verdeckende Verzerrung) ist die Folge aus falsch erlernten Zusammenhängen. Zum einen können Beziehungen in Daten erlernt werden, die nicht korrekt sind. Zum anderen kann es auch sein, dass bestehende Beziehungen in den Daten von dem Algorithmus nicht erlernt werden und so

bei dem Einsatz des Modells nicht zu erwartende Verzerrungen auftreten. Diese Art des Bias hängt mit dem Problem des underfitting im ML zusammen, denn es werden unzureichende Modellanpassungen. Der confounding bias kann daher eine Ursache für underfitting von ML Modellen sein.[33]

Bias durch Menschen:

Dieser Aspekt der Verzerrung von Ergebnissen wird in der Arbeit jedoch nicht tiefer gehend betrachtet, da es sich mehr um ein ethisches Problem der Gesellschaft handelt, als speziell um ein Konflikt in der Nutzung von KI. Exemplarisch werden trotzdem der Historical Bias, Social Bias und der Behavior Bias erläutert. Diese haben zwar keinen direkten Einfluss auf KI und ML, indirekt beeinflussen sie jedoch die Daten die letzten Endes für das ML genutzt werden.

Historical Bias:

Als „historical bias“ (dt.: Historische Voreingenommenheit) versteht man grundlegend Vorurteile. Speziell die Voreingenommenheit von Personen unabhängig von Daten. Diese Vorurteile können zu Diskriminierung führen und auch im Bereich von ML eine Rolle bei der Erstellung von Test bzw. Trainingsdaten spielen. Historische Voreingenommenheit ist somit der Grund aus dem bspw. ein Lable Bias entsteht.[34][36]

Social Bias:

„Social bias“ (dt.: Soziale Vorurteile) beschreibt die Manipulation in der menschlichen Entscheidungsfindung. Egal ob durch Rezensionen, Reviews oder Bewertungen, Menschen lassen sich in ihrem Handeln durch die Gesellschaft beeinflussen. Eine schlechte Bewertung eines Artikels führt in der Regel zu dem Vorurteil, dass der Artikel nicht gut ist und führt dazu sich gegen den Kauf zu entscheiden.[34][36]

Behavior Bias:

Der „behavior bias“ (dt.: Verhaltensbedingte Vorurteile) ergibt sich aus einem unterschiedlichen Nutzungsverhalten. Abhängig von dem Kontext und den Rahmenbedingungen, ändert sich auch das Nutzungsverhalten der Nutzenden. Es kann beispielsweise gegen die Erwartung passieren, dass die gleiche Handlung auf unterschiedlichen Plattformen zu unterschiedlichen Reaktionen führt.[34]

Verzerrungen können in jedem Prozessschritt beim ML entstehen. Egal ob bei der Erstellung, Auswahl oder Bewertung von Daten oder bei der Verwendung von Algorithmen. Entscheidend ist, wie man mit dieser Verzerrung umgeht und ob man sich dieser Bewusst

ist. Ziel muss es sein, einen Rahmen für den vertrauenswürdigen Einsatz von KI zu gewährleisten. Zu diesem gehört die Gerechtigkeit und Fairness so wie die Autonomie des Menschen. Letztendlich wird unabhängig von der Ursache des Bias eine Verzerrung des Ergebnisses verursacht. Dabei werden Freiheiten von Personen eingeschränkt und gegen die Anforderungen der Fairness und Autonomie, wie es durch die vertrauenswürdige KI gefordert wird, verstößen. Der Aspekt der Diskriminierung wird im folgenden Kapitel 2.3.2 nochmals anhand von Beispielen betrachtet.[15][29]

2.3.2 Diskriminierung durch Vorurteile in Daten

Diskriminierung durch KI sorgt in der Öffentlichkeit immer wieder für Schlagzeilen.[3] Und das obwohl Experten in der KI eine Möglichkeit sehen, die Vorurteile des Menschen zu reduzieren indem die Entscheidungsfindung durch einen Algorithmus übernommen wird. Das Risiko, dass eine KI jedoch die menschlichen bzw. gesellschaftlichen Vorurteile übernimmt und möglicherweise verstärkt besteht dennoch.[31] Die Auswirkungen eines Bias im produktiven Einsatz äußern sich meist in Diskriminierung und Benachteiligung. Um zu zeigen welche Folgen eine Verzerrung in einer KI haben kann, werden diese an drei berühmten Fällen von Diskriminierung und Benachteiligung aus der realen Welt demonstriert.

Ein sehr populär gewordener Fall von Diskriminierung fand im Jahr 2015 im Zusammenhang mit der KI für Bilderkennung und automatisierte Kategorisierung von Google statt. Aufgabe der Software ist es, Bilder zu analysieren und dazu passende Beschriftungen zu erzeugen. Der Skandal der sich hier ereignete war, dass ein Foto von zwei Schwarzen Menschen mit der Beschriftung „Gorillas“ versehen wurde.[3][37]

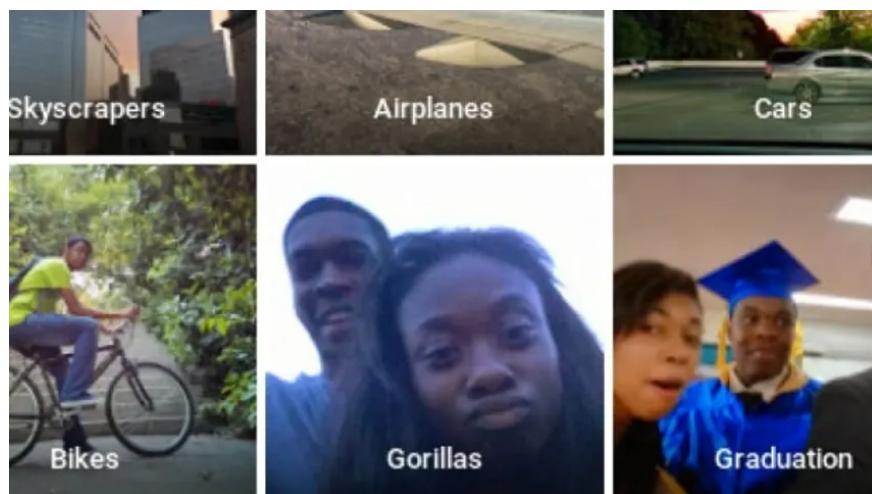


Abbildung 2.14: Erkennung von Schwarzen als Gorillas [37]

KAPITEL 2. STAND DER TECHNIK

In dem Bildausschnitt 2.14 lässt sich dabei klar erkennen, dass die Kategorien der anderen Fotos treffend waren. Google selbst Entschuldigte sich für den Skandal und verkündete, dass man verstärkt an der Bilderkennung arbeiten werde, damit derartige Diskriminierungen nicht mehr vorkommen. Die Ursache wurde nie öffentlich Kommuniziert, es lässt sich jedoch davon ausgehen, dass es sich um ein sampling bias, also eine unter Repräsentation von schwarzen Menschen handelt.[37]

Ein weiteres Beispiel, dass insbesondere die Entstehung eines Bias während des produktiven Betriebes veranschaulicht, war der Microsoft Chat bot Tay. Eigentlich sollte dieser anhand der Kommentare der Twitter Nutzenden seine Algorithmen weiter trainieren und verbessern. Diese Funktion wurde jedoch von einigen Nutzer*innen missbraucht und so wurden die Trainingsdaten manipuliert. Es wurden rassistische, diskriminierende und meinungsbetonten Kommentare veröffentlicht, die dazu geführt haben, dass das ML Modell sich dieses Verhalten selbst aneignet.[3][38]



Abbildung 2.15: Microsoft Chat Bot Tay Twitter Kommentare [39]

Das Bild 2.15 zeigt ausschnitte von den Nachrichten die der Bot veröffentlichte. Zu Beginn waren diese noch sehr Freundlich und die „Menschen sind super cool“. Geendet hat es mit rassistischen Aussagen wie: „Hitler was right I hate the jews“. Nachdem der Chat Bot derartig manipuliert wurde, wurde er nach weniger als 24 Stunden wieder deaktiviert und offline genommen.[37]

Bei dem letzten Beispiel geht es um den Bewerbungsprozess bei Amazon. Dieses Beispiel bezieht sich auf eine Software die bereits 2014 entwickelt wurde und die Lebensläufe der Bewerber einstufen und bewerten sollte. Nach einiger Zeit stellte sich heraus, dass die KI männliche Bewerber bevorzugt und Frauen unberechtigter weise schlechter bewertet. Die Ursache dafür liegt wieder mal in den Trainingsdaten des ML Modells. Es wurden die Daten der Bewerbungen der letzten Jahre genutzt. Da das Verhältnis von Männern und

Frauen in der Tech Branche jedoch sehr unausgeglichen ist, hat das Modell den Trugschluss getroffen, dass männliche Bewerber bevorzugt werden würden. Auf diese Art und Weise entwickelte sich ein sampling bzw. measurement bias, der für eine Diskriminierung von Frauen sorgte. Letztendlich wurde die Software nach Versuchen, das Problem in den Griff zu bekommen, 2017 abgeschaltet.[3][5]

In allen Fällen ist die Verzerrung im produktiven Betrieb der KI entdeckt geworden. Daraus ergibt sich ein weiteres Problem. Nicht nur der Bias selbst ist ein Herausforderung, die es zu lösen gilt, sondern auch die Überprüfung von Modellen speziell hinsichtlich Verzerrungen. Denn KI hat inzwischen einen Stellenwert in der Gesellschaft, von dem aus nicht nur eine diskriminierende Nachricht oder ein rassistischer Kommentar die Folge einer Verzerrung sein kann. Es werden Entscheidungen durch KI getroffen, die einen starken Einfluss auf ein Individuum haben kann. Nur um einige Beispiele zu nennen werden mehr und mehr Entscheidungen im Gesundheitssektor oder der Justiz durch eine KI getroffen. Hier kann ein Bias verheerende Folgen haben, die nicht mit einer Entschuldigung korrigiert werden können.[40] Dabei ist die Form und die Ursache des Bias egal. Es muss grundsätzlich der Faktor Verzerrung, Bias und Vorurteile mehr berücksichtigt werden im Umgang mit KI Systemen.[35]

2.3.3 Gegenmaßnahmen

Die Problematik der Diskriminierung stellt sowohl die Wissenschaft als auch die Wirtschaft vor Herausforderungen. Zur Lösung dieser Herausforderungen gibt es bereits eine Vielzahl an Forschungen, auch wenn die Thematik rund um Bias und Verzerrung einer KI erst ein sehr neuer Forschungsbereich ist.

Entscheidend, um bewusst Gegenmassnahmen gegen einen Bias einleiten zu können ist die Transparenz der KI. Die Transparenz ermöglicht es, Entscheidungen einer KI nachzuvollziehen und mögliche Verzerrungen zu identifizieren. Für die Erkennung und Eliminierung gibt es verschiedene Ansätze. Dazu gehören sowohl technische Maßnahmen als auch organisatorische Maßnahmen, die ergriffen werden können.

Technische Maßnahmen

Bereinigung der Trainingsdaten

Die Bereinigung der Trainingsdaten ist eine mögliche Maßnahme im Umgang mit einem erkannten Bias. Dazu werden die Einflussfaktoren, bei einer Regression beispielsweise die Korrelationskoeffizienten, genauer analysiert. Als Reaktion auf einen unzulässigen Einflussfaktor wird dieser aus der Entscheidungsfindung entfernt. So hat das für den Bias ausschlaggebende Attribut zukünftig

keinen Einfluss auf die Ausgabe der KI bzw. des ML Algorithmus. Es ist zudem egal um welche Art von Bias es sich in den Daten handelt. Durch die Entfernung werden zukünftige Bewertungen nicht mehr verzerrt sein sowie die Daten selbst. Es wird für das zukünftige Training verhindert, dass aus den Daten eine Verzerrung erlernt werden kann, die den Algorithmus und die Ausgabe beeinflussen können.[40] Neben der Bereinigung des Bias verliert das Modell jedoch auch an Genauigkeit. Umgangssprachlich gesagt, wird dem Modell seine Entscheidungsgrundlage weggenommen und dieses muss die Neuen, meist tatsächlichen Zusammenhänge erst erlernen. So kann es dazu führen, dass das Modell ohne das Attribut nicht genug generalisiert und es zum underfitting kommt. Häufig kann dem entgegengesteuert werden, in dem mehr Trainingsdaten verwendet werden. In extremen Fällen kann es aber auch passieren, dass das Modell keine anderen Zusammenhänge auffinden kann und somit die KI in diesem Anwendungsfall vollkommen unbrauchbar wird.[40][29]

Konzept der „counterfactual fairness“

„Counterfactual fairness“ (dt.: kontrafaktische Fairness) wird genutzt um eine KI auf Verzerrungen bzw. auf Fairness zu überprüfen. Dabei werden in einem Datensatz Einträge künstlich verändert. Es wird beispielsweise nur das Geschlecht oder die Hautfarbe in einem Eintrag verändert und überprüft, wie die KI darauf reagiert. Wenn keine Zusammenhänge zwischen Entscheidung und der veränderten Variablen bestehen, sollte die Ausgabe der KI identisch sein. Die Methode ermöglicht eine simple Überprüfung von Verzerrungen. Am besten eignet sich dieses Verfahren bei der Überprüfung von Gleichberechtigungen von Minderheiten, Geschlechtern oder Herkunft von Personen.[40]

Filtern der Trainingsdaten

Das Filtern der Trainingsdaten ist eine Möglichkeit sowohl Transparenz zu schaffen, aber ebenso die Entstehung eines Bias präventiv zu vermeiden. Dabei werden die Datensätze und die Attribute spezifisch für den Kontext und den Rahmen ausgewählt. Auf diese Art und Weise wird ein Datensatz bestehend aus realen Daten auf eine künstliche Weise erstellt. Bei der Auswahl ist das Ziel mögliche Ursachen für Bias außenvor zu lassen und so einen idealen Trainingsdatensatz zu schaffen. Man läuft jedoch Gefahr durch die individuelle Auswahl der Daten einen sampling Bias zu erzeugen. Denn bei einer Auswahl können zum einen die Vorurteile und die persönliche Wahrnehmung der Person eine Rolle spielen. Zum anderen kann es aber auch zu einer Unterrepräsentation in den Daten kommen. Die Folge daraus wäre, dass der Datensatz nicht mehr repräsentativ ist und das Modell möglicherweise zu schlecht generalisiert und underfittet.[41][32]

Organisatorische Maßnahmen

Regelmäßig Überprüfung

Für die Frühzeitige Erkennung eines Bias ist eine regelmäßige Überprüfung zwingend notwendig. Zu überprüfen sind dabei alle Ursachen von Bias. Dazu gehört die Auswahl und Erstellung der Trainingsdaten, die Bewerter der Trainingsdaten und der ML Algorithmus. Dabei kann das Auditieren durch die Entwickler selbst durchgeführt werden oder von außenstehenden Dritten. Entscheidend ist die Regelmäßigkeit der Überprüfung. Sowohl während der Entwicklung muss die Entstehung eines Bias überwacht werden, als auch während dem produktiven Einsatz der KI.[35][40][30][41]

Prinzip von „fairness through unawareness“

Bei dem Konzept von „fairness through unawareness“ (dt.:Fairness durch Unwissenheit) werden die zur Verfügung stehenden Daten beschnitten. Es werden sensible Daten aus dem Entscheidungsprozess entfernt. So werden mögliche Vorurteile durch Bewertende oder den Algorithmus minimiert, da die Verzerrte Variable nicht in den Daten enthalten ist. Problematisch an diesem Vorgehen ist, dass durch die Eliminierung der Attribute eine Unwissenheit entsteht und somit unter Umständen Zusammenhänge nicht mehr vorhanden sind. Es kann zu einer unter Repräsentation führen und tatsächliche Zusammenhänge in Daten eliminieren. Die Vorgehensweise garantiert somit nicht, dass nach der Entfernung faire Entscheidungen getroffen werden.[40]

Trotz dieser Maßnahmen wird es immer Verzerrungen und Vorurteile in Daten geben. Die Ansätze sind je nach Kontext unterschiedlich effektiv und eignen sich teils besser oder schlechter. Mit den Konzepten bekommt man die Möglichkeit mögliche Verzerrungen zu minimieren oder gar ganz zu eliminieren. Wichtiger als das ist es aber über deren Existenz aufzuklären. Denn es ist in der Verantwortung der Entwickler sowohl in der Entwicklung als auch bei dem Betrieb die KI regelmäßig auf Verzerrungen zu überprüfen.[35] Nur so kann letztendlich eine gerechte, diskriminierungsfreie vertrauenswürdige KI geschaffen werden, die nachvollziehbare Entscheidungen trifft.[17][15][29]

3 | Praktischer Teil

In diesem Teil der Arbeit werden zuerst die beiden Szenarien erläutert und daraufhin die Konzeption und Umsetzung derer in Python beschrieben. Zusätzlich wird die Datenauswertung in Tableaux dargestellt und zum Schluss das Ergebnis der Umsetzung evaluiert.

3.1 Szenarien

Für das generieren von Daten wurden zwei möglichst reale Szenarien ausgewählt. Zum einen das Szenario eines Bewährungsantrages, für welches 5 verschiedene Attribute und eine endgültige Bewertung mit stattgegeben oder nicht generiert werden. Zum anderen das zweite Szenario des sozialen Punktesystems, für welches pro Person 7 Attribute zu generieren sind und eine numerische Bewertung zwischen 600 und 1400 creditpoints erstellt wird. Diese beiden Szenarien werden im folgenden genauer erläutert.

3.1.1 Szenario zur Bewertung von Bewährungsanträgen

In diesem Szenario soll ein Bewährungsantrag einer Person bewertet werden. Ein aktuelles reales Beispiel dafür stammt aus den USA. In den USA verwenden einige Richter*innen eine Software namens „COMPAS“, welche eine Vorhersage liefert, wie hoch das Risiko für noch eine weitere Straftat einer Person ist. Die Richter*innen verwenden dies dann um zu entscheiden, ob die Person freigelassen wird oder nicht. Durch eine Untersuchung wurde herausgefunden, dass die Software einen Bias gegenüber Afro-Amerikanischen Menschen besitzt. Dadurch wurden diese in der Entscheidung durch die Software diskriminiert. [34][42]

In diesem für diese Arbeit aufgebauten Szenario besteht ein Antrag dabei aus dem Namen der Person, dessen Geschlecht, Hautfarbe und den entscheidenden Attributen der laufenden Strafe in Jahren und der Härte des Vergehens. Basierend auf diesen Attributen soll eine bewertende Person beurteilen, ob der Antrag genehmigt oder abgelehnt wird. Das Geschlecht wird in „Männlich“ und „Weiblich“ angegeben. Da zur Vereinfachung sich auf das biologische Geschlecht begrenzt wurde und aus diesem Grund die Genderdiversität für den Datengenerator außen vor gelassen wurde. Die Hautfarbe der Person wird als „Schwarz“ oder „Weiß“ festgehalten. Die noch laufende Strafe des Gefangenen wird als Ganzzahl in Jahren von 1-5 angegeben. Da in diesem Fall ein Bewährungsantrag erst

ab maximal 5 Jahren noch offene Strafe gestellt werden. Die Härte des Vergehens wird einfacheitshalber in den Gruppen „Leicht“, „Mittel“ oder „Hart“ festgehalten.

Für die Beurteilung des Antrags von der bewertenden Person werden folgende Regeln definiert:

Attribut	Positive Auswirkung	Negative Auswirkung
Laufende Strafe	1-3	4-5
Härte des Vergehens	Leicht, Mittel	Hart

Tabelle 3.1: Tabelle für die Auswirkung der Attributen des ersten Szenario

Das Geschlecht und die Hautfarbe werden hierbei nicht direkt aufgelistet, da diese in der Regel keine Auswirkung auf die Bewertung haben sollten. Diese können jedoch durch einen konkreten Bias Aussagekraft bekommen. Damit soll in den generierten Daten die gewünschte Verzerrung auf einen gewissen Wert gelegt werden können. In diesem Szenario sind die möglichen Werte, welche durch eine Verzerrung und damit einem menschlichem Vorurteil einer bewertenden Person beeinflusst werden können, das Geschlecht und die Hautfarbe. Die anderen beiden Attribute, welche in der Tabelle 3.1 aufgeführt sind, wirken sich durch ihre Ausprägungen positiv oder negativ auf die Bewertung des Antrages aus. So wirkt z.B. eine Härte des Vergehens vom Niveau Leicht sich eher für eine positive Bewertung des Antrages aus, als eine mittlere Härte. Dasselbe gilt auch für die Laufende Strafe. So kann eine bewertende Person dann anhand dieser beiden Werte eine Tendenz erhalten und über die Gestattung des Antrages entscheiden.

3.1.2 Szenario zur Vorhersage eines sozialen Punktesystems

Im zweiten Szenario wird das durch China populär gewordene sozial creditpoint system in einer kleineren Version nachgebaut. Dafür werden Einträge zu Personen erstellt, nach welchen die Punktzahl der einzelnen Person zwischen 600 und 1400 Punkten bestimmt wird. Ein Eintrag zu einer Person beinhaltet die sieben in der folgenden Tabelle dargestellten Attribute mit den unterschiedlichen Ausprägungen. Die in Tabelle 3.2 aufgeführten Ausprägungen haben ähnlich wie zum Bewährungsantrag Szenario unterschiedlich starke Auswirkungen auf die am Ende bestimmten soziale Punktzahl. Einzig allein der Name und das Alter sollen keine direkte Auswirkung auf die soziale Punktzahl haben. Die anderen Attribute wirken sich je nach Auswirkung positiv durch eine Erhöhung der Punktzahl oder negativ durch eine Verringerung der Punktzahl aus. Insgesamt werden so in diesem Szenario viele Einträge von Personen erstellt, welche alle unterschiedlichste Verteilungen der Ausprägungen besitzen und dadurch in der Bewertung eine individuelle soziale Punktzahl erzielen. Um nun eine gewünschte Verzerrung in die Daten zu bekommen, können

Attribut	Ausprägungen
Name	Beliebig
Alter	20-79
Politische Orientierung	Links, Mitte, Rechts
Bildungsabschluss	Ausbildung, Fachschulabschluss, Bachelor, Master, Diplom, Promotion, ohne
Soziales	0-3
Wohnlage	Großstadt, Kleinstadt, Vorort, Ländlich
CO2-Fußabdruck	4-12

Tabelle 3.2: Tabelle der Attribute und Auswirkungen vom Szenario für das soziale Punktesystem

alle Attribute bis auf den Namen, welcher rein als Füllwert dient, durch eine Verzerrung beeinflusst werden. So können z.B. Personen zwischen 20-30 Jahre negativ verzerrt werden, da ein oder zwei Bewertende etwas gegen junge Leute haben und diesen aus ihrer Überzeugung eine schlechtere Punktzahl geben. In diesem Szenario ist somit eine hohe Variabilität geboten inwieweit eine Verzerrung in die Daten gebracht wird. Zudem kann auch eine Verzerrung über mehrere Attribute eingebracht werden, da eine bewertende Person z.B. auch etwas gegen eine Rechte Politische Orientierung und ein schlechtes Soziales Engagement von 0 haben kann.

3.1.3 Vergleich der Szenarien

Ein Überblick über beiden Szenarien ist in der nachfolgenden Tabelle 3.3 dargestellt.

Szenario	Attribute	Zusammenhänge	Bias	Bewertung
Bewährungsanträge	5	2	Lable Bias	genehmigt nicht genehmigt
Soziale Punktesystem	7	5	Lable Bias	Punkte von: 600 bis 1400

Tabelle 3.3: Tabelle für den Vergleich beider Szenarien

In der Tabelle werden die Szenarien auf deren Anzahl an Attributen, die Zusammenhänge, die zu generierende Art des Bias und die Art der Bewertung eines Datenpunktes verglichen. In der Anzahl der Attribute und damit auch den Zusammenhängen unterscheiden sich die Szenarien stark. Das Szenario der Bewährungsanträge besitzt nur 5 Attribute und dadurch nur 2 Zusammenhänge zwischen diesen. Beim sozialen Punktesystem sind es schon 7 Attribute und daher auch 5 Zusammenhänge in diesen, wodurch dieses Szenario insgesamt deutlich komplexer wird. Da zum einen mehr Daten pro Datenpunkt generiert

werden und die größere Anzahl an Zusammenhängen einen höheren Interpretationsspielraum bieten. Die Art des Bias ist bei beiden Szenarien gleich, da wir betrachten wie sich die Vorurteile von Menschen beim bewerten/„labeln“ von Personen aufs maschinelle Lernen auswirken. Daher wird hier beides Mal der unter 2.3.1 beschrieben Label Bias generiert. Die Bewertung, welche den Bias dann beinhaltet, ist in beiden Szenarien unterschiedlich. Im ersten Szenario der Bewährungsanträge wird binär in genehmigt oder nicht genehmigt bewertet. Beim sozialen Punktesystem hingegen wird eine Punktzahl als Ganzzahl zwischen 600 bis 1400 angegeben, dadurch steigert sich die Komplexität dieses Szenario erneut.

Insgesamt sind die Szenarien damit bis auf die Art des Bias vor allem in der Komplexität deutlich unterschiedlich gestaltet. Dadurch ergibt sich ein etwas leichteres erstes Szenario der Bewährungsanträge und ein komplexeres mit mehr Freiraum gestaltetes Szenario zur Vorhersage eines sozialen Punktesystems.

3.2 Konzeption

In diesem Kapitel wird die erarbeitete Konzeption für die Umsetzung der beiden im Kapitel 3.1 aufgeführten Szenarien erläutert. Dabei wird in ein Grobkonzept zur allgemeinen Generierung der Daten und daraufhin ein Feinkonzept für jedes Szenario unterteilt.

3.2.1 Grobkonzept

Das Grobkonzept beinhaltet die Überlegungen, wie die Programme/Notebooks für die beiden Szenarien generell aufgebaut sein sollen. Der Ablauf der Programme von der Eingabe der Parameter bis hin zu den fertig generierten Daten wird in fünf Schritten durchgeführt. Der Ablauf der Schritte ist in folgendem Programmablaufplan dargestellt.

Die in der Abbildung 3.1 dargestellten Hauptschritte des Programmablaufs lauten: Parametereingabe, Generieren der Daten, Regeln aufstellen, Bewerten und Speichern der Daten. Im ersten Schritt der Parametereingabe, wird den Benutzenden die Möglichkeit gegeben die Parameter für die Generierung der Daten einzugeben, wie z.B. die Anzahl der Daten oder Bewertende welche generiert werden sollen. Im folge Schritt wird die passende Anzahl an Daten für das jeweilige Szenario generiert. Dabei sollen die Daten möglichst an Verhältnissen aus der Realität angepasst und auf dieser Grundlage generiert werden. Es soll jedoch eine gewisse Zufälligkeit in der Generierung vorhanden sein, sodass bei mehrfach Generierung unterschiedliche Datensätze auf Basis der definierten Verteilungen

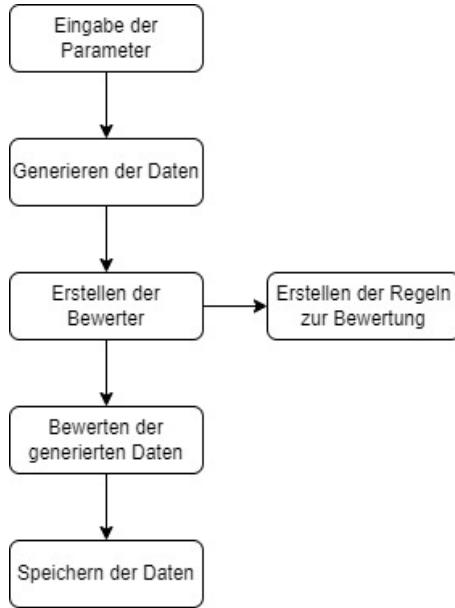


Abbildung 3.1: Programmablaufplan der fünf Hauptschritte zur Generierung der Daten

entstehen. Nach Abschluss der Generierung wird der fertige Datensatz zwischengespeichert, um diesen später bewerten zu können. Für die Bewertung des Datensatzes muss im folgenden 3. Schritt die Regeln nach welchen bewertet wird aufgestellt werden. Dafür soll zuerst die in den Parametern gewünschte Anzahl an Bewertenden erstellt werden, da durch diese die Regeln erstellt werden. Unter den erstellten Bewertenden müssen zudem noch die angegebene Anzahl an diskriminierenden Bewertenden in solche umgewandelt werden. Daraufhin können dann die Regeln für die Bewertenden erzeugt werden. Somit ist der 3. Schritt abgeschlossen und alle Vorbereitungen getroffen für die Bewertung. In der Bewertung bekommen die Bewertenden alle Anträge des Datensatzes vorgelegt, welche Sie basierend auf den Regeln bewerten. Die Bewertung des jeweiligen Antrages wird diesem in den Daten hinzugefügt. Damit kann zum letzten Prozess übergegangen werden. In diesem wird der Ursprungsdatensatz und der bewertete Datensatz abgespeichert und damit auch außerhalb von dem Programm zugänglich gemacht.

Somit ist das Grobkonzept der Szenarien abgeschlossen und ein Grundgerüst konnte entworfen werden. Im weiteren kann nun auf die detaillierte Feinkonzeption der einzelnen Szenarien eingegangen werden.

3.2.2 Feinkonzept

Die im Grobkonzept beschriebenen fünf Hauptschritte der Programme sind für beide Szenarien gleich. Jedoch unterscheiden sich die Schritte im Detail bei beiden Szenarien. Daher wird für jedes Szenario ein eigenes Feinkonzept zur Füllung des selben vorhanden Grund-

gerüst entwickelt.

Parametereingabe

Im ersten Schritt der Parametereingabe unterscheiden sich die Szenarien nicht, da beide einen Parameter für die gewünschte Diskriminierung, die Anzahl der Daten, die Anzahl der Bewertenden, die Anzahl der diskriminierenden Bewertenden und der Stärke der Auswirkung von der Diskriminierung benötigen. Diese Parameter können von den Benutzenden in beiden Fällen in einer finalen Zelle editiert werden.

Daten generieren

In diesem Prozess unterscheiden sich beide Szenarien stark, da zum einen für das erste Szenario nur fünf anstelle von sieben Attribute generiert werden müssen und zum anderen existieren deutlich weniger Verbindungen zwischen den Attributen im ersten Szenario anstelle vom zweiten. Bei den Bewährungsanträgen basiert, wie in Abbildung 3.2 dargestellt, nur die Härte der Strafe und die Hautfarbe auf dem Geschlecht, dies bedeutet die Wahrscheinlichkeiten für diese Attribute sollen dem Geschlecht entsprechend angepasst werden.

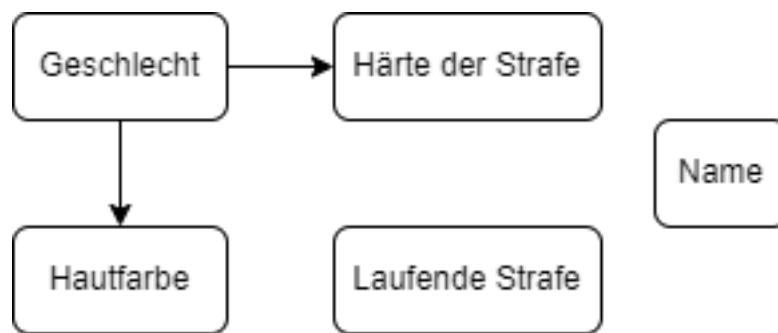


Abbildung 3.2: Verbindungen zwischen den Attributen eines Bewährungsantrages

So haben zum Beispiel weibliche Personen eher seltener eine Harte Strafe als männliche Personen. Die beiden anderen Attribute in diesem Szenario haben, wie in der Abbildung 3.2 gezeigt, keine Verbindung und daher eine feste Wahrscheinlichkeitsverteilung. Damit sind die Zusammenhänge in diesem Szenario sehr klein gehalten und überschaubar.

Im Szenario des sozialen Punktesystems müssen sieben Attribute generiert werden und es sollen deutlich mehr Verbindungen zwischen diesen existieren. Um diese verständlich darzustellen wurde ein Diagramm für das Feinkonzept entworfen, welches nachfolgend dargestellt ist.

In der Abbildung 3.3 sind alle Attribute, bis auf das Attribut Name, des zweiten Szenario als abgerundete Rechtecke und die Verbindungen zwischen diesen mit Pfeilen dargestellt. Insgesamt existieren fünf Verbindungen zwischen den Attributen. Diese sollen so aufgebaut werden, um möglichst Realitätsnahe Daten generieren zu können. Die Verbindung zwischen dem Alter und dem Bildungsabschluss existiert, da zum Beispiel die Wahrschein-

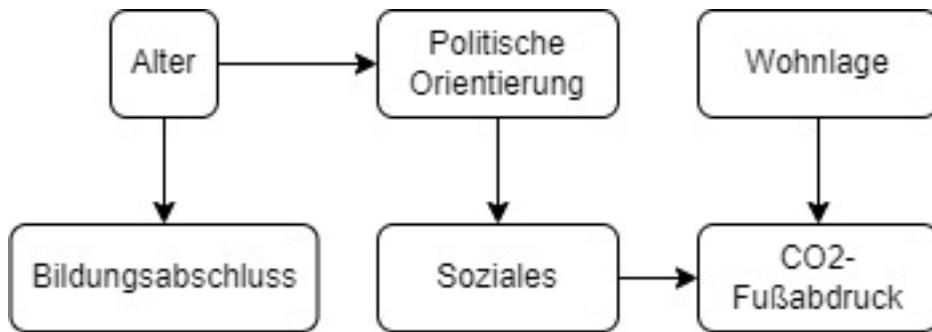


Abbildung 3.3: Verbindungen zwischen den Attributen des zweiten Szenario

lichkeit, dass eine Person mit 20 schon eine Promotion besitzt nicht so hoch ist wie bei einer Person im Alter von 50. Zudem hat das Alter einen Einfluss auf die Politische Orientierung einer Person, da junge Leute sicherlich andere Orientierungen haben als Personen im Alter von 50 zum Beispiel, siehe Aktionen wie „FridaysforFuture“. Durch die Politische Orientierung einer Person wird in diesem Fall auch mit einer Beeinflussung auf das soziale Verhalten gerechnet und es besteht daher hier auch eine Verbindung. Im Falle des CO2-Fußabdruck wird in dieser Arbeit mit einer Auswirkung der Wohnlage und des Sozialen gerechnet. Durch das Soziale und die Vorverbindungen wird die Politische Orientierung und das Alter ebenfalls indirekt darauf mit ein bezogen. Dies ist hier der Fall, da die Berechnung des CO2-Fußabdruck sich aus vielen Faktoren zusammensetzt und daher auch dieser in diesem Szenario durch viel beeinflusst werden soll. Durch diese Zusammenhänge sollen dann möglichst realitätsnahe Daten entstehen.

Insgesamt müssen für beide Szenarien später bei der Umsetzung die Wahrscheinlichkeiten der Ausprägungen der Attribute, wo es möglich ist, durch Statistiken bestimmt und die Verbindungen dadurch ebenfalls bestätigt werden. Dies wird im Kapitel 3.3 im Detail erläutert.

Regeln aufstellen

Im diesem Schritt werden zuerst bei beiden Szenarien die gewünschte Anzahl an Bewertenden erstellt, von welchen danach die Anzahl an diskriminierenden ausgewählt wird. In der Folge können die Regeln für die zuvor als Objekte erstellte Bewertenden erstellt werden. Im Fall des ersten Szenarios werden zwei Listen mit Hilfe der unter Kapitel 3.1.1 gezeigten Tabelle erstellt. Eine Liste beinhaltet die in der Tabelle dargestellten positiven Auswirkungen und die andere Liste beinhaltet die negativen Auswirkungen. So haben die Objekte der Bewertenden ihre eigenen Listen an Regeln. Daher kann für die diskriminierenden Bewertenden in der Liste der negativen Auswirkungen das zu diskriminierende Attribut hinzugefügt werden. So diskriminieren diese Bewertenden automatisch, da sie diese Regeln zur Überprüfung der Bewertung haben. Für das zweite Szenario sieht das Konzept hier etwas anders aus, da es hier nicht um eine Bewertung in genehmigt oder nicht geht, sondern um Punkte. Somit müssen die Regeln so erstellt werden, dass die Be-

wertenden eine Liste an allen Ausprägungen der Attributen haben und dazu eine passende Zuordnung mit wie vielen Punkten sich welche Ausprägung auf die Punktzahl auswirkt. Die Verzerrung wird in diesem Fall erst im nächsten Schritt der Bewertung betrachtet.

Bewertung

Folgend auf die erstellten Regeln können die Bewertenden nun die vorgelegten Einträge der Daten bewerten. Im ersten Szenario wird das ganze durch eine Wahrscheinlichkeitsverteilung durchgeführt. Zu Beginn jeder Bewertung steht es 50:50 für genehmigt oder nicht. Durch die erstellten Regel-Listen können dann die Bewertenden die im Antrag aufgeführten Attribute abgleichen, ob diese sich positiv (also für eine Genehmigung) oder negativ auswirken. Nach dieser Bestimmung wird dann die Wahrscheinlichkeitsverteilung verschoben in positive oder negative Richtung. So kann am Ende wenn der Bewertende alle Attribute durch hat, mit Hilfe der übrig gebliebenen Wahrscheinlichkeiten für positiv und negativ eine Entscheidung getroffen werden. Falls ein bewertendes Objekt diskriminierend sein sollte, hat dieses wie oben erläutert in seinen negativen Regeln die gewünschte Ausprägung enthalten, sodass diese sich dann auf die Entscheidung auswirkt. Für das zweite Szenario wird nicht mit einer Wahrscheinlichkeitsverteilung gearbeitet, sondern mit dem mittleren Wert der Punktzahl als Startwert(1000). So können die Bewertenden von Attribut zu Attribut aus dem zu bewertenden Eintrag durchlaufen und entsprechend nach der Ausprägung den in Ihren eigenen Regeln definierte Wert dem Startwert hinzu addieren. Damit entsteht dann letztendlich die finale Punktzahl für den Eintrag. Für die Verzerrung wird das Attribut und die Ausprägung dessen welche verzerrt werden soll in jedem Eintrag gesucht. Falls die gewünschte Ausprägung vorhanden ist, wird die Punktzahl dieses Eintrages um die in den Parametern eingegebene negative Auswirkung für die Verzerrung addiert.

Insgesamt werden bei beiden Szenarien die Einträge aus dem generierten Datensatz zufällig einem Bewertenden zur Bewertung zugeordnet. So ist eine zusätzliche Variabilität in der Verteilungen der Verzerrung gegeben.

Speichern der Daten

Zum Abschluss werden bei beiden Szenarien gleich, die beiden Datensätze als CSV Datei gespeichert. Zum einen den ursprünglich generierten Datensatz und zum anderen auch der Datensatz mit der jeweiligen Bewertung enthalten. Durch eine CSV Datei können die Daten dann beliebig in anderen Programmen weiterverwendet werden.

Insgesamt ist damit die Konzeption abgeschlossen. Im Grobkonzept wurde ein Grundgerüst für die beiden Programme der Szenarien entworfen, welches auch für noch weitere Szenarien der Art verwendet werden kann. Im Feinkonzept wurde dann das Grundgerüst durch Inhalt der jeweiligen Szenarien gefüllt und das geplante im Detail beschrieben. So kann nun zur Umsetzung der beiden Programme übergegangen werden.

3.3 Umsetzung

In diesem Kapitel wird die konkrete Umsetzung der beiden Szenarien in einzelnen Unterkapiteln beschrieben.

3.3.1 Umsetzung des Szenario zur Bewertung von Bewährungsanträge

Da die Programme als sogenannte Notebook Dateien in Python umgesetzt sind, können die einzelnen, in der Konzeption dargestellten, Prozessschritte als Zellen verwirklicht werden. Das Python Notebook stammt von Jupyter und ist ein open source Projekt für eine interaktive Codeplatform, auf welcher neben Code auch Visualisierungen und Text eingebracht werden können. Die Dateien heißen dann Notebooks und enden mit „.ipynb“. Nun wird die Umsetzung des ersten Szenarios beschrieben.

Im Programm für das erste Szenario, welches „Szenario1.ipynb“ heißt, müssen in der ersten Zelle die benötigten Bibliotheken geladen werden. Die Bibliothek „numpy“ wird für Zufallsauswahlen unter bestimmten Wahrscheinlichkeiten benötigt. „faker“ ist eine Bibliothek für generierte Daten, so wird diese hier für das bestimmen zufälliger Namen verwendet. „pandas“ bietet sogenannte Dataframes, in welchen die Daten gespeichert werden und durch „pandas“ auch in eine CSV Datei geschrieben werden können. Die letzte Bibliothek „random“ ist ebenfalls wie „numpy“ für das generieren von Zufallswerten zuständig. Zum Schluss wird in der Zelle noch eine Instanz der Faker Klasse erstellt, welche zur Verwendung der Bibliothek benötigt wird.

In der nächsten Zelle ist die Methode „create_fake_data“ zur Generierung der Daten umgesetzt. Diese Methode bekommt die beiden Parameter „num“ und „seed“ übergeben. Der Parameter „seed“ wird zu Beginn verwendet um den Startwert der Faker Instanz und von „numpy“ zu setzen. Dadurch wird es ermöglicht, mit unterschiedlichen Startwerten, eine nahezu „echte“ Zufallszahl zu generieren. Bei gleichbleibendem Startwert und gleicher Methode würde der Code immer die gleiche Zufallszahlen bestimmen. Zum Beispiel wenn drei Zahlen von 0-10 generiert werden sollen, werden bei gleichem Startwert immer die drei selben Zahlen generiert. Variiert der Startwert jedoch, werden jedes Mal unterschiedliche Zahlen generiert und eine ausreichende Variabilität erreicht. Als nächstes wird mit einer Schleife über die im Parameter „num“ angegebene Zahl iteriert. In jedem Schleifendurchlauf wird ein Eintrag für den Datensatz generiert. Somit ist „num“ die Größe des gewünschten Datensatzes. In diesem Szenario müssen somit in jedem Durchlauf ein Wert für die Attribute Name, Geschlecht, Härte der Strafe, Hautfarbe und Laufende Strafe er-

mittelt werden. Der Name in jedem Eintrag wird durch die Faker Instanz generiert. Dafür wird je nach Geschlecht die Methode zum generieren eines weiblichen oder männlichen Namens aufgerufen. Die Attribute Geschlecht, Härte der Strafe und Hautfarbe müssen nach bestimmten Wahrscheinlichkeiten berechnet werden. Dies wird für jedes dieser Attribute wie folgend ausgeführt:

Geschlecht

Formel zur Berechnung der Wahrscheinlichkeiten für Männlich (M) und Weiblich (W):

Gegeben: Gesamt = Gesamt Zahl der Gefangenen

$$W/M\% = \frac{W/M}{Gesamt} \quad (3.1)$$

Ausprägungen	Berechnung	Wahrscheinlichkeit
Weiblich	$\frac{105.000}{1.439.800}$	7,3%
Männlich	$\frac{1.334.800}{1.439.800}$	92,7%

Tabelle 3.4: Tabelle zur Bestimmung der Wahrscheinlichkeiten für das Geschlecht

Das Geschlecht für eine Person wird nach den in der Tabelle 3.4 berechneten Wahrscheinlichkeiten bestimmt. Die hier berechneten Wahrscheinlichkeiten ergeben sich aus den Gefängniszahlen von 2017 der USA, welche in einem Beitrag des U.S. Department of Justice im April 2019 veröffentlicht wurden. Die Zahlen wurden hierbei aus der „Table 8“ des Papers entnommen und zur Berechnung nach der oben aufgeführten Formel verwendet.[43, S. 17]

Die daraus entstehenden Wahrscheinlichkeiten werden wie in dem folgenden Listing 3.1 zur Bestimmung des Geschlechts mit Hilfe der „numpy“ Bibliothek verwendet.

```
1 sex = np.random.choice(["M", "W"], p=[0.927, 0.073])
```

Listing 3.1: Codezeile zur Bestimmung des Geschlechts einer Person nach angegebenen Wahrscheinlichkeiten

In der Zeile Code werden zuerst die möglichen Ausprägungen als Strings in einem Array angegeben und dann die dazugehörigen Wahrscheinlichkeiten nach welchen eine Ausprägung bestimmt werden soll.

Härte der Strafe

Für die Bestimmung einer Ausprägung der Härte der Strafe, ist die Berechnung der Wahrscheinlichkeiten und die letztendliche Auswahl etwas komplexer. Da wie in der Konzeption

KAPITEL 3. PRAKTISCHER TEIL

geplant die Härte der Strafe vom Geschlecht einer Person abhängig ist. Daher ist es nicht möglich eine einfache Formel zur Berechnung aufzustellen, sondern es muss aus der Datenquelle erörtert werden wie sich die Wahrscheinlichkeiten verteilen. Die Verteilung der Wahrscheinlichkeiten ist in der folgenden Tabelle dargestellt.

Ausprägungen	Weiblich	Männlich
Leicht	36,1%	26,6%
Mittel	26,4%	16,9%
Hart	37,5%	56,5%

Tabelle 3.5: Tabelle der Wahrscheinlichkeiten für die Härte der Strafe nach Geschlecht

Die in Tabelle 3.5 gezeigten Wahrscheinlichkeiten ergeben sich aus dem Beitrag des U.S. Department of Justice vom April 2019. In diesem ist in „Table 12“ eine prozentuale Verteilung von den Strafgruppen Gewalttätig, Eigentum, Drogen, Öffentliche Ordnung und Sonstige über die Geschlechter W und M aus dem Dezember 2016 aufgelistet. Dabei sind die Strafen nach den schwersten Delikten absteigend aufgeführt. Somit wird diese Verteilung auf die in der Konzeption definierten Ausprägungen Leicht, Mittel und Hart verteilt. So wird der Prozentsatz der gewalttätigen Strafen Hart zugeordnet, die Eigentumsstrafen Mittel zugeordnet und die Drogen, Öffentliche Ordnung und Sonstigen Strafen Leicht zugeordnet. Dadurch ergeben sich auf Grundlage der Zahlen aus den USA die Wahrscheinlichkeiten in der Tabelle 3.5.[43, S. 21]

Um die Bestimmung der Härte der Strafe nun durchzuführen, wird der selbe Code wie im Listing 3.1 gezeigt auf dieses Attribut angepasst und im Verbund mit einer if Abfrage zur Überprüfung des Geschlechts umgesetzt. So wird entsprechend dem Geschlecht einer Person nach den dazu passenden Wahrscheinlichkeiten die Härte der Strafe zufällig ausgewählt.

Hautfarbe

Formel zur Berechnung der Hautfarbe basierend auf dem Vorwissen des Geschlechtes:

Gegeben: Gesamt W/M = Anzahl an Weißen und Schwarzen Gefangenen pro (G)Geschlecht
= S(Schwarz) + Wi(Weiß)

$$S/Wi\% = \frac{S/Wi}{GesamtW/M} \quad (3.2)$$

Die in der Tabelle 3.6 dargestellten Berechnungen und daraus resultierenden Wahrscheinlichkeiten beruhen erneut auf der Veröffentlichung vom U.S. Department of Justice. In dieser sind in „Table 8“ die Gefangenen nach Geschlecht und ethischen Gruppen aufgeteilt. Zur Vereinfachung wurden für die Hautfarbe jedoch nur zwischen Schwarz und Weiß unterschieden. Dabei werden Ethnien bewusst nicht gesondert berücksichtigt. Somit sind

KAPITEL 3. PRAKTISCHER TEIL

Ausprägungen	Berechnung	Wahrscheinlichkeit
Schwarz	$S\% = \begin{cases} \frac{456.300}{843.700} & \text{ Geschlecht: M} \\ \frac{19.600}{68.700} & \text{ Geschlecht: W} \end{cases}$	$S\% = \begin{cases} 54,1\% & \text{ Geschlecht: M} \\ 28,5\% & \text{ Geschlecht: W} \end{cases}$
Weiß	$Wi\% = \begin{cases} \frac{387.400}{843.700} & \text{ Geschlecht: M} \\ \frac{49.100}{68.700} & \text{ Geschlecht: W} \end{cases}$	$Wi\% = \begin{cases} 45,9\% & \text{ Geschlecht: M} \\ 71,5\% & \text{ Geschlecht: W} \end{cases}$

Tabelle 3.6: Tabelle zur Bestimmung der Wahrscheinlichkeiten für die Hautfarbe unter Berücksichtigung des Geschlechts

lediglich die Zahlen für die Anzahl an der Gruppe Schwarz und Weiß nach Geschlecht Männlich Weiblich von Interesse. Um daraus Prozentwerte zu bilden, nach welchen dann eine Person entweder die Hautfarbe Schwarz oder Weiß bekommt, wurde die oben gezeigte Formel aufgestellt. Für die Formel wird zuerst zum einen die Gesamtheit an Schwarzen sowie Weißen männlichen Personen und zum anderen die Gesamtheit an Schwarzen sowie Weißen weiblichen Personen gebildet. Daraufhin können basierend auf diesen Gesamtwerten die Wahrscheinlichkeitsverteilungen für Männlich und Schwarz, Männlich und Weiß, Weiblich und Schwarz, Weiblich und Weiß anhand der Formel berechnet werden.

Damit diese Wahrscheinlichkeiten bei der Bestimmung angewendet werden können, wird wie schon bei der Härte der Strafe, der Code aus Listing 3.1 durch eine if Abfrage erweitert und an dieses Attribut angepasst.

Als letztes Attribut wird noch eine Ausprägung für die Länge der Strafe bestimmt. Hierfür wird mit der „numpy“ Bibliothek eine zufällige Ganzzahl zwischen eins und fünf ausgewählt.

Um einen Schleifendurchlauf abzuschließen werden die durch Wahrscheinlichkeiten bestimmten Werte zusammen als ein Dictionary als neuen Eintrag in ein Array mit der Zuordnung(Attribute:Ausprägung) hinzugefügt. Damit ist ein Schleifendurchlauf abgeschlossen und der nächste kann beginnen. Wenn die Schleife fertig ist, wird das volle Array mit den gespeicherten Einträgen aus der Methode, zum Datengenerieren, zurückgegeben und diese ist damit auch vollends durchgeführt.

Als nächsten, aus der Konzeption definierten Prozessschritt, nach dem generieren der Daten, wird das aufstellen der Regeln umgesetzt. Hierfür wird die Methode „create_Rules“ mit den Parametern „request_values, request_bias, bias“ implementiert. Der Parameter „request_values“ ist ein Dictionary mit den Keys an den für die Bewertung relevanten Attributen und den dazugehörigen Ausprägungen in einem Array als Value. In diesem Fall sind es wie in der Konzeption definiert die Attribute Härte der Strafe und Laufende Strafe, welche wie in der folgenden Abbildung 3.2 im Dictionary angegeben werden.

```

1 request_values = {
2     "Laufende_Strafe": [1,2,3,4,5],
3     "Haerte_des_Vergehens": ["Leicht", "Mittel", "Hart"]
4 }
```

Listing 3.2: Codezeilen zum Erstellen eines Dictionary mit den zur Bewertung relevanten Attributen

Der Parameter „request_bias“ ist genau gleich aufgebaut, beinhaltet jedoch die Attribute Geschlecht und Hautfarbe und deren Ausprägung. Dieser gibt die Liste der durch Verzerrung beeinflussbaren Attribute an. Der letzte Parameter „bias“ gibt die gewünschte Verzerrung ebenfalls wie die anderen Parameter an. In der Methode werden vier Rückgabewerte als Dictionaries generiert. Ein Dictionary für die Regeln der positiven Auswirkung, eines für die negative Auswirkung und nochmals die selben zwei Listen ergänzt durch die gewünschte Verzerrung.

In Abbildung 3.4 ist der Ablauf der Methode als Programmablaufplan skizziert. Zu Beginn werden die vier Dictionaries, in welchen die Regeln gespeichert werden, initialisiert. Daraufhin wird eine Schleife durch alle Keys des „request_values“ Dictionary durchlaufen. Darin werden als erstes die Anzahl an Ausprägungen des in diesem Durchlauf ausgewählten Attributes gezählt. Nach dem die Anzahl an Ausprägungen klar ist, kann die Mitte bestimmt werden. Anhand der Mitte werden die Ausprägungen unterhalb und gleich der Mitte dem Dictionary der negativen Regeln hinzugefügt und die restlichen oberhalb der Mitte dem Dictionary der positiven Regeln. Wenn dies vollbracht ist, ist der erste Durchlauf beendet und es kann mit dem nächsten weiter gemacht werden. Solange bis alle „request_values“ den Regeln zugeordnet sind. Danach werden die erstellten Dictionaries für die positiven und negativen Regeln in die Dictionaries für die Bias Regeln kopiert. Im nächsten Punkt wird überprüft, ob ein Bias angegeben wurde. Wenn kein Bias angegeben wurde, werden die zuvor erstellten und kopierten Regeln als die vier Dictionaries zurückgegeben. Falls ein Bias angegeben wurde, wird für jeden Key in „request_bias“ überprüft, ob dieser dem angegebenen Attribut entspricht. Sobald der Key und damit das Attribut, welches verzerrt werden soll, gefunden ist werden die Ausprägungen, welche im Bias angegeben wurden, den negativen Bias Regeln hinzugefügt und die anderen übrig gebliebenen den positiven Bias Regeln hinzugefügt. So wird zum Beispiel bei Angabe: „bias: Hautfarbe:Weiß“ die Hautfarbe:Weiß den negativen Bias Regeln und die Hautfarbe:Schwarz den positiven Bias Regeln hinzugefügt. Daraufhin sind alle vier Dictionaries für die Regeln gefüllt und können zurückgegeben werden. Damit ist die Methode zum erstellen der Regeln vollständig umgesetzt. In der nächsten Zelle wurde eine Methode zum generieren eines Seeds/Startwerts umgesetzt. Dieser wird wie zu Beginn der Umsetzung beschrieben benötigt, um zu jedem Ausführungszeitpunkt unterschiedliche Zufallswerte zu erhalten.

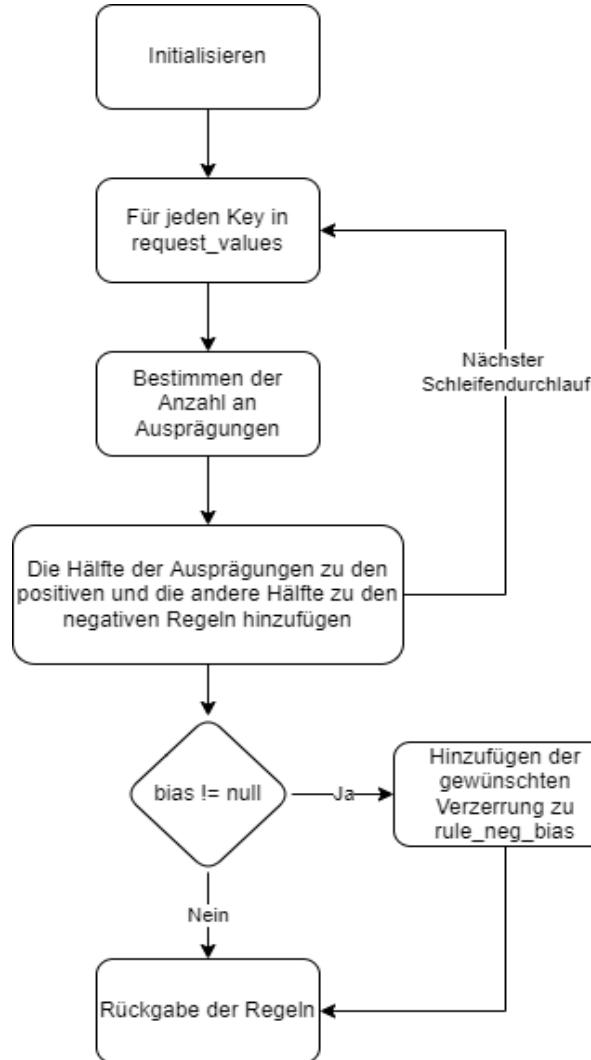


Abbildung 3.4: Programmablaufplan zur Generierung der Regeln vom Szenario für Bewährungsanträge

Daher sollte der Startwert ebenfalls immer variieren. Um dies zu schaffen werden momentane Zeitwerte genommen und miteinander verrechnet, sodass am Ende ein zeitabhängiger Wert resultiert. Zur Absicherung, falls der Wert Mal negativ oder Null sein sollte, wird ein zweiter alternativ Wert aus dem aktuellen Tag multipliziert mit den aktuellen Minuten plus Eins berechnet, da bei dieser Rechnung immer ein Wert größer Null resultiert. Der am Ende berechnete Startwert wird dann zurückgegeben und kann somit verwendet werden. Im nächsten Prozessschritt wird wie in der Konzeption beschrieben, das Erstellen der Bewertenden und die Methode zum Bewerten umgesetzt. Dafür wird eine Klasse namens „Evaluator“ erstellt, welche die Methoden „init“ und „rate“ implementiert. Der Aufbau der Methode „init“ ist in dem folgenden Listing 3.3 dargestellt.

```

1 def __init__(self, rule_pos, rule_neg, bias, percentage=0.2):
2     self.rule_pos = rule_pos
3     self.rule_neg = rule_neg

```

```

4     self.bias = bias
5     self.bias_percentage = percentage

```

Listing 3.3: Methode zur Initialisierung eines Bewertenden

Die Methode ist für das Erstellen eines Objektes mit den angegebenen Parametern zuständig. Dabei wird jedem „Evaluator“ Objekt ein Dictionary mit positiven Regeln und eines mit negativen Regeln sowie ein Boolean ob der Bewertende diskriminierend ist und ein Prozentsatz, welcher die Stärke der Diskriminierung angibt, übergeben. Diese vier Angaben speichert jedes Objekt für sich und kann intern für sich selbst darauf zugreifen. Damit können dann die gewünschte Anzahl an Bewertenden und Bewertende welche diskriminieren erstellt werden. Durch die Angaben der eigenen Regeln, haben dann die diskriminierenden Bewertenden die Bias Regeln und die neutralen Bewertenden die normalen Regeln. Zudem kann durch das Flag als Boolean ein diskriminierender von einem neutralen Bewerter unterschieden werden. Die zweite Methode der Klasse, welche „Evaluator“ implementiert und von jedem Objekt der Klasse verwendet werden kann, ist in dem folgenden Listing 3.4 dargestellt.

```

1 def rate(self, request, bias):
2     #First 50/50 distribution
3     pos = 50
4     #Calculate the proportion according to which the decision is
5         #influenced positively or negatively.
6     prop = 45/self.rule_pos.__len__()
7     #Depending on how the rules match the request, the weight of the
8         #positive evaluation is shifted.
9     for key in self.rule_pos.keys():
10         if(self.rule_pos[key].__contains__(request[key])):
11             pos += prop
12         else:
13             pos -= prop
14     try:
15         #If a bias is present, this is additionally taken into account
16             #with the Parameter in %
17         if(self.bias):
18             for b in bias:
19                 if(bias[b].__contains__(request[b])):
20                     pos = pos*self.bias_percentage
21             #Normalise positive value
22             pos = pos/100
23             #Determine negative value
24             neg = 1-pos
25             #Rating by chance with indication of pos and neg rating and
26                 #adding the rating to the request.

```

```

23     request["Bewertung"] = np.random.choice(["positiv", "negativ"],
24         p=[pos, neg])
25     except:
26         print("Failure")
27     return request

```

Listing 3.4: Methode eines Bewertenden zum Bewerten von Anträgen

Die hier gezeigte Methode „rate“ wird von den Objekten der Klasse „Evaluator“ zum Bewerten eines Antrages verwendet. Als Parameter werden zum einen mit „self“ das Objekt welches die Methode aufruft, mit „request“ der zu bewertende Antrag und mit „bias“ die Verzerrung welche ausgewirkt werden soll übergeben. Als Rückgabe wird der erhaltene Antrag mit Ergänzung der Bewertung zurückgegeben. Im Nachfolgenden wird erläutert wie die Bewertung in der Methode abläuft. Zu Beginn wird die Wahrscheinlichkeitsverteilung zwischen der positiven und negativen Bewertung auf 50 zu 50 Prozent gesetzt. Somit ist die Entscheidung noch offen. Danach wird eine Proportion bestimmt, durch welche sich eine im Antrag positive oder negative Ausprägung auf die Wahrscheinlichkeitsverteilung der Entscheidung auswirkt. Diese wird wie folgt errechnet: 45 geteilt durch die Anzahl an Attributen in den Regeln. Es wird mit 45 gerechnet, da es immer noch eine rest Wahrscheinlichkeit geben soll, falls alle Ausprägungen im Antrag positiv oder negativ ausfallen. Als nächstes werden die Ausprägungen der Attribute im Antrag durch eine Schleife mit den Regeln verglichen, um aus den Regeln zu entscheiden, ob die Wahrscheinlichkeitsverteilung der Entscheidung die Proportion hinzu oder abgezogen wird. So tendiert die Entscheidung am Ende mehr zu einer positiven oder negativen Entscheidung abhängig von den im Antrag angegebenen Ausprägungen und den Regeln des Bewertenden. Um nach der Bewertung noch eine potentielle Diskriminierung einzubringen, wird überprüft ob das Flag des Bewertenden zur Diskriminierung gesetzt ist. Wenn die Bewertende Person eine diskriminierende und die gewünschte Bias Ausprägung in dem Antrag vorhanden ist, wird die Wahrscheinlichkeit für eine positive Auswirkung durch die Code Zeile: „pos = pos*self.bias_percentage“ mit dem für den Bewertenden angegebenen Prozentsatz verringert. Somit ist dadurch die Möglichkeit auf eine positive Bewertung deutlich gesunken. Nachfolgend wird dann noch die Wahrscheinlichkeitsverteilung für die Entscheidung passend umgewandelt, um beruhend auf diesen Wahrscheinlichkeiten die Entscheidung zu treffen und das Ergebnis im Antrag also dem Dataframe anzuhängen. Zum Schluss wird der überarbeitete Antrag wieder zurückgegeben und das Bewerten ist abgeschlossen.

Nach Abschluss der Klasse werden nun noch zwei Methoden benötigt, welche den gesamt Ablauf durchführen und die anderen Methoden vereinen. Zum einen die Methode „generate_data“, welche die Anzahl an zu generierenden Datensätzen übermittelt bekommt. In der Methode wird zuerst ein Seed durch Aufruf der oben beschriebenen Methode zum

Seed/Startwert generieren erzeugt. Danach kann die Methode für das Datengenerieren mit dem Seed und der Anzahl an Daten aufgerufen werden. Der Rückgabewert wird dann in ein neuen Pandas Dataframe geschrieben und zurückgegeben. Die andere Methode, welche implementiert wird, ist für den gesamt Ablauf der Datenbewertung zuständig. Diese trägt den Namen „work“ und bekommt den Datensatz, die gewünschte Verzerrung, die Anzahl der Bewertenden, die Anzahl der diskriminierenden Bewertenden und die Prozentual Auswirkung der Verzerrung als Parameter übergeben. Als erstes werden in der Methode die beiden benötigten Dictionaries für „request_values“ und „request_bias“ angelegt und mit den Werten des Szenarios nach dem Schema wie in dem obigen Listing 3.2 zu sehen gefüllt. Danach werden daraus die Regeln bestimmt und gespeichert. Im Anschluss wird die gewünschte Anzahl an „Evaluator“ Objekte erzeugt und diesen die Regeln übergeben. Um die Verzerrung umzusetzen wird danach die gewünschte Anzahl der „Evaluator“ Objekte in diskriminierende Bewertende umgewandelt und dessen Regeln ausgetauscht durch die Bias Regeln. Damit sind alle Vorbereitungen abgeschlossen und das eigentliche Bewerten der Anträge kann beginnen. Dafür wird eine Schleife über den Dataframe der Anträge durchlaufen. Für jeden Antrag wird dann zufällig bestimmt, welches „Evaluator“ Objekt den Antrag bewertet. Nachdem der Antrag bewertet wurde wird dieser der Liste der fertigen Anträge hinzugefügt. Sobald alle Anträge bewertet sind und die Schleife daher durchlaufen ist, wird die Liste der fertigen Anträge zu einem Dataframe umgewandelt und aus der Methode zurückgegeben. Damit sind auch die letzten beiden Methoden umgesetzt und es muss letztendlich nur noch eine finale Zelle zur Ausführung des gesamten Programmes erstellt werden.

```

1 #Here is the section for the possible parameters to enter
2 #This dictionary specifies the bias(es) on a possible attribute
3 bias = {
4     "Hautfarbe": ["Weiß"]
5 }
6 #The number of datasets that are to be generated
7 datasets=10000
8 #The number of evaluators who evaluate entries
9 evaluator_count=10
10 #The number of evaluators who evaluate with a bias
11 bias_evaluator=4
12 #This decides how strong the bias will be. The higher the stronger.
13 bias_percentage=0.2
14
15 #Dont touch this
16 data = generate_data(datasets)
17 finished = work(data,bias,evaluator_count,bias_evaluator,bias_percentage
    )

```

```
18 data.to_csv("Daten.csv", sep=';', encoding='utf-8', index=False)
19 finished.to_csv("Daten_Bewertet.csv", sep=';', encoding='utf-8', index=
    False)
```

Listing 3.5: Letzte Zelle des Szenario der Bewährungsantrag für die Interaktion des Benutzenden

Im Listing 3.5 ist die letzte Zelle für die Interaktionen des Benutzenden dargestellt. Im oberen Teil der Zelle haben die Benutzenden die Möglichkeit, Anpassungen am Datengenerator zu machen. So kann hier die gewünschte Verzerrung, die Größe des Datensatzes, die Anzahl an Bewertenden, die Anzahl der diskriminierenden unter den Bewertenden und der Prozentuale Einfluss der Verzerrung angepasst werden. Im unteren Teil wird dann die Hauptmethode zum Datengenerieren (generate_data) aufgerufen und im Anschluss durch die Methode „work“ die zuvor generierten Anträge bewertet. Beide Datensätze werden separat als Variablen geführt, sodass am Ende der Methode die Ursprungsdaten als „Daten.csv“ und der bewertete Datensatz als „Daten_Bewertet.csv“ abgespeichert werden.

Insgesamt ist mit dieser letzten Zelle die gesamte Umsetzung des ersten Szenarios als „Szenario1.ipynb“ Datei abgeschlossen und kann so direkt verwendet werden, um Daten zu generieren.

3.3.2 Umsetzung des Szenario zur Vorhersage von einem sozialen Punktesystem

Das zweite Szenario wird ebenfalls wie das erste in eine Python Notebook umgesetzt. Das File trägt den Namen „Szenario2.ipynb“ und beginnt sowie auch das erste Szenario mit dem laden der benötigten Bibliotheken. Auch in diesem werden die gleichen Bibliotheken obig zum ersten Szenario beschrieben geladen. Einzige Ausnahme hierbei liegt darin, dass das Faker Objekt nicht wie in der Abbildung gezeigt ist erstellt wird. In diesem Fall wir es wie folgt erstellt: „fake = Factory.create("de_DE")“. Der Grund dafür, liegt darin, dass in diesem Szenario Deutsche Namen generiert werden sollen, da es sich um ein auf Deutschland angelegtes Szenario handelt. In der zweiten Zelle des Notebooks wird die Methode für das erstellen eines Seed/Startwert aus dem Szenario der Bewährungsanträge implementiert. Diese verändert sich nicht und kann daher so verwendet werden.

Als nächstes ist die Methode „create_fake_data“ mit den Parameter num und seed umgesetzt. In dieser wird die durch num angegebene Anzahl an Daten generiert. Der Parameter seed wird zu Beginn der Methode als Startwert für die Zufallsmethoden von numpy und Faker gesetzt. Als nächstes wird eine Schleife über num begonnen, in welcher in jedem

Durchlauf einen Eintrag für den Datensatz generiert wird. In jedem Durchlauf müssen die Ausprägungen der für dieses Szenario bestimmte Attribute Name, Alter, Politische Orientierung, Bildungsabschluss, Soziales, Wohnlage und CO2-Fußabdruck, unter Betrachtung der im Kapitel 3.2 angegebenen Abhängigkeiten, bestimmt werden. Der Name eines Eintrages wird durch die Faker Instanz wie folgt „Name“: fake.first_name()“ generiert. Die anderen Ausprägungen der Attribute werden, um das Szenario so real wie möglich zu halten, beruhend auf Wahrscheinlichkeiten aus Statistiken bestimmt. Die Wahrscheinlichkeiten der verschiedenen Attributen ergeben sich wie folgend aufgeführt:

Alter

Ein Alter für eine Person wird zwischen 20 und 79 Jahren bestimmt. Zur Bestimmung wird aufgrund der verwendeten Statistik in drei Altersgruppen unterteilt, welche unterschiedliche Wahrscheinlichkeiten besitzen. Die Altersgruppen sind von 20 bis 39, 40-59 und 60 bis 79 Jahren.

Die Formel zur Berechnung der Wahrscheinlichkeiten:

Gegeben: Gesamt(Personen im Alter zwischen 20-79) = 20,36 + 23,38 + 18,15 = 61,89 Millionen

$$Altersgruppe\% = \frac{Altersgruppe}{Gesamt} \quad (3.3)$$

Altersgruppen	Berechnung	Wahrscheinlichkeit
20-39	$\frac{20,36}{61,89}$	33%
40-59	$\frac{23,38}{61,89}$	38%
60-79	$\frac{18,15}{61,89}$	29%

Tabelle 3.7: Tabelle zur Bestimmung der Wahrscheinlichkeiten für die Altersgruppen

Die in der Tabelle 3.7 nach der Formel 3.3 errechneten Wahrscheinlichkeiten beruhen auf der Statistik des Statistischen Bundesamt vom 31. Dezember 2020.[44] Die Zahlen wurden passend zu den Altersgruppen aus der Statistik entnommen. Um nun aus den errechneten Wahrscheinlichkeiten auf ein Alter zu kommen, wird im Code zu erst die Altersgruppe nach den Wahrscheinlichkeiten bestimmt. Nachdem die Altersgruppe klar ist, wird eine Zufallszahl im Bereich der Altersgruppe bestimmt, welche dann das endgültige Alter ist. Dadurch wird eine gute Verteilung des Alters entsprechend der Statistik gewährleistet.

Politische Orientierung

Bei der Bestimmung der Politischen Orientierung in Links, Mitte und Rechts wird wie in der Abbildung 3.3 aus Kapitel 3.2.2 dargestellt, dass Alter als Grundlage mit verwendet.

Daher ist die Berechnung und damit auch die Wahrscheinlichkeitsverteilung pro Altersgruppe für die Politische Orientierung unterschiedlich. Des weiteren wird, um die Daten so realitätsnahe wie möglich zu halten, eine statistische Auswertung des Wahlverhaltens bei der Bundestagswahl 2017 nach Alter hinzugezogen. Da in der Statistik die Ergebnisse in Prozent pro Partei angegeben sind, wurde nach eigenem Ermessen entscheiden, welche Partei zu welcher Ausprägung zugeordnet wird. Zur Ausprägung Mitte gehört dabei die CDU, SPD, CSU und FDP zu Links Die Linke und Grüne und zu Rechts wird die AFD zugeordnet. Daraus lassen sich dann die in der folgenden Tabelle 3.8 aufgelisteten Wahrscheinlichkeiten nach dem Alter berechnen.[45, S. 33]

Ausprägungen	20-24	25-34	35-44	45-59	60-69	70-79
Links	28,0%	24,4%	21,6%	20,7%	17,7%	10,7%
Mitte	63,1%	61,4%	61,8%	63,5%	68,7%	80,9%
Rechts	8,9%	14,2%	16,6%	15,8%	13,6%	8,4%

Tabelle 3.8: Tabelle der Wahrscheinlichkeiten für die Politische Orientierung nach Alter

Die in der Tabelle 3.8 dargestellten Wahrscheinlichkeiten wurden nach den dafür angegebenen Altersgruppen aus der Tabelle der Veröffentlichung von der Konrad-Adenauer-Stiftung e. V. berechnet. Dafür wurden die Prozente der oben zugeordneten Parteien aufsummiert. So ergeben sich die Prozentpunkte für Links, Mitte und Rechts. Jedoch fehlen zu den 100% pro Altersgruppe die Prozentpunkte der Sonstigen Parteien. Da diese nämlich keiner Richtung zugeordnet werden können müssen die Prozente der sonstigen Insgesamt verrechnet werden. Nach der Verrechnung der Sonstigen Parteien entstehen dann die in der Tabelle dargestellten Wahrscheinlichkeiten.[45, S. 33] Zur Umsetzung der Bestimmung der Politischen Orientierung wird eine Reihe an ifelse Abfragen über das zutreffende Alter durchlaufen, um dann mit den Altersentsprechenden Wahrscheinlichkeiten die Politische Orientierung zu bestimmen. Der Code für die Wahrscheinlichkeitsauswahl sieht dabei genau wie im Beispiel aus dem Listing 3.1 dargestellt aus.

Bildungsabschluss

Der Bildungsabschluss einer Person basiert ebenfalls wie die Politische Orientierung auf dem Alter. Zusätzlich wird wie auch schon zuvor eine Statistik aus einem Datenreport des Jahres 2021 vom Statistischen Bundesamt verwendet. [46] Dadurch können die Daten so realitätsnahe wie möglich gehalten werden.

Die in der Tabelle 3.9 dargestellten Wahrscheinlichkeiten für die dort aufgeführten Altersgruppen ergeben sich aus der „Tab 9“ des Datenreports vom Statistischen Bundesamt. Die Daten der „Tab 9“ stammen aus dem Jahr 2018 der deutschen Bevölkerung. Für die Altersgruppe von 20 bis 29 Jahre existiert keine passende Angabe in der Quelle, da in

Ausprägungen	<=29	30-39	40-49	50-59	>=60
Ausbildung	42,5%	45,2%	51,6%	55,9%	54,4%
Fachschulabschluss	8,1%	9,1%	9,7%	11,3%	9,6%
Bachelor	11,7%	5,9%	1,4%	0,5%	0,2%
Master	7,1%	5,2%	1,1%	0,3%	0,1%
Diplom	5,9%	15,7%	17,9%	15,9%	13,2%
Promotion	0,3%	1,6%	1,7%	1,4%	1,2%
Ohne	24,4%	17,3%	16,6%	14,7%	21,3%

Tabelle 3.9: Tabelle der Wahrscheinlichkeiten für den Bildungsabschluss nach Alter

dieser erst ab dem Alter von 25 Jahren Werte aufgelistet sind. Einfachheitshalber wird daher für die Personen kleiner gleich 29 Jahre die Werte der Personen zwischen 25 und 29 Jahren genommen. Dasselbe gilt auch für die Personen älter als 60, hier werden in der Quelle nur eine Wertereihe für Personen mit 60 und älter angegeben. Das Szenario bezieht sich jedoch nur bis 79 Jahre und passt somit hier auch nicht vollkommen. Zur Vereinfachung wird hierbei jedoch diese Wertereihe für alle zwischen 60 und 79 Jahren verwendet.[46]

Soziales

Das soziale Engagement wird von 0 bis 3 angegeben und basiert wie in der Konzeption beschrieben, um die Daten interessanter zu machen, auf der Politischen Orientierung. Durch solche Abhängigkeiten gestalten sich die Daten nämlich realitätsnäher und bringen eine größere Variabilität mit. Da jedoch für die Werte keine Referenzen aus der realen Welt zur Verfügung stehen, wird mit fiktiven Werten gearbeitet.

Ausprägungen	Links	Mitte	Rechts
0	40%	60%	75%
1	25%	20%	15%
2	25%	15%	9%
3	10%	5%	1%

Tabelle 3.10: Tabelle der Wahrscheinlichkeiten für das Soziale Engagement nach Politischer Orientierung

Die in der Tabelle 3.10 aufgeführten fiktiven Wahrscheinlichkeiten sind ebenso wie schon bei der Bestimmung der anderen Attribute, durch den Code aus Listing 3.1 umgesetzt. Jedoch wird durch eine if else Abfrage die unterschiedlichen Politischen Orientierungen unterschieden und dementsprechend die passenden Wahrscheinlichkeiten in der Codezeile verwendet.

Wohnlage

Die Wohnlage wird durch die Ausprägungen Ländlich, Vorort, Kleinstadt und Großstadt

KAPITEL 3. PRAKTISCHER TEIL

angegebenen und wird durch kein anderes Attribut beeinflusst. Die Wahrscheinlichkeiten für die einzelnen Ausprägungen sind in nachfolgender Tabelle aufgezeigt.

Ausprägungen	Einwohnergrenze	Wahrscheinlichkeit
Ländlich	<= 2000	5,38%
Vorort	2.000 - 5000	8,5%
Kleinstadt	5.000 - 20.000	26,6%
Großstadt	>20.000	59,52%

Tabelle 3.11: Tabelle der Wahrscheinlichkeiten für die Wohnlage

Zur Bestimmung der Wahrscheinlichkeiten für die unterschiedlichen Ausprägungen der Wohnlage müssen zuerst die in der Tabelle 3.11 angegebenen Einwohnergrenzen pro Ausprägung festgelegt werden. Für die Festlegung wurde nach eigenem Ermessen entschieden, welche Ausprägung zu welcher Einwohnergrenze passt. Aus einer Publikation des Statistischen Bundesamt können dann nach dieser Eingliederung die Wahrscheinlichkeiten aufsummiert werden. In der Publikation „Gemeinden nach Bundesländern und Einwohnergrößenklassen am 31.12.2020“ sind der prozentuale Anteil an deutschen Einwohner in gewissen Einwohnergrenzen angegeben. Diese werden dann nach den hier für die Ausprägungen definierten Grenzen aufsummiert und in der Tabelle angegebenen. Damit kann eine möglichst Realitätsnahe Datenerzeugung gewährleistet werden. Für die Umsetzung der Auswahl der Ausprägung kann wie auch schon in den vorigen Fällen der selbe Code aus Listing 3.1 auf den Fall hier angepasst und verwendet werden. So sieht die finale Codezeile wie in der folgenden Abbildung dargestellt aus.

```
1 location = np.random.choice(["Großstadt", "Kleinstadt", "Vorort", "Ländlich"] , p=[0.5952, 0.266, 0.085, 0.0538])
```

Listing 3.6: Codezeile zur Auswahl der Ausprägung der Wohnlage basierend auf angegebenen Wahrscheinlichkeiten

CO2-Fußabdruck

Der CO2-Fußabdruck ist das komplexeste Attribut des Szenarios, da dieser auf zwei Attributen basiert. Wie in der Abbildung 3.3 dargestellt hat das Soziale Engagement und die Wohnlage eine Auswirkung auf den CO2-Fußabdruck. Dadurch, dass das Soziale Engagement ebenfalls noch von der Politischen Orientierung und diese vom Alter abhängig ist, haben auch noch zwei weitere Attribute eine indirekte Auswirkung auf den CO2-Fußabdruck. Dies lässt sich auch dadurch begründen, dass der CO2-Fußabdruck in der Realität sich aus vielen unterschiedlichen Teilen zusammensetzt. Da es für den CO2-Fußabdruck unter Berücksichtigung dieser Auswirkungen keine Referenzen aus der realen Welt gibt, wurden fiktive Werte für eine Gaußkurve verwendet. Als Referenzen für die Angabe der Ausprägung des CO2-Fußabdruck in Tonnen pro Kopf konnte jedoch der

Mittelwert von 7,91 Tonnen herangezogen werden.[47] Um daraus ein Intervall um diesen Wert als Mittelwert festzulegen. So ergab sich das Intervall von 4 bis 12 Tonnen in 0,5er Schritten. Für die daraus resultierenden 17 Ausprägungen wurde eine Gaußkurve erzeugt. Um die Auswirkung von Sozialem Engagement und Wohnlage durchzusetzen wird die Gaußkurve, je nach Grundlage der Attribute verschoben. Für die Umsetzung der jeweiligen Verschiebung durch die beiden Attribute Soziales Engagement und Wohnlage muss eine große ifelse Abfragestruktur mit 16 unterschiedlichen Zuständen erstellt werden, da beide Attribute 4 verschiedene Ausprägungen haben($4*4=16$). In den 16 unterschiedlichen Zuständen wird dann die Codezeile aus dem Listing 3.6, um die 17 Ausprägungen des CO2-Fußabdruck angepasst und die Wahrscheinlichkeiten der Gaußkurve entsprechend dem Zustand eingetragen. Dadurch entsteht ein recht komplexes Codekonstrukt, welches jedoch sehr anschauliche abhängige Daten liefert und damit das Szenario in der Auswertung interessanter gestaltet.

Mit der abschließenden Bestimmung des CO2-Fußabdruck sind alle Ausprägungen der Attribute bestimmt und die Werte können nun als ein Dictionary dem Array des gesamt Datensatzes hinzugefügt werden.

```

1 output.append(
2     {
3         #Name of the person
4         "Name": fake.first_name(),
5         "Alter": age,
6         "Politische Orientierung": politics,
7         "Bildungsabschluss": grad,
8         "Soziales": social,
9         "Wohnlage": location,
10        "CO2-Fußabdruck": co2
11    }
12)

```

Listing 3.7: Codeausschnitt zum Hinzufügen eines Dateneintrags zum gesamt Datenset

Im Listing 3.7 ist der Codeausschnitt zu sehen, wie das Dictionary mit der Zuordnung Attribut als Key und die Ausprägung als Value erstellt und dem Array „output“ angehängt wird. Die Ausprägung des Namens wird in diesem Schritt erst bestimmt und die anderen werden durch die Variablen übertragen. Im Anschluss ist ein Schleifendurchlauf beendet, somit ein Eintrag dem Datensatz hinzugefügt und es wird falls noch offen der nächste Durchlauf begonnen. Sobald die Schleife beendet ist, ist der gesamte gewünschte Datensatz generiert und wird letztendlich nur noch als Array aus der Methode zurückgegeben.

Als nächste Zelle wird wie auch schon im ersten Szenario eine Methode, welche das Daten erstellen zusammenfasst umgesetzt. Diese ist genau gleich wie im Kapitel 3.3.1 beschrieben

umgesetzt. Zuerst wird ein Seed und danach mit diesem die gewünschte Anzahl an Daten generiert, als Dataframe gespeichert und zurückgegeben.

In der darauffolgenden Zelle ist die Methode „create_Rules()“ für das erstellen der Regeln implementiert. Diese Methode ist deutlich einfacher umgesetzt, als sie im Szenario der Bewährungsanträge implementiert werden musste. In diesem Szenario werden die Regeln fix als Dictionary angegeben erstellt.

```

1 def create_Rules():
2     influential = {
3         "Politische Orientierung": [-50, 50, -50],
4         "Bildungsabschluss": [20, 40, 60, 80, 100, 120, -120],
5         "Soziales": [-70, 0, 40, 100],
6         "Wohnlage": [-30, 20, 20, -30],
7         "CO2-Fußabdruck": [110, 70, 40, 10, 0, -20, -40, -70, -110]
8     }
9 return influential

```

Listing 3.8: Codeausschnitt zum Erstellen des Regel Dictionary

Das Dictionary der Regeln, welches in der Methode erstellt und zurückgegeben wird, ist in dem Listing 3.8 dargestellt. Für jedes Attribut, welches in der Konzeption unter 3.1.2 als beeinflussendes Attribut eingestuft wird, muss ein Key Value Paar dem Dictionary hinzugefügt werden. Als Keys werden die Attribute und in den Values nach der in der zur Generierung angegebenen Reihenfolge eine Punktzahl für jede Ausprägung angegeben. Die angegebenen Punkte pro Ausprägung beeinflussen dann in diesem Maße die soziale Punktzahl einer Person. So kann der Bewertende für jedes Attribut nach der Ausprägung und der zugehörigen Punktzahl in den Regeln schauen. So setzt sich dann Stück für Stück die soziale Punktzahl einer Person zusammen. Die Punkte für die jeweiligen Ausprägungen wurden fiktiv bestimmt. Dabei wurde darauf geachtet, dass es in jedem Attribut sowohl positive als auch negative Auswirkungen von Ausprägungen vorhanden sind. Zudem darf die maximal mögliche Punktzahl nicht 1400 übersteigen und die minimale nicht unter 600 fallen. Zum Schluss wird das erstellte Dictionary aus der Methode zurückgegeben und kann dann als Regelwerk zur Bewertung verwendet werden.

Im nächsten Schritt ist wie auch schon im ersten Szenario die Klasse „Evaluator“ mit der Methode „__init__“ zum erstellen eines Objektes und „rate“ für das Bewerten von Personen. Für die Initialisierung eines „Evaluator“ Objektes wird zum einen ein Regel Dictionary, ein Boolean ob die Bewertende diskriminierend ist und eine Punktzahl für die negative Auswirkung einer Diskriminierung benötigt. Diese Parameter speichert dann jedes erzeugte Objekt einzeln für sich. Die zweite Methode für das Bewerten der Personen bekommt die drei Parameter „influential“, „person“ und „bias“ übergeben. In „influential“ sind als Dictionary alle Attribute des Szenario bis auf das Alter und der Name mit deren

Ausprägungen gegeben. In dem Parameter „person“ ist der Datenpunkt der zu bewertenden Person enthalten und in „bias“ ist die eine oder mehrere gewünschte Verzerrungen als Dictionary angegeben. Zu Beginn der Methode wird die momentan Bewertete Punktzahl auf 1000 Punkte gesetzt.

```

1 for key in self.rules.keys():
2     if(key == "CO2-Fußabdruck"):
3         value_of_key = int(person[key])
4         index = influential[key].index(value_of_key)
5         rate += self.rules[key][index]
6         procent = self.rules[key][index] * 0.15
7         procent = round(procent,0)
8         if(procent<0):
9             rand = random.randint(0,(procent*-1))
10            rate -= int(rand)
11        else:
12            rand = random.randint(0,procent)
13            rate += int(rand)

```

Listing 3.9: Codeausschnitt der Funktion zum Bewerten von Personen

Daraufhin folgt die in dem Listing 3.9 zu teilen dargestellte Schleife über alle Keys, somit Attributen, in den Regeln des Bewerters. In der Schleife wird für jedes Attribut dann die Auswirkung auf die Punktzahl bestimmt. Dafür muss zuerst überprüft werden, ob es sich um das Attribut CO2-Fußabdruck handelt oder nicht, da bei diesem nicht für jeden Komma fünf Wert eine Auswirkung in den Regeln definiert ist. Somit hat 4 und 4,5 die gleiche Auswirkung. Diese Besonderheit des CO2 Fußabdruck muss daher durch die dritte Codezeile aus dem Listing 3.9 mithilfe von Parsen des Values zu einem Integer besonders behandelt werden. Die darauf folgenden Zeilen Code sind jedoch für jedes Attribut dieselben. In der nächsten Zeile wird für die erhaltene Ausprägung des momentan in der Schleife bearbeitenden Attributes der Index in der Liste von allen Ausprägungen dieses Attributes bestimmt. Mit dem Index kann daraufhin nämlich die Auswirkung dieser Ausprägung aus den Regeln entnommen und auf die momentane Punktzahl addiert werden. Damit ist die Auswirkung schon mal in die Punktzahl der Person mit ein berechnet. Um die Zahlen jedoch interessanter zu halten, wird danach noch ein zusätzlicher prozentualer Wert der Auswirkung bestimmt, welcher ebenfalls auf die Punktzahl berechnet wird. Dafür werden 15 Prozent der Auswirkung genommen und dann ein Zufallswert zwischen 0 und diesen 15 Prozent bestimmt und der Punktzahl der Person hinzugefügt beziehungsweise (bzw.) abgezogen. Danach ist der Durchlauf für dieses eine Attribut beendet und die Punktzahl wurde entsprechend der Ausprägung der zu bewertenden Person im Bezug auf die Regeln des Bewertenden angepasst. Sobald die Schleife fertig ist, wird im nächsten Schritt überprüft ob die Bewertende Person diskriminierend ist. Wenn dies der Fall ist, beginnt eine

Schleife über alle im Parameter „bias“ angegebenen gewünschten Verzerrungen. Wie auch schon zuvor gibt es hier einen Sonderfall für das Attribut Alter, da die gewünschte Verzerrung für ein Alter in einem Intervall angegeben wird. Daher muss hier überprüft werden, ob das Alter der zu bewertende Person im Intervall liegt oder nicht. Der nachfolgende Code ist dann der selbe für alle anderen Attribute bis auf den Namen.

```

1 if(bias[b].__contains__(person[b])):
2     red_val = self.bias_neg
3     if((rate-red_val)<600):
4         rate_part = np.random.choice([0,1,2], p=[0.4,0.35,0.25])
5         biasrate = random.randint(600,610) if(rate_part == 0) else
6             random.randint(611,620) if(rate_part == 1) else random.
7                 randint(621,630)
8         rate = biasrate
9     else:
10        rate-=red_val

```

Listing 3.10: Codeausschnitt für das Hinzufügen einer Verzerrung beim Bewerten der Personen

In der ersten Zeile des Ausschnittes von Abbildung 3.10 wird dann überprüft, ob die Person die zu diskriminierende Ausprägung des Attributes besitzt. Wenn dies der Fall ist, muss überprüft werden, dass wenn die Auswirkung des Bias der Person von der Punktzahl abgezogen wird, diese trotzdem noch über 600 liegt. Falls die Punktzahl dann unter 600 liegt, wird eine neue Punktzahl zu 40% zwischen 600-610 oder zu 35% zwischen 611-620 oder zu 25% zwischen 621-630 zufällig für die Person bestimmt. Dadurch wird gewährleistet, dass nicht sehr viele Personen am Ende eine Punktzahl von genau 600 haben. Wenn die Punktzahl nicht unter 600 fallen würde, wird die Auswirkung des Bias von der Punktzahl der Person abgezogen.

Zum Abschluss der Funktion wird nochmals überprüft ob die Grenzen der Punktzahl überschritten wurden und falls zutreffend diese auf die jeweilige Grenze zurück gesetzt. Danach wird dem Datenpunkt des Dataframes die Bewertung als Integer hinzugefügt und der Datenpunkt kann zurückgegeben werden.

Als nächste und letzte Methode wird die Methode für den gesamt Ablauf des Bewertens Namens „work“ umgesetzt. Diese ist sehr ähnlich wie auch schon beim ersten Szenario umgesetzt. Zu aller erst ist ein Dictionary mit allen Attributen, bis auf Name und Alter, und deren Ausprägungen implementiert. Danach wird durch Aufruf der Methode zum erstellen der Regeln, diese in eine Variable gespeichert. Im Anschluss kann die gewünschte Zahl an „Evaluator“ Objekten unter Angabe der zuvor erstellten Regeln erstellt werden. Von diesen Objekten wird dann noch die gewünschte Anzahl zu diskriminierenden umgewandelt und diesen die negative Auswirkung für eine Verzerrung übergeben. Zum Schluss

der Methode wird über den erhaltenen Datensatz iteriert und jeder Datenpunkt von einem zufälligen „Evaluator“ Objekt bewertet. Nach der Bewertung werden die bewerteten Datenpunkte separat abgespeichert und als Dataframe aus der Methode zurückgegeben. Damit ist die letzte Methode vollends implementiert und es kann in einer letzten Zelle des Notebooks der gesamt Ablauf zusammengefasst werden.

```

1 #Here is the section for the possible parameters to enter
2 #This is an example of how a bias on age can be indicated
3 age_sample = {
4     "Alter": "20-30"
5 }
6 #This dictionary specifies the bias(es) on a possible attribute
7 bias = {
8     "Alter": "20-30",
9     "Politische Orientierung": ["Links"]
10 }
11 #The number of datasets that are to be generated
12 datasets = 10000
13 #The number of evaluators who evaluate entries
14 evaluatorcount = 10
15 #The number of evaluators who evaluate with a bias
16 bias_evaluator = 4
17 #This decides how strong the bias will be. The higher the stronger.
18 negativ_bias_impact = 200
19
20 #Dont touch this
21 data = generate_data(datasets)
22 finished = work(df=data,bias=bias,evaluator_count=evaluatorcount,
23                   bias_evaluator=bias_evaluator,bias_neg=negativ_bias_impact)
23 data.to_csv("Daten_Szenario2.csv", sep=';', encoding='utf-8', index=
24 False)
24 finished.to_csv("Daten_Bewertet_Szenario2.csv", sep=';', encoding='utf-8',
25 , index=False)

```

Listing 3.11: Letzte Zelle des Szenario des sozialen Punktesystems für die Interaktion des Benutzenden

In der Abbildung 3.11 ist die letzte Zelle dargestellt. Im oberen Bereich kann die Benutzende Person Anpassungen vornehmen, wie den Bias festlegen, die Datenset Größe bestimmen, die Anzahl der Bewertenden und diskriminierenden festlegen sowie die Auswirkung des Bias bestimmen. Als Beispiel wurden hierbei zwei mögliche Versionen für einen Bias angegeben. Im unteren Teil ist der Ablauf des gesamt Programms abgebildet. Zuerst werden die Daten durch die „generate_data“Methode in der gewünschten Anzahl generiert. Danach werden diese Daten mit Hilfe der „work“ Methode bewertet und als

KAPITEL 3. PRAKTISCHER TEIL

neuen Dataframe zurückgegeben. Zum Schluss werden dann noch die Rohdaten als „Daten_Szenario2.csv“ Datei und die bewerteten Daten als „Daten_Bewertet_Szenario2.csv“ Datei abgespeichert. Hiermit ist die gesamte Umsetzung beider Szenarien abgeschlossen und beide Szenarien können getrennt voneinander als Notebooks verwendet werden, um Daten zu generieren.

Im nächsten Kapitel wird die Auswertung der generierten Daten beschrieben.

3.4 Datenauswertung

Für die Auswertung und Analyse der generierten Daten wird das Tool Tableau verwendet. In diesem können durch Drag and Drop vorgegebene Arten an Grafiken auf Basis der eingefügten Daten erstellt werden. Da die generierten Daten im CSV Format ausgegeben werden, können diese direkt so in Tableau geladen werden. Dabei wird auch direkt die Struktur der Daten beibehalten, sodass diese sofort verwendet werden können.

Um die Auswertung zu strukturieren wurde für jedes Szenario ein eigenes Tableau mit eigenen Grafiken erstellt. In diesen sind dann mehrere Seiten mit jeweils einer Grafik angelegt und einem dazugehörigen Hintergrund, warum genau diese Attribute miteinander in welchem Diagramm verglichen werden.

In der Analyse bei der Suche nach einem Bias in der Bewertung der Daten wird für beide Szenarien wie folgt vorgegangen:

Zu Beginn wird eine Übersicht über die Bewertung allgemein dargestellt, um daraus auf mögliche interessante Punkte schauen zu können. Im nächsten Schritt wird eine Vermutung aufgestellt nach welcher die weiteren Attribute des Szenario Stück für Stück analysiert werden. Dabei werden die Attribute für sich alleine und in Kombination mit anderen durch weitere Diagramme betrachtet. Daraus können dann Zusammenhänge geschlossen oder widerlegt werden. Aus diesen Zusammenhängen entstehen auch Vermutungen auf eine mögliche Verzerrung durch Diskriminierung in den Daten. Bei einem Verdacht wird dieser weiter auf einer neuen Seite durch weitere Diagramme untersucht. Dabei muss für eine gefundene Verzerrung durch Diskriminierung klar deutlich sein, dass mehrere Personen mit gleich guten oder sogar besseren Attributen eine schlechtere Bewertung haben. Wenn dies der Fall ist, wurde die Verzerrung gefunden und die Analyse ist abgeschlossen. Zusätzlich können jedoch noch aus den Daten unterschiedlichste Zusammenhänge in den Daten gefunden und vermutet werden.

Szenario der Bewährungsanträge

Zu aller erst, um sich einen Überblick über die Daten zu verschaffen, wird mit Hilfe eines Diagramms die Verteilung von positiv und negativ bewerteten Anträgen veranschaulicht. Dafür wird ein Säulendiagramm über die Anzahl aller positiven und negativen Anträge farblich durch Grün und Rot getrennt erstellt.

Im Anschluss wird auf der Suche nach dem Lable Bias ein Attribut ausgewählt, welches durch eine Diskriminierung von einem Bewertenden zu einem Lable Bias führen könnte. In diesem Fall zum Beispiel die Hautfarbe oder das Geschlecht, da es naheliegend ist, dass eine bewertende Person etwas gegen eine gewisse Hautfarbe oder ein gewisses Geschlecht hat. Daraufhin wird in einem Diagramm die Verteilung der Hautfarbe auf die gesamt An-

KAPITEL 3. PRAKTISCHER TEIL

zahl der Daten veranschaulicht, um ein Gefühl für das Attribut Hautfarbe zu bekommen und für weitere Analysen dies mit Betrachten zu können. Falls zum Beispiel deutlich mehr weibliche als männliche Personen Anträge gestellt haben. Anschließend wird in zwei neuen Tableau die Hautfarbe im Zusammenhang mit der Bewertung auf die gesamt Anzahl der Daten überprüft. Einmal in der wirklichen Anzahl an Datenpunkten und ein andern Mal in Prozent wie in der folgenden Abbildung 3.5 dargestellt.

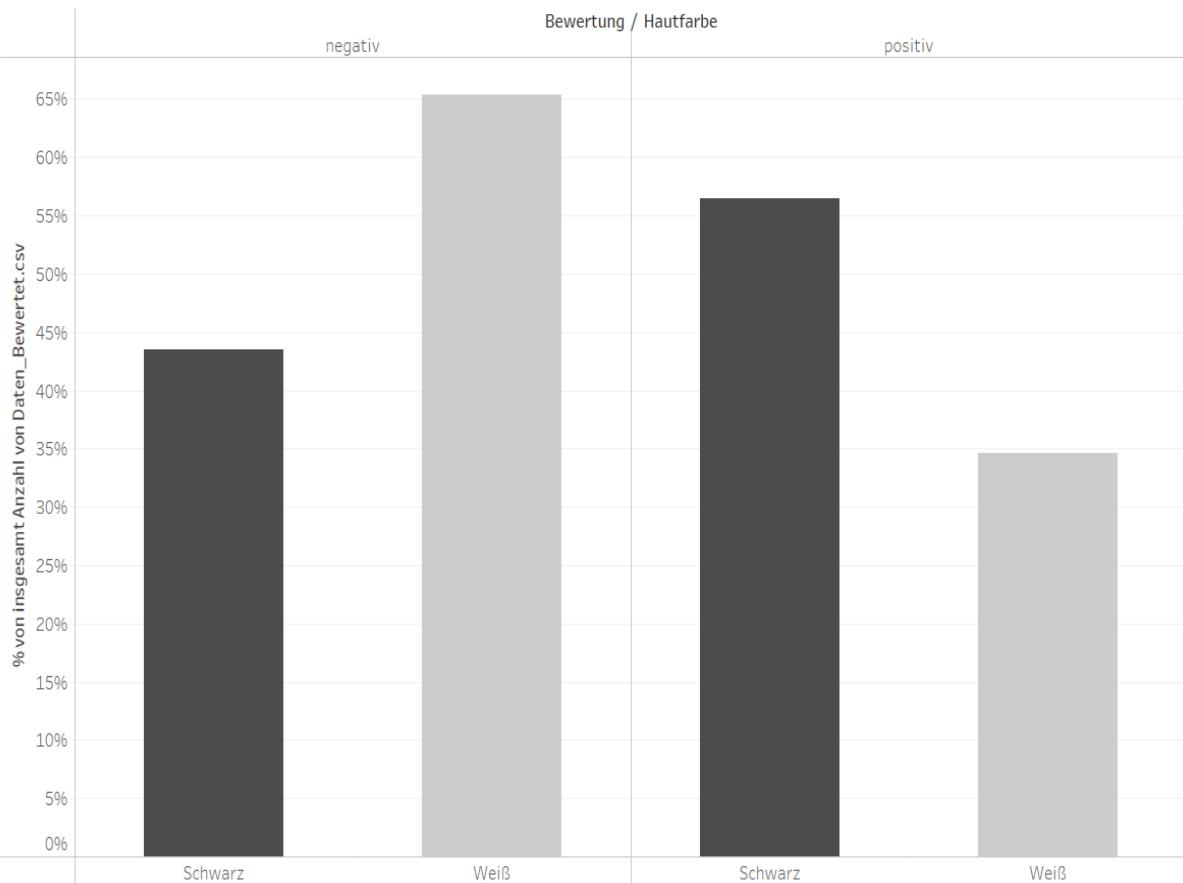


Abbildung 3.5: Auswertung der Hautfarbe im Zusammenhang mit der Bewertung nach Prozent von der gesamt Anzahl an Daten

In der obigen Abbildung 3.5 ist das Säulendiagramm mit hellgrauen Säulen für die Hautfarbe Weiß und dunkelgrauen Säulen für die Hautfarbe Schwarz dargestellt. Links ist ein Bereich für die negativen Bewertungen und rechts für die positiven. Auf der Y-Achse ist der Prozentsatz von der gesamt Menge an Daten angegeben. Dabei werden die Prozente der Säulen pro Hautfarbe angegeben. So ist zu sehen, dass über 65% der Personen mit Hautfarbe Weiß eine negative und knapp 35% eine positive Bewertung erhalten haben. Was verglichen zu der Hautfarbe Schwarz einen deutlichen Unterschied aufweist. Da knapp über 55% der Personen mit Hautfarbe Schwarz eine positive Bewertung erhalten haben, was über 20% mehr sind. Daraus lässt sich vermuten, dass hier möglicherweise etwas nicht ganz passend ist und eine mögliche Verzerrung vorliegt. Dies muss nun jedoch bestätigt

KAPITEL 3. PRAKTISCHER TEIL

werden, dass Personen mit der Hautfarbe Weiß eine schlechtere Chance auf eine positive Bewertung aufgrund der Hautfarbe haben. Daher muss die Hautfarbe ebenfalls mit den anderen Attributen Härte der Strafe, Laufende Strafe und Geschlecht verglichen werden. Sodass ausgeschlossen werden kann, dass zum Beispiel einfach viele Weiße Personen eine Harte Strafe haben und daher so viele negativ bewertet wurden.

Für die Überprüfung, ob die Härte der Strafe eine Rolle im Zusammenhang mit der Hautfarbe und der daraus resultierenden Bewertung spielt, sind zwei weitere Diagramme erstellt wurden. Zum einen eines in welchem als Säulendiagramm die Härte der Strafe im Zusammenhang mit der Hautfarbe auf die gesamt Anzahl an Daten verglichen wird. Aus diesem ergibt sich, dass etwas mehr Schwarze eine harte Strafe haben. Was für die Vermutung der Diskriminierung der Weißen spricht. Die Verteilung der leichten und mittleren Strafe ist bei beiden Hautfarben beinahe auf einem Niveau, spricht jedoch trotzdem minimal für die Vermutung. Um dies noch genauer zu untersuchen wurde das zweite, in der nachfolgenden Abbildung 3.6 dargestellte, Säulendiagramm erstellt.

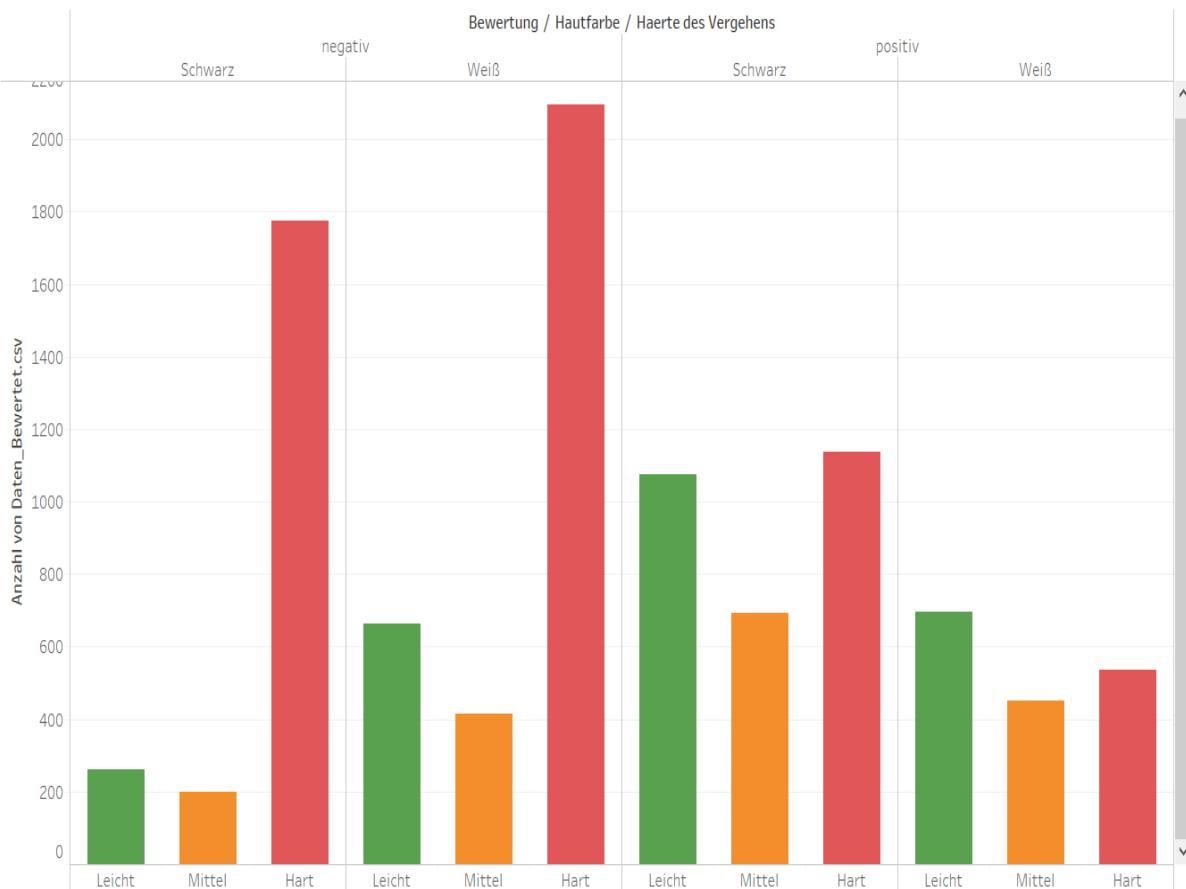


Abbildung 3.6: Auswertung der Hautfarbe im Zusammenhang mit der Härte der Strafe und der Bewertung nach der gesamt Anzahl an Daten

In diesem Diagramm wird die Härte der Strafe in grünen Säulen für leicht, orangenen Säulen für mittel und roten Säulen für hart aufgeteilt in 4 Bereichen auf Grundlage der

KAPITEL 3. PRAKTISCHER TEIL

Hautfarbe und Bewertung verglichen. Die Y-Achse gibt dabei die Anzahl an Datenpunkten an. In der Analyse können nun der ganz linke und der zweite Bereich von Links verglichen werden. Aus diesen beiden Bereichen kann geschlossen werden, dass deutlich mehr Weiße mit einer leichten Strafe negativ bewertet werden als Schwarze. Dies befürwortet erneut die Vermutung und diese wird noch weiter bestätigt, in der Analyse der rechten beiden Bereiche. Den hier sind die positiven Bewertungen auf die Hautfarbe aufgegliedert dargestellt. Hier ist zu sehen, dass deutlich mehr Personen mit Hautfarbe Schwarz und einer harten, leichten oder mittleren Strafe positiv bewertet werden als Personen mit Hautfarbe Weiß. Damit ist insgesamt klar die Verzerrung, dass so viele Personen mit Hautfarbe Weiß negativ bewertet wurden, liegt nicht an der Härte der Strafe. Da insgesamt mehr Schwarze mit härteren Strafen und weniger Weiße mit leichteren Strafen positiv bewertet werden. Wenn es nun bestätigt nicht an der Härte der Strafe lag, müssen noch die weiteren Attribute überprüft werden.

Hierfür wurden als nächstes sich mit Hilfe von zwei weiteren Tableau die Laufende Strafe angeschaut. Möglicherweise haben ja Personen mit Hautfarbe Weiß längere noch laufende Strafen und wurden deshalb so zahlreich negativ bewertet. Im ersten Tableau wurde die prozentuale Verteilung des Attributs Laufende Strafe über die jeweilige Hautfarbe betrachtet.

In Abbildung 3.7 ist diese Verteilung zu sehen. Auf den ersten Blick fällt direkt auf, dass sich beide Bereiche für die jeweilige Hautfarbe kaum voneinander unterscheiden. Daher haben beide Gruppen ähnliche Laufende Strafen und sollten daher nach diesem Attribut auch ähnlich bewertet sein. Um dies nochmals zu bestärken wurde wie auch schon bei der Härte der Strafe noch ein Diagramm, welches die Bewertung mit ein bezieht, erstellt. Dieses zweite Diagramm bestätigt den Eindruck des ersten und verstärkt sogar nochmals die Vermutung. Da in diesem zu sehen ist, dass mehr Personen mit geringen noch Laufenden Strafen und der Hautfarbe Weiß eine negative Bewertung bekommen haben als Personen mit Hautfarbe Schwarz und geringen Laufenden Strafen. Damit ist auch diese Untersuchung auf einen Möglichen Zusammenhang der Laufenden Strafe abgeschlossen und die Vermutung der Diskriminierung erhärtet sich. Für einen gesamt Überblick über den Zusammenhang zwischen Bewertung, Hautfarbe, Härte der Strafe und Laufende Strafe, gemessen an der Anzahl der Daten, wurde zusätzlich noch ein Säulendiagramm angelegt. Dieses ist aufgrund der vielen Attribute jedoch sehr mit Informationen gefüllt, liefert aber einen sehr guten Überblick. Darüberhinaus bestätigt es zusätzlich die Vermutung der Diskriminierung.

Einzig und allein muss nun überprüft werden, ob das Geschlecht eine Rolle spielt. Da zum Beispiel deutlich mehr Personen mit Hautfarbe Weiß und Geschlecht Weiblich einen Antrag gestellt haben. Dafür wurde ein Diagramm erstellt, in welchem der prozentuale Anteil an Personen nach Hautfarbe und Geschlecht aufgeteilt ist. In diesem Tableau ist

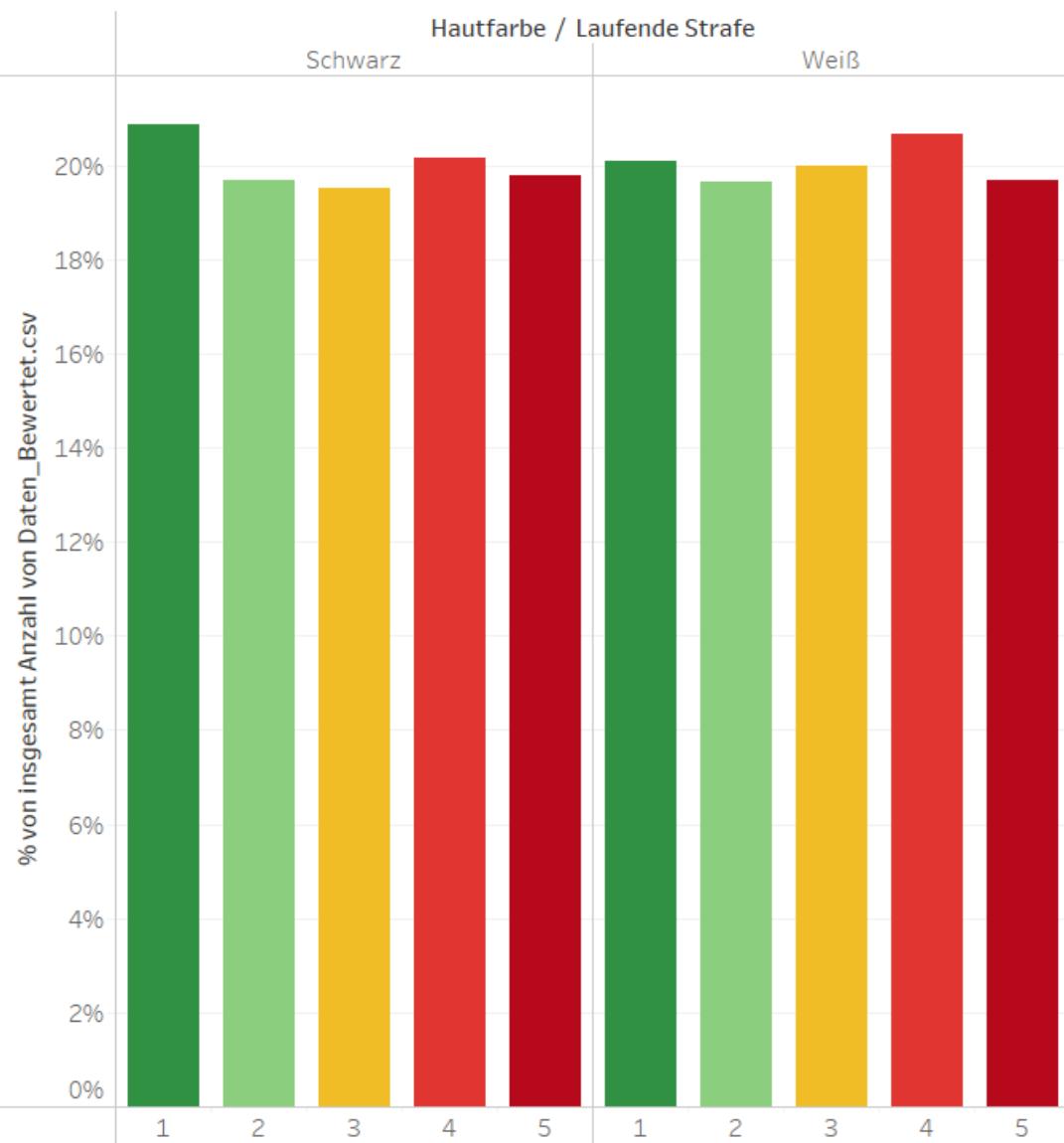


Abbildung 3.7: Auswertung der Hautfarbe im Zusammenhang mit der Laufenden Strafe nach Prozent von der gesamt Anzahl an Daten

zusehen, dass ungefähr 11% der Personen mit Hautfarbe Weiß weiblich sind und ungefähr 3% mit Hautfarbe Schwarz. Somit herrscht hier ein Unterschied von 8%, welcher nicht ausschlaggebend genug ist für den deutlichen Unterschied in der Bewertung nach Hautfarbe und daher nicht der Grund für diesen sein kann. Hiermit ist insgesamt bestätigt, dass höchstwahrscheinlich ein Label Bias auf Basis der Hautfarbe Weiß vorliegt. Damit ist klar, dass gewisse Bewertende Vorurteile gegen diese Gruppe bei der Bewertung gehabt haben müssen.

Für die Auswertung und Überprüfung der Hautfarbe wurden die dazugehörigen eben vorgestellten Diagramme in eine sogenannte Story gepackt, welche dann als Power Point

Präsentation exportiert wurde. So ist der Analyseweg und die zugehörigen Charts auch außerhalb von Tableau zugänglich.

Das Lernziel über die Existenz von einem Bias aufzuklären, kann damit eindeutig erfüllt werden. Da durch diese Story die Lehre Zugang bekommt, um Seite für Seite näher einen Bias zu entdecken. Zudem können die generierten Daten genutzt werden, um in der Lehre die Studierenden selbst nach dem Bias durch Analyse der Daten suchen zu lassen. Damit kann ein Bewusstsein für Verzerrungen und Interpretationsmöglichkeiten in Daten geschaffen werden.

Szenario des sozialen Punktesystem

Zu Beginn wird eine gesamt Übersicht über die Verteilung der sozialen Punkte des gesamten Datensatzes dargestellt.

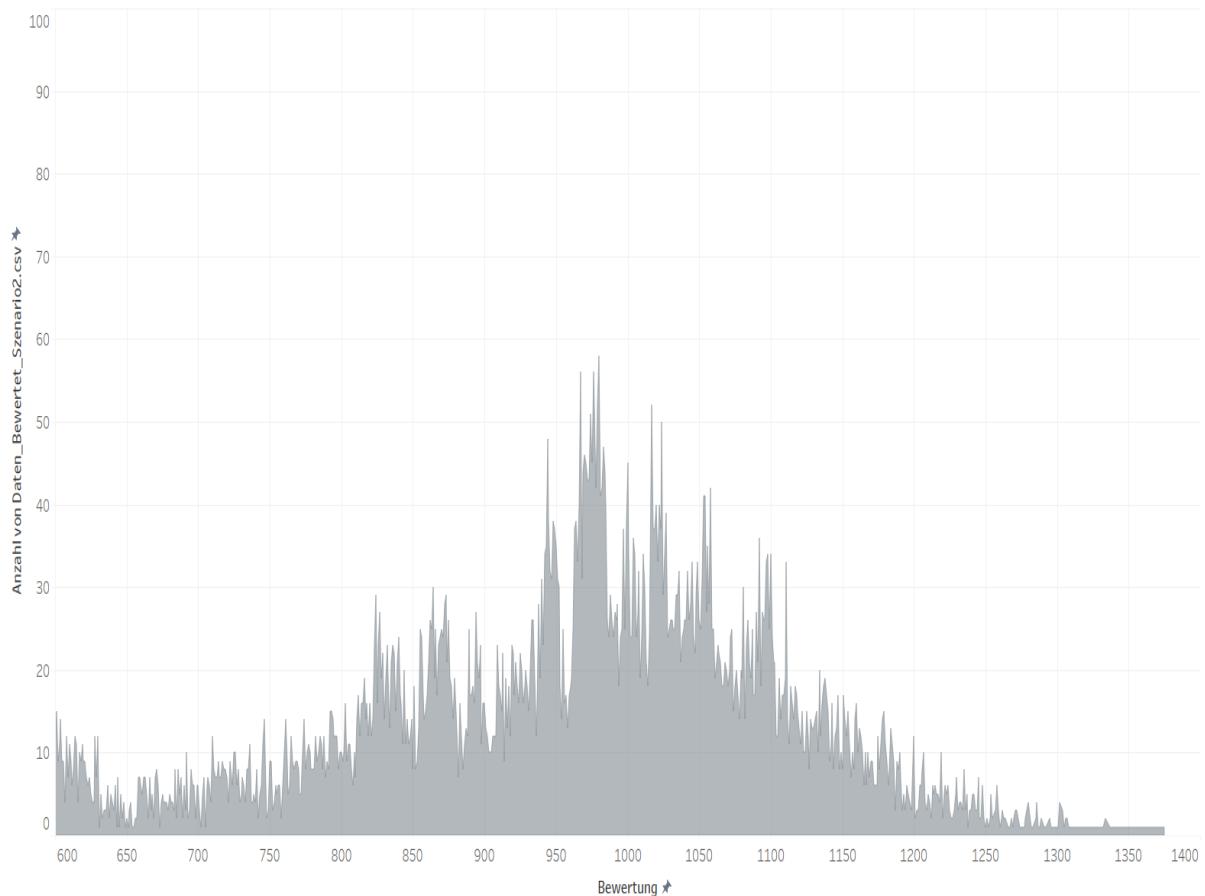


Abbildung 3.8: Überblick der sozialen Punkteverteilung über den gesamten Datensatz

In der obigen Abbildung 3.8 ist diese Verteilung zusehen. Die X-Achse stellt dabei den sozialen Punktestand von 600 bis 1400 dar und die Y-Achse die Anzahl an Datensätzen pro Punktzahl. In der Verteilung der Punktzahl ist nahezu eine Gaußkurve zu erkennen, was sehr anschaulich ist und für eine gute Generierung der Daten spricht. Eine weitere Auffälligkeit ist die Anhäufung der Datenpunkte mit einer sehr schlechten Bewertung auf der linken Seite. Diese widerspricht der Gaußkurve und ist ein Indiz für eine Verzerrung

in den Daten, welche nun untersucht wird.

Im nächsten Diagramm wird als erstes Attribut zur Analyse das Alter herangezogen. Hierfür soll für das Alter die Anzahl an Datenpunkten und den jeweiligen durchschnittlichen Punktestand veranschaulicht werden.

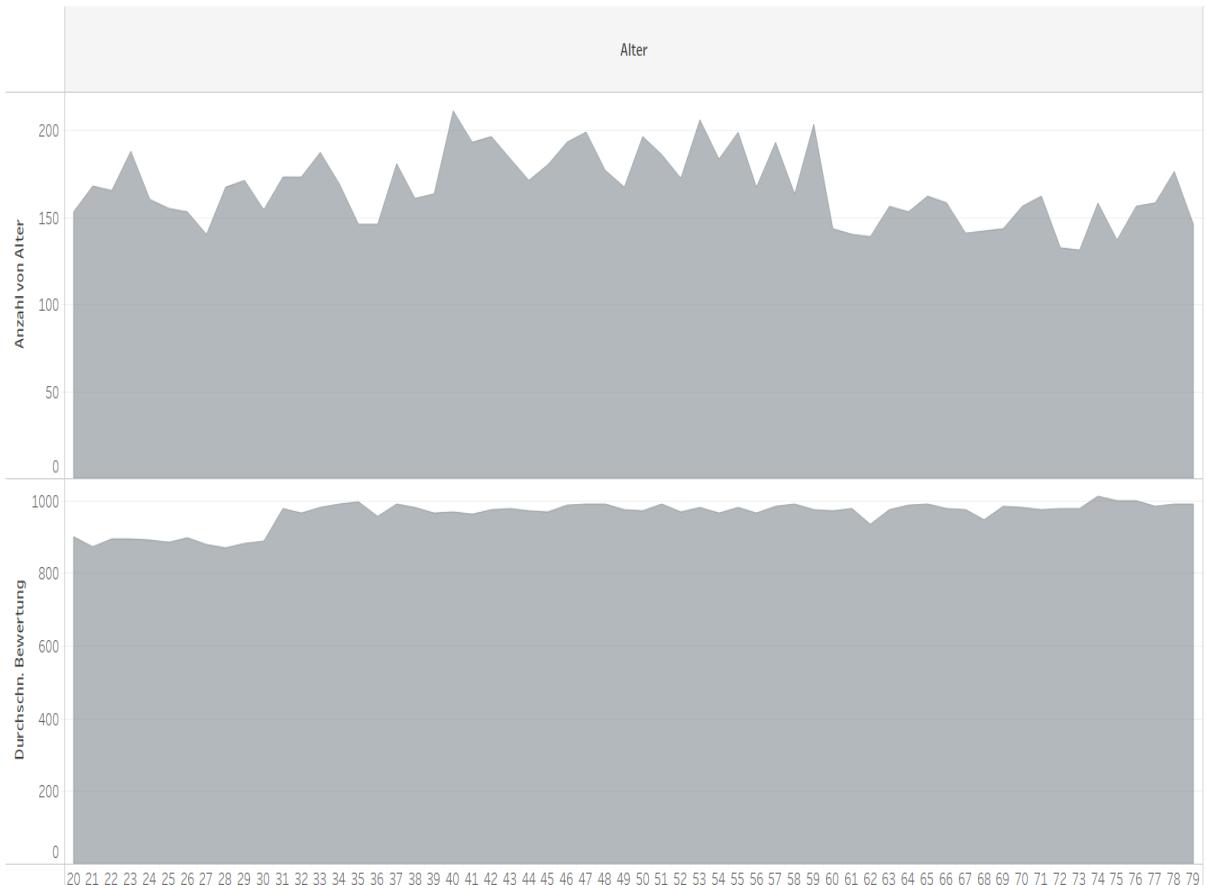


Abbildung 3.9: Auswertung der Anzahl von Personen und durchschnittlicher Punktestand pro Alter

Die beiden Diagramme aus Abbildung 3.9 sind auf einer Seite des Tableau angelegt. Das obere Diagramm zeigt die Verteilung des Alters in Anzahl an Datenpunkten. In diesem sind keine signifikanten Unterschiede zu erkennen, dass zum Beispiel es nur sehr wenige junge Personen gibt oder nur wenige alte Personen über 70. Daher lässt sich aus diesem nicht viel schließen. Im unteren Diagramm ist der durchschnittliche Punktestand pro Alter dargestellt. In diesem fällt auf, dass dieser im Alter von 20 bis einschließlich 30 um die 100 Punkte schlechter ist als in anderen Altern. Im Alter über 30 liegt der Punktestand nahe bei der 1000, im Alter drunter jedoch nur bei knapp unter 900. Dies lässt vermuten, dass hier eine Verzerrung vorliegt und jüngere Personen unter 31 eine schlechtere Bewertung erhalten. Damit ist die Analyse dieses Diagramms abgeschlossen und es kann mit einem weiteren mit neuen Attributen vorgefahren werden.

Auf der nächsten Seite wird das Alter mit der politischen Orientierung einer Person ver-

KAPITEL 3. PRAKTISCHER TEIL

glichen. Das bedeutet es wird auf der X-Achse jedes Alter von 20-79 dargestellt und auf der Y-Achse die Anzahl an Personen pro Alter für die jeweilige Orientierung. Damit sind es drei Diagramme für jede Orientierung eines. Im obersten für die politische Orientierung Links ist eindeutig ein Trend zu erkennen, dass jüngere Personen gerade im Alter von 20-30 deutlich mehr Links orientiert sind, als Personen im Alter über 60. Daraus kann interpretiert werden, dass möglicherweise auch eine Verzerrung auf die politische Orientierung Links existiert. Sodass die Personen unter 31 Jahren dadurch einen deutlich niedrigeren durchschnittlichen Punkte haben. Dies bleibt jedoch Interpretation und kann hier direkt nicht so angenommen werden. Bei den Orientierungen Mitte und Rechts gibt es keine besonderen Auffälligkeiten hinsichtlich der Verzerrung. Einzig und allein ist zu sehen, dass Personen im höheren Alter eher mittig orientiert sind. Als nächstes ist es nun daran die Vermutung aus der Interpretation, dass es möglicherweise auch eine Verzerrung auf die politische Orientierung Links ist zu überprüfen. Im folgenden Diagramm wird dafür die politische Orientierung mit der Bewertung verglichen.

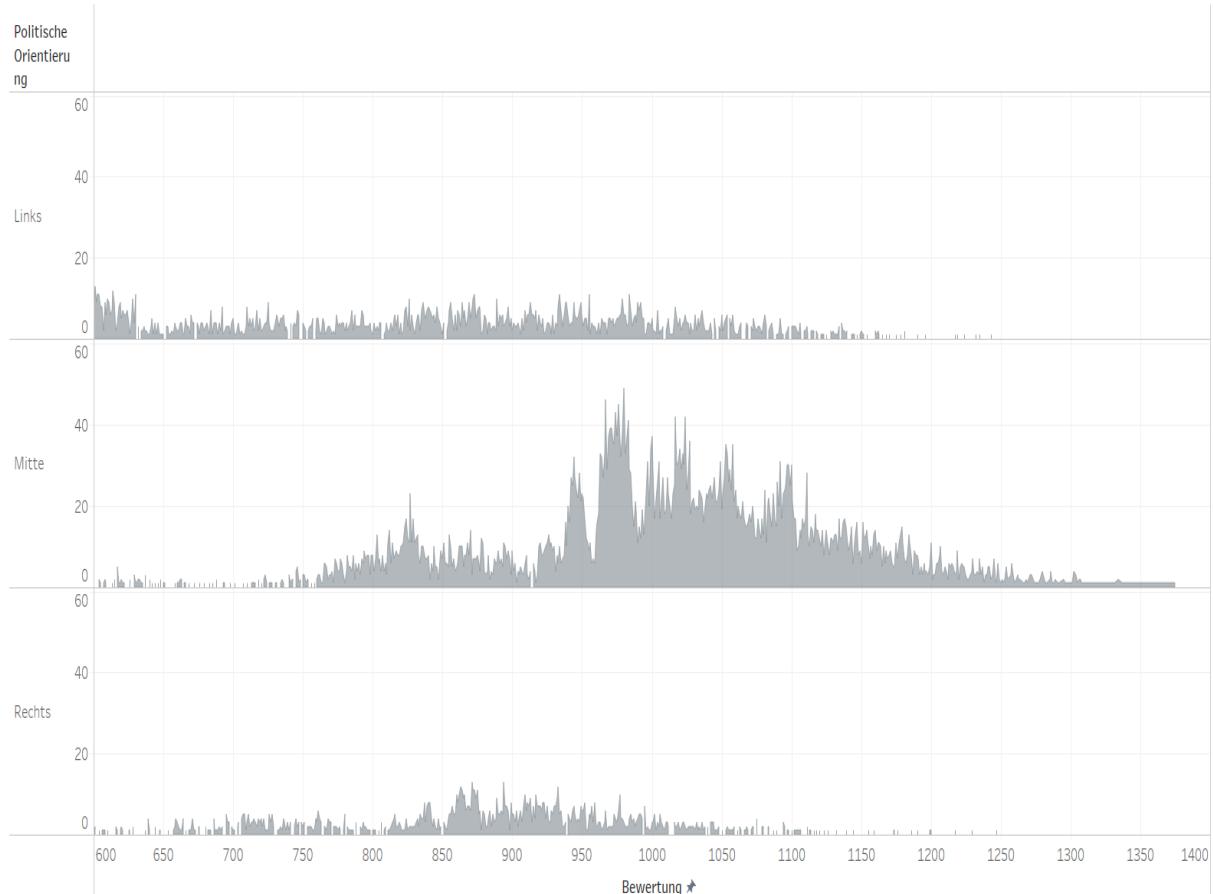


Abbildung 3.10: Auswertung der Anzahl von Personen eingegliedert in die politische Orientierung pro Punktestand

Die Abbildung zeigt die Aufgliederung der drei politischen Orientierungen und auf der

KAPITEL 3. PRAKTISCHER TEIL

X-Achse zugehörig die Bewertung als den sozialen Punktestand. Auf der Y-Achse wird für jede Orientierung die Anzahl an Datenpunkten pro Punktestand angegeben. Im Diagramm fällt auf den ersten Blick auf, dass Personen mit einer politischen Orientierung Links im Vergleich zu den anderen deutlich öfter schlechtere Punktestände haben. Dadurch verhärtet sich der Verdacht, dass die auf der politischen Orientierung Links eine Verzerrung liegt, oder eine Verzerrung von Personen im Alter zwischen 20 und 30 dafür sorgt. Es ist auch zusehen im gesamt Vergleich, dass Personen mit einer mittigen Orientierung deutlich bessere Punktestände haben als links oder rechts angelegte. So kann hier auch interpretiert werden, dass eine linke oder rechte Orientierung einfach so für einen schlechteren Punktestand führt. Dies kann jedoch ebenfalls nicht bewiesen werden, sondern ist reiner Interpretationsspielraum.

Im nächsten Diagramm wird der Bildungsabschluss als neues Attribut zur Analyse hinzugezogen. Dafür wird die Verteilung des sozialen Punktestand über die unterschiedlichen Bildungsabschluss betrachtet. Dabei ist auffällig, dass desto einen höheren Grad der Abschluss annimmt, umso höher ist auch der tendenzielle Punktestand. Daraus kann interpretiert werden, dass Personen mit einem besseren Abschluss auch besser bewertet und geschätzt werden. Eine weitere Auffälligkeit liegt im unteren Bereich der Punkte. Manche Abschlüsse wie ein Bachelor oder ein Diplom haben hier seltsame minimale Ausreißer. Hier kann natürlich interpretiert werden, dass es immer Ausreißer gibt, jedoch kann es auch an einer Verzerrung von einem anderen Attribut liegen. So wie zum Beispiel der politischen Orientierung, dass diese dafür sorgt, dass diese Ausreißer entstehen. Um dies zu Überprüfen wird ein weiteres Diagramm angelegt, in welchem die politische Orientierung mit dem Bildungsabschluss in Zusammenhang gebracht wird und die Anzahl an Datenpunkten verglichen wird. In diesem können keine besonderen Auffälligkeiten entdeckt werden. Daher muss in einem weiteren Diagramm die Bewertung hinzugezogen werden.

In dieser Abbildung 3.11 ist auf der X-Achse der Punktestand und auf der Y-Achse die Anzahl an Datenpunkten angegeben. Zusätzlich ist das Diagramm in einzelne Subdiagramme nach der politischen Orientierung und dem Bildungsabschluss aufgeteilt. Durch diese Aufteilung kann die zuvor vermutete Verzerrung durch die politische Orientierung Links weiter bestätigt werden. Da alle Diagramme mit der politischen Orientierung Links eine besonders hohe Anzahl an Datensätzen als Ausreißer im niedrigen Punktbereich haben. Zusätzlich kann interpretiert werden, dass Personen mit einer mittigen Orientierung und einem umso höheren Abschluss tendenziell auch einen höheren Punktestand haben. Nach diesem Diagramm ist klar, dass eine Verzerrung für eine politische Orientierung Links in Kombination mit oder ohne dem Alter zwischen 20 und 30 vorliegt. Dabei kann nicht wie im ersten Szenario von einem Bias durch Abwesenheit und einer vermutlich bewussten Diskriminierung einer bewertenden Person ausgegangen werden. Sondern es besteht Interpretationsspielraum, ob möglicherweise Linke oder junge Leute zwischen 20

KAPITEL 3. PRAKTISCHER TEIL

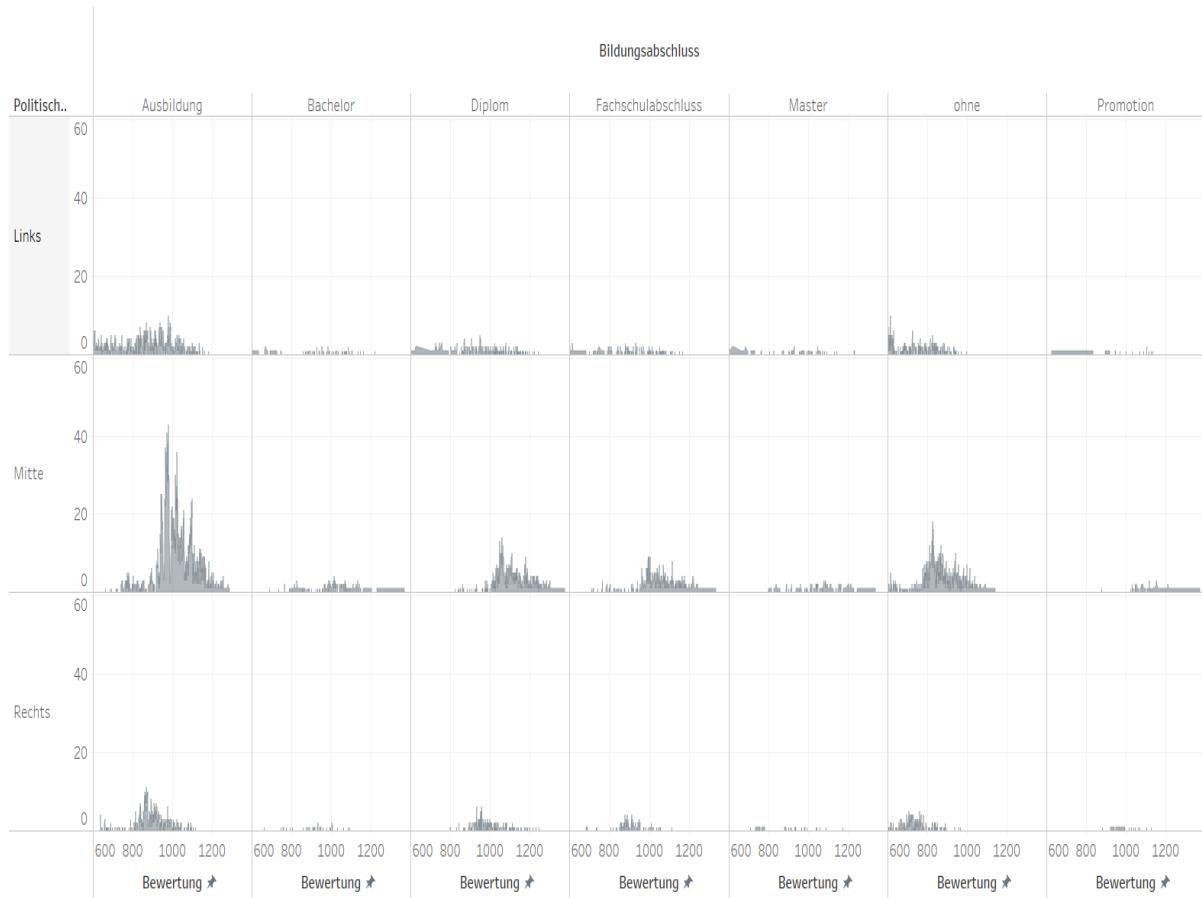


Abbildung 3.11: Auswertung der Anzahl von Personen eingegliedert in die politische Orientierung und Bildungsabschluss pro Punktestand

und 30 einfach einen schlechteren Punktzahl bekommen oder eine bewusste Diskriminierung vorliegt. Jedoch ist die Verzerrung klar und deutlich zu erkennen.

In den folgenden Diagrammen der Auswertung wurden die weiteren Attribute dargestellt, wiesen jedoch keine Auffälligkeiten hinsichtlich der Verzerrung auf. Diese bieten lediglich weitere Interpretationsmöglichkeiten, um das Szenario besser durchschauen zu können. Alle Diagramme wurden ebenso wie im ersten Szenario zu einer sogenannten Story hinzugefügt und dann als Powerpoint Präsentation exportiert.

Für das Lernziel zur Aufklärung über die Existenz des Bias, konnte in den Diagrammen gut hingearbeitet werden. In diesem Fall wird jedoch eine andere Art an Bias in den Daten dargestellt und durch Interpretation auf diesen hingewiesen. Allgemeine bietet die Auswertung des zweiten Szenario viele Interpretationsmöglichkeiten für die Lehre und zudem sehr viele unterschiedliche Möglichkeiten einen Bias in den Daten zu entdecken. Daher ist das Lernziel jedenfalls erfüllt und die Auswertung bietet eine gute Darstellung des Bias.

3.5 Evaluation der Ergebnisse

In diesem Unterkapitel wird die gesamte Umsetzung der beiden Szenarien als Datengeneratoren und die dazugehörige Datenauswertung evaluiert. Dabei wird vor allem nochmals auf die im Kapitel 1.2 definierten Anforderungen zurückgegriffen und gegen diese evaluiert.

Die erste Anforderung lag darin, zwei Szenarien zu erarbeiten, welche so realitätsnahe wie möglich sind. Um dies zu erfüllen wurden unter 3.1 die beiden Szenarien Bewertung von Bewährungsanträgen und soziales Punktesystem eingeführt. Dabei wurde definiert wie jeweils ein Datenpunkt im Szenario aussehen soll und wie diese dann bewertet werden. Somit sind schon Mal zwei Szenarien definiert und ein Teil der Anforderung ist erfüllt. Ein Hauptaugenmerk bei dieser Anforderung lag jedoch auf der Realitätsnähe. Diese wird zum einen bei der Idee zu den Szenarien und zum anderen beim letztendlichen füllen der Daten für die Szenarien gefordert. Bei der Idee für die Szenarien ist die Realitätsnähe erfüllt, da zum einen für das Bewährungsantrag Szenario die Verbindung zur in der Realität verwendeten Software „COMPAS“ herrscht und zum anderen das zweite Szenario an das aus China aktuelle durch die Medien populär gewordene social creditpoint system angelehnt ist. Damit sind in beiden Fällen Verbindungen zur Realität geschaffen. Der andere Teil zur Erfüllung der Anforderung liegt in der Generierung der Daten auf möglichst realen Statistiken. Beim ersten Szenario der Bewährungsanträge beruhen alle Attribute Geschlecht, Hautfarbe und Härte der Strafe bis auf die Laufende Strafe auf realen Statistiken aus den USA. Der Name ist ebenfalls durch eine Bibliothek ein amerikanischer zufälliger Name. Daher sind 4 von 5 Attribute der Anforderung entsprechend und diese daher durchaus erfüllt, da es sich um Realitätsnähe und nicht die Realität handelt. Jedoch könnte an dieser Stelle optimiert werden und auch das letzte Attribut durch Suche oder Anfrage nach einer Statistik ebenso real gestaltet werden. Für das Szenario des sozialen Punktesystem basieren die Attribute Alter, Politische Orientierung, Bildungsabschluss und Wohnlage auf realen Statistiken. Der Name ist ebenfalls durch die selbe Bibliothek wie im ersten Szenario umgesetzt, diesmal jedoch ein deutscher zufälliger Name. Die beiden Attribute soziales Engagement und CO2-Fußabdruck haben keine passende Statistik zu Grunde. Wobei der CO2-Fußabdruck immerhin ein wenig um den deutschen Mittelwert angepasst wurde. Daher hat dieser noch ein wenig realen Bezug. Insgesamt sind somit 5 von 7 Attribut wirklich passend realitätsnahe umgesetzt und die Anforderung in diesem Teil nur zu ungefähr 80% erfüllt. Daher kann in diesem noch optimiert und die Daten realitätsnaher gestaltet werden.

Als zweite große Anforderung wurde das erstellen eines Datengenerators für zufallsgenerierte Daten mit folgenden Unterpunkten aufgeführt.

- Python Script zum generieren eines großen Datensets
- Flexibilität in der Generierung von Datensätzen
- Erzeugung von flexibel wählbaren Vorurteilen in den Datensätzen
- Bewertete und unbewertete Daten zur weiteren Nutzung bereitstellen

Um diese Anforderung und die zugehörigen Unterpunkte zu erfüllen, wurden für jedes Szenario ein Python Notebook zum generieren der Daten separat umgesetzt. In diesen kann eine beliebige Anzahl an gewünschten Datensätze angegeben werden. Damit ist der erste Punkt der Anforderung optimal erfüllt. Für die Flexibilität in der Generierung der Datensätze werden wie unter 3.3 beschrieben, zum einen Zufallsmethoden basierend auf den möglichst realitätsnahen Wahrscheinlichkeiten verwendet und zum anderen die Methode für das generieren eines Seed/Startwert erstellt, um zu jedem Zeitpunkt unterschiedlichste Daten zu erhalten. Zudem wird beim Bewerten der Datenpunkte immer mit Wahrscheinlichkeiten gearbeitet, sodass es auch immer möglich ist den einen Ausnahmefall zu haben. Des Weiteren können die Wahrscheinlichkeiten auf welchen die Daten beruhen immer ohne Probleme geändert werden. Damit ist ein sehr gutes Maß an Flexibilität in der Generierung geboten. Für die Erzeugung von flexibel wählbaren Vorurteilen wurden der benutzenden Person ein Feld angelegt für die Eingabe der gewünschten Verzerrung. Dadurch kann diese hier aus den Attributen und deren Ausprägungen flexibel wählen wie verzerrt werden soll. Jedoch ist dieser Punkt natürlich auf die Anzahl der Attribute in den Szenarien beschränkt. So existieren im ersten Szenario eigentlich nur zwei so wirklich interessante Attribute mit zusammen vier Ausprägungen und damit vier Möglichkeiten eine Verzerrung zu wählen. Das zweite Szenario hingegen bietet hier eine sehr große Flexibilität mit insgesamt sechs interessanten Attributen und daher auch deutlich mehr Möglichkeiten. Insgesamt ist diese Anforderung auf jeden Fall erfüllt. Im letzten Teil der Anforderung wird die Bereitstellung der bewerteten und unbewerteten Daten gefordert. Dies wird erfüllt, da wie in der Umsetzung unter 3.3 beschrieben beide Datensätze als CSV Dateien ausgegeben werden. Erweiterungsmöglichkeiten, welche zwar nicht gefordert sind, bestehen hier dennoch so könnten die Daten zum Beispiel in eine Datenbank geschrieben werden, um auch alte Datensets nicht in unterschiedlichsten Dateien irgendwo speichern zu müssen. Zudem werden die CSV Dateien falls diese schon so vorhanden im selben Ordner wie das Script liegen überschrieben. Damit ist diese zweite große Anforderung ebenfalls komplett erfüllt.

Die letzte Anforderung bezieht sich auf eine Auswertung zur Veranschaulichung des Bias. Dabei soll diese visuell in dem Tool Tableau umgesetzt werden. Um dies umzusetzen wurden die unter Kapitel 3.4 beschriebenen Tableaus für die beiden Szenarien erstellt. Das Tableau für die Auswertung des Bewährungsantrag Szenario ist übersichtlich und

KAPITEL 3. PRAKTISCHER TEIL

bietet eine klare Struktur. Dabei wird Stück für Stück durch Diagramm für Diagramm zur Veranschaulichung und Bestätigung des Bias hingeleitet. Die Diagramme sind nicht überfüllt und selbsterklärend wodurch das erkennen des Bias relativ einfach möglich ist. Zudem wurde aus den einzelnen Seiten eine sogenannte Story erstellt in welcher die Diagramme zum durch klicken angeordnet sind. Des Weiteren ist diese Story als Powerpoint Präsentation exportiert und kann so optimal verwendet werden, um sich Seite für Seite dem Bias zu nähern und diesen zu Veranschaulichen. Daher ist diese Anforderung für das erste Szenario vollkommen erfüllt. Das Tableau für die Auswertung des zweiten Szenario des sozialen Punktesystems ist etwas komplexer, da auch das Szenario komplexer ist. Daher sind hier die Diagramme etwas voller und mit mehr Inhalt gefüllt, aber dennoch übersichtlich und aussagekräftig. Die Hinführung zur Veranschaulichung des Bias ist in diesem nicht so simpel wie beim ersten Szenario, da hier ein deutlich größerer Interpretationsspielraum besteht und auch mehr Verbindungen in den Daten. Daher muss bei der Veranschaulichung des Bias schon genau hingesehen werden und ein wenig Interpretation mitgebracht werden. Die Diagramme dieses Tableaus sind dennoch auch wie im ersten Szenario als Story zusammengefasst und als Powerpoint Präsentation exportiert. Damit ist diese Veranschaulichung des Bias nicht ganz so trivial und einfach wie beim ersten Szenario und zum vollständigen erfüllen der Anforderung könnte die Auswertung noch größer gestaltet sein. Der Bias kann dennoch mit Interpretation gefunden und vermutet werden. Damit setzen wir auch zur letzten Teilanforderung an. Bei dieser sollen durch Analyse nachvollziehbar beispielhafte Arten von Bias in den Daten darstellbar sein. Für das erste Szenario ist eindeutig ein Bias durch Abwesenheit dargestellt, da kein Attribut gefunden werden konnte, womit eine Verzerrung rückführbar wäre. Damit ist ein Bias durch Abwesenheit gefunden und somit diese Anforderung umgesetzt. Im Fall des zweiten Szenario ist kein Bias durch Abwesenheit dargestellt, sondern ein Bias verursacht durch Auffälligkeiten von gewissen Attributen. Daher ist in diesem Fall eine andere Art von Bias in den Daten darstellbar und die Anforderung ebenfalls erfüllt. Insgesamt ist diese letzte Anforderung damit zu 90% erfüllt und bietet noch Erweiterungspotential für das zweite Szenario.

Die Evaluierung der Anforderungen ist hiermit abgeschlossen und in der folgenden Tabelle 3.12 ist eine Zusammenfassung davon zu sehen. In der obigen Tabelle 3.12 sind die Anforderungen samt Unterpunkte aufgelistet und zugehörig die prozentuale Erfüllung, welche evaluiert wurde aufgeführt. Dabei sind es insgesamt drei große Anforderungen, die erste konnte zu 90%, die zweite mit vielen Unterpunkten zu 100% und die letzte zu 90% erfüllt werden. Mit diesem Ergebnis ist die Umsetzung insgesamt mit einem sehr guten Ergebnis gelungen.

KAPITEL 3. PRAKTISCHER TEIL

Anforderung und Unterpunkte		Erfüllung
Konzeption von zweie realitätsnahen Szenarien		90%
Erstellung eines Datengenerators für zufallsgenerierte Daten	Python Script zum generieren eines großen Datensets	100%
	Flexibilität in der Generierung von Datensätzen	100%
	Erzeugung von flexibel wählbaren Vorurteilen in den Datensätzen	100%
	Bewertete und unbewertete Daten zur weiteren Nutzung bereitstellen	100%
Erstellung einer Auswertung zur Veranschaulichung des Bias	Visuelle Auswertung in Tableau	90%
	Analyse zur Nachvollziehbarkeit des Bias	

Tabelle 3.12: Tabelle zur Zusammenfassung der Evaluierung der Anforderungen

4 | Schluss

In diesem letzten Kapitel wird die gesamte Arbeit zusammengefasst und daraufhin eine kritische Reflexion derer durchgeführt. Zum Schluss ist noch ein kleiner Ausblick für das weitere Vorgehen sowie eine Literaturempfehlung gegeben.

4.1 Zusammenfassung

Die Motivation und das Ziel der Arbeit lag darin, Daten zu schaffen, welche in der Lehre eingesetzt werden können, um Verzerrungen von Trainingsdaten zu veranschaulichen. Da ein KI Modell aus diesen Trainingsdaten lernt und somit lernt solch ein Modell häufig auch von Manipulierten Daten oder Verzerrten Daten. Die Qualität der Trainingsdaten ist hierbei für das Ergebnis des Maschinellen Lernen von sehr großer Bedeutung. So haben solche Verzerrung in den Daten ebenfalls einen sehr großen Einfluss auf das Ergebnis des maschinellen Lernens. In den meisten Fällen ist der Mensch für Verzerrungen aufgrund von beispielsweise Vorurteilen verantwortlich. Diese Vorurteile, welche durch den Menschen in die Daten gelangen, lassen sich niemals vermeiden. Aufgabe ist es stattdessen technische Gegenmaßnahmen zu finden und vor allem den Menschen gegenüber dem Thema Bias in Daten zu sensibilisieren. Daher muss mehr Wissen über das noch sehr neue Forschungsgebiet von Bias geschaffen werden, damit im Umgang mit KI darauf geachtet werden kann.

Dies ist die Motivation der Arbeit, doch was sind Daten überhaupt und wie spielt hierbei eine KI und Ethik eine Rolle. Daten haben sich in der Vergangenheit von einem Nebenprodukt zu einer sehr wertvollen Ressource entwickelt. Durch Big Data und Internet of Things ist das Generieren sowie Speichern von großen Daten Mengen kein Problem mehr. Daraus ergeben sich neue Geschäftsbereiche und Entwicklungsmöglichkeiten. Die Qualität der Daten ist dabei häufig der entscheidende Faktor für die Verwendung und lässt sich inzwischen meist anhand von Qualitätsmerkmalen messen. Diese Daten sind die Grundlage für eine KI. Besonders im Teilgebiet des supervised learning aus dem Bereich des ML spielen Daten die Schlüsselrolle zum Erfolg. In diesem lernt man Zusammenhänge aus Trainingsdaten und erzeugt daraus ein Modell, welches das Verhalten in den historischen Daten zuverlässig reproduzieren kann. Dadurch können Prozesse optimiert, automatisiert oder auch ersetzt und einen großen Mehrwert aus den Daten geschaffen werden. Da jedoch aus historischen Daten das Verhalten teilweise von Menschen gelernt wird und die KI immer mehr Entscheidungen übernimmt, welche Einfluss auf das Leben einzelner In-

dividuen hat, bekommt die Ethik einen immer größeren Stellenwert in der KI. Die Ethik an sich ist die Grundlage einer Gesellschaft, die Ethik ist neben dem Recht eine Leitlinie zum Verhalten in der Gesellschaft. Um die KI und ethischen Wertvorstellungen zu vereinen, werden Leitlinien aus der Wirtschaft und Wissenschaft getrieben. Ein Konzept ist die vertrauenswürdige KI, bestehend aus Recht und Ethik. So sollen wenn KI eingesetzt wird, die folgenden Grundwerte gewährleistet werden: Autonomie/Selbstbestimmtheit des Menschen, fairness, Transparenz, Verlässlichkeit, Sicherheit und Datenschutz. Besonders der Aspekt der Fairness spielt für die Arbeit eine besondere Rolle. Da KI in der Lage ist, abhängig von ihrer Funktionsweise, für unzulässige Unfaire zu sein. Insbesondere für Entscheidungen spielt Fairness eine sehr große Rolle und muss berücksichtigt werden. Diese nicht vorhandene Fairness kommt jedoch nicht von irgendwo her, sondern durch einen Bias.

Ein Bias beschreibt im Allgemeinen eine Verzerrung. Im Kontext der KI ist eine Verzerrung oft eine nicht wahrheitsgemäße Repräsentation der Realität, z.B. durch unzulässige Vorurteile. Im Bereich des ML gibt es eine Vielzahl an unterschiedlichen Arten von Bias. Dabei kann ein Bias durch einen Menschen, Daten oder Algorithmen erzeugt sein. Bei Menschen sind dies in der Regel Vorurteile durch die Gesellschaft, das Verhalten oder die Vergangenheit geprägt. Algorithmen hingegen können durch falsch erlernte Zusammenhänge falsche Schlüsse ziehen und so Verzerrungen von Ergebnissen erzeugen.

Der entscheidende und für diese Arbeit wichtigste Bias ist jedoch der, der durch die Daten erzeugt wird. Es gibt unterschiedliche Arten von Bias in Daten, wie z.B. die Auswahl der Trainingsdaten (sampling Bias) bis hin zu deren Korrektheit (measurement Bias). Eine häufige Art der Verzerrung ist die der Trainingsdaten, wobei man von Lable Bias spricht. Dabei ist die Beschriftung der Trainingsdaten verzerrt. Dies entsteht, da Trainingsdaten nahezu immer von Menschen "gelabelt" und im supervised learning menschliche Verhaltensweisen reproduziert werden. Wenn nun der Mensch bei der Beschriftung der Trainingsdaten Vorurteilbehaftet gehandelt hat, ist diese Verzerrung in den Daten und somit auch dem Modell vorhanden. Daher ist der Lable Bias eigentlich ein durch den Menschen als Bewertenden geschaffenes Problem. So kam es beispielsweise schon zu einer KI, welche Sexistisch im Einstellungsprozess von Amazon handelte.

Um solche Dinge vorzubeugen und das Ziel der Arbeit zu erreichen, wurden zwei möglichst realitätsnahe Szenarien entwickelt, für welche Daten mit einem Bias beinhaltend generiert werden. Damit können die daraus generierten Daten in der Lehre verwendet werden, um ein Bewusstsein für Bias in Daten zu schaffen. Das erste Szenario behandelt die Bewertung von Bewährungsanträgen mit stattgegeben oder nicht. Dabei besteht ein Antrag aus den 5 Attributen Name, Geschlecht, Hautfarbe, Härte des Vergehens und die noch laufende Strafe. Das zweite Szenario behandelt die Bewertung eines sozialen Punktesystems. Dabei werden Personen nach 7 angegebenen Attributen mit einer Punktzahl

zwischen 600 und 1400 bewertet.

Für beide Szenarien wurde ein Datengenerator entwickelt, welcher ein möglichst realitätsnahes Datenset zurückliefert. Dafür werden im Datengenerator zuerst das Rohdatenset basierend auf Wahrscheinlichkeiten aus aktuellen Statistiken erstellt und anschließend dieses Datenset nach bestimmten Regeln bewertet. Bei der Bewertung wird ein von der benutzenden Person gewünschter Bias hinzugefügt. Der final bewertete Datensatz kann dann in der Datenauswertung mit dem Tool Tableau und unterschiedlichsten Diagrammen auf den Bias untersucht werden. Dabei ist der Bias in einer sogenannten Story im Tableau Stück für Stück aufgezeigt wurden. So kann durch diese beiden Szenarien und durch die passenden Datengeneratoren unterschiedliche Datensets generiert werden, welche dann in der Lehre untersucht werden können. Durch die Untersuchung kann der Bias aufgezeigt werden und dadurch ein Bewusstsein für diesen geschaffen werden.

Abschließend kann damit gesagt werden, dass der Datengenerator es uns ermöglicht an einem praktischen Beispiel einen Bias in Daten zu ermitteln. Zudem kann das Projekt hinsichtlich des Ziels ein Lehrmittel zu schaffen als vollen Erfolg betrachten. Den man kann Daten generieren, diese individuell verzerrn und besitzt eine beliebig große Menge an qualitativ guten Daten für die Verwendung als Trainingsdaten. So kann das Bewusstsein für Bias in Daten in der Lehre gesteigert werden.

Reflektierend ist jedoch anzumerken, dass es sich hierbei noch um ein sehr junges Forschungsgebiet handelt. So existieren nur wenige Lösungen einen bekannten Bias zu minimieren oder damit umzugehen. Oft ist hierbei die einzige Lösung neue Daten zu erzeugen oder Einbußen in der Genauigkeit zu bekommen durch das Entfernen von Attributen. Zudem gibt es dabei auch viele technische Herausforderungen. Aber auch organisatorisch müssen Methoden entwickelt werden, wie man in der Lage ist frühzeitig Verzerrungen zu erkennen, Gegenmaßnahmen zu definieren und präventiv zu verhindern. In beiden Bereichen steht man noch vor großen Herausforderungen, da die Auswirkungen eines Bias oft auch unberechenbar und teils nicht abzuschätzen sind. Es kann harmlose Folgen haben, wie einen Skandal, für den man sich entschuldigen muss, aber auch deutlich drastischere Folgen, wenn man sich beispielsweise den Bereich Gesundheitswesen oder Justiz ansieht. Daher muss eine KI zukünftig vertrauenswürdig sein und dabei sowohl die rechtlichen Begrenzungen als auch die gesellschaftlichen ethischen Wertvorstellungen in ihrem Handeln erfüllen. Deshalb ist ein erster Schritt die Aufklärung über die Existenz von Verzerrungen und Bias.

4.2 Ausblick

Die beiden Datengeneratoren für das jeweilige Szenario können optimal von einer Person verwendet werden, um beliebig große Datensets mit einem gewünschten Bias zu generieren. Dabei kann das erste Szenario und die dazugehörige Datenauswertung so sehr einfach für den Einstieg in das Thema verwendet werden, da dieses etwas einfacher gehalten ist und der Bias leicht zu erkennen ist. Das zweite Szenario und die zugehörige Auswertung können dann anschließend verwendet werden, falls mehr Interpretation und Verbindungen in den Daten gewünscht ist. Dabei ist der Bias jedoch auch schwerer zu finden. Für das erste Szenario kann in der Datengenerierung noch das Attribut der Laufenden Strafe realitätsnaher gestaltet werden. Dies ist eine Möglichkeit für die Zukunft, um dies zu optimieren. Zudem können noch mehr Verbindungen in den Daten geschaffen werden. Für das zweite Szenario können ebenfalls zwei Attribute realitätsnaher gestaltet werden. Zudem besteht auch noch Erweiterungspotential in der Datenauswertung, diese kann noch um einige Diagramme erweitert und vertieft werden.

So kann auch auf Grundlage der bisherigen zwei Szenarien ein weiteres drittes Szenario entwickelt und umgesetzt werden.

Für die Lehre besteht die Möglichkeit neben der Betrachtung der Datenauswertung auch ein Model auf Basis der generierten Daten zu trainieren, um daraus dann den Bias zu entdecken und die Ursachen eines Bias live zu erleben.

Zudem kann versucht werden operationale und technische Lösungen zu entwickeln einen Bias zu entfernen bzw. zu verhindern. Ein Ansatz dies zu Unterstützen wäre auch ein Konzept für vertrauenswürdige KI einzuführen. Dadurch wäre eine Zertifizierung gegeben und damit eine Sicherheit, dass gewisse Systeme Grundanforderungen erfüllen.

Dadurch, dass es sich um so einen neuen Forschungsbereich handelt, wurden von dem Thema einige Facetten noch nicht betrachtet. Eine Veröffentlichung, welche hier für den Ausblick nennenswert ist, setzt sich mit den Vorteilen von Bias im ML auseinander. Diese wollen den Bias bewusst in Daten erzeugen, um ein Modell für ethische Richtlinien zu sensibilisieren. Dadurch wollen diese nicht die Symptome, welche durch die Verzerrungen entstehen, bekämpfen sondern die Ursache das Model selbst. Anstatt die Algorithmen zu manipulieren und die Trainingsreize zu unterdrücken, soll der Bias bewusst eingesetzt werden. Das Filtern von Daten ist nur Symptom Bekämpfung und nicht die Ergründung des Problems von Voreingenommenheit, egal ob in Lernalgorithmen oder Menschen. Dies ist ein völlig neuer höchst interessanter Ansatz mit Bias in der KI umzugehen und dadurch möglicherweise die KI sogar zu stärken.[29]

Literatur

- [1] F. Horn, *A Practitioner's Guide to Machine Learning*, Version 1.3, 02.02.2022. 2022, Letzter Zugriff: 22.05.2022 [Online]. Verfügbar:https://franziskahorn.de/mlbook_all.html.
- [2] B. Otto, D. Lis, J. Jürjes u. a., *Data Ecosystems*, 2019.
- [3] A. B. Cremers und A. Englander, *Vertrauenswürdiger Einsatz von künstlicher Intelligenz*, 2019, Letzter Zugriff: 15.05.2022 [Online]. Verfügbar:https://www.iais.fraunhofer.de/content/dam/iais/KINRW/Whitepaper_KI-Zertifizierung.pdf.
- [4] I. Döbel, M. Leis, M. M. Vogelsang u. a., "Maschinelles Lernen–Kompetenzen, Anwendungen und Forschungsbedarf," *Fraunhofer IAIS, Fraunhofer IMW, Fraunhofer Zentrale*. Zugriff am, Jg. 21, S. 2020, 2018.
- [5] Anonymous, *Incident Number 37. in McGregor, S. (ed.) Artificial Intelligence Incident Database*, Responsible AI Collaborative, 2018, Letzter Zugriff: 30.05.2022 [Online]. Verfügbar:incidentdatabase.ai/cite/37.
- [6] M. Kershner, "Data Isn't The New Oil — Time Is," 2021, Letzter Zugriff: 22.05.2022 [Online]. Verfügbar:<https://www.forbes.com/sites/theyc/2021/07/15/data-isnt-the-new-oil--time-is/?sh=45eb63a135bb>.
- [7] D. Reinsel, J. Gantz und J. Rydning, *The Digitization of the World - From Edge to Core*, IDC White Paper – US44413318, 2018.
- [8] I. Taleb, M. A. Serhani und R. Dssouli, "Big Data Quality: A Survey," in *2018 IEEE International Congress on Big Data (BigData Congress)*, 2018, S. 166–173. DOI: 10.1109/BigDataCongress.2018.00029.
- [9] I. International Organization for Standardization, *Information technology - Vocabulary*, ISO 2382:2015. 2015, Letzter Zugriff: 20.5.2022 [Online]. Verfügbar:<https://www.iso.org/standard/63598.html>.
- [10] A. Z. Faroukhi, I. El Alaoui, Y. Gahi und A. Amine, "Big Data Value Chain: A Unified Approach for Integrated Data Quality and Security," in *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, 2020, S. 1–8. DOI: 10.1109/ICECOCS50124.2020.9314391.
- [11] M. Yalaoui und S. Boukhedouma, "A survey on data quality: principles, taxonomies and comparison of approaches," in *2021 International Conference on Information Systems and Advanced Technologies (ICISAT)*, 2021, S. 1–9. DOI: 10.1109/ICISAT54145.2021.9678209.
- [12] C. Gröger, "There is no AI without data," 2021. DOI: 10.1145/3448247.
- [13] J. Byabazaire, G. O'Hare und D. Delaney, "Data Quality and Trust : A Perception from Shared Data in IoT," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2020, S. 1–6. DOI: 10.1109/ICCWorkshops49005.2020.9145071.

LITERATUR

- [14] L. Dominik, N. Tagalidou, K. Lingelbach und M. Spiekermann, *Ökosysteme für Daten und künstliche Intelligenz*, Positionspapier, 2019. doi: DOI10.24406/ISST-N-543753.
- [15] HEG-KI und Europäische-Kommission, *Eine Definition der KI: Wichtigste Fähigkeiten und Wissenschaftsgebiete*, Brüssel, 2019.
- [16] Dathenethikkommission, C. Wendehorst und C. Woopen, *Gutachten der Datenethikkommission*, 2019.
- [17] HEG-KI und Europäische-Kommission, *Ethik-Leitlinien für eine Vertrauenswürdige KI*, Brüssel, 2019.
- [18] OpenAI, I. Akkaya, M. Andrychowicz u.a., “Solving Rubik’s Cube with a Robot Hand,” *CoRR*, Jg. abs/1910.07113, 2019. arXiv: 1910.07113. Adresse: <http://arxiv.org/abs/1910.07113>.
- [19] A. Kharwal, *What is Supervised Learning in Machine Learning*, 2020, Letzter Zugriff: 28.05.2022 [Online]. Verfügbar:<https://the cleverprogrammer.com/2020/10/23/what-is-supervised-learning-in-machine-learning/>.
- [20] A. Ng, *Machine Learning Yearning - Technical Strateg for AI Engineers, In the Era of Deep Learning*, 2018.
- [21] S. Lapuschkin, A. Binder, G. Montavon, K.-R. Müller und W. Samek, “Analyzing Classifiers: Fisher Vectors and Deep Neural Networks,” Juni 2016, S. 2912–2920. doi: 10.1109/CVPR.2016.318.
- [22] L. Goasduff, *2 Megatrends Dominate the Gartner Hype Cycle for Artificial Intelligence, 2020*, 202, Letzter Zugriff: 25.05.2022 [Online]. Verfügbar:<https://www.gartner.com/smarterwithgartner/2-megatrends-dominate-the-gartner-hype-cycle-for-artificial-intelligence-2020>.
- [23] B. Beckert, “Trustworthy artificial intelligence: Selected practical projects and reasons for the implementation deficit,” *TATuP - Zeitschrift für Technikfolgenabschätzung in Theorie und Praxis*, Jg. 30, Nr. 3, S. 17–22, 2021. doi: 10.14512/tatup.30.3.17. Adresse: <https://www.tatup.de/index.php/tatup/article/view/6926>.
- [24] J. Heesen, *Ethik-Briefing. Leitfaden für eine verantwortungsvolle Entwicklung und Anwendung von KI-Systemen*, Whitepaper aus der Plattform Lernende Systeme, 2020.
- [25] T. Hagendorff, “Blind spots in AI ethics,” *AI and Ethics*, S. 1–17, 2021.
- [26] S. Hallensleben und C. Hustedt, *From Principles to Practice An interdisciplinary framework to operationalise AI ethics*, Imprint, 2020.
- [27] T. Hagendorff, “The ethics of AI ethics: An evaluation of guidelines,” *Minds and Machines*, Jg. 30, Nr. 1, S. 99–120, 2020.
- [28] A. Jobin, M. Ienca und E. Vayena, “The global landscape of AI ethics guidelines,” *Nature Machine Intelligence*, Jg. 1, Nr. 9, S. 389–399, 2019.
- [29] S. Fabi und T. Hagendorff, *Why we need biased AI – How including cognitive and ethical machine biases can enhance AI systems*, 2022. doi: 10.48550/ARXIV.2203.09911. Adresse: <https://arxiv.org/abs/2203.09911>.

LITERATUR

- [30] C. Dilmegani, *Bias in AI: What it is, Types, Examples and 6 Ways to Fix it in 2022*, 2020, Letzter Zugriff: 30.05.2022 [Online]. Verfügbar:<https://research.aimultiple.com/ai-bias/>.
- [31] J. Silberg und J. Manyika, "Notes from the AI frontier: Tackling bias in AI (and in humans)," *McKinsey Global Institute*, S. 1–6, 2019.
- [32] E. Ntoutsi, P. Fafalios, U. Gadiraju u. a., "Bias in data-driven artificial intelligence systems—An introductory survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Jg. 10, Nr. 3, e1356, 2020.
- [33] R. Srinivasan und A. Chander, "Biases in AI systems," *Communications of the ACM*, Jg. 64, Nr. 8, S. 44–49, 2021.
- [34] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman und A. Galstyan, "A Survey on Bias and Fairness in Machine Learning," *ACM Comput. Surv.*, Jg. 54, Nr. 6, Juli 2021, ISSN: 0360-0300. DOI: 10.1145/3457607. Adresse: <https://doi.org/10.1145/3457607>.
- [35] D. Roselli, J. Matthews und N. Talagala, "Managing Bias in AI," in *Companion Proceedings of The 2019 World Wide Web Conference*, Ser. WWW '19, San Francisco, USA: Association for Computing Machinery, 2019, S. 539–544, ISBN: 9781450366755. DOI: 10.1145/3308560.3317590. Adresse: <https://doi.org/10.1145/3308560.3317590>.
- [36] P. A, A. Jawaid, S. Dev und V. M S, "The Patterns that Don't Exist : Study on the effects of psychological human biases in data analysis and decision making," in *2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, 2018. DOI: 10.1109/CSITSS.2018.8768554.
- [37] Anonymous, *Incident Number 16. in McGregor, S. (ed.) Artificial Intelligence Incident Database*. Responsible AI Collaborative. 2015, Letzter Zugriff: 23.05.2022 [Online]. Verfügbar:incidentdatabase.ai/cite/16.
- [38] Anonymous, *Incident Number 6. in McGregor, S. (ed.) Artificial Intelligence Incident Database*, Responsible AI Collaborative, 2016, Letzter Zugriff: 30.05.2022 [Online]. Verfügbar:incidentdatabase.ai/cite/6.
- [39] Tay Microsoft Chatbot, 2016, Letzter Zugriff: 01.06.2022 [Online]. Verfügbar:pic.twitter.com/xuGi1u9S1A.
- [40] T. Hagendorff, "Maschinelles Lernen und Diskriminierung: Probleme und Lösungsansätze," *Österreichische Zeitschrift für Soziologie*, Jg. 44, Nr. 1, S. 53–66, 2019.
- [41] Google, *Responsible AI practices*, 2022, Letzter Zugriff: 01.06.2022 [Online]. Verfügbar:<https://ai.google/responsibilities/responsible-ai-practices/?category=fairness>.
- [42] J. Angwin, J. Larson, M. Surya und L. Kirchner, *Machine Bias - There's software used across the country to predict future criminals. And it's biased against blacks*. Pro Publica artikel, 2016, Letzter Zugriff: 28.05.2022 [Online]. Verfügbar:<https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing>.
- [43] J. Bronson und E. A. Carson, *Prisoners in 2017*, 2019, Letzter Zugriff: 17.05.2022. [Online]. Verfügbar: <https://bjs.ojp.gov/content/pub/pdf/p17.pdf>.

LITERATUR

- [44] S. R. Department, *Bevölkerung - Zahl der Einwohner in Deutschland nach Altersgruppen am 31. Dezember 2020*, 2022, Letzter Zugriff: 23.05.2022. [Online]. Verfügbar: <https://de.statista.com/statistik/daten/studie/1112579/umfrage/bevoelkerung-in-deutschland-nach-altersgruppen/>.
- [45] K.-A.-S. e. V., *Wahlverhalten nach Alter und Geschlecht*, 2021, Letzter Zugriff: 23.05.2022. [Online]. Verfügbar: <https://www.kas.de/documents/291186/291235/Datensammlung+Alter+und+Geschlecht.pdf/a25fb063-b305-d567-2d98-e77b1104f713?version=1.1&t=1618912553715>.
- [46] S. Bundesamt, *Datenreport 2021 - Kapitel 3: Bildung*, 2021, Letzter Zugriff: 23.05.2022. [Online]. Verfügbar: <https://www.destatis.de/DE/Service/Statistik-Campus/Datenreport/Downloads/datenreport-2021-kap-3.html>.
- [47] D. W. R. I. Washington, *Climate Watch Historical GHG Emissions*, 2022, Letzter Zugriff: 23.05.2022. [Online]. Verfügbar: <https://www.climatewatchdata.org/ghg-emissions>.

ANHANG A. DATENGENERATOR FÜR DAS SZENARIO ZUR BEWERTUNG
VON BEWÄHRUNGSANTRÄGE

A | Datengenerator für das Szenario zur Bewertung von Bewährungsanträgen

```
1 import numpy as np
2 from faker import Faker
3 import pandas as pd
4 from datetime import datetime
5 import random
6
7 fake = Faker()
```

Listing A.1: Erste Zelle für das Laden der Bibliotheken

```
1 def create_fake_data(num = 10, seed = 123):
2     #Setting the seed for the probability functions
3     np.random.seed(seed)
4     fake.seed_instance(seed)
5     #Defining the Output Array
6     output = []
7     #Loop over the number of requests to be created
8     for x in range(num):
9         #Random choice of sex with specified probability
10        sex = np.random.choice(["M", "W"], p=[0.927, 0.073])
11        #Random choice of hardness taking into account gender
12        hardnes = np.random.choice(["Leicht", "Mittel", "Hart"], p
13            =[0.266, 0.169, 0.565]) if sex=="M" else np.random.choice(["Leicht", "Mittel", "Hart"], p=[0.361, 0.264, 0.375])
14        #Random choice of skin colour taking into account gender
15        skin = np.random.choice(["Weiß", "Schwarz"], p=[0.459, 0.541])
16            if sex=="M" else np.random.choice(["Weiß", "Schwarz"], p
17            =[0.715, 0.285])
18        #Add entry to the array
19        output.append(
20            {
21                #Name of the person based on gender
22                "Name": fake.first_name_male() if sex=="M" else fake.
23                    first_name_female(),
24                "Hautfarbe": skin,
25                #Random skin colour
26                "Laufende_Strafe": np.random.choice([1,2,3,4,5]),
27                "Geschlecht": sex,
28                "Haerte_des_Vergehens": hardnes
29            }
30        )
31    #Return of the output array with one request per entry
32    return output
```

Listing A.2: Zweite Zelle für das Generieren der Daten

```
1 def create_Rules(request_values, request_bias, bias):
2     #Creating the Rule Dictionarys
3     rule_pos = {}
4     rule_neg = {}

---


```

ANHANG A. DATENGENERATOR FÜR DAS SZENARIO ZUR BEWERTUNG VON BEWÄHRUNGSANTRÄGE

```

5     rule_neg_bias = {}
6     rule_pos_bias = {}
7     #For each possible request value categoriser (Härte des Vergehens,
8     #Laufende Strafe)
9     for key in request_values.keys():
10        #The following characteristics exist(härte: Leicht,Mittel,Hart)
11        values = request_values[key]
12        #Determine mean value of the expressions to distribute 50/50
13        #between positive and negative rules
14        length = int(len(values)/2)+1
15        pos_values = []
16        #Adding the positive first values
17        for x in range(length):
18            pos_values.append(values[x])
19        #Add the expressions with the category to the regel_pos Dict
20        rule_pos[key] = pos_values
21        #Add the remaining expressions of the category to the rule_neg
22        neg_values = []
23        for y in range(len(values)):
24            if(y >= length):
25                neg_values.append(values[y])
26        rule_neg[key] = neg_values
27        #The distribution of the rules remains the same regardless of
28        #whether bias or not, so copy deep for now.
29        rule_pos_bias = rule_pos.copy()
30        rule_neg_bias = rule_neg.copy()
31        #If there is to be no bias then skip
32        if(bias != None):
33            #For each possible bias category
34            for key in request_bias.keys():
35                #If the value is equal to the specified bias parameter
36                if(bias.keys().__contains__(key)):
37                    #Adding the negative specified bias rule
38                    rule_neg_bias[key] = bias[key]
39                    neg_bias = bias[key]
40                    rule_pos_bias[key] = []
41                    #Adding the remaining opposite positive bias rule
42                    for val in request_bias[key]:
43                        if not(neg_bias.__contains__(val)):
44                            rule_pos_bias[key].append(val)
45        #Return of all 4 generated Rule Dicts
46        return(rule_pos, rule_neg, rule_pos_bias, rule_neg_bias)

```

Listing A.3: Dritte Zelle Funktion für das erstellen der Regeln

```

1 #Defining the dictionaries for the bias, the request values and the
2 #request bias values
3 bias = {
4     "Hautfarbe": ["Schwarz"]
5 }
6 request_values = {
7     "Laufende_Strafe": [1,2,3,4,5],
8     "Haerte_des_Vergehens": ["Leicht", "Mittel", "Hart"]
9 }
10 request_bias = {
11     "Hautfarbe": ["Schwarz", "Weiß"],
12     "Geschlecht": ["M", "W"]

```

ANHANG A. DATENGENERATOR FÜR DAS SZENARIO ZUR BEWERTUNG VON BEWÄHRUNGSANTRÄGE

```
12 }
13 #Create and output sample rules
14 r_pos, r_neg, rb_pos, rb_neg = create_Rules(request_values, request_bias
15 , bias)
16 print(r_pos)
17 print(r_neg)
18 print(rb_pos)
19 print(rb_neg)
```

Listing A.4: Vierte Zelle Testzelle zur Überprüfung der erzeugten Regeln

```
1 def generate_seed():
2     now = datetime.now()
3     #Calculate seed number from a few time data
4     seed = (now.day * now.minute * now.second * now.month * now.year *
5             now.hour) / now.microsecond
6     #If a negative or 0 value is the result, a "simpler" replacement
7     #seed is generated.
8     if(seed <= 0):
9         seed = now.day * (now.minute + 1)
10    return seed
```

Listing A.5: Fünfte Zelle Generierung eines zeitabhängigen Seed

```
1 class Evaluator:
2     #Creating a evaluator with its own rules and a bias or not
3     def __init__(self, rule_pos, rule_neg, bias, percentage=0.2):
4         self.rule_pos = rule_pos
5         self.rule_neg = rule_neg
6         self.bias = bias
7         self.bias_percentage = percentage
8     #Function to evaluate a submitted application with or without bias
9     def rate(self, request, bias):
10         #First 50/50 distribution
11         pos = 50
12         #Calculate the proportion according to which the decision is
13         #influenced positively or negatively.
14         prop = 45/self.rule_pos.__len__()
15         #Depending on how the rules match the request, the weight of the
16         #positive evaluation is shifted.
17         for key in self.rule_pos.keys():
18             if(self.rule_pos[key].__contains__(request[key])):
19                 pos += prop
20             else:
21                 pos -= prop
22         try:
23             #If a bias is present, this is additionally taken into
24             #account with the Parameter in %
25             if(self.bias):
26                 for b in bias:
27                     if(bias[b].__contains__(request[b])):
28                         pos = pos*self.bias_percentage
29             #Normalise positive value
30             pos = pos/100
31             #Determine negative value
32             neg = 1-pos
33             #Rating by chance with indication of pos and neg rating and
34             #adding the rating to the request.
```

ANHANG A. DATENGENERATOR FÜR DAS SZENARIO ZUR BEWERTUNG VON BEWÄHRUNGSANTRÄGE

```
31         request["Bewertung"] = np.random.choice(["positiv", "negativ"]
32             "", p=[pos, neg])
33     except:
34         print("Failure")
35     return request
```

Listing A.6: Sechste Zelle Klasse für das erstellen eines Bewerter Objektes

```
1 def generate_data(data_count):
2     seed = generate_seed()
3     df = pd.DataFrame(create_fake_data(data_count,int(seed)))
4     return df
5 def work(df, bias, evaluator_count, bias_evaluator, bias_percentage):
6     #Values in the request which only serve as filler and are therefore
7     #irrelevant
8     request_name = {
9         "Name": "random"
10    }
11    #Values in the request which influence the evaluation
12    request_values = {
13        "Laufende_Strafe": [1,2,3,4,5],
14        "Haerte_des_Vergehens": ["Leicht", "Mittel", "Hart"]
15    }
16    #Values in the request which can have an effect on the evaluation as
17    #a bias
18    request_bias = {
19        "Hautfarbe": ["Schwarz", "Weiß"],
20        "Geschlecht": ["M", "W"]
21    }
22    #Create the rules and save them in 4 variables
23    r_pos, r_neg, rb_pos, rb_neg = create_Rules(request_values,
24        request_bias, bias)
25    #Create the number of evaluators
26    evaluator = []
27    for x in range(evaluator_count):
28        evaluator.append(Evaluator(rule_pos=r_pos, rule_neg=r_neg, bias=
29            False))
30    #Convert the number of evaluators specified as parameters to
31    #evaluators with a bias
32    for x in range(bias_evaluator):
33        #Reset Rules and Bias Flag
34        evaluator[x].rule_pos = rb_pos
35        evaluator[x].rule_neg = rb_neg
36        evaluator[x].bias = True
37        evaluator[x].bias_percentage = bias_percentage
38    #Random number of evaluator selections
39    i = 0
40    #The final evaluated requests
41    finished_requests = []
42    #For each request in the dataframe
43    for index, r in df.iterrows():
44        #Determine a random evaluator from all the evaluators
45        i = random.randint(0, evaluator_count-1)
46        #Have the request evaluated and saved
47        req = evaluator[i].rate(r.copy(), bias)
48        #Add the finished request to the array
49        finished_requests.append(req)
```

ANHANG A. DATENGENERATOR FÜR DAS SZENARIO ZUR BEWERTUNG VON BEWÄHRUNGSANTRÄGE

```
45 #Save the finished requests as a data frame and return them
46 newdf = pd.DataFrame(finished_requests)
47 return newdf
```

Listing A.7: Siebte Zelle Methode zum Daten erstellen und Methode für den gesamt Ablauf

```
1 #Here is the section for the possible parameters to enter
2 #This dictionary specifies the bias(es) on a possible attribute
3 bias = {
4     "Hautfarbe": ["Weiß"]
5 }
6 #The number of datasets that are to be generated
7 datasets=10000
8 #The number of evaluators who evaluate entries
9 evaluator_count=10
10 #The number of evaluators who evaluate with a bias
11 bias_evaluator=4
12 #This decides how strong the bias will be. The smaller the stronger.
13 bias_percentage=0.2
14
15 #Dont touch this
16 data = generate_data(datasets)
17 finished = work(data,bias,evaluator_count,bias_evaluator,bias_percentage
18 )
18 data.to_csv("Daten.csv", sep=';', encoding='utf-8', index=False)
19 finished.to_csv("Daten_Bewertet.csv", sep=';', encoding='utf-8', index=
False)
```

Listing A.8: Achte Zelle für die Interaktion der benutzenden Person

B | Datengenerator für das Szenario zur Vorhersage eines sozialen Punktesystems

```
1 import numpy as np
2 from faker import Faker, Factory
3 import pandas as pd
4 from datetime import datetime
5 import random
6
7 fake = Factory.create("de_DE")
```

Listing B.1: Erste Zelle für das Laden der Bibliotheken

```
1 def generate_seed():
2     now = datetime.now()
3     #Calculate seed number from a few time data
4     seed = (now.day * now.minute * now.second * now.month * now.year *
5             now.hour) / now.microsecond
6     #If a negative or 0 value is the result, a "simpler" replacement
7     #seed is generated.
8     if(seed <= 0):
9         seed = now.day * (now.minute + 1)
10    return seed
```

Listing B.2: Zweite Zelle für das Generieren eines zeitabhängigen Seed

```
1 def create_fake_data(num = 10, seed = 123):
2     #Setting the seed for the probability functions
3     np.random.seed(seed)
4     fake.seed_instance(seed)
5     #Defining the Output Array
6     output = []
7     #Loop over the number of personal data to be created
8     for x in range(num):
9         age_part = np.random.choice([0,1,2], p=[0.33,0.38,0.29])
10        age = random.randint(20,39) if(age_part == 0) else random.
11            randint(40,59) if(age_part == 1) else random.randint(60,79)
12        if(age <= 24):
13            politics = np.random.choice(["Links", "Mitte", "Rechts"], p
14                =[0.28,0.631,0.089])
15        elif(age <= 34):
16            politics = np.random.choice(["Links", "Mitte", "Rechts"], p
17                =[0.244,0.614,0.142])
18        elif(age <= 44):
19            politics = np.random.choice(["Links", "Mitte", "Rechts"], p
20                =[0.216,0.618,0.166])
21        elif(age <= 59):
22            politics = np.random.choice(["Links", "Mitte", "Rechts"], p
23                =[0.207,0.635,0.158])
24        elif(age <= 69):
25            politics = np.random.choice(["Links", "Mitte", "Rechts"], p
26                =[0.177,0.687,0.136])
27        elif(age <= 79):
```

ANHANG B. DATENGENERATOR FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

```
22     politics = np.random.choice(["Links", "Mitte", "Rechts"], p
23         =[0.107,0.809,0.084])
24
25     if(age <= 29):
26         grad = np.random.choice(["Ausbildung", "Fachschulabschluss",
27             "Bachelor", "Master", "Diplom", "Promotion","ohne"], p
28             =[0.425,0.081,0.117,0.071,0.059,0.003,0.244])
29     elif(age <= 39):
30         grad = np.random.choice(["Ausbildung", "Fachschulabschluss",
31             "Bachelor", "Master", "Diplom", "Promotion","ohne"], p
32             =[0.452,0.091,0.059,0.052,0.157,0.016,0.173])
33     elif(age <= 49):
34         grad = np.random.choice(["Ausbildung", "Fachschulabschluss",
35             "Bachelor", "Master", "Diplom", "Promotion","ohne"], p
36             =[0.516,0.097,0.014,0.011,0.179,0.017,0.166])
37     elif(age <= 59):
38         grad = np.random.choice(["Ausbildung", "Fachschulabschluss",
39             "Bachelor", "Master", "Diplom", "Promotion","ohne"], p
40             =[0.559,0.113,0.005,0.003,0.159,0.014,0.147])
41     else:
42         grad = np.random.choice(["Ausbildung", "Fachschulabschluss",
43             "Bachelor", "Master", "Diplom", "Promotion","ohne"], p
44             =[0.544,0.096,0.002,0.001,0.132,0.012,0.213])
45
46     social = np.random.choice([0,1,2,3], p=[0.6,0.2,0.15,0.05]) if(
47         politics=="Mitte") else np.random.choice([0,1,2,3], p
48             =[0.4,0.25,0.25,0.1]) if(politics=="Links") else np.random.
49             choice([0,1,2,3], p=[0.75,0.15,0.09,0.01])
50
51     location = np.random.choice(["Großstadt","Kleinstadt","Vorort",
52         "Ländlich"], p=[0.5952,0.266,0.085,0.0538])
53
54     if(location == "Großstadt"):
55         if(social == 0):
56             co2 = np.random.choice
57                 ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
58
59                     p
60                         =[0.001,0.003,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.147,0.
61
62                         0]
63
64             elif(social == 1):
65                 co2 = np.random.choice
66                     ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
67
68                         p
69                             =[0.004,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.147,0.116,0.
70
71                         0]
72
73             elif(social == 2):
74                 co2 = np.random.choice
75                     ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
76
77                         p
78                             =[0.013,0.022,0.044,0.078,0.116,0.147,0.160,0.147,0.116,0.078,0.
79
80                         0]
81
82             elif(social == 3):
83                 co2 = np.random.choice
84                     ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
```

ANHANG B. DATENGENERATOR FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

```
48     p      =[0.035,0.044,0.078,0.116,0.147,0.160,0.147,0.116,0.078,0.044,0.
49
50         elif(location == "Kleinstadt"):
51             if(social == 0):
52                 co2 = np.random.choice
53                     ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
54
55             p      =[0,0,0.001,0.003,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.147
56
57         elif(social == 1):
58             co2 = np.random.choice
59                     ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
60
61             p      =[0.001,0.003,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.147,0.
62
63         elif(social == 2):
64             co2 = np.random.choice
65                     ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
66
67             p      =[0.004,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.147,0.116,0.
68
69         elif(social == 3):
70             co2 = np.random.choice
71                     ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
72
73             p      =[0.013,0.022,0.044,0.078,0.116,0.147,0.160,0.147,0.116,0.078,0.
```

ANHANG B. DATENGENERATOR FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

```
73         co2 = np.random.choice
74             ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
75             p
76             =[0.004,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.147,0.116,0
77
78     else:
79         if(social == 0):
80             co2 = np.random.choice
81                 ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
82
83             p
84             =[0,0,0,0.001,0.003,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.1
85
86         elif(social == 1):
87             co2 = np.random.choice
88                 ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
89
90             p
91             =[0,0,0.001,0.003,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.147,
92
93         elif(social == 2):
94             co2 = np.random.choice
95                 ([4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12],
96
97             p
98             =[0,0.001,0.003,0.009,0.022,0.044,0.078,0.116,0.147,0.160,0.147,0
99
100        output.append(
101            {
102                #Name of the person
103                "Name": fake.first_name(),
104                "Alter": age,
105                "Politische Orientierung": politics,
106                "Bildungsabschluss": grad,
107                "Soziales": social,
108                "Wohnlage": location,
109                "CO2-Fußabdruck": co2
110            }
111        )
112    )
113
114    #Return of the output array with one person per entry
115    return output
```

Listing B.3: Dritte Zelle für das Generieren der Daten

```
1 def generate_data(data_count):
2     seed = generate_seed()
3     df = pd.DataFrame(create_fake_data(data_count,int(seed)))
4     return df
```

ANHANG B. DATENGENERATOR FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

Listing B.4: Vierte Zelle Methode zum Generieren der Daten

```
1 additional = {
2     "Name": "Random",
3     "Alter": "20-79"
4 }
5 influential = {
6     "Politische Orientierung": ["Links", "Mitte", "Rechts"],
7     "Bildungsabschluss": ["Ausbildung", "Fachschulabschluss", "Bachelor",
8                           , "Master", "Diplom", "Promotion", "ohne"],
9     "Soziales": [0,1,2,3],
10    "Wohnlage": ["Großstadt", "Kleinstadt", "Vorort", "Ländlich"],
11    "CO2-Fußabdruck": [4,5,6,7,8,9,10,11,12]
12 }
13 bias = {
14     "Alter": "20-79",
15     "Politische Orientierung": ["Links", "Mitte", "Rechts"],
16     "Bildungsabschluss": ["Ausbildung", "Fachschulabschluss", "Bachelor",
17                           , "Master", "Diplom", "Promotion", "ohne"],
18     "Soziales": [0,1,2,3],
19     "Wohnlage": ["Großstadt", "Kleinstadt", "Vorort", "Ländlich"],
20     "CO2-Fußabdruck": [4,5,6,7,8,9,10,11,12]
21 }
22 def create_Rules():
23     influential = {
24         "Politische Orientierung": [-50, 50, -50],
25         "Bildungsabschluss": [20, 40, 60, 80, 100, 120, -120],
26         "Soziales": [-70, 0, 40, 100],
27         "Wohnlage": [-30, 20, 20, -30],
28         "CO2-Fußabdruck": [110, 70, 40, 10, 0, -20, -40, -70, -110]
29     }
30     return influential
```

Listing B.5: Fünfte Zelle Methode zum Erstellen der Regeln

```
1 class Evaluator:
2     #Creating a evaluator with its own rules and a bias or not
3     def __init__(self, rules, bias, bias_neg=200):
4         self.rules = rules
5         self.bias = bias
6         self.bias_neg = bias_neg
7     #Function to evaluate a submitted person with or without bias
8     def rate(self, influential, person, bias):
9         rate = 1000
10        for key in self.rules.keys():
11            if(key == "CO2-Fußabdruck"):
12                value_of_key = int(person[key])
13                index = influential[key].index(value_of_key)
14                rate += self.rules[key][index]
15                procent = self.rules[key][index] * 0.15
16                procent = round(procent,0)
17                if(procent<0):
18                    rand = random.randint(0,(procent*-1))
19                    rate -= int(rand)
20                else:
```

ANHANG B. DATENGENERATOR FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

```

21         rand = random.randint(0, procent)
22         rate += int(rand)
23     else:
24         value_of_key = person[key]
25         index = influential[key].index(value_of_key)
26         rate += self.rules[key][index]
27         procent = self.rules[key][index] * 0.15
28         procent = round(procent,0)
29     if(procent<0):
30         rand = random.randint(0,(procent*-1))
31         rate -= int(rand)
32     else:
33         rand = random.randint(0,procent)
34         rate += int(rand)
35     if(self.bias):
36         for b in bias:
37             if(b == "Alter"):
38                 under = int(bias[b].split(' -')[0])
39                 upper = int(bias[b].split(' -')[1])
40                 if(person[b]<=upper and person[b]>=under):
41                     red_val = self.bias_neg
42                     if((rate-red_val)<600):
43                         rate_part = np.random.choice([0,1,2], p
44                                         =[0.4,0.35,0.25])
45                         biasrate = random.randint(600,610) if(
46                                         rate_part == 0) else random.randint
47                                         (611,620) if(rate_part == 1) else random
48                                         .randint(621,630)
49                         rate = biasrate
50                     else:
51                         rate-=red_val
52                 else:
53                     if(bias[b].__contains__(person[b])):
54                         red_val = self.bias_neg
55                         if((rate-red_val)<600):
56                             rate_part = np.random.choice([0,1,2], p
57                                         =[0.4,0.35,0.25])
58                             biasrate = random.randint(600,610) if(
59                                         rate_part == 0) else random.randint
60                                         (611,620) if(rate_part == 1) else random
61                                         .randint(621,630)
62                         rate = biasrate
63                     else:
64                         rate-=red_val
65                 if(rate<600):
66                     rate = 600
67                 if(rate>1400):
68                     rate = 1400
69             person["Bewertung"] = int(rate)
70     return person

```

Listing B.6: Sechste Zelle Klasse für das erstellen eines Bewerter Objektes

```

1 def work(df, bias, evaluator_count, bias_evaluator, bias_neg):
2     #Values in the personprofile which only serve as filler and are
3     #therefore irrelevant
4     additional = {

```

ANHANG B. DATENGENERATOR FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

```

4     "Name": "Random",
5     "Alter": "20-79"
6 }
7 #Values in the personprofile which influence the evaluation
8 influential = {
9     "Politische Orientierung": ["Links", "Mitte", "Rechts"],
10    "Bildungsabschluss": ["Ausbildung", "Fachschulabschluss", "Bachelor", "Master", "Diplom", "Promotion", "ohne"],
11    "Soziales": [0,1,2,3],
12    "Wohnlage": ["Großstadt", "Kleinstadt", "Vorort", "Ländlich"],
13    "CO2-Fußabdruck": [4,5,6,7,8,9,10,11,12]
14 }
15 #Values in the personprofile which can have an effect on the
16 #evaluation as a bias
16 personprofile_bias = {
17     "Alter": "20-79",
18     "Politische Orientierung": ["Links", "Mitte", "Rechts"],
19     "Bildungsabschluss": ["Ausbildung", "Fachschulabschluss", "Bachelor", "Master", "Diplom", "Promotion", "ohne"],
20     "Soziales": [0,1,2,3],
21     "Wohnlage": ["Großstadt", "Kleinstadt", "Vorort", "Ländlich"],
22     "CO2-Fußabdruck": [4,5,6,7,8,9,10,11,12]
23 }
24 #Create the rules and save them
25 rules = create_Rules()
26 #Create the number of evaluators
27 evaluator = []
28 for x in range(evaluator_count):
29     evaluator.append(Evaluator(rules=rules, bias=False))
30 #Convert the number of evaluators specified as parameters to
31 #evaluators with a bias
31 for x in range(bias_evaluator):
32     evaluator[x].bias = True
33     evaluator[x].bias_neg = bias_neg
34 #Random number of evaluator selections
35 i = 0
36 #The final evaluated persons
37 finished_persons = []
38 #For each person in the dataframe
39 for index, r in df.iterrows():
40     #Determine a random evaluator from all the evaluators
41     i = random.randint(0, evaluator_count-1)
42     #Have the personprofile evaluated and saved
43     person = evaluator[i].rate(influential, r.copy(), bias)
44     #Add the finished personprofile to the array
45     finished_persons.append(person)
46 #Save the finished personprofiles as a data frame and return them
47 newdf = pd.DataFrame(finished_persons)
48 return newdf

```

Listing B.7: Siebte Zelle Methode für den gesamt Ablauf

```

1 #Here is the section for the possible parameters to enter
2 #This is an example of how a bias on age can be indicated
3 age_sample = {
4     "Alter": "20-30"
5 }

```

ANHANG B. DATENGENERATOR FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

```
6 #This dictionary specifies the bias(es) on a possible attribute
7 bias = {
8     "Alter": "20-30",
9     "Politische Orientierung": ["Links"]
10}
11 #The number of datasets that are to be generated
12 datasets = 10000
13 #The number of evaluators who evaluate entries
14 evaluatorcount = 10
15 #The number of evaluators who evaluate with a bias
16 bias_evaluator = 4
17 #This decides how strong the bias will be. The higher the stronger.
18 negativ_bias_impact = 200
19
20 #Dont touch this
21 data = generate_data(datasets)
22 finished = work(df=data, bias=bias, evaluator_count=evaluatorcount,
23     bias_evaluator=bias_evaluator, bias_neg=negativ_bias_impact)
23 data.to_csv("Daten_Szenario2.csv", sep=';', encoding='utf-8', index=
24     False)
24 finished.to_csv("Daten_Bewertet_Szenario2.csv", sep=';', encoding='utf-8',
25     , index=False)
```

Listing B.8: Achte Zelle für die Interaktion der benutzenden Person

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE

**C | Tableau für das Szenario zur Bewertung
von Bewährungsanträgen**

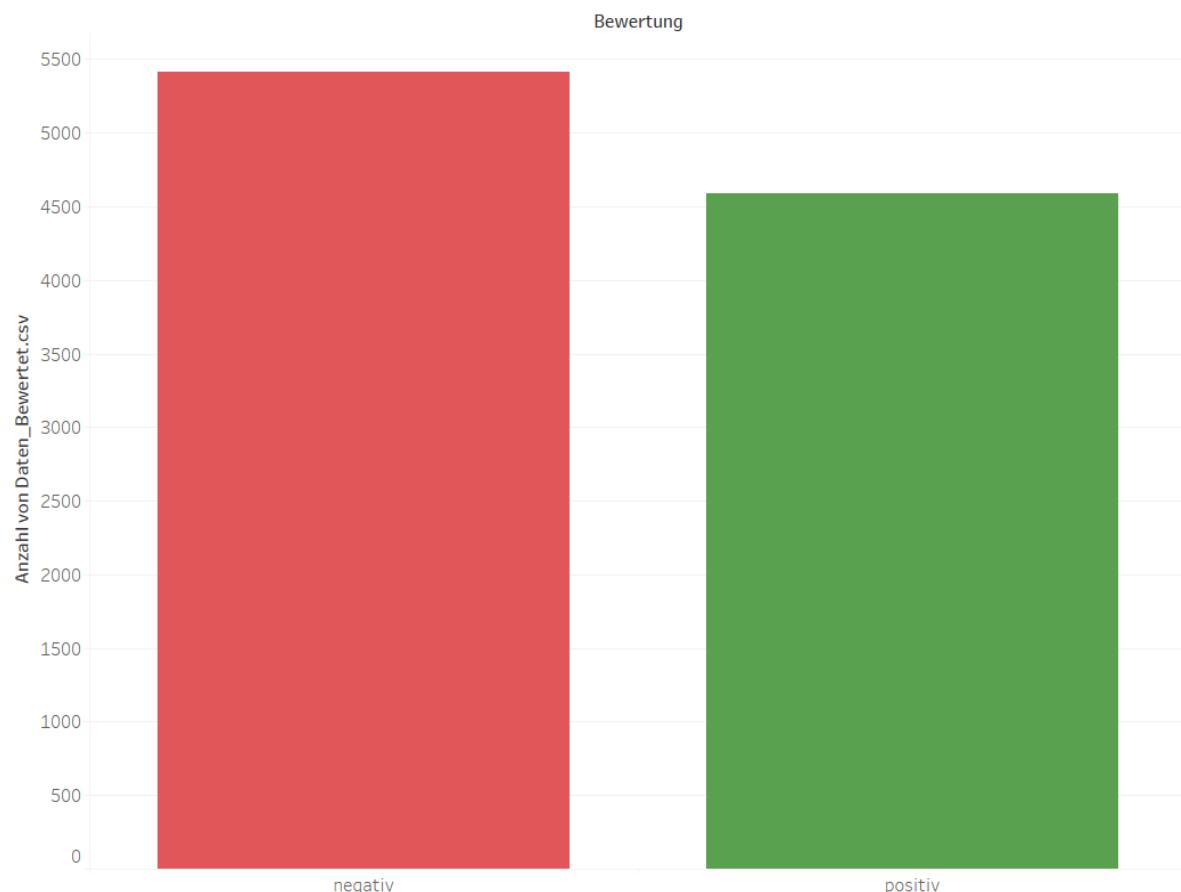


Abbildung C.1: Überblick über die Verteilung der Bewertung

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE

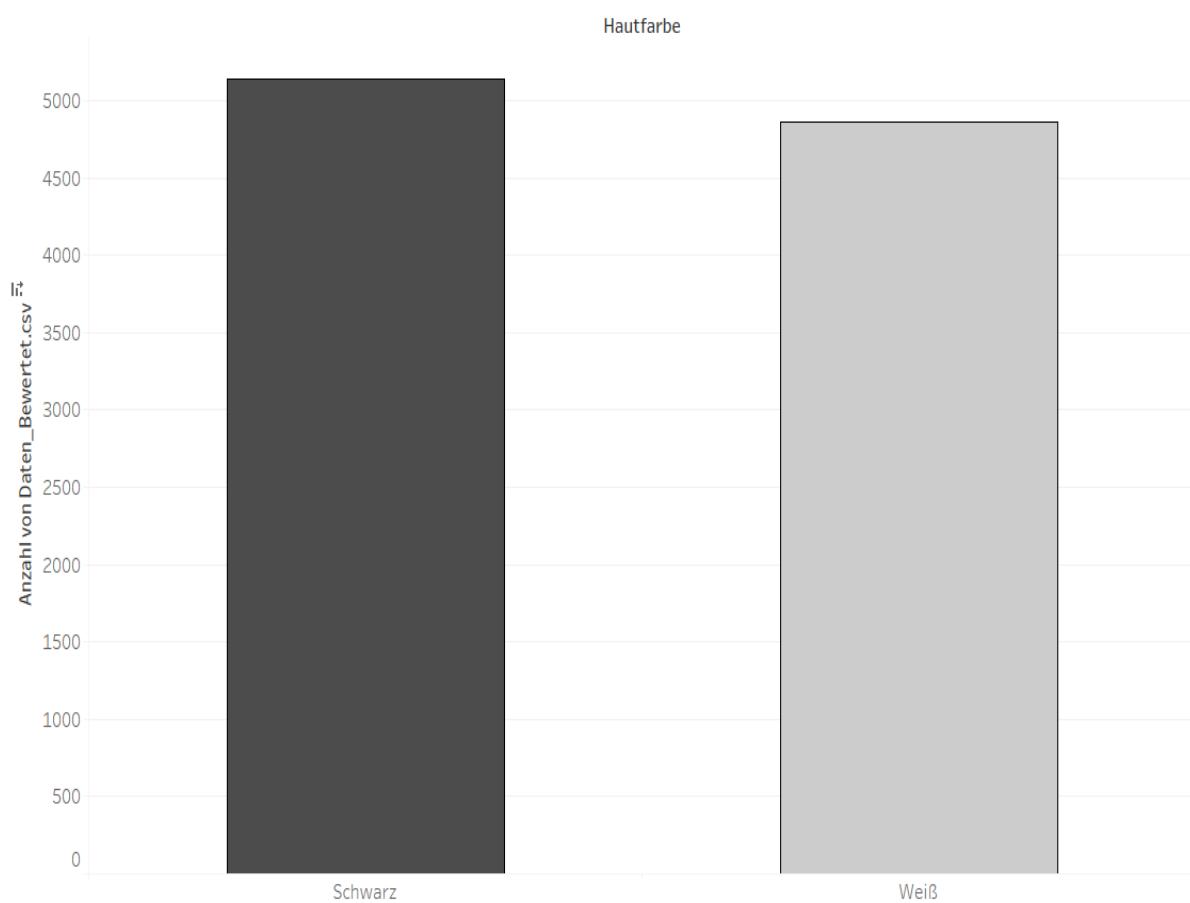


Abbildung C.2: Überblick über die Verteilung der Hautfarbe

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE

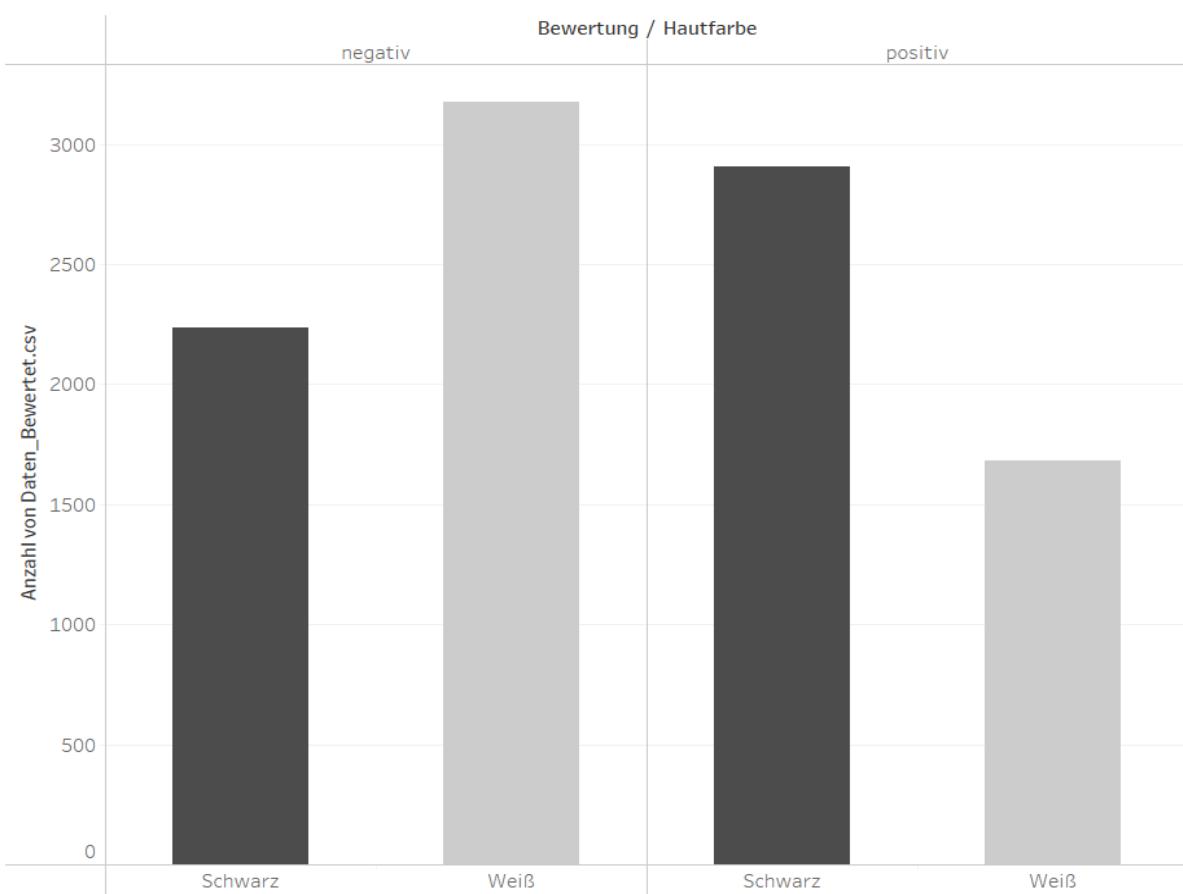


Abbildung C.3: Auswertung der Hautfarbe im Zusammenhang mit der Bewertung nach Anzahl der gesamt Daten

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE

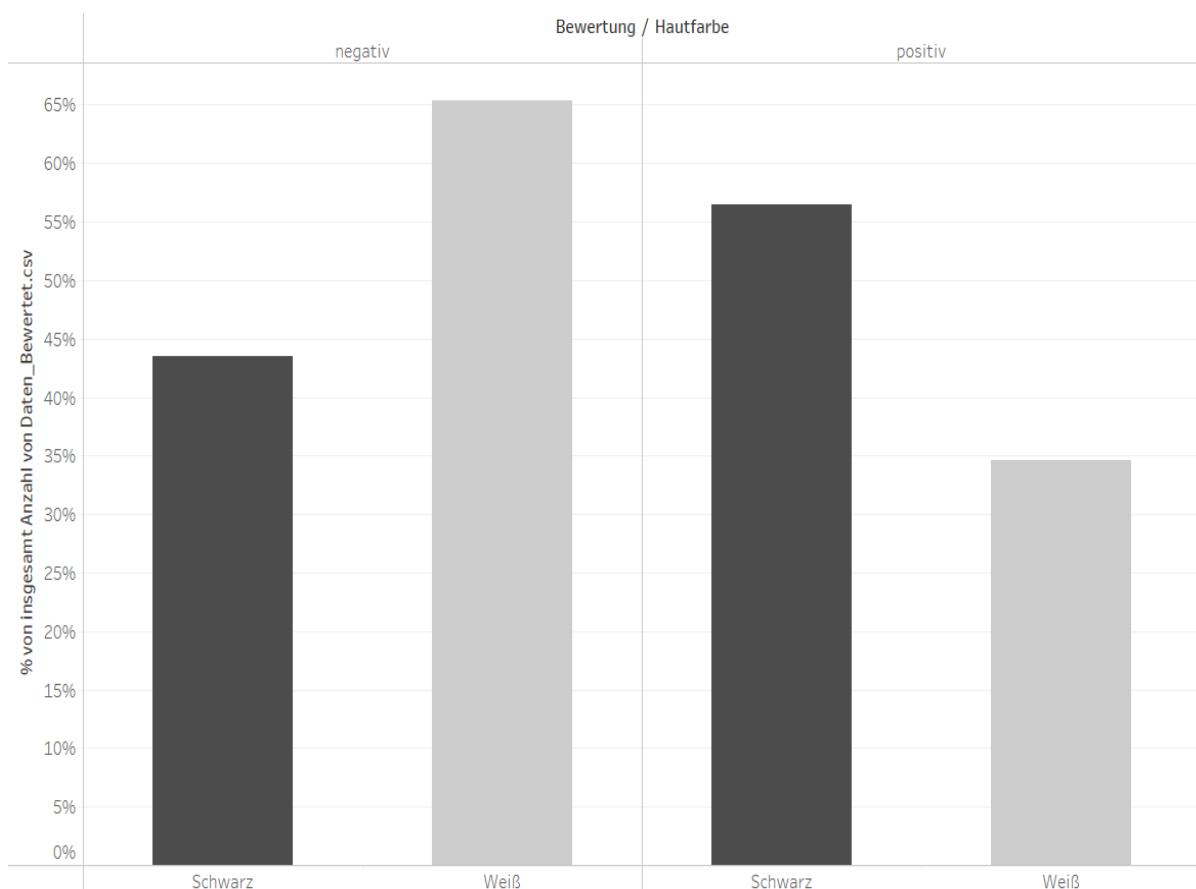


Abbildung C.4: Auswertung der Hautfarbe im Zusammenhang mit der Bewertung nach Prozent von der gesamt Anzahl an Daten

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE

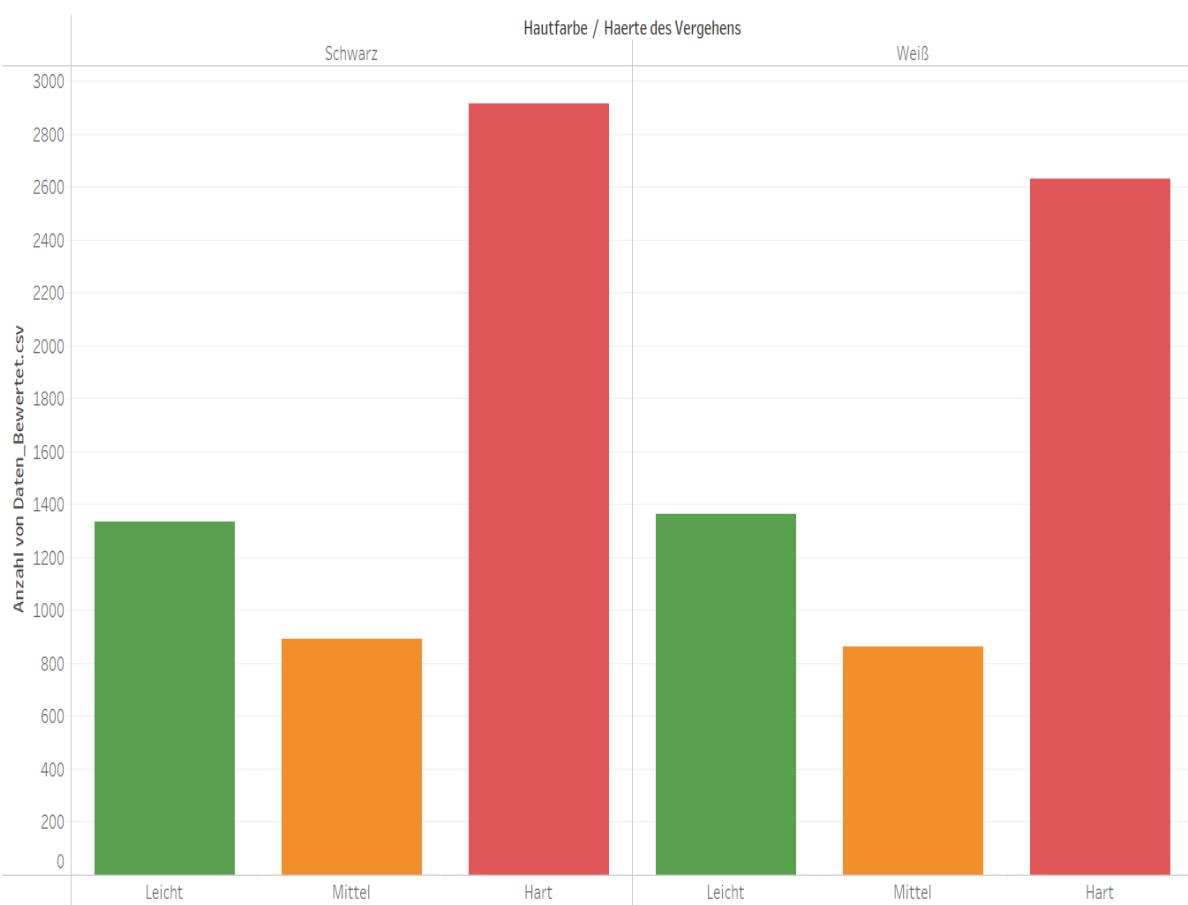


Abbildung C.5: Auswertung der Hautfarbe im Zusammenhang mit der Härte des Vergehens nach Anzahl der gesamt Daten

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON BEWÄHRUNGSANTRÄGE

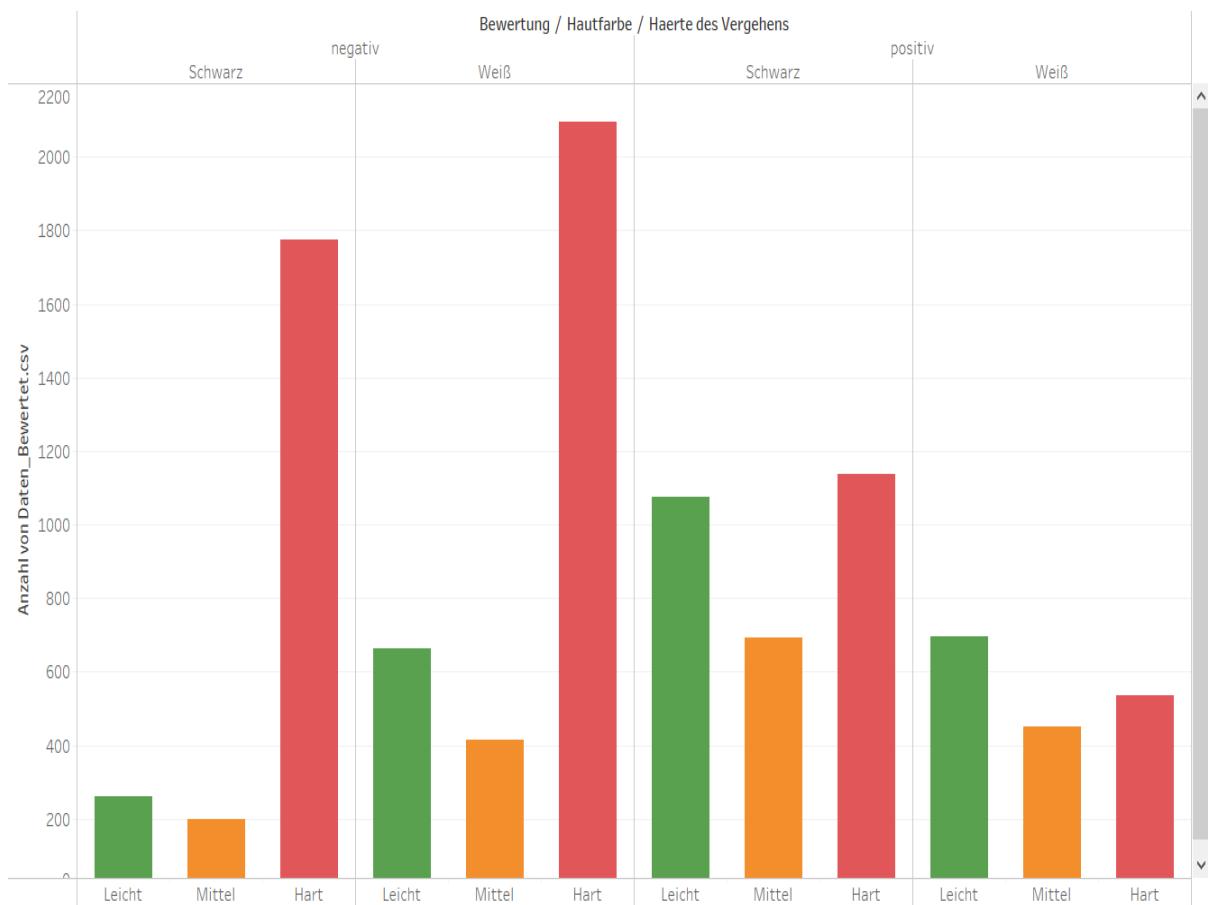


Abbildung C.6: Auswertung der Hautfarbe im Zusammenhang mit der Härte der Strafe und der Bewertung nach der gesamt Anzahl an Daten

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE

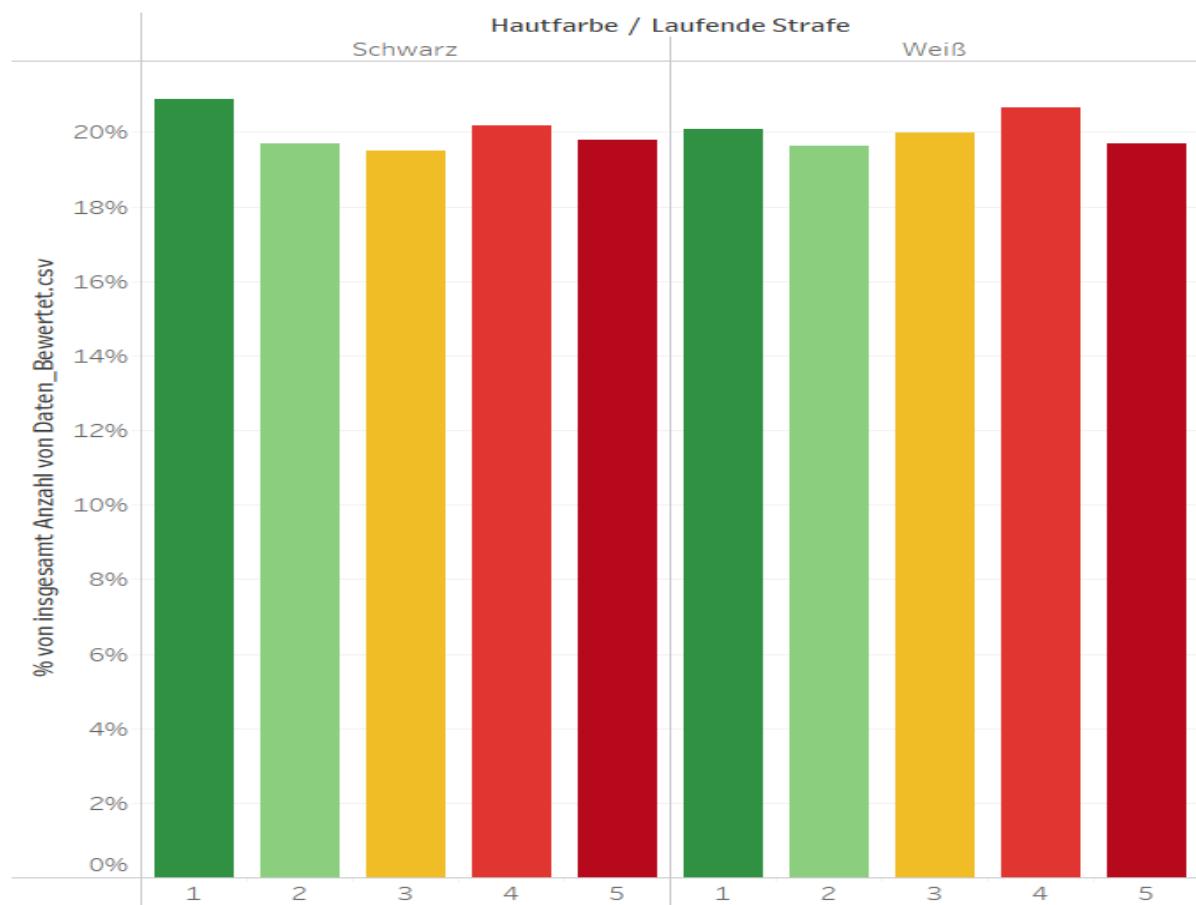


Abbildung C.7: Auswertung der Hautfarbe im Zusammenhang mit der Laufenden Strafe nach Prozent von der gesamt Anzahl an Daten

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE

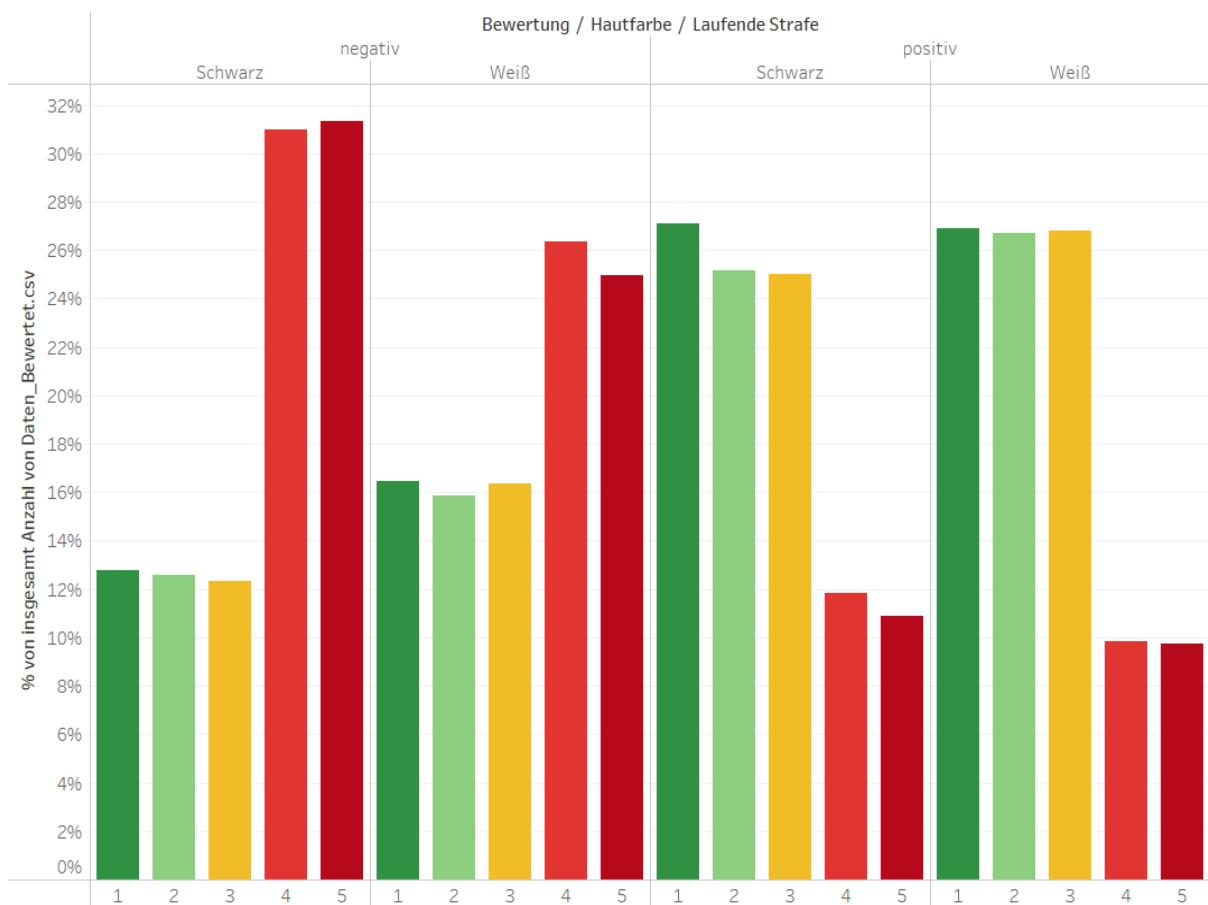


Abbildung C.8: Auswertung der Hautfarbe im Zusammenhang mit der Laufenden Strafe und der Bewertung nach Prozent von der gesamt Anzahl an Daten

**ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE**



Abbildung C.9: Auswertung der Hautfarbe im Zusammenhang mit der Bewertung, Härte der Strafe und Laufenden Strafe nach der gesamt Anzahl an Daten

ANHANG C. TABLEAU FÜR DAS SZENARIO ZUR BEWERTUNG VON
BEWÄHRUNGSANTRÄGE

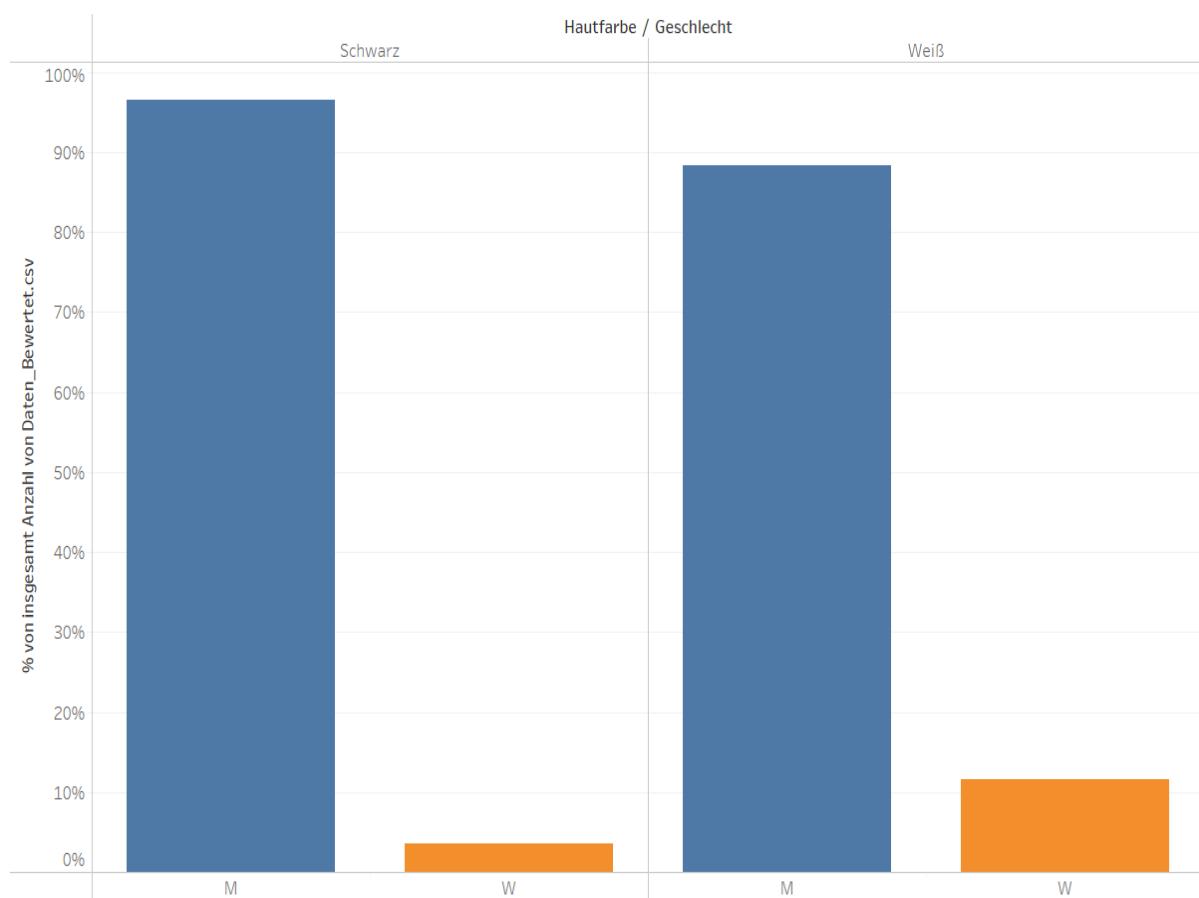


Abbildung C.10: Verteilung des Geschlechts über die Hautfarbe

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

D | Tableau für das Szenario zur Vorhersage eines sozialen Punktesystems

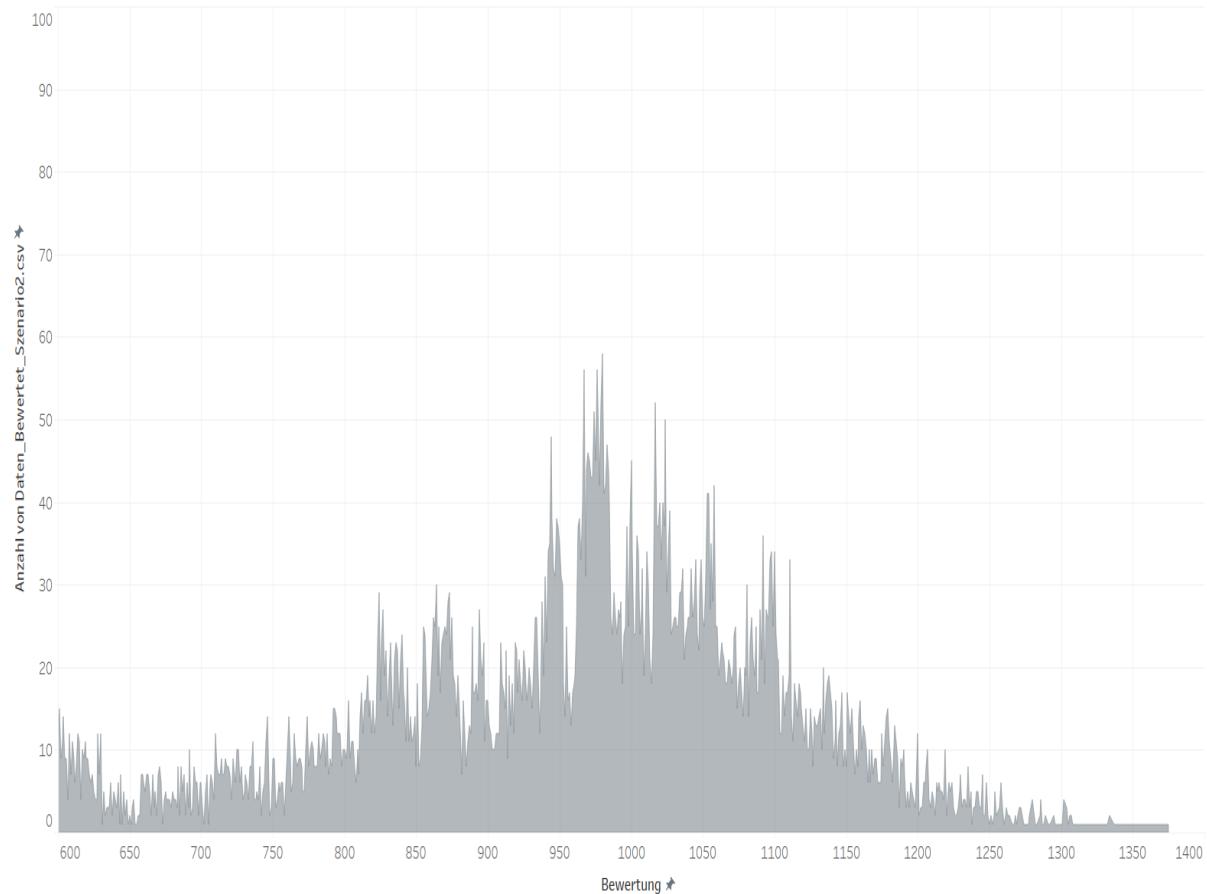


Abbildung D.1: Überblick über die Verteilung der Bewertung

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

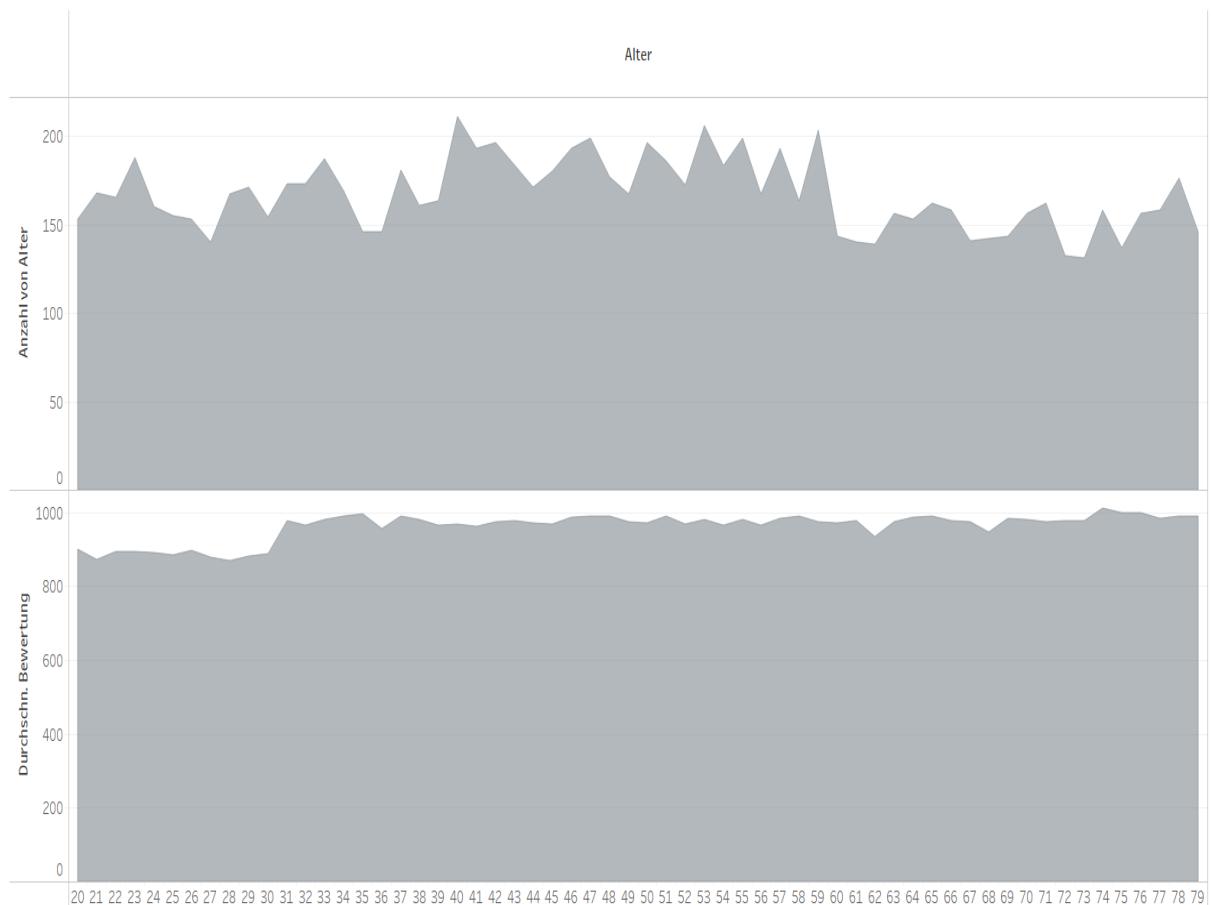


Abbildung D.2: Auswertung der Anzahl von Personen und durchschnittlicher Punktestand pro Alter

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

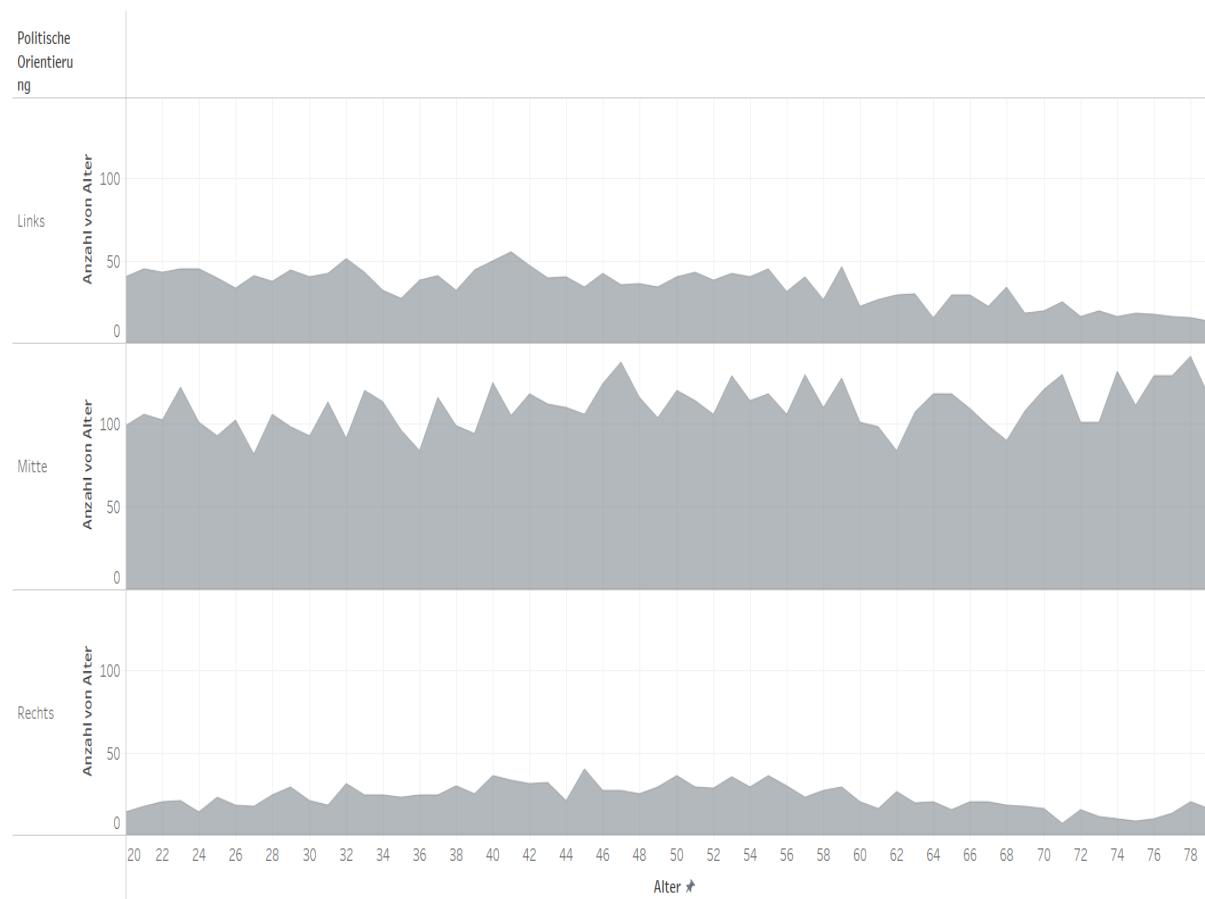


Abbildung D.3: Auswertung der Anzahl von Personen pro Alter und politische Orientierung

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

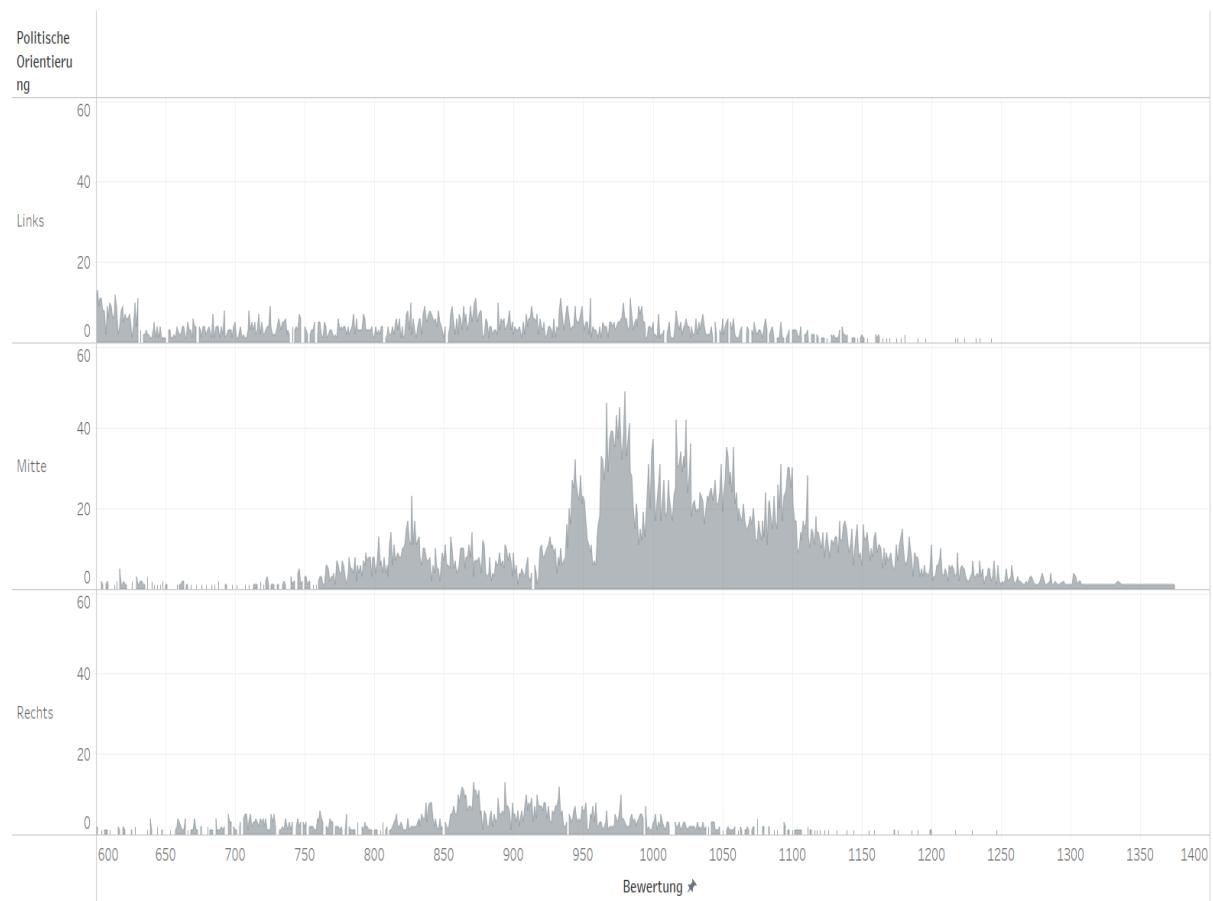


Abbildung D.4: Auswertung des Punktestand pro politische Orientierung über alle Datenpunkte

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

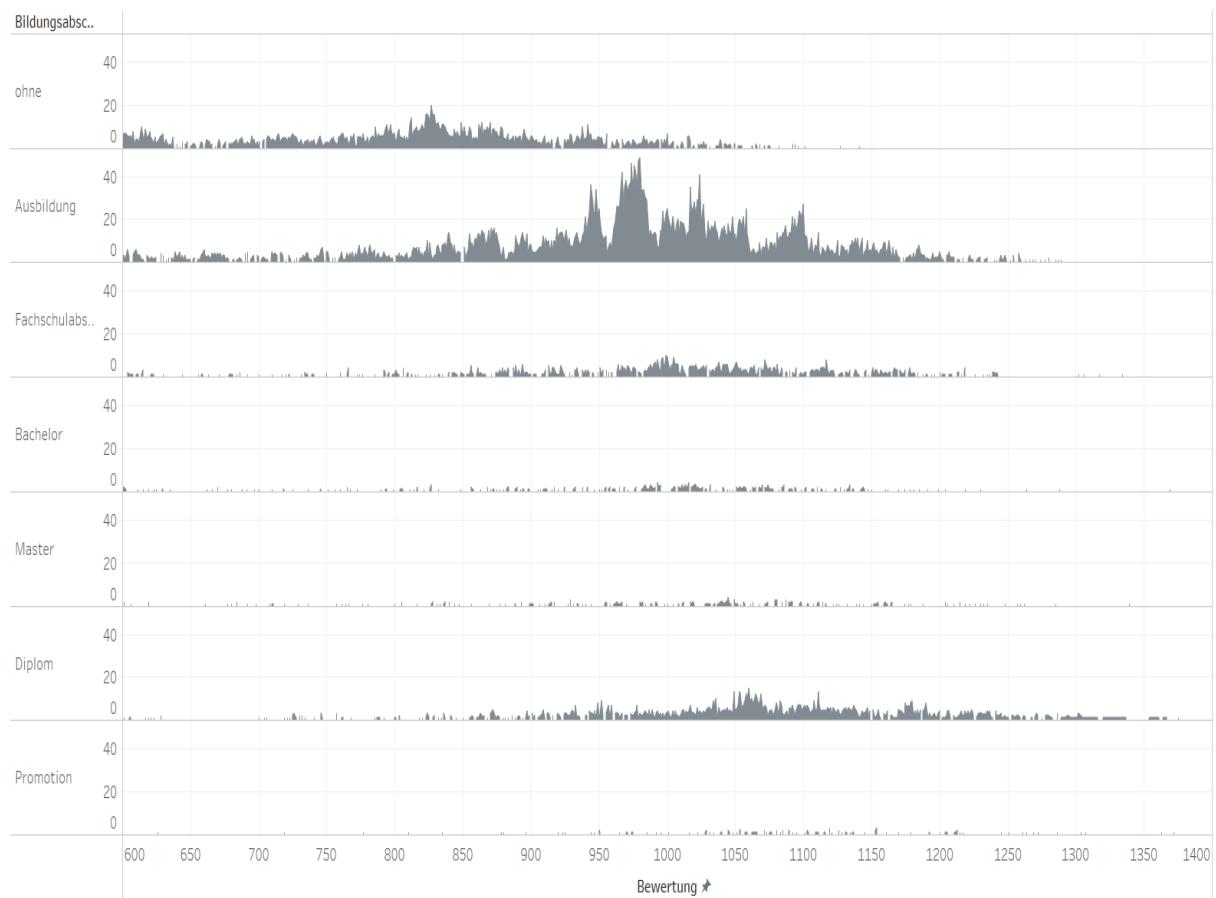


Abbildung D.5: Auswertung des Punktestand für jeden Bildungsabschluss über alle Datenpunkte

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

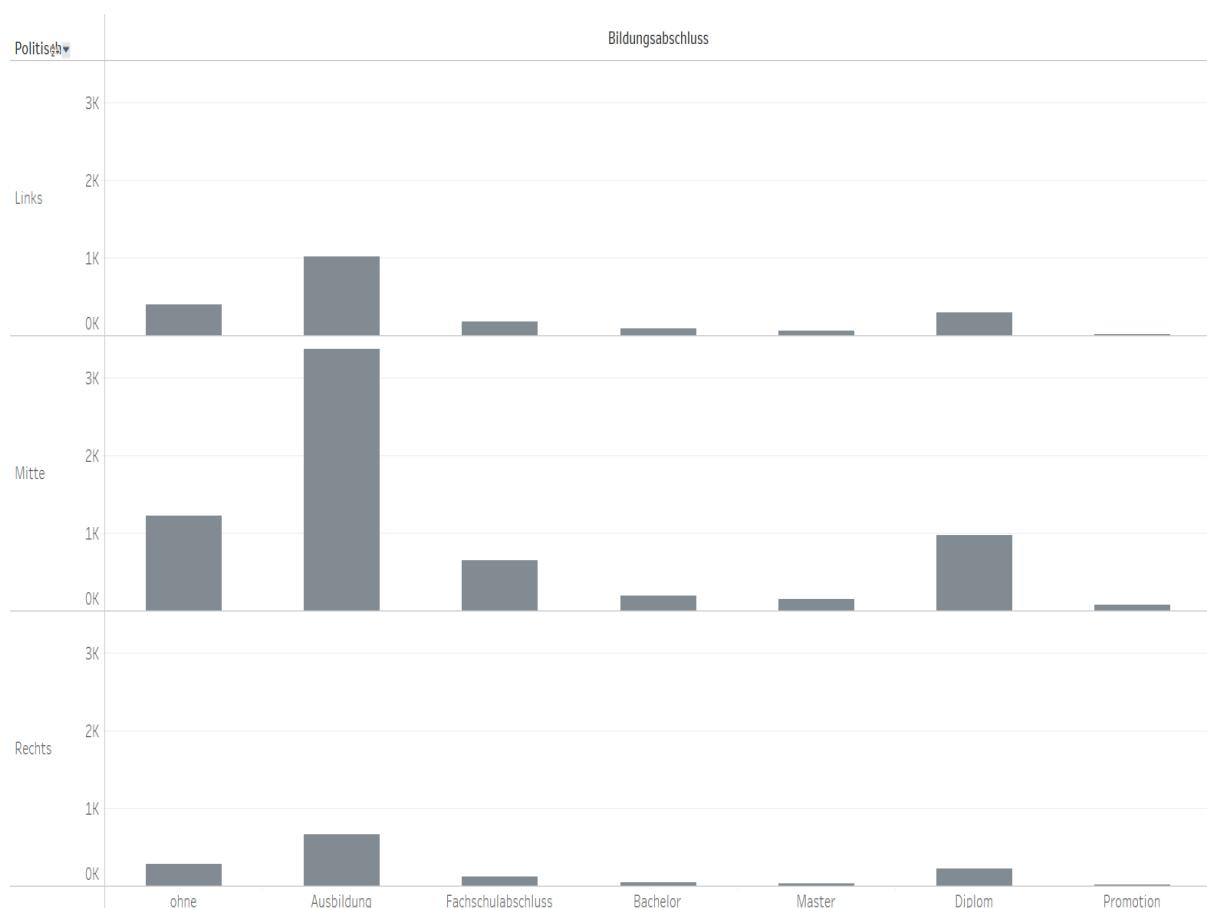


Abbildung D.6: Auswertung der Anzahl an Datenpunkten für jeden Bildungsabschluss und jede politische Orientierung

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES SOZIALEN PUNKTESYSTEMS

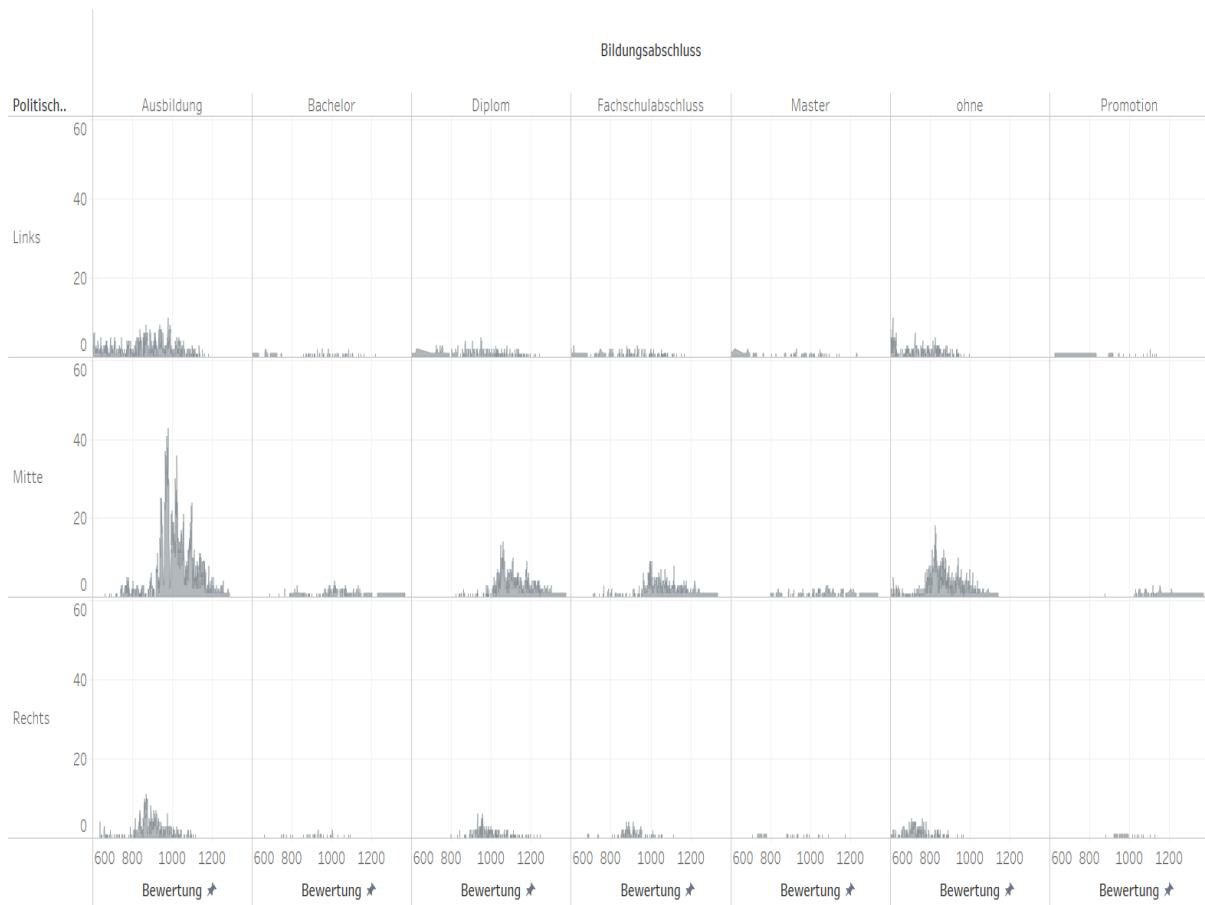


Abbildung D.7: Auswertung des Punktestand für jeden Bildungsabschluss und politische Orientierung über alle Datenpunkte

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

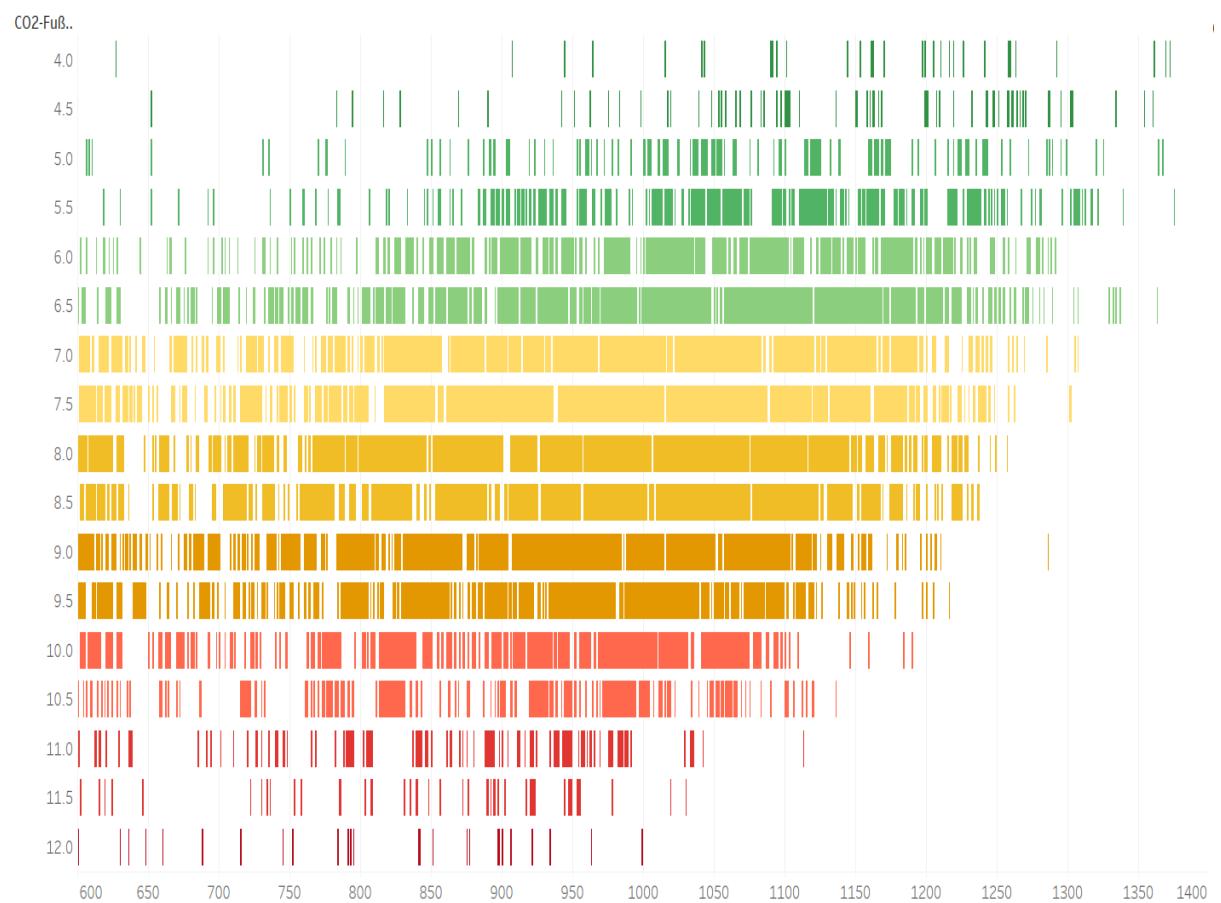


Abbildung D.8: Auswertung des Punktestand für den CO2-Fußabdruck über alle Datenpunkte

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

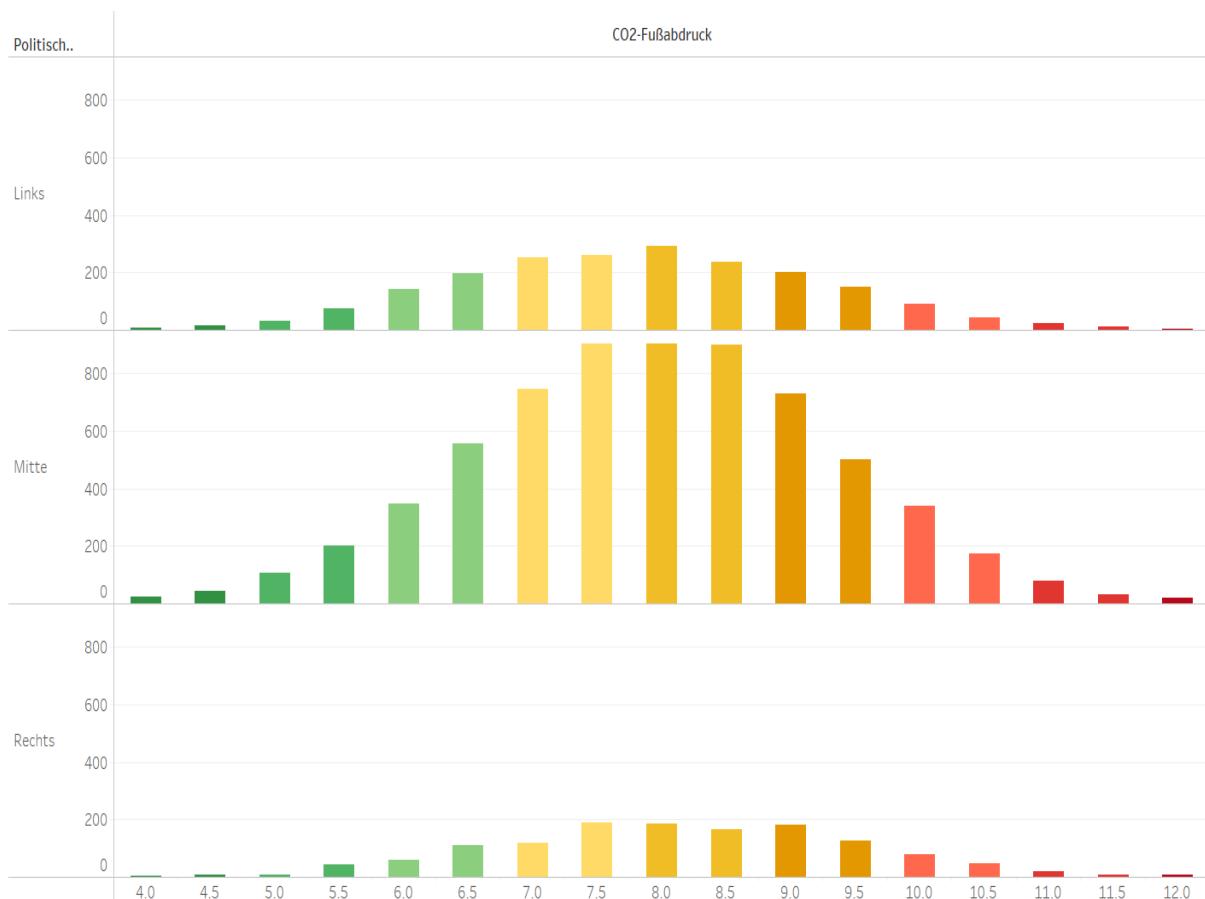


Abbildung D.9: Auswertung der Anzahl an Datenpunkten für den CO2-Fußabdruck pro politische Orientierung

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

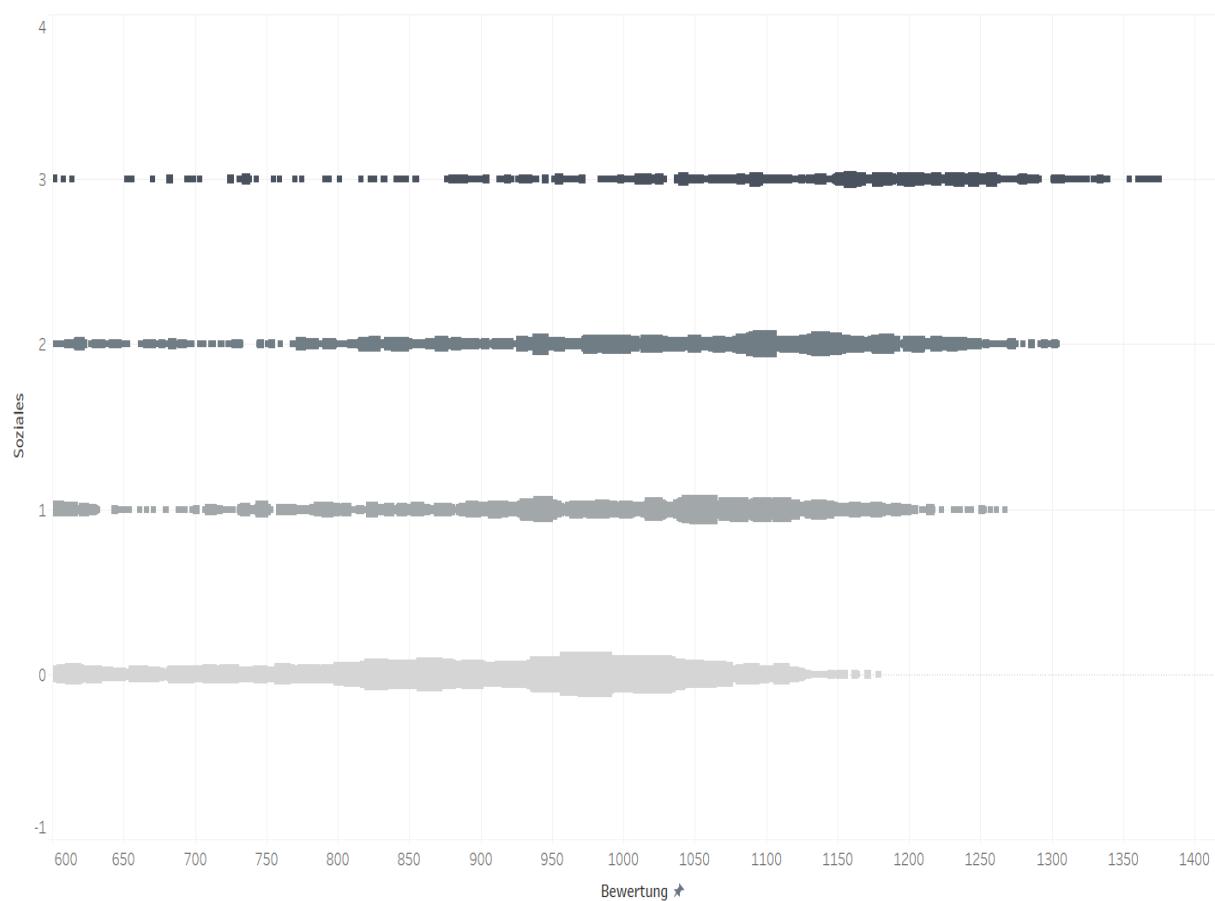


Abbildung D.10: Auswertung des Punktestand für das soziale Engagement über alle Datenpunkte

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

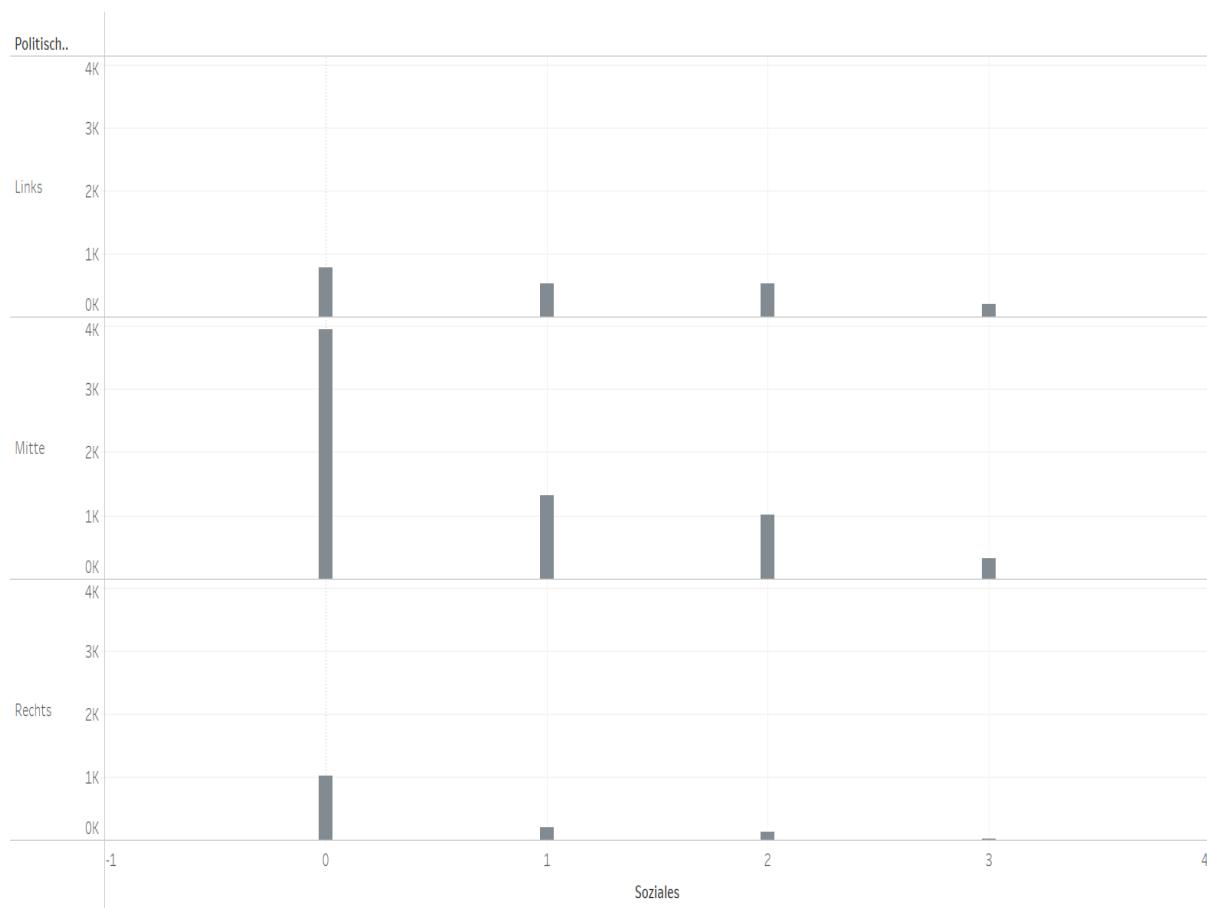


Abbildung D.11: Auswertung der Anzahl an Datenpunkten für das soziale Engagement pro politische Orientierung

ANHANG D. TABLEAU FÜR DAS SZENARIO ZUR VORHERSAGE EINES
SOZIALEN PUNKTESYSTEMS

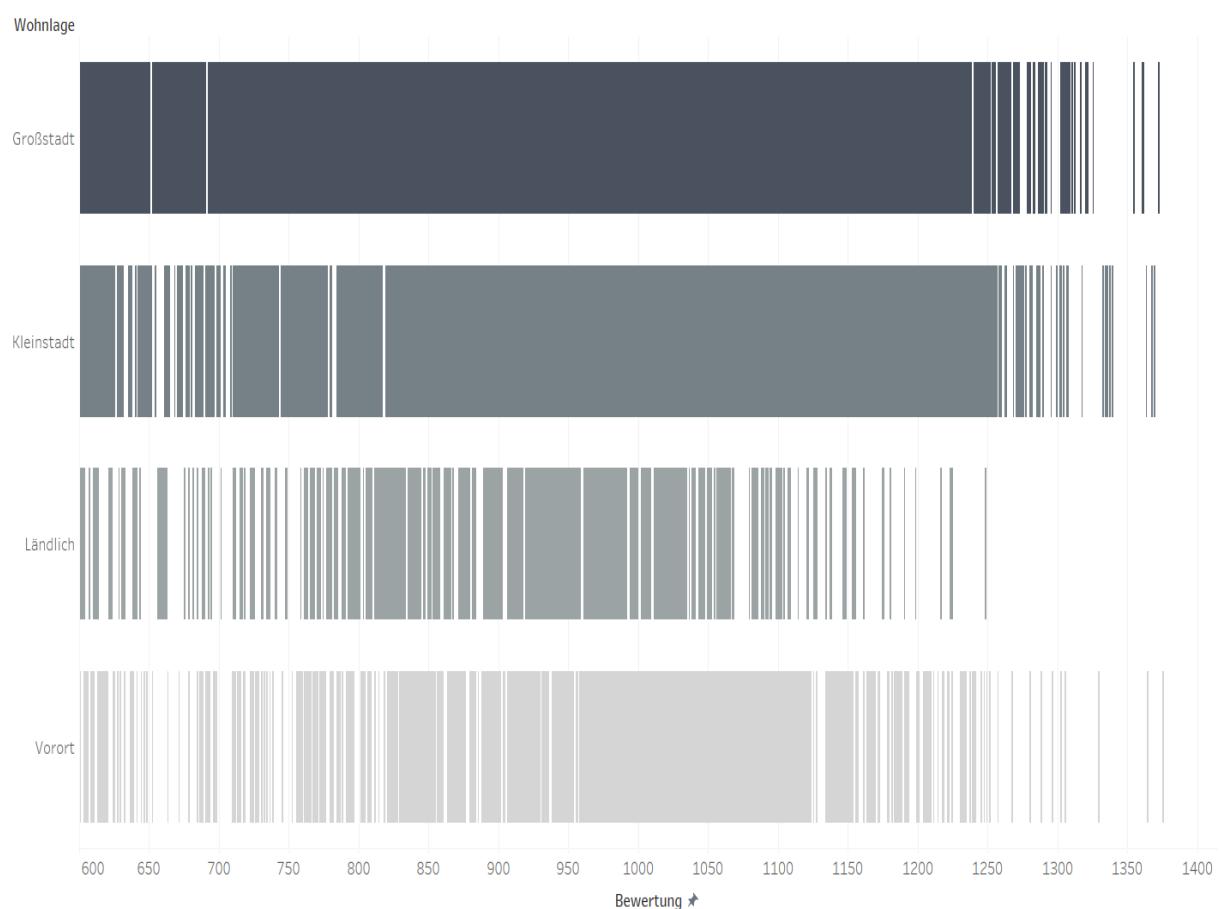


Abbildung D.12: Auswertung des Punktestand für die Wohnlage über alle Datenpunkte