Institut für Parallele und Verteilte Systeme

Universität Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Bachelorarbeit

Entwicklung eines interaktiven Systems für das FALCC Framework

Linus Zaiser

Studiengang: Informatik

Prüfer/in: Prof. Dr. rer. nat. Melanie Herschel

Betreuer/in: Nico Lässig, M.Sc.

Beginn am: 20. September 2023

Beendet am: 20. März 2024

Kurzfassung

In der heutigen digitalisierten Welt werden immer öfter KI-Modelle verwendet, um Entscheidungen zu treffen, von denen Menschen betroffen sind. Es ist erwiesen, dass künstliche Intelligenz gute Leistungen erbringen kann. Es ist allerdings wichtig, die Eignung von KI-Modellen nicht nur hinsichtlich ihrer Genauigkeit zu bewerten, sondern auch bezüglich ihrer Fairness. In diesem Aspekt gibt es große Bedenken. Es wurde gezeigt, dass Modelle häufig voreingenommen gegenüber unterschiedlichen gesellschaftlichen Gruppen sind. Dieses Problem wird aktuell von vielen Forschern untersucht und es wurden bereits einige Ansätze für Verbesserungen entwickelt. Einer dieser Ansätze ist das FALCC-Framework [Läs23] der Universität Stuttgart. Trotz der vielversprechenden Ergebnisse des FALCC-Algorithmus besteht eine Lücke in der Nutzbarkeit und Erklärbarkeit durch die Abwesenheit einer benutzerfreundlichen Schnittstelle. Diese Arbeit stellt FALCC-IS, ein interaktives System für das FALCC-Framework vor, das speziell dafür entwickelt wurde, den FALCC-Algorithmus zugänglicher zu machen. FALCC-IS bietet eine intuitive Benutzeroberfläche, die es Benutzern ermöglicht, das FALCC-Framework zu erforschen. Es verfügt über einige Analyse- und Evaluationswerkzeuge, die nicht nur das Aufdecken von Bias ermöglichen, sondern auch die Erklärbarkeit des FALCC-Algorithmus verbessern. Ein besonderes Merkmal ist die Möglichkeit, Datensätze zweidimensional darzustellen, wodurch Benutzer neue Vorhersagen treffen und kontrafaktische Untersuchungen anstellen können. Zudem können neue Metriken kreiert werden, welche sich aus mehreren bereits implementierten Metriken zusammensetzen.

Inhaltsverzeichnis

1 Einleitung **11**
1.1 Motivation interaktiver Systeme 12
1.2 Aufgabenstellung 14

2 Grundlagen **15**
2.1 Fairness 15
2.2 FALCC-Framework 18

3 Verwandte Arbeiten **23**
3.1 Interaktive KI-Werkzeuge 23
3.2 Übersicht von interaktiven Systemen 25

4 Interaktives System **31**
4.1 Anforderungen 31
4.2 Verwendete Technologien 32
4.3 Bedienablauf 32
4.4 Unterstützte Datensätze 32
4.5 Implementierte Werkzeuge 33

5 Demonstration und Ergebnisse **45**
5.1 Auswertung der Trainingsstrategien 45
5.2 Auswertung der Proxy-Minderung-Techniken 47
5.3 Weitere Anwendungsfälle 49

6 Zusammenfassung und Ausblick **51**
6.1 Ausblick 51

Literaturverzeichnis **53**

Abbildungsverzeichnis

1.1 Genauigkeit vs. Erklärbarkeit [GA19] 13

2.1 Schlechte lokale Fairness 18

4.1 Bedienablauf von FALCC-IS 33
4.2 Eingabeformular des interaktiven Systems 34
4.3 2D Darstellung des *Communities and Crime*-Datensatzes inklusive Cluster 37
4.4 2D Darstellung des *Communities and Crime*-Datensatzes inklusive korrekt vorhergesagte Punkte und sensitiven Gruppen 37
4.5 Attribut-Werte eines ausgewählten Punktes 38
4.6 Vorhergesagen der FALCC-Modelle und Submodelle 38
4.7 Linearkombination von Metriken 39
4.8 Balkendiagramm der FALCC-Modelle mit der AdaBoost- und Optimal-AdaBoost-Strategie 40
4.9 Balkendiagramm der AdaBoost-Modelle zur Demonstration der globalen Demographic Parity 41
4.10 Streudiagramm für die FALCC-Modelle 41
4.11 Boxplot für den Optimal-AdaBoost Run 42
4.12 Auswertung der Submodelle für Cluster 0 43

5.1 Genauigkeit und Fairness der FALCC-Modelle und Submodelle 46
5.2 Genauigkeit und Fairness der Fair-Input-Strategie 47
5.3 Ungenauigkeits-Fairness-Metrik der Fair-Input-Strategie 47
5.4 Darstellung der Cluster und der sensitiven Gruppen 49
5.5 Darstellung der sensitiven Gruppen 50
5.6 Vergleich der Proxy-Minderung-Techniken 50

Tabellenverzeichnis

2.1 Parameter-Tabelle 22

3.1 Übersicht verschiedener interaktiver Systeme und deren Eigenschaften 27

4.1 Verwendungszwecke der Werkzeuge 44

1 Einleitung

Durch die rasante Verbesserung der Genauigkeit von Künstlicher Intelligenz, werden immer öfter KI-Modelle verwendet, um Entscheidungen in verschiedensten Bereichen zu treffen. Für Bereiche bei denen Menschen von den Entscheidungen künstlicher Intelligenz betroffen sind, ist neben der Genauigkeit, auch die Fairness von Algorithmen von großer Bedeutung. Das ist eine zentrale Herausforderung in der modernen Forschung der Informatik. Die Wichtigkeit der Forschung im Bereich der fairen Algorithmen lässt sich durch zahlreiche Fallstudien verdeutlichen. In [LI16] wurde die Verwendung von KI-Systemen für die Vorhersage von zukünftigen Drogen-Straftätern kritisch hinterfragt. Es wurde festgestellt, dass Menschen mit dunkler Hautfarbe und geringem Einkommen mit höherer Wahrscheinlichkeit als verdächtig eingestuft werden. Eine weitere Fallstudie stellte die Benachteiligung von Frauen bei sogenannten Anzeigebereitstellungs-Algorithmen fest. Lamprecht und Tucker untersuchten einen Algorithmus, der dazu diente, ausgewählten Benutzern Werbung für MINT Berufe anzuzeigen [LT19]. Ihr Ziel bestand darin, ihre Werbung besonders geschlechtsneutral zu gestalten. Jedoch wurde festgestellt, dass der Algorithmus Frauen seltener Werbung anzeigte. Dieses Ungleichgewicht konnte nicht durch ein geringeres Interesse der Frauen erklärt werden, da Frauen sogar prozentual häufiger auf die Werbung klickten. Vielmehr resultiert die Diskriminierung daraus, dass der Algorithmus tatsächlich weniger Erfolg darin sah, Frauen diese Werbung anzuzeigen. Diese Fälle und noch viele weitere machen deutlich, dass Algorithmen nicht frei von Vorurteilen sind. Um unfaire Auswirkungen auf bestimmte Bevölkerungsgruppen zu vermeiden, ist eine kritische Betrachtung und Verbesserung dieser Algorithmen notwendig. Auf diesen Erkenntnissen baut das FALCC (Efficiently performing, locally fair and accurate classifications)-Framework [LH24] auf. Das Framework stellt unter anderem einen effizienten und fairen Algorithmus für binäre Klassifizierung vor, der sich zum Hauptziel gesetzt hat, die Fairness zu erhöhen. Hierbei werden mehrere verschiedene Fairnessdefinitionen und Metriken berücksichtigt, die genauer in Abschnitt 2 erläutert werden. Die Grundidee hinter dem FALCC-Algorithmus basiert auf einer fairen, dynamischen Modellauswahl. Das heißt, für jeden neuen Datenpunkt, der klassifiziert werden soll, wird neu entschieden, welches Modell am geeignetsten für diesen Punkt ist. Für die Bewertung des besten Modells wird nicht nur die Genauigkeit des Modells, sondern auch die Fairness des Modells berücksichtigt. Es konnte gezeigt werden, dass für viele Anwendungsfälle der Algorithmus schnelle und faire Ergebnisse liefert. Allerdings fehlt es dem FALCC-Framework noch an einer interaktiven Benutzeroberfläche. Das erschwert unerfahrenen Benutzern die Verwendung und die Analyse des FALCC-Algorithmus. Ziel dieser Arbeit ist die Entwicklung von FALCC-IS (interactive System), einer Benutzeroberfläche, die speziell auf das FALCC-Framework zugeschnitten ist.

1.1 Motivation interaktiver Systeme

Es gibt eine Reihe an Gründen, weshalb ein interaktives System nützlich für das FALCC-Framework ist. Dazu gehören unter anderem die Benutzerfreundlichkeit, die Erklärbarkeit, die Möglichkeit manuell eine gute Parameterkonfiguration zu finden und vor allem auch das Erkennen von Bias, also unfairen Modellen und Daten. Auf diese Vorteile wird im folgenden genauer eingegangen.

1.1.1 Benutzerfreundlichkeit

Der offensichtlichste Grund ist, dass die Benutzung eines visuellen Interface deutlich einfacher ist, als die Bedienung des Frameworks über Code oder Kommandozeile. Davon profitieren vor allem Benutzer ohne Programmiererfahrung. Es gibt zum Beispiel Eingabeformulare, die es dem Benutzer verbieten, unerlaubte Eingaben zu tätigen. Oder einen automatischen Modellspeicher, der dem Anwender das ständige neu trainieren seines Modells erspart. Es gibt sehr viele Möglichkeiten, die in FALCC-IS integriert werden können, um die Benutzung angenehmer, fehlertoleranter und intuitiver zu gestalten. Dadurch wird natürlich auch die Arbeitseffizienz gesteigert. Das FALCC-Framework hat eine große Komplexität und der Einarbeitungsaufwand in FALCC ist groß. Dieser wird durch FALCC-IS reduziert.

1.1.2 Erklärbarkeit

Das Problem, dass komplexe KI-Algorithmen für den Benutzer weitgehend wie eine Blackbox wirken, ist durchaus problematisch. Aktuell gestaltet sich die Nutzung des KI-Systems oft wie folgt: Der Benutzer gibt seine Eingabe ein, der Algorithmus verarbeitet diese und gibt schließlich das Ergebnis aus. Jedoch bleibt die Arbeitsweise des Algorithmus undurchsichtig. Der Benutzer hat keine Möglichkeit nachzuvollziehen, wie das Ergebnis zustande kam und was anders hätte sein müssen, damit man eine positive Entscheidung erhält und ob man dem Ergebnis überhaupt trauen kann. Eine geringe Erklärbarkeit des Modells bewirkt auch, dass möglicher Bias im Modell schwieriger erkannt werden kann. Besonders in Bereichen, in denen wichtige Entscheidungen getroffen werden sollen, wie beispielsweise bei Jobeinstellungen [Mil15], in denen heutzutage KI-Algorithmen eingesetzt werden, ist die Steigerung der Transparenz im Entscheidungsfindungsprozess von großer Bedeutung. Eine alternative Lösung besteht darin, auf nachvollziehbarere Algorithmen wie zum Beispiel Entscheidungsbäume zurückzugreifen. Allerdings konnte eine Studie feststellen, dass die Genauigkeit von KI-Algorithmen tendenziell steigt, wenn die Erklärbarkeit abnimmt [GA19]. Abbildung 1.1 zeigt, dass komplexere Trainingsstrategien wie beispielsweise neuronale Netzwerke eine hohe Genauigkeit aufweisen. Allerdings geht mit der erhöhten Genauigkeit auch eine geringere Erklärbarkeit einher. Die hohe Komplexität verhindert, dass nachvollzogen werden kann, warum das Modell bestimmte Entscheidungen trifft. Das bedeutet, dass Modelle, die für den Benutzer weniger verständlich oder offensichtlich sind, häufig besser funktionieren. Ein wichtiges Ziel von interaktiven Systemen, ist es also die Erklärbarkeit von Modellen zu verbessern, ohne dabei Verluste in der Performanz zu erhalten. Auch der FALCC-Algorithmus kann, abhängig von den individuell trainierten Modellen, eine beträchtliche Komplexität aufweisen. Benutzt man als Submodelle eine Trainingsstrategie, die mit neuronalen Netzen funktioniert, würde sich der FALCC-Algorithmus in der Grafik noch weiter links als der Punkt für neuronale Netze befinden. Denn die Komplexität des FALCC-Algorithmus geht über die Komplexität eines einzelnen Modells hinaus. Es werden

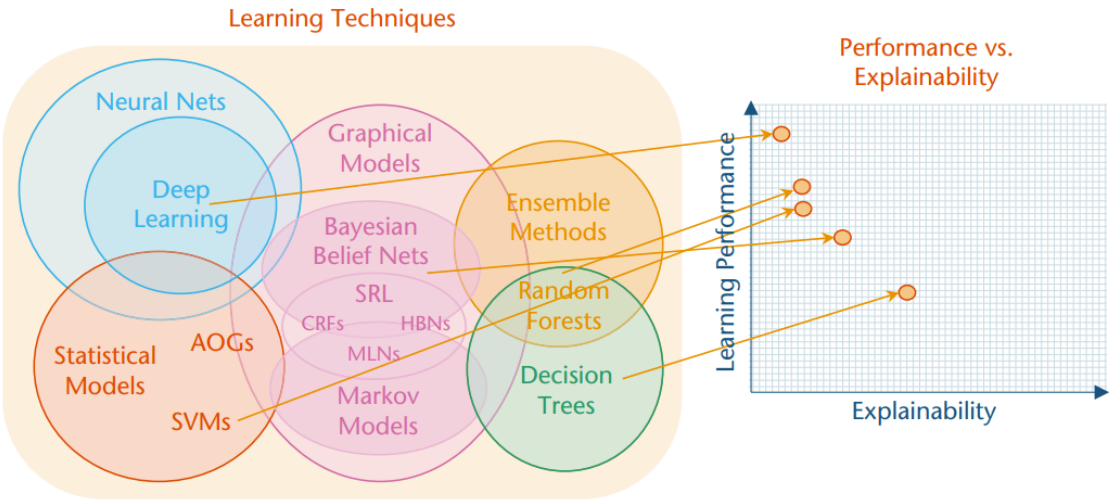


Abbildung 1.1: Genauigkeit vs. Erklärbarkeit [GA19]

lokale Regionen gebildet und verschiedene Klassifizierungsmodelle kommen zum Einsatz, was insgesamt die Komplexität erhöht. FALCC-IS ermöglicht Benutzern ein tieferes Verständnis und eine verbesserte Kontrolle über den FALCC-Algorithmus zu erreichen. Unter anderem können hypothetische Veränderungen in den Eingabedaten und deren Auswirkung auf die Vorhersage untersucht werden. Für Vorhersagen können somit die relevanten Attribute bestimmt werden. Auch die Bereitstellung von Modell spezifischen Provenance-Daten kann dabei helfen, ein System besser zu verstehen. Und es gibt es eine Reihe verschiedener Visualisierungsmethoden, die dabei helfen, die Erklärbarkeit zu verbessern [CMJK20].

1.1.3 Parameter Optimierung

Komplexere Modelle wie zum Beispiel beim FALCC-Algorithmus, haben oftmals eine große Anzahl an Hyperparametern. Je nach Konfiguration erzielen sie ganz unterschiedliche Ergebnisse. FALCC-IS ermöglicht das Analysieren, Vergleichen und Optimieren dieser Parameter. Es können Modelle trainiert werden und deren Performanz analysiert und schließlich Veränderungen an der Konfiguration gemacht werden. Das manuelle Optimieren der Parameter ist essenziell, um ein geeignetes Modell für einen bestimmten Anwendungsfall zu finden.

1.1.4 Unfairness erkennen

Trotz des Ziels des FALCC-Algorithmus, ein höheres Maß an Fairness zu erreichen, ist die Erzielung perfekter Fairness eine Herausforderung, die nicht garantiert werden kann. Einige Werkzeuge von FALCC-IS können dabei helfen, Bias zu erkennen und zu verringern. Dabei nutzt FALCC-IS visuelle Werkzeuge, die es den Benutzern ermöglicht, Unfairness sowohl im Modell, als auch in den zugrunde

liegenden Daten zu identifizieren. Zusätzlich zur Analyse von Fairness-Metriken ermöglicht FALCC-IS eine umfassende Evaluation der Modellgenauigkeit. Diese Funktionalität ist wichtig, da es oftmals einen Trade-Off zwischen der Fairness und der Genauigkeit gibt [LH24]. FALCC-IS erlaubt es, Zusammenhänge und Korrelationen zwischen Fairness- und Leistungsmetriken aufzudecken.

1.2 Aufgabenstellung

Die Aufgabe dieser Arbeit besteht darin, ein interaktives System für das FALCC-Framework zu entwickeln (FALCC-IS). Es soll ein visuelles Interface für die Eingabe von Parametern, Modell-Interaktionen und verschiedene Evaluations-Werkzeuge enthalten. Das Ziel ist es, die zuvor beschriebenen Vorteile eines interaktiven Systems für das FALCC-Framework zu realisieren.

Struktur

Im zweiten Kapitel werden alle Grundlagen für die Interaktion mit dem System erläutert. Dies beinhaltet die Grundlagen und den aktuellen Stand im Bereich der Fairness. Außerdem wird die Funktionsweise des FALCC-Algorithmus genauer betrachtet, um einige FALCC spezifischen Elemente des Systems besser nachvollziehen zu können. Im dritten Kapitel werden existierende interaktive Systeme für KI-Modelle behandelt. Es wird untersucht und gruppiert, welche Werkzeuge vorhanden sind und welchen Zweck diese verfolgen. Danach werden diese Erkenntnisse genutzt, um im vierten Kapitel das interaktive System für das FALCC-Framework vorzustellen. Es werden die im Rahmen dieser Arbeit implementierten Werkzeuge beschrieben und auch einige Implementierungs-Details erklärt. Das fünfte Kapitel illustriert die Funktionsweise und die Einsatzmöglichkeiten von FALCC-IS durch konkrete Beispiele. Es dient dazu, ein Gefühl zu entwickeln wie mit dem interaktiven System gearbeitet werden kann und um die Effektivität von FALCC-IS in realen Szenarien zu demonstrieren. Im abschließenden Kapitel wird ein Fazit gezogen, in dem nicht nur die erzielten Ergebnisse und Erkenntnisse zusammengefasst werden, sondern auch zukünftige Verbesserungen und Weiterentwicklungen des Systems vorgeschlagen werden.

2 Grundlagen

FALCC-IS greift einige grundlegende Konzepte im Bereich der Datenvisualisierung und der erklärbaren KI auf. Zudem visualisiert es einige spezifische Aspekte des FALCC-Frameworks. Die Hauptmotivation für den FALCC-Algorithmus besteht darin, die Fairness in Klassifizierungsproblemen zu verbessern. Damit der Benutzer des Systems versteht, welche Informationen angezeigt werden, ist es notwendig, einige Grundlagen im Bereich der Fairness kennenzulernen. In den kommenden Abschnitten des zweiten Kapitels werden unterschiedliche Fairness-Typen genauer definiert und anschließend die Funktionsweise des FALCC-Algorithmus grundlegend behandelt.

2.1 Fairness

Ein Algorithmus kann als fair betrachtet werden, wenn er keine Individuen und Gruppen aufgrund ihrer angeborenen und sozialisierten Eigenschaften benachteiligt. Eigenschaften die potenziell diskriminierende Vorurteile auslösen können, werden als **sensitive Attribute** bezeichnet [MMS+21]. Darunter fallen zum Beispiel das Geschlecht und die Religion. **Sensitive Gruppen** bestehen aus Personen, die aufgrund eines gemeinsamen sensitiven Attributes ein erhöhtes Risiko haben, benachteiligt zu werden. Die sensitiven Attribute bilden durch Kombination sensitive Gruppen. Zum Beispiel entstehen durch die Kombination der sensitiven Attribute Hautfarbe und Geschlecht (zur Vereinfachung jeweils nur zwei Ausprägungen) vier sensitive Gruppen. Diese sind schwarze Männer, weiße Männer, schwarze Frauen und weiße Frauen. Um Fairness in Klassifizierungsproblemen zu verbessern, ist es zunächst einmal notwendig, Fairness mathematisch zu definieren. Da Fairness eine komplexe und subjektive Eigenschaft ist, haben sich im Laufe der Jahre verschiedene Definitionen ergeben, die je nach Situationen unterschiedlich angemessen sein können. Im folgenden Unterabschnitt werden die übergeordneten Konzepte Gruppen-Fairness, individuelle Fairness und lokale Fairness betrachtet.

2.1.1 Globale Gruppen-Fairness

Die Gruppen-Fairness strebt an, dass unterschiedliche Gruppen gleich behandelt werden [LH24; MMS+21]. Keine der sensitiven Gruppen soll also ein erhöhtes Risiko haben, benachteiligt zu werden.

Es gibt verschiedene Metriken zur Bewertung von Gruppen-Fairness. In den folgenden Berechnungsvorschriften der Metriken für die Gruppen-Fairness bezeichnet G die sensitiven Gruppen. $\hat{Y}$ ist ein trainiertes Klassifizierungsmodell, wobei $\hat{Y} = 1$ eine positive Vorhersage bedeutet, während Y das tatsächliche, korrekte Label darstellt.

Demographic Parity

Das Konzept der Demographic Parity möchte sicherstellen, dass der Anteil von positiven Vorhersagen für alle Gruppen gleich ist. Mathematisch lässt sich das durch folgende Formel messen:

$$dp = \frac{1}{|G|} \cdot \sum_{j \in G} |P(\hat{Y} = 1|G = j) - P(\hat{Y} = 1)|$$

Wenn alle Gruppen die gleiche Wahrscheinlichkeit für eine positive Vorhersage haben, bedeutet dies, dass die Wahrscheinlichkeit für eine positive Vorhersage unabhängig von der Gruppenzugehörigkeit ist. Gemäß dem Gesetz der totalen Wahrscheinlichkeit wäre in diesem Fall der Wert der Demographic Parity 0, da keine Ungleichheit in den Vorhersagen zwischen den Gruppen besteht. Ein niedriger Wert bedeutet also, dass $\hat{Y}$ ein faires Klassifizerungsmodell in Bezug auf die Demographic Parity ist. Beispielsweise strebt Demographic Parity für den Werbungsverteilungs-Algorithmus an, dass möglichst gleiche Anteile von Frauen und Männer die Werbung bekommen.

Equal Opportunity

Equal Opportunity berücksichtigt die Korrektheit der Klassifizierung. Die Equal Opportunity wird somit nicht nur durch die Wahrscheinlichkeit positiver Vorhersagen beeinflusst, sondern durch die Wahrscheinlichkeit, dass die positiven Vorhersagen korrekt sind. Die Formel modifiziert sich damit zu:

$$eq_{op} = \frac{1}{|G|} \cdot \sum_{j \in G} |P(\hat{Y} = 1|G = j, y = 1) - P(\hat{Y} = 1|y = 1)|$$

Das bedeutet, dass Equal Opportunity klein ist, wenn die Wahrscheinlichkeit eine korrekte positive Klassifizierung zu erhalten, für beide Gruppen ähnlich ist . In unserem Beispiel würde das bedeuten, dass versucht wird, prozentual gleich vielen Frauen wie Männern, die tatsächlich an einer MINT Karriere interessiert sind, die Werbung anzuzeigen.

Equalized Odds

Equalized Odds berücksichtigt nicht nur die Wahrscheinlichkeit einer korrekten positiven Vorhersage, sondern auch die Wahrscheinlichkeit einer falschen positiven Vorhersage.

$$eq_{od} = \frac{1}{2} \cdot \sum_{k=0}^{1} \cdot (\frac{1}{|G|} \cdot \sum_{j \in G} |P(\hat{Y} = 1|G = j, y = k) - P(\hat{Y} = 1|y = k)|)$$

Es soll also ähnliches wie bei Equal Opportunity erreicht werden. Zusätzlich soll jedoch auch sichergestellt werden, dass die Wahrscheinlichkeit einer positiven Vorhersage für Punkte mit einem tatsächlich negativen Label in jeder Gruppe gleich groß ist. In unserem Beispiel strebt man also danach, dass prozentual gleich viele Frauen wie Männer, die tatsächlich an einer MINT Karriere interessiert sind, die Werbung sehen. Zusätzlich sollte berücksichtigt werden, dass prozentual genauso viele Frauen wie Männer, die eigentlich nicht an einer MINT Karriere interessiert sind, die Werbung trotzdem erhalten.

Treatment Equality

Die Treatment Equality [BHJ+21] untersucht das Verhältnis zwischen falschen negativen Vorhersagen (FN) und falschen positiven Vorhersagen (FP). Wenn das Verhältnis zwischen FN und FP gleich für alle sensitiven Gruppen ist, ist die Treatment Equality erfüllt. Definieren lässt sie sich also durch folgende Formel:

$$tr_{eq} = \frac{1}{|G|} \cdot \sum_{j \in G} \left| \frac{FP_{G=j}}{FP_{G=j} + FN_{G=j}} - \frac{FP_{total}}{FP_{total} + FN_{total}} \right|$$

Im Kontext des Werbungsverteilungs-Algorithmus zielt die Treatment Equality darauf ab, dass das Verhältnis zwischen der Anzahl von Personen, die Werbung für MINT-Berufe angezeigt bekommen, obwohl sie eigentlich nicht interessiert sind (FP) und der Anzahl an Personen die an MINT-Berufen interessiert wären, aber die Werbung nicht erhalten (FN), für beide Geschlechter gleich sein sollte.

2.1.2 Individuelle Fairness

Eine andere Fairness-Klasse ist die individuelle Fairness. Individuelle Fairness zielt darauf ab, Personen die sich ähnlich sind auch ähnlich zu behandeln [PS22]. Dies erfordert einen Algorithmus, der zwei Tupel aufnimmt und quantifiziert wie ähnlich sich diese sind. Die Implementierung eines solchen Algorithmus ist komplex und erfordert besondere Überlegungen im Bereich der Datenvorverarbeitung. Eine beliebte Metrik für individuelle Fairness ist die Consistency [ZWS+13]. Sie wird für einen gesamtem Datensatz bestimmt, indem die Vorhersage für jedes Tupel mit den Vorhersagen seiner k nächsten Nachbarn verglichen wird. Dabei wird geprüft, ob die Vorhersagen für ähnliche Individuen konsistent sind. In der folgenden Formel, bezeichnet k die Anzahl der betrachteten nächsten Punkte und N die Gesamtzahl der Datenpunkte.

$$\text{Consistency} = 1 - \frac{1}{N \cdot k} \sum_{n} \left| \hat{Y}_n - \frac{1}{k} \sum_{j \in kNN(x_n)} \hat{Y}_j \right|$$

Es gibt neben der Consistency Metrik noch weitere Ansätze, um zu erreichen, dass ein Algorithmus individuell faire Klassifizierungen durchführt, die hier nicht behandelt werden [MMS+21; PS22]. Die Schwachstelle aller Ansätze liegt aber in den „Proxy-Attributen“. Proxy-Attribute sind Merkmale, die zwar nicht selbst als sensitives Attribut gelten, die aber stark mit den sensitiven Attributen korrelieren. Ein klassisches Beispiel für ein Proxy-Attribut ist die Körpergröße, die mit dem Geschlecht korrelieren kann. Das FALCC-Framework zielt unter anderem auf die Verbesserung der individuellen Fairness. Das Framework implementiert zwei verschiedene Techniken, um den Einfluss von Proxy Attributen zu verringern, welche in Abschnitt 2.2.1 beschrieben werden.

2.1.3 Lokale Fairness

Die letzte hier behandelte Fairness-Klasse ist die lokale Fairness [LOH22]. Die lokale Fairness kombiniert die Ideen der Gruppen-Fairness mit denen der individuellen Fairness. Genauer gesagt wird eine Metrik der Gruppen-Fairness nicht auf den gesamten Datensatz, sondern auf Teilgruppen angewandt. Die Teilgruppen enthalten dabei Punkte, die sich insgesamt ähnlich sind. Dabei wird

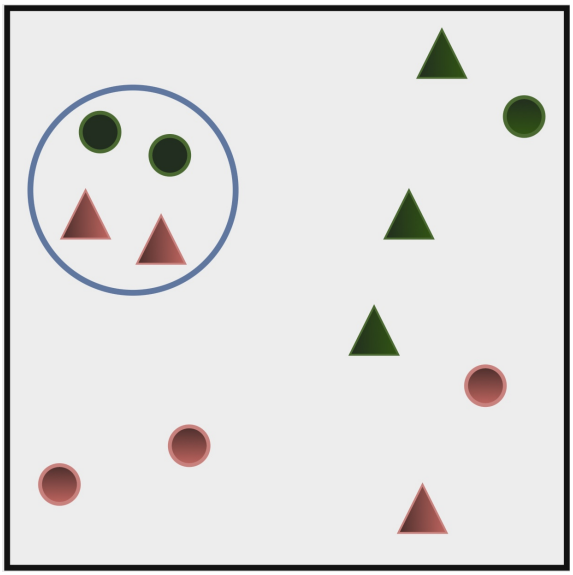


Abbildung 2.1: Schlechte lokale Fairness

die Definition der Gruppen-Fairness in den einzelnen Teilgruppen berücksichtigt. Die Motivation hinter der lokalen Fairness ist, dass selbst wenn die globale Gruppen-Fairness gewährleistet ist, es keine Garantie gibt, dass die Gruppen-Fairness auch lokal wirkt. Diese Beobachtung wird in Abbildung 2.1 verdeutlicht. Diese Graphik basiert nicht auf realen Daten, vielmehr soll sie nur die Relevanz der lokalen Fairness verdeutlichen. Die Farben der Punkte repräsentieren das vorhergesagte Label. Rot ist ein negatives und grün ein positives Label. Zusätzlich zeigt ein Dreieck an, ob eine Person weiblich ist. Ein Kreis bedeutet, dass die Person männlich ist. Wenn wir nun eine globale Gruppen-Fairness-Metrik, zum Beispiel Demographic Parity anwenden, sehen wir, dass drei von sechs Männern eine positive Vorhersage erhalten, und ebenfalls drei von sechs Frauen. Dieses Ergebnis erfüllt damit perfekt die Anforderungen an die Demographic Parity. Unfairer wird das Ergebnis, wenn man die lokale Fairness betrachtet, zum Beispiel die Demographic Parity in der durch den blauen Kreis hervorgehobenen Teilgruppe. Die Personen in dieser Teilgruppe sind sich alle sehr ähnlich, was durch ihren geringen Abstand visualisiert wird. Hier erhalten alle Männer eine positive Vorhersage, jedoch keine Frau, so dass die lokale Fairness hier fehlt. Daher bedeutet eine gute globale Gruppen-Fairness nicht unbedingt auch eine gute lokale Fairness.

2.2 FALCC-Framework

Das FALCC-Framework [Läs23; LH24; LOH22] bietet einige Werkzeuge zur Untersuchung und Verbesserung von Fairness in binären Klassifikationsproblemen. Unter anderem implementiert es den FALCC-Algorithmus, dessen Ziel darin besteht, die Fairness zu optimieren. Der Hauptfokus liegt hierbei bei der lokalen Fairness, aber auch die globale Fairness und die individuelle Fairness werden berücksichtigt. Außerdem, soll es durch die Verbesserung der Fairness keine Verschlechterung der Genauigkeit geben. Die grundlegende Idee des FALCC-Algorithmus basiert auf einer dynamischen Modellauswahl. Hierbei wird der Datensatz in verschiedene lokale Regionen unterteilt. Wenn ein neues Tupel klassifiziert werden soll, verwendet der Algorithmus ein Modell, das besonders

gut in der zugehörigen Region abgeschnitten hat. Der FALCC-Algorithmus ist, ähnlich wie die meisten KI-Algorithmen, in zwei Phasen unterteilt: Die Offline-Phase und die Online-Phase. Die Offline-Phase, welche nur einmalig durchgeführt wird, berechnet alle benötigten Informationen und Modelle. Wohingegen die Online-Phase jedes Mal ausgeführt wird, wenn ein neues Tupel klassifiziert werden soll. Insbesondere für die Online-Phase ist die Laufzeit von entscheidender Bedeutung.

2.2.1 Offline-Phase

Während der Offline-Phase werden verschiedene Modelle trainiert und anschließend in lokalen Regionen ausgewertet. Die Modell-Kombinationen, die in jeder lokalen Region die besten Ergebnisse erzielen, werden beibehalten. Die Offline-Phase ist unterteilt in das Modelltraining, eine Phase zur Verminderung der Diskriminierung durch *Proxy-Attribute*, dem Clustering-Schritt und die Modellbewertung.

Modelltraining

Das Modelltraining dient dazu, eine bestimmte Anzahl von Modellen mit dem Trainingsdatensatz zu trainieren. Der Input des Algorithmus ist also der Trainingsdatensatz mit den sensitiven Attributen G. Zunächst werden verschiedene Modelle trainiert, diese seien $m_1, ..., m_n$. Diese Modelle werden fortan als Submodelle bezeichnet. Welche Trainingstrategie dabei angewendet wird, spielt keine Rolle für die Funktionalität, jedoch für die Qualität. Das FALCC-Framework implementiert eine angepasste Form der *AdaBoost-Strategie* [LH24]. Während jeder Iteration in der Trainingsphase werden nicht nur die Trainingsdaten angepasst, sondern auch das Basismodell verändert. Es kommen also verschiedene Modelltypen zum Einsatz. Zudem umfasst das FALCC-Framework eine optimierte version der *AdaBoost* und *Random-Forest*, die durch Hyperparameter-Optimierung das diverseste Modell-Ensemble findet [Nie23]. Ein diverses Modell-Ensemble unterscheidet sich in den Vorhersagen häufiger gegenüber nicht diversen Modellen. Außerdem bietet das FALCC-Framework eine *Single-Classifiers-Strategie*, die unterschiedliche Standardtypen von Modellen individuell trainiert. Die letzte verfügbare Strategie ist die *Fair-Input-Strategie*. Diese Strategie trainiert bereits faire Klassifizierungsmodelle als Submodelle. Hierfür werden der LFR-Algorithmus[ZWS+13], Fair-SMOTE-Algorithmus [CMM21] und FAX-Algorithmus [GPM22] verwendet. In Kapitel 5.1 werden alle verfügbaren Trainingstrategien mithilfe von FALCC-IS ausgewertet.

Das Ergebnis des Modelltrainings ist eine Liste MC_{cand}, wobei ein Eintrag in der Liste als Modellkombination bezeichnet wird. Eine Modellkombination ist eine Menge an Paaren, die jeder sensitiven Gruppe ein Modell zuordnen. Einfacher gesagt, enthält ein Eintrag für jede sensitive Gruppe ein Modell, das auf diese sensitive Gruppe trainiert wurde. Als Beispiel haben wir die sensitiven Gruppen g_d und g_f, sowie drei Modelle m_1, m_2, m_3. Die Ergebnisliste lautet dann $MC_{\text{cand}} = \{\{(m_1, g_d), (m_1, g_f)\}, \{(m_1, g_d), (m_2, g_f)\}, \{(m_2, g_d), (m_1, g_f)\}, ..., \{(m_3, g_d), (m_3, g_f)\}\}$. Man erhält also mögliche Modellkombinationen, um ein neues Tupel zu klassifizieren. Das Ziel der nächsten Schritte ist es, diese Liste auf die besten Modellkombinationen für jede lokale Region zu reduzieren.

Minderung der Diskriminierung durch Proxy-Attribute

In diesem Schritt des FALCC-Frameworks liegt der Fokus darauf, das in Abschnitt 2.1.3 beschriebene Problem der Proxy-Attribute zu verbessern. Hierzu wird zunächst eine Messung der Korrelation zwischen sensitiven und nicht sensitiven Attributen vorgenommen, wobei die Pearson-Korrelation [BCHC09] als Messmethode verwendet wird. Es existieren zwei implementierte Strategien, um mit einer großen Korrelation umzugehen. Die erste Strategie gewichtet die Attributwerte in Abhänigkeit ihrer Pearson Korrelation neu. Sie wird im Framework „Reweigh“ genannt. Die zweite Strategie, „Remove“ zielt darauf ab, Attribute mit einer zu hohen Pearson-Korrelation vollständig zu entfernen. Dieser Schritt ist zwar für die Funktionalität nicht zwingend erforderlich, es hat sich aber herausgestellt, dass er zu Verbesserungen führen kann. Diese Techniken zur Minderung der Diskriminierung durch Proxy-Attribute finden nach der Phase des Modelltrainings statt. Somit liegt die Relevanz nicht darin, dass das Training der Submodelle weniger von den Proxy Attributen beeinflusst wird, sondern darin, die Auswirkungen der Proxy-Attribute auf die nächste Phase, also auf die Clusterbildung, zu minimieren. Die Auswirkung dieses Schrittes wird in Kapitel 5.2 demonstriert.

Clustering

In dieser Phase erfolgt die eigentliche Bildung der lokalen Regionen. Der durch die vorherige Phase veränderte Evaluationsdatensatz wird einem Cluster-Algorithmus übergeben. Das Ergebnis ist ein Datensatz mit lokalen Regionen, die für die Berechnung der lokalen Fairness notwendig sind. Hierbei wird angestrebt, dass die lokalen Regionen unabhängig von den sensitiven Attributen gebildet werden. Für die Clusterbildung wird der k-means-Algorithmus verwendet. Die Anzahl der lokalen Regionen kann dabei entweder frei festgelegt oder durch einen Hyperparameter-Optimierungsalgorithmus wie zum Beispiel den LOGMeans- [FBS20] oder den Elbow-Algorithmus [Tho53] optimiert werden. Die Anzahl der Cluster steht stark in Korrelation mit der lokalen Fairness. Je mehr Cluster gebildet werden, desto lokaler werden die betrachteten Gebiete und desto ähnlicher wird die lokale Fairness zur einer individuellen Fairness-Metrik. Falls nur ein einziges Cluster gebildet wird, wird vom Algorithmus nur die globale Gruppen-Fairness berücksichtigt.

Modellbewertung

In der Modellbewertungs-Phase werden die berechneten Informationen aus den drei vorherigen Schritten verwendet. Ziel ist es, für jedes der im zweiten Schritt erstellten Cluster, die optimale Modellkombination zu finden, die im ersten Schritt erstellt wurde. Dafür werden für jedes Cluster alle Modellkombinationen ausgewertet, wobei nur die beste beibehalten wird. Im Gegensatz zu vielen anderen Algorithmen berücksichtigt der FALCC-Algorithmus für die Evaluation nicht nur die Genauigkeit, sondern auch die Fairness. Der Algorithmus versucht, folgende Funktion zu minimieren:

$$\hat{L} = \lambda \cdot \text{Ungenauigkeit} + (1 - \lambda) \cdot \text{Unfairness} \quad | \quad \lambda \in [0, 1] \tag{2.1}$$

Der Algorithmus versucht also, die Ungenauigkeit und die Unfairness abhängig von einem Gewichtungsparameter λ zu verringern. Die Ungenauigkeit kann mit folgender Formel berechnet werden:

$$\text{Ungenauigkeit} = \frac{1}{|D|} \cdot \sum_{i \in D} |y_i - z_i|$$

Dabei bezeichnet $|D|$ die Anzahl der Tupel, y_i ist der korrekte Wert und z_i der vom Modell vorhergesagte Wert. Die Unfairness kann durch eine der in Abschnitt 2.1.1 erwähnten Metriken bestimmt werden. λ ist ein Parameter, der die Gewichtung zwischen Genauigkeit und Fairness bestimmt. Je niedriger λ ist, desto mehr Gewicht liegt auf der Unfairness. Das heißt, dass die Minimierung der Funktion dann erreicht wird, wenn die ausgewählten Modelle fair sind. Umgekehrt bedeutet ein großes λ, dass die Genauigkeit entscheidender für die Minimierung der Funktion ist.

Das Ergebnis dieses letzten Schritts der Offline-Phase ist eine Datenstruktur, die jedem erstellten Cluster die Modellkombination zuweist, welche die Formel 2.1 minimiert.

2.2.2 Online-Phase

In der Online-Phase wird das Label für ein bislang noch nicht gesehenes Tupel vorhergesagt. Dazu werden nicht nur die in der Online-Phase erstellten Modellkombinationen verwendet, sondern auch die Cluster. Auch diese Phase ist in mehrere Schritte aufgeteilt. Im ersten Schritt wird auf das neue Tupel die gleiche Technik zur Minderung der Diskriminierung durch Proxy-Attribut angewendet wie im zweiten Schritt der Offline-Phase. Dadurch soll die Auswirkung der Proxy Attribute auf die Cluster Zuordnung verringert werden. Der nächste Schritt ist die Cluster Zuordnung, bei der das neue Tupel einer lokalen Region zugeordnet wird. Das zugeordnete Cluster enthält Punkte, die dem neuen Punkt am ähnlichsten sind. Nun wird die Modellkombination aus der Datenstruktur geholt, die für das erhaltene Cluster am besten abgeschnitten hat. Schließlich wird aus der Modellkombination das Modell für die Vorhersage benutzt, das für die sensitive Gruppe geeignet ist.

2.2.3 Parameter

Wie bereits deutlich wurde, gibt es viele verschiedene, den Ablauf des FALCC-Algorithmus, beeinflussende Parameter. Eine passende Parameterkonfiguration ist entscheidend, um genaue und faire Ergebnisse zu erzielen. Ein wichtiger Verwendungszweck von FALCC-IS liegt darin, manuell eine geeignete Parameterkonfiguration für den Anwendungsfall des Benutzers zu finden. In der Tabelle 2.1 werden alle Parameter zusammengefasst, die eine Auswirkung auf die Qualität des Ergebnisses haben.

Parameter	Bedeutung	Domain
sensitive Attribute	Liste von Sensitiven Attributen	-
Trainingsstrategie	verwendete Trainingstrategie	Opt-AdaBoost, Opt-Random-Forest, AdaBoost, Single-Classifier, Fair-Input
Combination Amount	Anzahl an trainierten Submodellen	
Testsize	Anteil des Testdatensatzes	[0, 1]
Predsize	Anteil des Vorhersagedatensatzes	[0, 1]
Sbt	Option die Modelle seperat für alle sensitiven Gruppen zu trainieren	True, False
Ignore Sensitive	Option, dass das sensitive Attribut nichts zum Modell Training beiträgt	True, False
Proxy	Verwendete Proxy Minderung Strategie	reweigh, remove, no
Allowed	Liste von Attributen die nicht von der Proxy Minderungs Strategie verändert werden sollen	-
Metric	Verwendete Gerechtigkeits Metrik	Demographic parity, Equalized odds, Equal opportunity, Treatment equality
Weight	Gewichtungs Parameter λ	[0 , 1]
Cluster Algorithmus	Verwendeter Algorithmus zum Ermitteln der Cluster Anzahl	LOGmeans, elbow
ccr	Manuelle Anzahl der Cluster	[min, max]

Tabelle 2.1: Performanz beeinflussende Parameter

3 Verwandte Arbeiten

Es gibt bereits eine Reihe von interaktiven Systemen für KI-Modelle. Im Folgenden wird analysiert, welche Werkzeuge sie implementieren und ob diese für das interaktive System des FALCC-Frameworks nützlich sind. Dafür werden die Werkzeuge in vier verschiedene Gruppen gegliedert. Zum Schluss des Kapitels wird noch auf zwei Frameworks mehr im Detail eingegangen. Diese sind vor allem wegen ihrer vielen Analysemöglichkeiten mit Schwerpunkt auf die Fairness von Modellen interessant.

3.1 Interaktive KI-Werkzeuge

Die in den betrachteten Systemen integrierten Werkzeuge können in vier Hauptgruppen wie folgt unterteilt werden: Es gibt Werkzeuge, (1) für die Datensatz-Interaktion, (2) für die Konfigurations-Auswahl, (3) für die Modell-Interaktion und (4) für die Evaluation der Ergebnisse.

3.1.1 Datensatz-Interaktion

Heutige Modelle werden häufig auf sehr großen Datensätzen trainiert. Daher sind effiziente Analyse-Werkzeuge für die Analyse der Daten wichtig. Interaktive Systeme bieten verschiedene Optionen zur Visualisierung des Datensatzes. Das kann in tabellarischer Form [ZWLC19] oder durch ein Streudiagramm [WPB+19] erfolgen. Bei großen Datensätzen stößt eine Tabelle jedoch schnell an ihre Grenzen. In grafischer Form lassen sich Muster und Verteilungen besser erkennen. Für die Darstellung durch ein Streudiagramm muss der multidimensionale Datensatz auf einen zwei- oder drei-dimensionalen Graphen reduziert werden. Eine einfache Möglichkeit ist die manuelle Auswahl eines Attributes für jede Achse. Zum Beispiel kann man auf der x-Achse das Alter und auf der y-Achse das Gehalt anzeigen lassen. Diese Darstellung ignoriert jedoch alle anderen Attribute. Die Alternative sind sogenannte Dimensionsreduktions-Algorithmen [ASS21], deren Ziel es ist, die Dimension des Datensatzes mit möglichst geringem Informationsverlust zu reduzieren. Dadurch ist es möglich, ein Streudiagramm aus einem multidimensionalen Datensatz zu erstellen, wobei zwei nahe beieinander liegende Punkte darauf hindeuten, dass sie sich ähnlich sind. Eine geeignete Form der Datenvisualisierung und die Visualisierung der Klassifizierungsergebnisse helfen dabei, wichtige Strukturen, Trends und Ausreißer im Modell zu finden [DK18]. Auch weitere beschreibende Statistikfunktionen, wie zum Beispiel das arithmetische Mittel, können genutzt werden, um den Datensatz zu verstehen. Eine wichtige Funktion ist es, den Datensatz gruppieren zu können. Zum Beispiel will man sich anzeigen lassen, wie hoch das Durchschnittseinkommen ist, wenn man nur Frauen berücksichtigt. Solche Untersuchungen ermöglichen es, den Datensatz und das Modell auf Fairness zu überprüfen, indem die Statistiken für alle sensitiven Gruppen miteinander verglichen werden. Dementsprechend helfen solche Abfragen dabei, die Daten und sein Modell besser im Bereich der Fairness zu analysieren und sind somit wichtig für FALCC-IS.

3.1.2 Konfigurations-Auswahl

Die Konfigurations-Auswahl bietet die Möglichkeit, verschiedene Modelle basierend auf ihren Parametern auszuwählen. Diese Funktion ist vor allem dann wichtig, wenn es verschiedene Möglichkeiten gibt, das Modell zu konfigurieren und noch unklar ist, was die beste Konfiguration für den aktuellen Anwendungsfall ist. Das Ziel dieses Schrittes ist die Erstellung eines visuellen Eingabeformulars, welches bei der Eingabe von Parameterkonfigurationen hilft. Dieser Schritt ist entscheidend, da er das Optimieren von Parametern erleichtert und somit die Suche nach der richtigen Parameterkonfiguration für einen bestimmten Datensatz unterstützt. Für das FALCC-Framework gibt es, wie in Tabelle 2.1 zu sehen ist, eine Vielzahl verschiedener Parameter. Daher ist es wichtig, ein einfaches Eingabeformular bereitzustellen, um schrittweise die Möglichkeit zu haben, eine optimale Konfiguration zu finden. Das FALCC-Framework erfordert also ein sehr spezielles Eingabeformular, weshalb es eventuell inkompatibel mit bereits existierenden Systemen ist. Besonders wichtig ist die Möglichkeit, nicht nur ein Modell auszuwählen, sondern viele mit jeweils unterschiedlichen Konfigurationen. Somit können gleichzeitig mehrere Modelle miteinander verglichen werden, wodurch das Optimieren der Parameter erleichtert wird. Eine spezifische Möglichkeit geht in die Richtung des interaktiven Ensemble-Modell-Auswahlverfahrens, wie es in [TLKT09] beschrieben wird. Dort wird dem Benutzer ermöglicht, interaktiv mehrere Klassifikationsmodelle zu kombinieren, um einen Ensemble-Klassifikator zu erhalten. Diese Grundidee könnte in leicht abgewandelter Form auch für FALCC-IS relevant sein. Es wäre sinnvoll, wenn der Benutzer die Möglichkeit hätte, selbst zu wählen, welche Modelle als Grundmodelle in das FALCC-Framework einfließen sollen.

3.1.3 Modell-Interaktionen

Modell-Interaktionen sind interaktive Untersuchungen und Anpassungen von Modellentscheidungen. Diese Werkzeuge sind für das Erkennen von Unfairness und für die Erklärbarkeit unerlässlich. Dabei wird die Erklärbarkeit in globale und lokale Erklärbarkeit unterteilt [DK17]. Die globale Erklärbarkeit versucht, das Modell auf einer globalen Ebene zu erklären. Es sollen Strukturen und Muster vom Modell transparent werden. Im Gegensatz dazu spricht man von lokaler Erklärbarkeit, falls die Gründe einer einzigen bestimmten Vorhersage deutlich sind. Werkzeuge, die zur Verbesserung der Erklärbarkeit genutzt werden können, lassen sich also in folgende Gruppen unterteilen:

- **Werkzeuge zur Verbesserung der globalen Fairness:** Diese Werkzeuge dienen dazu, die Struktur und Muster des Modells klarzumachen. Darunter fällt, das Bereitstellten modellspezifischer Informationen. Beispielsweise kann durch [VV11] der ganze Entscheidungsbaum angezeigt werden, was das Modell auf einer globalen Ebene erklärt. Auch eine zweidimensionale Darstellung des Datensatzes inklusiver erhaltener Vorhersagen verdeutlichen Muster im Modell und dient zu einer Verbesserung der globalen Fairness.
- **Werkzeuge zur Verbesserung der lokalen Fairness:** Diese Werkzeuge ermöglichen eine Untersuchung der Gründe für eine bestimmte Vorhersage. Dazu gehören What-if Analyse [WPB+19], die untersuchen, wie sich Änderungen in den Attributwerten auf die Vorhersage auswirken. Auch kontrafaktische Erklärungen fallen in diesen Bereich. Kontrafaktische Erklärungen zeigen an, welcher Punkt am ähnlichsten zum betrachteten Punkt war, aber eine andere Vorhersage erhalten hat. Dadurch wird bewusst, welche Werte verändert werden müssen, um eine andere Entscheidung zu erhalten. Weitere Werkzeuge beschäftigen sich mit

einer Untersuchung, wie relevant Attribute für eine Entscheidung sind. Partielle Abhängigkeitsdiagramme zeigen zum Beispiel auf der x-Achse das zu untersuchende Attribut und auf der y-Achse die Wahrscheinlichkeit, dass die Vorhersage positiv ist. Dadurch sieht man, wie stark die Auswirkung von dem Attribut auf die Vorhersage ist.

Außerdem können durch die Verwendung solcher Analysewerkzeuge unfaire Entscheidungen identifiziert werden. Wenn beispielsweise eine hypothetische Änderung des Geschlechts zu einer anderen Vorhersage führt, könnte dies ein Anzeichen dafür sein, dass der Algorithmus unfair ist. Auch das Bereitstellen von Provenance-Daten erhöhen die Transparenz eines Modells. Beispielsweise können im FALCC-Algorithmus die gebildeten Cluster visualisiert werden, wodurch der Benutzer weiß, welches Modell für welche Vorhersagen verwendet wird.

3.1.4 Evaluation der Ergebnisse

Werkzeuge zur Evaluation der Ergebnisse ermöglichen es, die Leistung der ausgewählten Modelle kritisch zu hinterfragen. Diese Gruppe von Werkzeugen, zusammen mit der Auswahl der Konfiguration, ist essenziell für das Optimieren der Modellkonfigurationen. Entscheidend ist dabei, dass eine umfangreiche Sammlung an Metriken angeboten wird. Für FALCC-IS sind sowohl Leistungsmetriken als auch Fairness-Metriken relevant, um eine vollständige Performanzanalyse durchzuführen. Die FairLearn-Bibliothek [BDE+20] bietet unter anderem die Möglichkeit, dass die Benutzer neue Metriken definieren können.

Die im interaktiven System zu Verfügung stehenden Metriken, können dann durch diverse Darstellungsformen betrachtet werden, beispielsweise durch Balkendiagramme oder Streudiagramme. Streudiagramme, bei denen auf den Achsen jeweils eine Metrik ausgewählt werden kann, sind wichtig um einen Trade-Off zwischen mehreren Metriken darzustellen. Die sogenannte “Konfusionsmatrix“ wird ebenfalls in vielen Systemen bereitgestellt, da sie ein wichtiger Bestandteil ist, um die Leistung eines Modells zu messen [BDH+18; KLTH10; WPB+19]. Die Konfusionsmatrix zeigt jeweils an, wie viele Punkte richtig positiv, richtig negativ, falsch positiv und falsch negativ vorhergesagt wurden. Wenn die Konfusionsmatrix separat für jede sensitive Gruppe bereitgestellt wird, kann sie auch zur Fairness-Untersuchung beitragen, beispielsweise durch die Treatment Equality Metrik (siehe Kapitel 2.1.1).

Auch interaktive Evaluationsmethoden können verwendet werden. In [KLTH10] kann der Benutzer eine Konfusionsmatrix nach seinen Vorstellungen definieren. Das interaktive System rekalkuliert das Modell dabei so, dass die angegeben Wunschwerte möglichst optimal erzielt werden.

3.2 Übersicht von interaktiven Systemen

Tabelle 3.1 bietet eine Übersicht über verschiedene interaktive Systeme. Die Systeme werden auf die zuvor beschriebenen Einsatzgebiete analysiert. Dabei werden sie in folgenden Kategorien bewertet.

- Datensatz-Interaktion: Bietet das System eine Visualisierung des Datensatzes oder weiterer Kenngrößen im Bereich der deskriptiven Statistik ?
- Modell-Typ: Werden Klassifizierungs-Modelle oder Regressions-Modelle erlaubt ?

- Konfigurations-Auswahl: Wie viele verschiedene Konfigurationen können gleichzeitig betrachtet werden oder gibt es ein visuelles Eingabe-Formular ?
- Modell-Interaktion: Kann das System genutzt werden um die globale oder lokale Erklärbarkeit zu verbessern ?
- Unterstützte Metriken: Werden Fairness- und Leistungs-Metriken bereitgestellt oder bietet das System ein Werkzeug für benutzerdefinierte Metriken an ?
- Verwendungszweck: Was ist das Ziel des interaktiven Systems ?

Die grün markierte Spalte am Ende der Tabelle repräsentiert das in dieser Arbeit entwickelte FALCC-IS. Dieses bietet eine Kombination von Werkzeugen aus allen Gruppen an. Diese Werkzeuge werden in Kapitel 4.5 vorgestellt. Allerdings ist FALCC-IS auf das FALCC-Framework spezialisiert und kann nicht für andere Klassifizierungsmodelle verwendet werden.

	WIT [WPB+19]	FairLearn [BDE+20]	FairPilot [DNAA23]	AIF 360 [BDH+18]	Gamut [HHC+19]	BaobabView [VV11]	iForest [ZWLC19]	FALCC-IS
Datensatz-Visualisierung	Ja*	Nein	Nein	Nein	Ja	Nein	Ja	Ja
Deskriptive Statistik	Ja*	Ja	Nein	Nein	Ja	Ja	Ja	Nein
Klassifikation	Ja	Ja	Ja	Ja	GAM	Ja	Ja	Ja
Regression	Ja	Ja	Nein	Nein	GAM	Nein	Nein	Nein
Anzahl	2	Beliebig	Beliebig	Beliebig	1	1	1	Beliebig
Visuelles-Formular	Nein	Ja	Ja*	Ja	Nein	Interaktive-Erstellung	Nein	Ja
Globale Erklärbarkeit	Ja*	Nein	Nein	Nein	Ja*	Ja*	Ja*	Ja
Lokale Erklärbarkeit	Ja*	Nein	Nein	Nein	Ja*	Ja	Ja	Ja
Fairness-Metriken	Ja	Ja*	Ja	Ja*	Nein	Nein	Nein	Ja
Leistungsmetriken	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
Benutzerdefinierte Metriken	Nein	Ja*	Nein	Nein	Nein	Nein	Nein	Ja
Ziel: Vebesserung der	Genauigkeit, Fairness, Erklärbarkeit	Fairness	Genauigkeit, Fairness	Fairness	Erklärbarkeit	Erklärbarkeit von Decision-trees	Erklärbarkeit von iForest	Genauigkeit, Fairness, Erklärbarkeit

Tabelle 3.1: Übersicht verschiedener interaktiver Systeme und deren Eigenschaften. Ein * kennzeichnet eine hohe Vielfalt an Methoden die vom System unterstützt werden.

In den folgenden Abschnitten wird auf einige Systeme aus der Tabelle näher eingegangen. Es wird ein tieferer Einblick in das What-If-Tool und das Fairlearn-Framework gegeben, da sie auch auf die Untersuchung von Fairness spezialisiert sind.

3.2.1 What-If-Tool

Das What-If-Tool [WPB+19] von Google ist ein Open-Source-Visualisierungstool, das für die Untersuchung von Klassifizerungs- und Regressions-Modellen entworfen wurde. Was es besonders macht ist, dass es im Gegensatz zu einigen seiner Konkurrenten [BDE+20; BDH+18; DNAA23] auch viele Möglichkeiten bietet um mit dem Modell zu interagieren. Das System ermöglicht den Benutzer „What-if"-Analysen durchzuführen. Dabei werden spezifische Punkte ausgewählt und deren Attributwerte manipuliert, damit dann eine Vorhersage auf die neuen Werte durchgeführt werden kann. Das What-If-Tool ermöglicht es auch den kontrafaktischen Partner von einem ausgewählten Punkt anzuzeigen. Außerdem kann für jedes Attribut das Partielle Abhängigkeitsdiagramm angezeigt werden. All diese Werkzeuge verdeutlichen die Relevanz verschiedener Attribute für eine bestimmte Vorhersage und verbessern somit die lokale Erklärbarkeit.

Durch eine zweidimensionale Visualisierung des Datensatzes inklusive der erhaltenen Vorhersagen für jeden Punkt, können Muster und Strukturen im Modellverhalten erkannt werden. Somit verbessert das System auch die globale Erklärbarkeit. Außerdem ermöglicht diese Darstellung, dass systematisch Fehler oder Diskriminierungen im gesamten Datensatz erkannt werden können. Zusätzlich zu der zweidimensionalen Darstellung, bietet das What-If-Tool einige Kenngrößen des Datensatzes an. Es erlaubt den Datensatz zu gruppieren, damit Kenngrößen separat für beliebige Gruppen ausgewertet werden können, wodurch auch ein Bias in den Daten erkannt werden kann. Zusammengefasst werden im What-If-Tool folgende Werkzeuge angeboten, die auch für das FALCC-Framework relevant sind:

- Vorhersage-Analysen (What-If Analysen, kontrafaktische Untersuchungen).
- Partielle Abhängigkeitsdiagramme.
- Visualisierung des Datensatzes und der Kenngrößen.
- Gruppierungs-Funktion des Datensatzes.
- Neue Datenpunkte hinzufügen und Datenmanipulation.
- Anpassung an Modell Parameter in Echtzeit, um die Auswirkungen auf die Ergebnisse zu sehen.

Das What-If-Tool ist in den beschriebenen Punkten gut geeignet, um den FALCC-Algorithmus zu analysieren. Allerdings trainiert der FALCC-Algorithmus nicht nur ein einziges Modell, das analysiert werden soll, sondern beruht auf der Kombination vieler Submodelle. Viele spezifische Details können also durch die Verwendung des What-If-Tools nicht analysiert werden. Ein Beispiel sind die gebildeteren Cluster, die relevant sind um den FALCC-Algorithmus zu untersuchen. Ein weiterer negativer Punkt ist, dass das What-If-Tool kein visuelles Konfigurations-Auwahl-Formular bereitstellt, das auf die Parameter des FALCC-Algorithmus zugeschnitten ist. Trotzdem ist das Tool in vielen Bereichen sehr fortgeschritten und einige der erwähnten Werkzeuge wurden in FALCC-IS übernommen.

3.2.2 Fairlearn

Fairlearn [BDE+20] ist eine Python-Bibliothek, die ursprünglich in älteren Versionen auch ein Visualisierungs-Interface angeboten hat. Die Bibliothek dient dazu, die Unfairness in einem Modell zu erkennen. Fairlearn ist relevant, da es eine umfangreiche Sammlung von Algorithmen und Metriken für die Bewertung von Fairness bereitstellt. Eine zentrale Fähigkeit liegt in der Möglichkeit, den Trade-Off zwischen Fairness und Genauigkeit zu visualisieren, was eine tiefgreifende Einsicht in die Auswirkungen verschiedener Fairness-Maßnahmen ermöglicht. Ein weiterer interessanter Punkt ist die Option, angepasste Metriken aus alten Metriken zu erstellen, die spezifischen Anforderungen entsprechen. Für das FALCC-Framework könnte diese Idee übertragen werden und es könnte beispielsweise eine lineare Kombination aus der globalen und lokalen Fairness erstellt werden, um das Modell auf beide Fairness-Arten gleichzeitig zu analysieren. Jedoch ist das Visualisierungs-Interface und die Interaktionsmöglichkeiten in Fairlearn begrenzt oder teilweise nicht vorhanden. Trotz dieser Einschränkungen bleibt Fairlearn ein wertvolles Werkzeug im Bereich der Fairnessanalyse, insbesondere durch die Vielzahl an Metriken und durch das einfache Erstellen von Graphen.

3.2.3 Weitere Systeme

FairPilot [DNAA23] ist ein interaktives System, welches speziell darauf ausgelegt wurde, die Leistung von verschiedenen Klassifizierungsmodellen und Hyperparametern zu untersuchen. Die Besonderheit des Systems ist die Darstellung der Modell-Performanz durch ein Streudiagramm, das die Pareto-Front visualisiert. Die Pareto-Front enthält alle Modelle, für die es keine anderen Modelle gibt, die in allen Metriken besser sind. Diese Darstellung ermöglicht eine schnelle Filterung der Modelle, die als dominiert betrachtet werden können. AIF360 [BDH+18] ist ein Toolkit zur Erkennung und Minderung von Bias. Ein wichtiger Verwendungszweck von AIF ist das Untersuchen der Effektivität von verschiedenen Fairness-Verbesserungsansätzen. Gamut [HHC+19] ist ein visuelles Analyse-System, das einige Werkzeuge zur Verbesserung der Erklärbarkeit integriert. Baobab View [VV11] bietet ein Werkzeug, um interaktiv einen Entscheidungsbaum aufzubauen und zu visualisieren. Dadurch werden die Entscheidungswege des Modells transparent. iForest[ZWLC19] ist ein Analyse-Tool, das speziell dazu entwickelt wurde, die Erklärbarkeit von Random-Forest-Modellen zu verbessern.

4 Interaktives System

In diesem Abschnitt wird FALCC-IS, ein interaktives System für das FALCC-Framework vorgestellt. Zunächst werden die technischen Implementierungsdetails und die Architektur des Systems behandelt (siehe Kapitel 4.2). Danach wird der generelle Arbeitsablauf mit dem System beschrieben (siehe Kapitel 4.3). Anschließend werden die bereits implementierten Werkzeuge präsentiert und diskutiert, wie sie bei der Auswertung des FALCC-Algorithmus, bei der Erkennung von Unfairness und zur Steigerung der Erklärbarkeit helfen können (siehe Kapitel 4.5). Dabei wurden einige Ideen aus dem vorherigen Kapitel aufgegriffen und angewendet. Darüber hinaus werden spezielle Visualisierungskonzepte eingeführt, die auf das FALCC-Framework zugeschnitten sind. Zur Vereinfachung wird in den folgenden Abschnitten eine Parameterkonfiguration und das daraus resultierende trainierte FALCC-Modell als „Run “bezeichnet.

4.1 Anforderungen

- **Benutzung des FALCC-Frameworks:** Es soll eine einfachere Handhabung des Algorithmus ermöglicht werden. Selbst unerfahrene Benutzer ohne Programmiererfahrung sollen in der Lage sein, das Framework zu nutzen. Dies beinhaltet das automatische Starten der Offline und Online-Phase und das Bereitstellen der Ergebnisse.
- **Erkennen von Ungenauigkeit und Bias:** Da das FALCC-Framework speziell darauf abzielt,die Fairness zu verbessern, ist es besonders wichtig, dass der Benutzer analysieren kann, ob der Algorithmus fair arbeitet. Eine hohe Fairness kommt oftmals zulasten der Genauigkeit, daher ist es relevant, die Modelle gleichzeitig auf die Genauigkeit zu überprüfen. Ebenfalls wichtig ist das Erkennen von Bias in den Datensätzen.
- **Verbesserung der Erklärbarkeit:** Die Erklärbarkeit des FALCC-Algorithmus ist eher niedrig (siehe Kapitel 1.1.2). FALCC-IS soll Werkzeuge beinhalten, die es erlauben, die Entscheidungen des Algorithmus besser nachzuvollziehen.
- **Parameter Optimierung:** Parameterkonfigurationen können ausgewertet und verglichen werden, mit dem Ziel, gute Parameterkonfigurationen zu erarbeiten.

4.2 Verwendete Technologien

FALCC-IS ist ein web-basiertes interaktives System. Es ist im Frontend mit *Angular*[1] in Typescript implementiert. Zusätzlich wird *Bootstrap*[2] verwendet, um das Designen von Bedien-Elementen zu vereinfachen. *Angular* bietet eine Vielzahl von nützlichen Bibliotheken an. Für FALCC-IS werden zudem grafische Darstellungen benötigt, daher wurde *Charts.js*[3] verwendet. *Charts.js* eignet sich gut zur Darstellung von Datensätzen mit vielen Punkten und bietet einige verschiedene Arten von Graphen an. Das Backend wurde mit *Flask*[4] in Python implementiert, weil es sich dabei um ein sehr flexibles Webserver-Framework handelt. Es ermöglicht einfach und schnell die Erstellung einer REST API und sorgt für die Auslieferung der HTML-Seiten.

4.3 Bedienablauf

Ein typischer Ablauf bei der Benutzung von FALCC-IS wird in der Abbildung 4.1 dargestellt. Der Benutzer macht zuerst Eingaben in der Konfigurations-Auswahl im Frontend. Die ausgewählten Parameterkonfigurationen werden im JSON-Format über die HTTP-REST-Schnittstelle an das Backend gesendet. Die Konfigurationen werden dann nacheinander durch die Offline-, Online- und Evaluationsphase des FALCC-Frameworks geschickt. In der Evaluationsphase werden die Leistungen des Modells berechnet. Dies basiert auf den Vorhersagen, die in der Online-Phase gemacht wurden. Alternativ kann der Benutzer einen bereits trainierten Run mit der gleichen Konfiguration wieder verwenden, anstatt einen neuen Run zu trainieren. Wird ein weiterer Run trainiert, werden die Ergebnisse aus der Offline-, Online- und Evaluationsphase abgespeichert, um zukünftige Abfragen zu beschleunigen. Der Benutzer kann nun direkt über die REST-Schnittstelle mit dem in der Offline-Phase trainierten Modell interagieren. Die in der Evaluationsphase ausgewerteten Metriken können in verschiedenen Graphen im Frontend angezeigt werden. Die Ergebnisse und die Modell-Interaktionen können dann manuell interaktiv vom Benutzer ausgewertet und beurteilt werden. Mit den gewonnenen Erkenntnissen kann der Benutzer seine Parameterkonfigurationen optimieren. Dieser Vorgang kann so oft wiederholt werden, bis die nötige Performanz, also zum Beispiel die gewünschte Fairness erreicht wurde. Daher könnte man diesen Ablauf auch als Optimierungszyklus oder Iteration bezeichnen. Die Abbildung 4.1 stellt die Ereignisse in vereinfachter Form dar, indem sie sich auf die Hauptkomponenten konzentriert. Tatsächlich umfasst FALCC-IS jedoch weitere Komponenten, darunter beispielsweise eine Komponente zur Verarbeitung des Datensatzes für eine zweidimensionale Darstellung.

4.4 Unterstützte Datensätze

Das FALCC-Framework und FALCC-IS bieten einige bereits vorverarbeitete Datensätze zur Analyse an. Diese Datensätze enthalten nur numerische Daten und können somit für den FALCC-Algorithmus genutzt werden. Alle Datensätze, die bereits zur Verfügung stehen, werden in [LH24] beschrieben.

[1] https://angular.io/docs
[2] https://getbootstrap.com/docs/5.3/getting-started/introduction/
[3] https://www.chartjs.org/docs/latest/
[4] https://flask.palletsprojects.com/en/3.0.x/

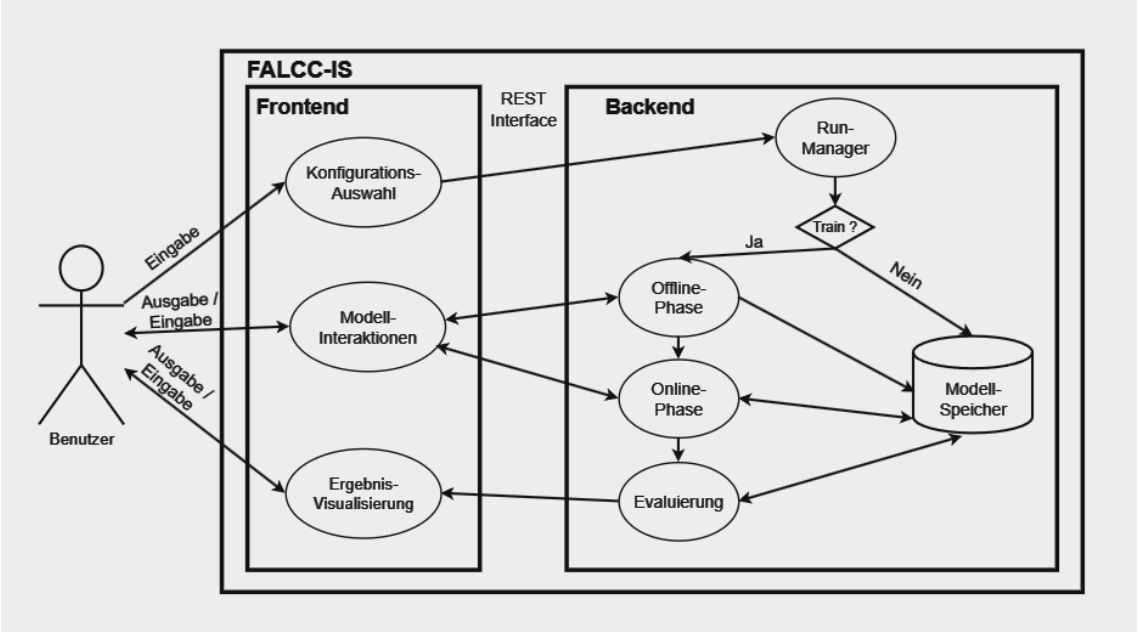


Abbildung 4.1: Bedienablauf von FALCC-IS

Die Integration neuer Datensätze ist einfach möglich, solange diese ausschließlich numerische Daten enthalten. In diesem Kapitel wird der vorverarbeitete *Communities and Crime*-Datensatz [Red09] zur Demonstration der Werkzeuge verwendet. Der Datensatz enthält Informationen über gewalttätige Verbrechen in verschiedenen Gemeinden der USA und umfasst 2.000 Einträge sowie 91 verschiedene Attribute. Als sensitives Attribut für den FALCC-Algorithmus wird die Ethnizität gewählt. Das Label gibt an, ob eine Gemeinde eine überdurchschnittlich hohe Kriminalitätsrate aufweist.

4.5 Implementierte Werkzeuge

In den vorherigen Kapiteln haben wir einige Werkzeuge und Konzepte kennengelernt. In diesem Abschnitt liegt der Fokus darauf, welche dieser Werkzeuge realisiert wurden und wie die Implementierung umgesetzt ist. Außerdem werden auch neue Visualisierungskonzepte eingeführt, die speziell für das FALCC-Framework konzipiert wurden.

4.5.1 Konfigurations-Auswahl

Diese Komponenten sollen die Selektion einer geeigneten Parameterkonfiguration erleichtern. Momentan stehen dafür zwei Konzepte zur Verfügung.

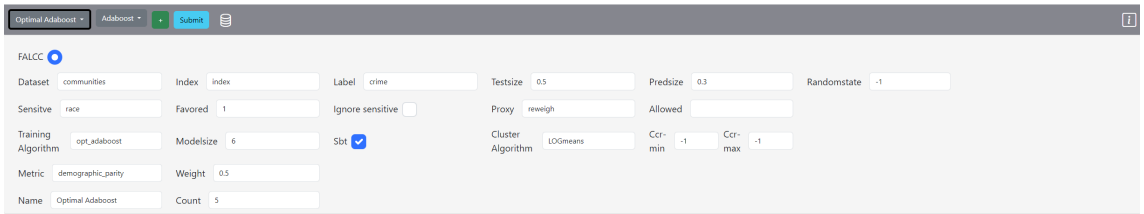


Abbildung 4.2: Eingabeformular des interaktiven Systems

Eingabeformular

FALCC-IS verfügt über ein umfangreiches Eingabeformular (siehe Abbildung 4.2), das alle Parameter aus Tabelle 2.1 sowie einige Datensatz-spezifische Parameter abfragt. Jedem Run muss vom Benutzer ein Name zugewiesen werden. Dieser Name bestimmt, wie das Modell später in den Grafiken bezeichnet wird. Es sollte deshalb ein eindeutiger Namen vergeben werden, der die gewählten Parameter gut beschreibt. Darüber hinaus kann der Benutzer festlegen, wie oft eine bestimmte Konfiguration trainiert werden soll, indem man den Parameter „Count“ verwendet. Für die Modellvalidierung ist es üblich, dass nicht nur ein einziges Modell trainiert und ausgewertet wird, sondern gleich mehrere. Dabei verwenden alle Runs einen unterschiedlichen Teil des Datensatzes für das Training, die Validierung und die endgültigen Vorhersagen. Eine höhere Anzahl an trainierten Runs führt zu mehr Robustheit und Generalisierbarkeit und somit zu einer besseren Bestätigung, dass der ausgewählte Parametersatz gute Ergebnisse erzielt. Da eine einzelne Parameterkonfigurationen in den meisten Anwendungsfällen nicht ausreicht, kann der Benutzer durch Verwendung des Plus-Symbols in der Navigationsleiste weitere hinzufügen. In der Navigationsleiste kann zwischen allen ausgewählten Konfigurationen gewechselt werden. Im abgebildeten Beispiel möchte der Benutzer zwei unterschiedliche Konfigurationen miteinander vergleichen. Eine Konfiguration verwendet den Trainingsalgorithmus AdaBoost und die andere verwendet den optimierten AdaBoost. Dementsprechend wurde dem ersten Run vom Benutzer der Name „AdaBoost“ und dem zweiten Run der Name „Optimal AdaBoost“ zugewiesen. Beide Konfigurationen haben den Parameter Count auf fünf gesetzt, was bedeutet, dass jeweils fünf Runs trainiert werden.

Modell-Speicher

Das ständige Trainieren von Modellen wird für große Datensätze und bei komplizierten Parameterkonfigurationen schnell langwierig. Deswegen wurde ein neues Konzept hinzugefügt, der sogenannte Modell-Speicher. Durch Klicken auf das Datenbank-Symbol in der Navigationsleiste wird ein Fenster geöffnet, das Informationen über alle bereits trainierten Konfigurationen enthält. Wenn Runs für die gewünschten Konfigurationen bereits trainiert wurden, kann man diese auswählen und sie werden automatisch als neue Konfiguration in das Eingabefeld geschrieben. Dort wird auch angezeigt, für wie viele Runs diese Konfigurationen trainiert und gespeichert sind. Wenn dem Benutzer die Anzahl der trainierten Modelle noch nicht ausreicht, kann man weitere trainieren, indem man den Count Parameter auf den gewünschten Wert setzt. Der Count Parameter gibt also an, wie viele Runs neu trainiert werden sollen.

Nach der Auswahl der Konfigurationen können die entsprechenden Berechnungen gestartet werden. Insbesondere für noch nicht trainierte Durchläufe kann der Prozess, bestehend aus der Offline-Phase, der Online-Phase, der Evaluationsphase sowie der Datenaufbereitung besonders bei großen Datensätzen lange dauern. Nachdem der Server die Berechnungen abgeschlossen hat, bietet FALCC-IS verschiedene Visualisierungstechniken an, um den Vorhersagedatensatz inklusive der Modellvorhersagen zu untersuchen.

4.5.2 Visualisierung des Datensatzes

Zuerst muss der Benutzer auswählen, mit welchem der trainierten Runs interagiert werden soll. Beispielsweise möchte man im Fall von Abbildung 4.3 mit dem Run interagieren, der AdaBoost genannt wurde. Das interaktive System bietet zwei Möglichkeiten zur Reduktion eines multidimensionalen Datensatzes auf zwei Dimensionen:

- **Verwendung eines Dimensionsreduktionsalgorithmus [ASS21]**: Im FALCC-System wird der t-SNE-Algorithmus auf einen durch die Proxy-Minderung-Strategie veränderten Vorhersagedatensatz angewandt. Das Ergebnis des t-SNE-Algorithmus ist eine 2D Darstellung des Datensatzes, wobei eine geringe Distanz zwischen zwei Punkten bedeutet, dass sie sich ähnlich sind. Der Dimensionsreduktionsalgorithmus wird nach der Proxy-Minderung-Strategie angewandt, damit die gebildeten Cluster mit der zweidimensionalen Darstellung möglichst optimal übereinstimmen. In der Visualisierung des *Communities and Crime*-Datensatzes (4.3) aus dem t-SNE-Algorithmus sieht man, dass die Cluster größtenteils übereinstimmen. Allerdings gibt es Abweichungen, etwa bei den Ausreißern in Cluster 0. Das liegt daran, dass bei einem Dimensionsreduktionsalgorithmus Informationen verloren gehen, da versucht wird, alle Attribute im 2D-Raum darzustellen. Es wurden einige andere Dimensionsreduktionsalgorithmen getestet, aber keiner von ihnen hat bessere Darstellungen geliefert.
- **Individuelle Auswahl von Attributen für jede Achse**: Eine weitere Möglichkeit, den Datensatz zu visualisieren, besteht darin, für jede Achse einzeln auszuwählen, welches Attribut angezeigt werden soll. Für Attribute, die nur eine beschränkte Anzahl an Werten annehmen können, stößt dieser Ansatz allerdings auf das Problem, dass viele Punkte dann aufeinanderliegen.

Die Visualisierung des Datensatzes dient im Allgemeinen dazu, Muster und Korrelationen zwischen den Daten zu erkennen. Zusätzlich gibt es noch viele weitere Verwendungszwecke, abhängig von den gewählten Visualisierungsoptionen. Die Auswahl an Visualisierungsoptionen umfasst die Visualisierung von Clustern, vorhergesagten Labels, tatsächlichen Labels, sensitiven Attributen und korrekt vorhergesagten Punkten. Die Abbildung 4.3 zeigt die Visualisierung des *Communities and Crime*-Datensatzes, wobei die Cluster als Visualisierungsoption gewählt wurden. FALCC-IS erlaubt auch die Kombination von Optionen. In Abbildung 4.4 zeigt die Form der Punkte die sensitiven Gruppen und die Farbe, ob ein Punkt korrekt vorhergesagt wurde. Dadurch wird eine detailliertere Analyse in folgenden Punkten ermöglicht:

Untersuchung der Fairness

Durch die Visualisierung der vorhergesagten Labels und der sensitiven Gruppen kann untersucht werden, wie fair das Modell gemäß der Demographic Parity ist. Zur Erinnerung: Demographic Parity ist dann erfüllt, wenn der Anteil an positiven Vorhersagen für alle sensitiven Gruppen gleich groß ist. Demnach kann der Benutzer zählen, wie oft eine sensitive Gruppe eine positive Vorhersage erhalten hat.

Was diese Visualisierungsmöglichkeit aber erst relevant macht, ist das diese Betrachtung der Fairness sowohl global als auch in jeder möglichen lokalen Region getätigt werden kann und somit umfangreicher als der vom Computer berechneten Wert der lokalen Fairness-Metrik sein kann. Dasselbe kann für die Equalized Opportunity oder Equal Opportunity Metrik untersucht werden, wenn anstatt der Option, die vorhergesagten Labels zu visualisieren, die Option gewählt wird, die korrekt vorhergesagte Labels hervorhebt. Das Visualisieren der im Datensatz vorhandenen Labels kann zudem zur Beurteilung der Fairness der Daten dienen.

Verbesserung der Erklärbarkeit

Die Visualisierungen verbessern die Erklärbarkeit des Modells. Durch die Anzeige der gebildeten Cluster kann unter anderem besser verstanden werden, welche Punkte dieselbe Vorhersagekombination nutzen somit wird transparent, welches Modell verwendet wurde, um die Vorhersage für einen Punkt zu tätigen. Die lokale Erklärbarkeit (siehe Kapitel 3.1.3) kann verbessert werden, indem die Relevanz eines Attributes auf die Vorhersage analysiert wird. Dafür wird dieses Attribut auf eine Achse gelegt und als Visualisierungsoption das vorhergesagte Label gewählt. Dadurch kann die direkte Korrelation zwischen dem untersuchten Attribut und dem Label festgestellt werden. Ebenfalls können hier kontrafaktische Erklärungen gemacht werden, indem untersucht wird, welcher Punkt die geringste Distanz zum betrachteten Punkt, aber ein anderes Label hat. Das Modell kann auch auf einer globalen Ebene untersucht werden, indem die t-SNE-Darstellung inklusive erhaltener Vorhersagen verwendet wird. Dadurch können Muster und Strukturen in den Vorhersagen erkannt werden.

4.5.3 Modell-Interaktionen

Der Benutzer hat die Möglichkeit, gezielt einen spezifischen Punkt zur Betrachtung auszuwählen. Dadurch werden alle Attribute und deren Werte vor und nach der Proxy-Minderungs-Technik in eine Tabelle geschrieben (siehe Abbildung 4.5). Der Benutzer kann manuell neue Punkte definieren oder gezielt die Attributwerte bestehender Punkte ändern. In dieser Komponente kann daher mit dem trainierten Modell interagiert werden, indem neue Vorhersagen getätigt werden. Der ausgewählte Datenpunkt wird an den Server geschickt und eine Vorhersage wird initiiert, was bedeutet, dass die Online-Phase des FALCC-Algorithmus durchgeführt wird. Falls der Benutzer mehrere Konfigurationen mit demselben Datensatz ausgewählt hat, werden die Vorhersagen für alle Modelle durchgeführt. Tabelle 4.6 zeigt die Vorhersage für den ausgewählten Punkt. Neben den Vorhersagen der FALCC-Modelle können auch die Vorhersagen der einzelnen Submodelle angezeigt werden. In unserer Tabelle wurden beispielsweise für alle sechs Modellkombinationen Vorhersagen getätigt. Die Zeile mit dem blauen Hintergrund repräsentiert das Submodell, das im Modellbewertungschritt der Offline-Phase des FALCC-Algorithmus für das entsprechende Cluster ausgewählt wurde. Dieses

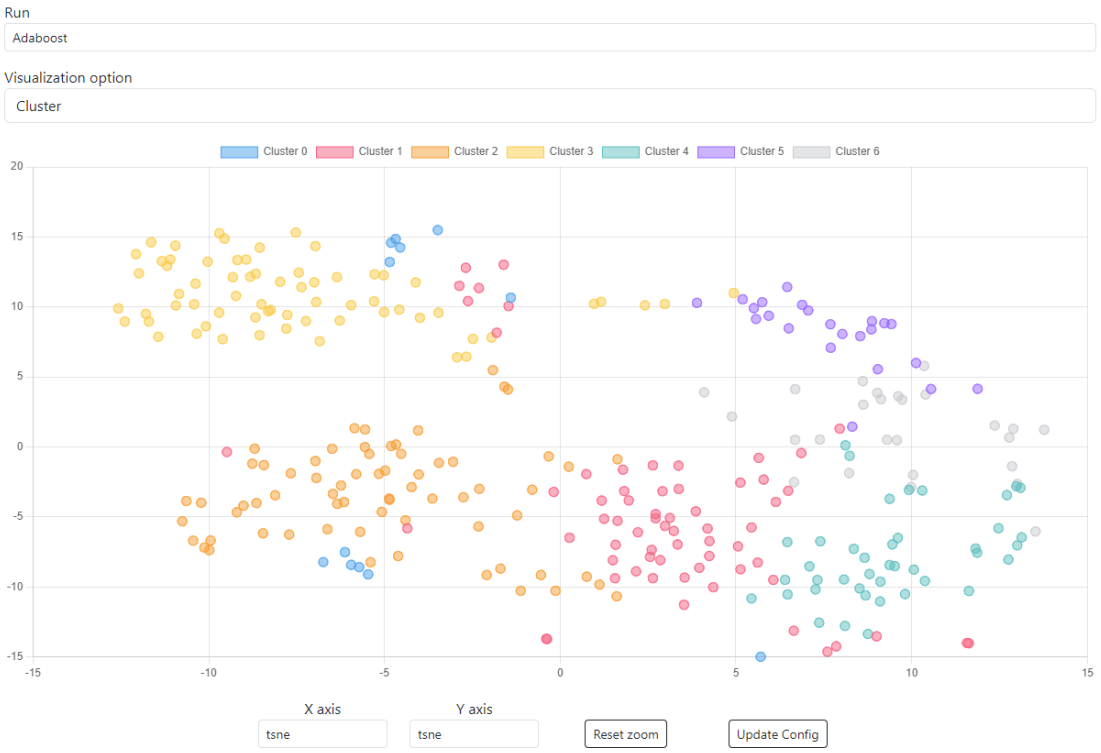


Abbildung 4.3: 2D Darstellung des *Communities and Crime*-Datensatzes inklusive Cluster

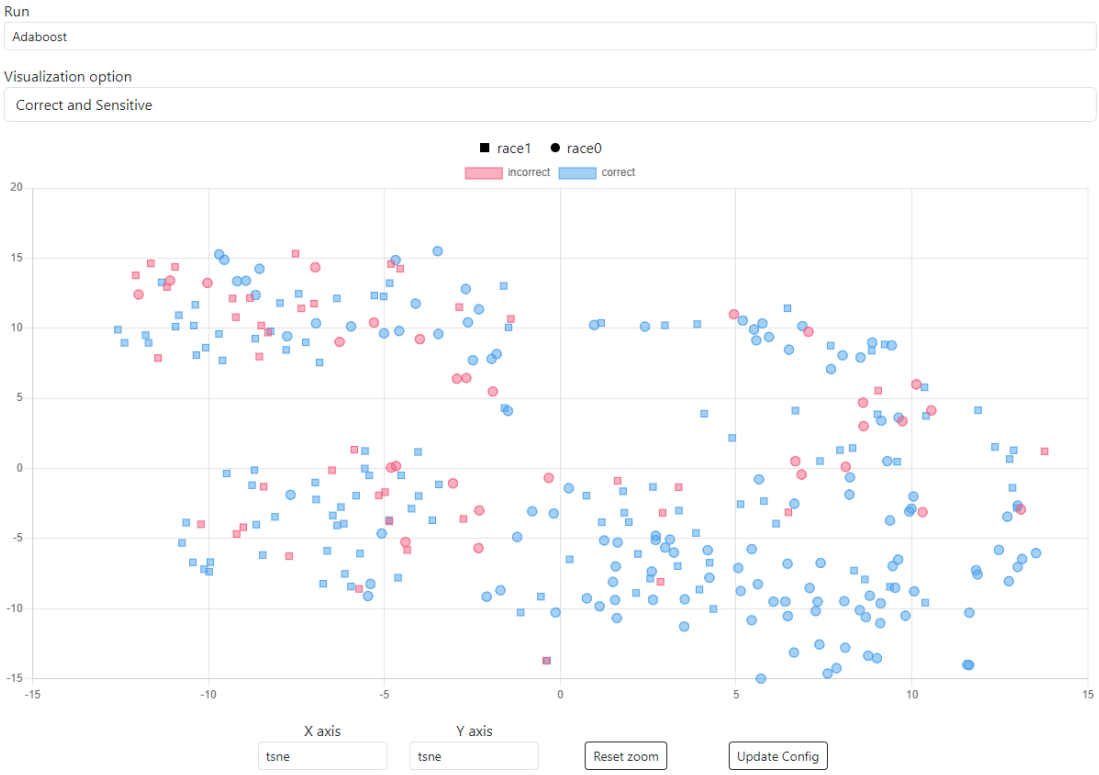


Abbildung 4.4: 2D Darstellung des *Communities and Crime*-Datensatzes inklusive korrekt vorhergesagte Punkte und sensitiven Gruppen

attribute	Start Prediction	Counterfactual	Proxy mitigation
FemalePctDiv	0,56		0,35019838297754785
HousVacant	0,04		0,029984288401813834
LandArea	0,03		0,025555916202559363
LemasPctOfficDrugUn	0		0
MalePctDivorce	0,45		0,28781349422689817
MalePctNevMarr	0,39		0,3081187179769526
MedNumBR	0		0

Abbildung 4.5: Attribut-Werte eines ausgewählten Punktes

Run Name	Prediction
Adaboost	1
1_adaboost_0_model.pkl	1
1_adaboost_1_model.pkl	1
1_adaboost_2_model.pkl	1
1_adaboost_3_model.pkl	1
1_adaboost_4_model.pkl	1
1_adaboost_5_model.pkl	1
Optimal Adaboost	1

Abbildung 4.6: Vorhergesagen der FALCC-Modelle und Submodelle

Modell ist somit für die Klassifizierung verwendet worden. Hauptzweck dieser Komponente ist die Verbesserung der lokalen Erklärbarkeit des Modells. Dazu kann der Benutzer What-If Analysen durchführen. Man kann untersuchen, ob das gezielte Ändern eines Wertes eine Auswirkung auf die Vorhersage hat. Darüber hinaus kann eine kontrafaktische Erklärung durchgeführt werden, indem im Graphen der zum ausgewählten Punkt nächstgelegene Punkt dick hervorgehoben wird, der eine andere Vorhersage erhalten hat. Dies gibt den Benutzern die Möglichkeit nachzuvollziehen, welche Wertänderungen erforderlich gewesen wären, um ein anderes Ergebnis zu erzielen. Diese kontrafaktische Erklärung könnte auch durch das im vorherigen Kapitel beschriebene Werkzeug mit den Distanzen der t-SNE-Datenpunkte erfolgen. Allerdings basiert der in dieser Komponente berechnete kontrafaktische Partner auf der tatsächlichen Ähnlichkeit zwischen zwei Punkten. Wobei hier der euklidische Abstand verwendet wird. Daher ist es durchaus möglich, dass der angezeigte kontrafaktische Punkt nicht derselbe ist wie der, der im Datenvisualisierungswerkzeug am nächsten erscheint.

4.5.4 Evaluation der Ergebnisse

Die letzte Kategorie von Werkzeugen, die FALCC-IS bietet, dient der Analyse und dem Vergleich der Leistung des Algorithmus. Wie auch beim FALCC-Framework liegt ein starker Fokus auf der Fairness. Die folgenden Visualisierungswerkzeuge helfen, die Leistung des FALCC-Algorithmus zu untersuchen und darauf aufbauend eine passende Parameterkonfiguration zu finden. Es können unfaire Modelle identifiziert und verbessert werden.

Abbildung 4.7: Linearkombination von Metriken

Metriken

Das wichtigste Mittel, um die Leistung zu bewerten, sind geeignete Metriken. FALCC-IS ermöglicht die Analysen von folgenden Metriken:

- Gruppen Fairness-Metriken: Demographic Parity, Equalized Odds, Equal Opportunity, Treatment Equality
- Individuelle Fairness-Metriken: Consistency
- Lokale Fairness-Metriken: Lokale Demographic Parity, Lokale Equalized Odds, Lokale Equal Opportunity, Lokale Treatment Equality, Lokale Consistency
- Leistungsmetriken: Genauigkeit, Ungenauigkeit, False Negative, False Positve, F1 Score

Diese Metriken wurden weitestgehend in Kapitel 2 behandelt. Die Lokale Fairness-Metriken berechnen die Gruppen Fairness für lokale Regionen. Der finale Wert wird berechnet, indem die einzelnen lokalen Fairnesswerte für jede Region addiert und relativ zu der Größe der Region gewichtet werden. Die False Negative und False Positiv Werte können separat für alle sensitiven Gruppen betrachtet werden und dienen somit nicht nur als Leistungsmetriken, sondern können auch als Fairness-Metriken betrachtet werden [PS22]. Außerdem ist eine Komponente integriert, die es ermöglicht, eine neue Metrik zu erstellen, die als Linearkombination aus bereits existierenden Metriken darstellbar ist. Der Benutzer wählt die Metriken $m_1, ..., m_n$ mit dazugehörigen Gewichten $w_1, ..., w_n$. Die neue, linear kombinierte Metrik berechnet sich wie folgt:

$$m_{neu} = \sum_{i=1}^{n} \frac{w_i \cdot m_i}{\sum_{i=1}^{n} w_i}$$

Der erstellten Metrik wird ein Name zugewiesen, unter dem der Benutzer darauf zugreifen kann, bis der Server neu gestartet wird. Dieses Werkzeug ist dann relevant, wenn man die Modelle nicht nur in einem Bereich vergleichen will, sondern untersuchen will, welches Modell über viele Metriken hinweg konstant gute Ergebnisse liefert. Als Beispiel, wie das Werkzeug benutzt werden kann, wird in Abbildung 4.7 eine Linearkombination aus drei verschiedenen Metriken ausgewählt. Es wird mit doppelter Gewichtung die Ungenauigkeit und mit einfacher Gewichtung die globale und lokale Demographic Parity betrachtetet. Durch die gewählten Gewichte hat die gesamte Fairness einen gleichstarken Einfluss wie die Genauigkeit auf das Ergebnis. Wichtig ist dabei, dass nicht jede Kombination von Metriken sinnvoll ist. Die Kombination einer Metrik, bei der ein hoher Wert ein Indikator für ein gutes Ergebnis ist, ist nicht kompatibel mit einer Metrik, bei der ein hoher Wert ein Indikator für ein schlechtes Ergebnis ist. Die nächsten Abschnitte behandeln Darstellungsformen der erzielten Werte der Metriken.

Balkendiagramm

Die einfachste Darstellungsform bietet das Balkendiagramm. FALCC-IS stellt zwei verschiedene Darstellungsformen zur Verfügung. Das erste Balkendiagramm in Abbildung 4.8 setzt alle ausgewählten Run-Namen auf die x-Achse und kann bis zu drei verschiedene Metriken visualisieren. Diese Darstellungsform dient zum Vergleich der verschiedenen Konfigurationen. Das zweite Balkendiagramm in Abbildung 4.9 visualisiert für einen ausgewählte Run, und dessen Submodelle, die Werte einer einzigen Metrik. Der Erfolg des FALCC-Algorithmus kann also nicht nur anhand von konkreten Zahlen betrachtet werden, sondern es kann auch der direkte Vergleich vom FALCC-Modell zu seinen Submodellen gemacht werden. Ist die finale FALCC-Modell Performanz schlechter als die der Submodelle, deutet das darauf hin, dass eine ungünstige Parameterkonfiguration gewählt wurde. Beide Balkendiagramme visualisieren den Durchschnittswert aller trainierten Runs dieser Konfiguration.

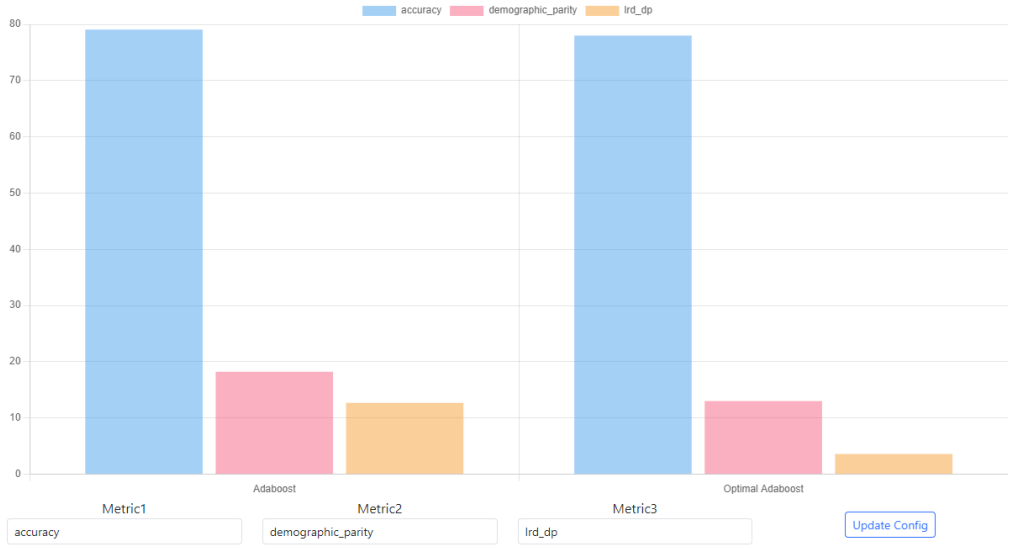


Abbildung 4.8: Balkendiagramm der FALCC-Modelle unter Verwendung der AdaBoost- und Optimal-AdaBoost-Strategie zur Demonstration der Genauigkeit, globalen Demographic Parity und lokalen Demographic Parity

Metrik-Streudiagramm

Einen Komplexitätsschritt weiter geht das Streudiagramm. Das Streudiagramm kann jeweils ein Attribut pro Achse anzeigen. Die Punkte repräsentieren dabei die Ergebnisse der einzelnen Runs. Wobei die kleineren Punkte die im FALCC-Algorithmus trainierten Submodelle sind und die großen Punkte die tatsächlichen Werte des FALCC-Algorithmus darstellen. Es stehen zwei weitere Einstellungsmöglichkeiten zur Verfügung: Erstens kann ausgewählt werden, ob jeweils nur der Durchschnitt aller trainierten Runs mit der gleichen Konfiguration angezeigt werden soll oder jeder einzelne Run separat. Zweitens kann bestimmt werden, ob alle Submodelle angezeigt werden sollen oder nur die Endergebnisse des FALCC-Algorithmus. Diese beiden Optionen können auch kombiniert werden. In der Grafik 4.10 wurde für die x-Achse die Genauigkeit und für die y-Achse

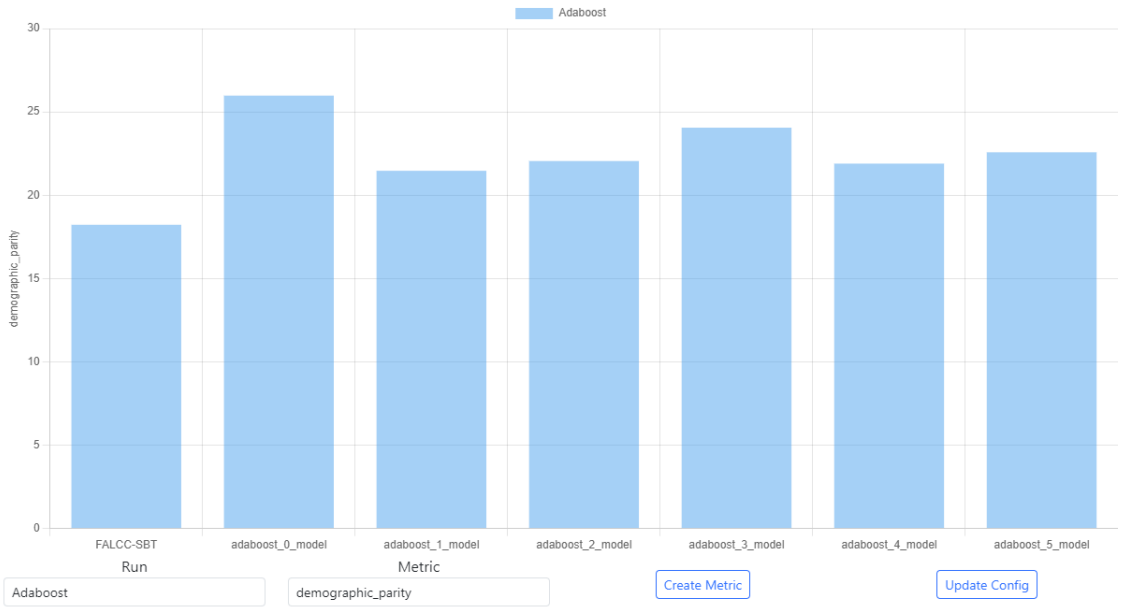


Abbildung 4.9: Balkendiagramm der AdaBoost-Modelle zur Demonstration der globalen Demographic Parity

Abbildung 4.10: Streudiagramm der FALCC-Modelle unter Verwendung der AdaBoost- und Optimal-AdaBoost-Strategie zur Demonstration der Genauigkeit und Demographic Parity

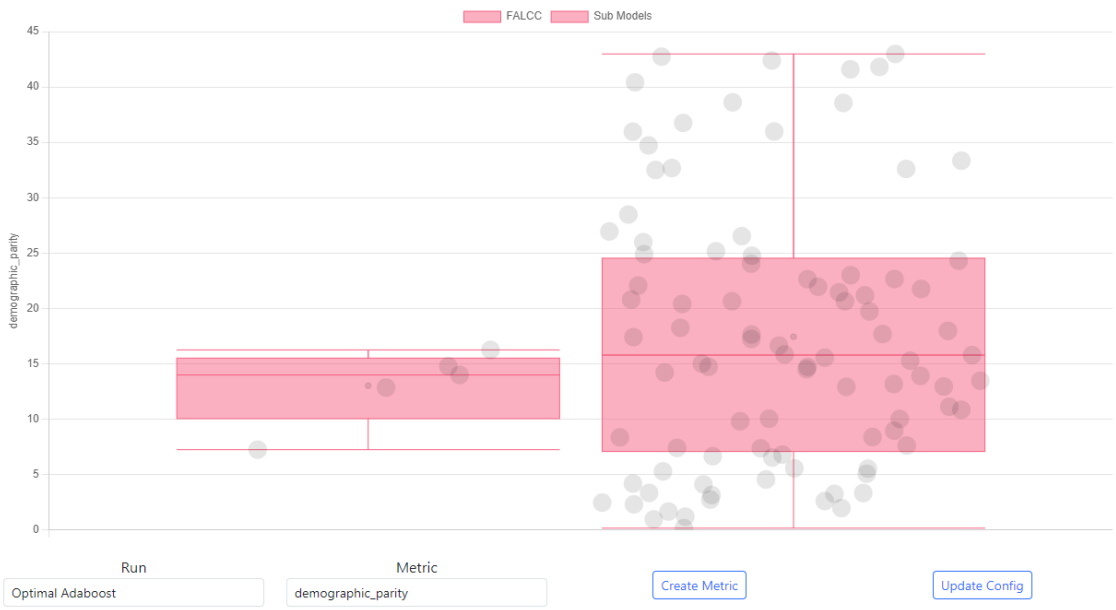


Abbildung 4.11: Boxplot für den Optimal-AdaBoost Run

die Demographic Parity Metrik ausgewählt. Es sollen die Ergebnisse der einzelnen Runs angezeigt werden, aber nicht die einzelnen Submodelle. Somit sind insgesamt zehn Punkte in dem Diagramm vorhanden. Das Punktediagramm hat einige Anwendungen im Bereich der Fairnessuntersuchung. Es können grundlegende Eigenschaften der Metriken untersucht werden. Beispielsweise verschiedene Trade-Offs zwischen Metriken (siehe Kapitel 5). Anhand der Diagramme lassen sich die Pareto-Fronten (siehe Kapitel 3.2.3) einfach grafisch bestimmen. Beispielsweise für Metriken, bei denen ein hoher Wert ein Indikator für eine gute Leistung ist, liegt ein Punkt dann in der Pareto-Front, falls kein anderer Punkt sowohl weiter rechts als auch weiter oben wie der betrachtete Punkt liegt.

Boxplot

Der Boxplot illustriert die Leistungsverbesserung des finalen FALCC-Modells im Vergleich zu seinen Submodellen (siehe Abbildung 4.11). Im Beispiel werden zwei Boxplots generiert. Der erste Boxplot enthält nur die Werte vom FALCC-Algorithmus und der zweite Boxplot nur die Werte der einzelnen Submodelle. Der Boxplot zeigt den niedrigsten Wert, den höchsten Wert, das obere und untere Quartil sowie den Median an. Dadurch wird visualisiert, wie stark das FALCC-Prinzip die Leistungen der einzelnen Submodelle verändert.

4.5.5 Cluster-Informationen

In der Modellauswertungsphase des FALCC-Algorithmus (siehe Kapitel 2.2.1) werden die einzelnen Modellkombinationen für jedes Cluster ausgewertet. FALCC-IS bietet eine Komponente, um diese Auswertungen anzuzeigen. Die Tabelle in Abbildung 4.12 zeigt die Auswertung der Modellkombinationen im Cluster 0. Diese Informationen geben einen Einblick in die Blackbox des Algorithmus. Für jedes Modell werden die Ungenauigkeit, die Wahrscheinlichkeit einer positiven Vorhersage und

Select Cluster

0

Metric: demographic_parity

Weight: 0.5

Submit

model	race	num_of_element	num_of_element_y0	num_of_element_y1	inacc_sum	inacc_sum_y0	inacc_sum_y1	inaccuracy	inaccuracy_y0	inaccuracy_y1	pppv	pppv_y0	pppv_y1	ppv_sum	ppv_sum_y0	ppv_sum_y1	fn	fp
0_adaboost_0	0	129	118	11	10	0	10	0.07751938	0	0.90909091	0.00775194	0	0.09090909	1	0	1	10	0
0_adaboost_1	0	129	118	11	13	4	9	0.10077519	0.03389831	0.81818182	0.04651163	0.03389831	0.18181818	6	4	2	9	4
0_adaboost_2	0	129	118	11	18	8	10	0.13953488	0.06779661	0.90909091	0.06976744	0.06779661	0.09090909	9	8	1	10	8
0_adaboost_3	0	129	118	11	12	1	11	0.09302326	0.00847458	1	0.00775194	0.00847458	0	1	1	0	11	1
0_adaboost_4	0	129	118	11	11	3	8	0.08527132	0.02542373	0.72727273	0.04651163	0.02542373	0.27272727	6	3	3	8	3
0_adaboost_5	0	129	118	11	19	9	10	0.14728682	0.07627119	0.90909091	0.07751938	0.07627119	0.09090909	10	9	1	10	9
1_adaboost_0	1	42	29	13	7	1	6	0.16666667	0.03448276	0.46153846	0.19047619	0.03448276	0.53846154	8	1	7	6	1
1_adaboost_1	1	42	29	13	11	4	7	0.26190476	0.13793103	0.53846154	0.23809524	0.13793103	0.46153846	10	4	6	7	4
1_adaboost_2	1	42	29	13	15	7	8	0.35714286	0.24137931	0.61538462	0.28571429	0.24137931	0.38461538	12	7	5	8	7
1_adaboost_3	1	42	29	13	9	1	8	0.21428571	0.03448276	0.61538462	0.14285714	0.03448276	0.38461538	6	1	5	8	1
1_adaboost_4	1	42	29	13	10	4	6	0.23809524	0.13793103	0.46153846	0.26190476	0.13793103	0.53846154	11	4	7	6	4
1_adaboost_5	1	42	29	13	15	3	12	0.35714286	0.10344828	0.92307692	0.0952381	0.10344828	0.07692308	4	3	1	12	3

Abbildung 4.12: Auswertung der Submodelle für Cluster 0

die Anzahl der inkorrekt positiv und inkorrekt negativ vorhergesagten Punkte angegeben. Zusätzlich kann in dieser Komponente eine neue Fairness-Metrik und ein neues Gewicht $\lambda \in [0, 1]$ angegeben werden, und das System berechnet erneut, welche Modellkombination für die neue Metrik und das Gewicht die Minimierungsfunktion (siehe 2.1) optimal erfüllt. Diese Modellkombination wird in der Tabelle durch einen blauen Hintergrund hervorgehoben. Der Benutzer kann dadurch untersuchen, welche Auswirkungen eine Veränderung dieser Parameter auf die Modellauswertungsphase des FALCC-Algorithmus hat, ohne dafür Trainingsläufe für neue Konfigurationen durchführen zu müssen.

4.5.6 Benutzerfreundlichkeit

In FALCC-IS sind einige Konzepte integriert, die die Benutzung des Systems angenehmer gestalten.

- Alle Werkzeuge verfügen über eine umfangreiche Beschreibung der zu Verfügung stehenden Optionen.
- Fehlgeschlagene Berechnungen und inkompatible Konfigurationen werden dem Benutzer unmittelbar über Pop-Up-Fenster angezeigt.
- Jede Komponente kann aus und eingeblendet werden, um den Benutzer nicht mit Informationen zu überfluten
- Es gibt für jeden Graphen die Option einen Screenshot zu erstellen oder die Daten in eine CSV-Datei zu schreiben. Für die Streudiagramme, gibt es außerdem eine Zoom-Funktion.

4.5.7 Übersicht der implementierten Werkzeuge

Eine Übersicht der implementierten Werkzeuge und ihres Verwendungszwecks bietet Tabelle 4.1. Dabei wird unterschieden, ob das Werkzeug zur Verbesserung der Bedienbarkeit, der Fairness oder der Erklärbarkeit beiträgt. Letztere wird in drei verschiedene Aspekte unterteilt: Werkzeuge, die eine Verbesserung der lokalen Erklärbarkeit ermöglichen, Werkzeuge, die eine Verbesserung der globalen Erklärbarkeit ermöglichen und Werkzeuge, die Provenance-Daten des FALCC-Algorithmus visualisieren, um die Transparenz zu erhöhen (siehe Kapitel 3.1.3). Mit diesen Werkzeugen existieren bereits einige Möglichkeiten, um das FALCC-Framework interaktiv zu erforschen.

	Eingabeformular	**Modellspeicher**	**Datensatz-Streudiagramm**	**Kontrafaktische Erklärungen**	**Metrikkombinierer**	**Balkendiagramm**	**Metrik-Streudiagramm**	**Boxplot**	**Cluster Informationen**
Verbesserung der Bedienbarkeit	✓	✓	✗	✗	✗	✗	✗	✗	✗
Fairness-Analyse	✗	✗	✓	✓	✓	✓	✓	✓	✓
Verbesserung der lokalen Erklärbarkeit	✗	✗	✓	✓	✗	✗	✗	✗	✗
Verbesserung der globalen Erklärbarkeit	✗	✗	✓	✗	✗	✗	✗	✗	✗
Provenance-Daten	✗	✗	✓	✗	✗	✗	✗	✗	✓

Tabelle 4.1: Verwendungszwecke der Werkzeuge

Eine gründliche Modell- oder Fairness-Untersuchung erfordert jedoch ein Zusammenspiel der verschiedenen Werkzeuge. Im nächsten Kapitel wird anhand von zwei Beispielen gezeigt, wie so ein Zusammenspiel aussehen kann und welche Erkenntnisse daraus gewonnen werden können.

5 Demonstration und Ergebnisse

Bislang haben wir nur die Werkzeuge des interaktiven Systemes kennengelernt. In diesem Abschnitt wird gezeigt, welche Erkenntnisse durch die Benutzung der Werkzeuge gewonnen werden können und wie die Werkzeuge dazu eingesetzt werden. Dabei beschränken wir uns auf zwei Anwendungsfälle. Das Ziel dieses Abschnitts besteht darin, interessante Ergebnisse zusammenzutragen und zugleich die breite Palette an Fähigkeiten von FALCC-IS zu veranschaulichen. Es wird gezeigt, wie man mehrere Werkzeuge kombiniert, um die gewünschten Ergebnisse zu erhalten.

5.1 Auswertung der Trainingsstrategien

Der erste Anwendungsfall, der in dieser Arbeit angesprochen wird, ist die Untersuchung der unterschiedlichen Trainingsstrategien. Ziel ist es nicht nur die Methoden auf Genauigkeit und Fairness zu bewerten, sondern auch das generelle Analysieren der Eigenschaften der Fairness-Metriken. Unter anderem werden mögliche Trade-Offs zwischen den einzelnen Metriken verdeutlicht.

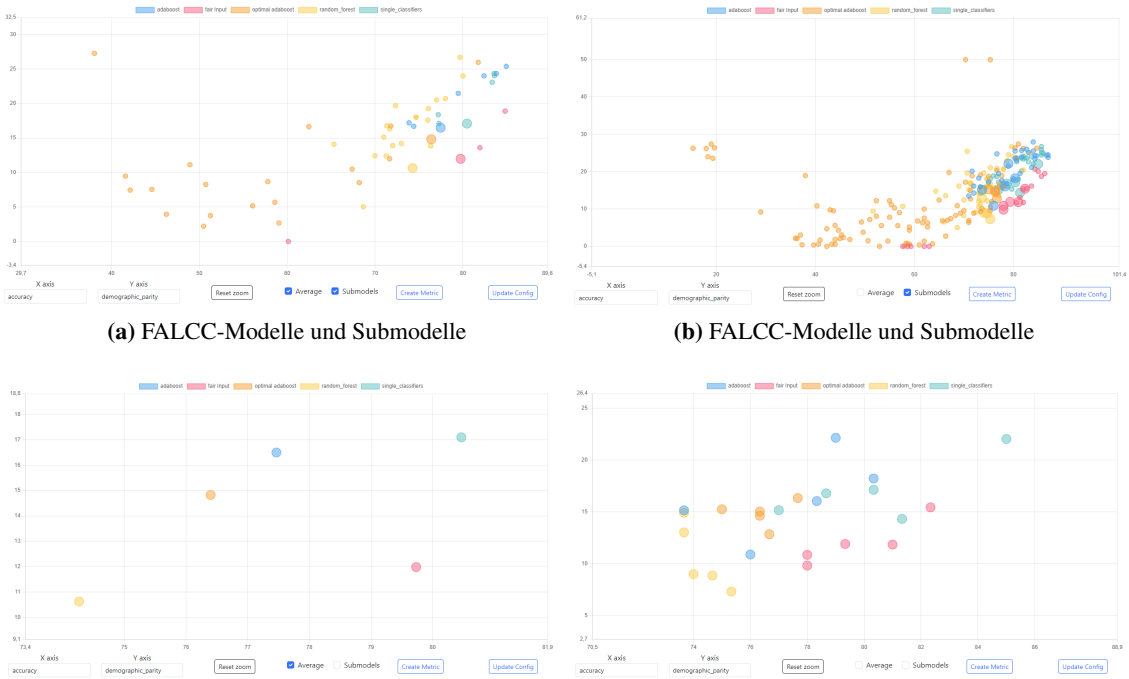
5.1.1 Eingabe

Dazu geben wir als Eingabe in das Eingabeformular fünf unterschiedliche Konfigurationen, die sich nur in der verwendeten Trainingstrategie unterscheiden: (1) AdaBoost, (2) Optimal-AdaBoost (3) Single-Classifier, (4) Optimal-Random-Forest und (5) Fair-Input. Diese Strategien wurden in Kapitel 2.2.1 behandelt. Für jede der Konfigurationen werden fünf Runs trainiert, damit die Ergebnisse aussagekräftig sind. Wir benutzten den *Communities and Crime*-Datensatz und verwenden die Demographic Parity als Fairness-Metrik für die Minimierungsfunktion (siehe 2.1). Der Rest der Konfiguration ist auf die Standardwerte gestellt.

5.1.2 Ergebnisse

Da wir zunächst alle Konfigurationen vergleichen wollen, wählen wir das Metrik-Streudiagramm. In der Abbildung 5.1 wurde auf der x-Achse die Genauigkeit und auf der y-Achse die Demographic Parity ausgewählt. In allen Grafiken kann man erkennen, dass es in unseren gewählten Trainingsstrategien einen Zusammenhang zwischen den beiden Metriken gibt. Wird die Genauigkeit größer, wird der Wert der Demographic Parity ebenfalls größer. Genaue Modelle haben meistens eine schlechtere globale Fairness. Ähnliche Ergebnisse erhält man, wenn man die lokale statt globale Demographic Parity betrachtet.

Interessiert sich der Benutzer nur für die finalen Ergebnisse des FALCC-Algorithmus, sind Abbildung 5.1c und 5.1d relevant. In diesen Abbildungen sieht man nur die Endergebnisse des FALCC-Algorithmus, einmal im Durchschnitt und einmal von allen trainierten Runs. Auffällig



(a) FALCC-Modelle und Submodelle

(b) FALCC-Modelle und Submodelle

(c) FALCC-Modelle

(d) FALCC-Modelle

Abbildung 5.1: Genauigkeit und Fairness: a), c) Durchschnittliche Werte b), d) Alle Werte. ● AdaBoost ● Fair-Input ● Pptimal-AdaBoost ● Optimal-Random-Forest ● Single-Classifier

gute Ergebnisse erzielt die Fair-Input-Konfiguration. Sie erzielte eine durchschnittliche Genauigkeit von ungefähr 71% , aber auch eine durchschnittliche Demographic Parity von 12. Ebenfalls in der Pareto-Front liegt die Optimal-Random-Forest-Strategie und die Single-Classifier-Strategie. Somit gelten die AdaBoost- und die Optimal-AdaBoost-Strategien für die betrachteten Metriken als dominiert. Betrachtet man die lokale Demographic Parity, dann liegen nur die Fair-Input-Strategie und die Single-Classifier-Strategie in der Pareto-Front. Wobei anzumerken ist, dass die Single-Classifier-Strategie nur minimal genauer als die Fair-Input-Strategie ist, aber deutlich unfairer. Deswegen wird im folgenden nur noch weiter auf die Fair-Input-Strategie eingegangen.

Eine naheliegende, auf diese Erkenntnisse aufbauende Frage könnte sein, ob die guten Werte der Fair-Input-Strategie durch den FALCC-Algorithmus zu erklären sind oder ob die guten Werte hauptsächlich den zu Grunde liegenden fairen Submodellen zu verdanken sind. Ein nächster Analyseschritt wäre demnach, die finalen Ergebnisse des FALCC-Algorithmus mit den benutzten Submodellen zu vergleichen. Dazu blendet Abbildung 5.2 alle anderen Konfigurationen aus und zusätzlich wird in Abbildung 5.3 der Boxplot auf der neu erstellten Metrik aus Abbildung 4.7 angewendet. In 5.2 sieht man das der LFR-Algorithmus sehr faire, aber ungenaue Ergebnisse erzielt. Die FALCC-Performanz (dicker Punkt) ist ähnlicher zu den beiden anderen Submodellen, was darauf hindeutet, dass diese beiden häufiger in den einzelnen Clustern als zugehöriges Modell gewählt wurden. Um das genauer zu untersuchen, kann die Cluster-Informations-Komponente benutzt werden. Der Boxplot in Abbildung 5.3, der die neu definierte Metrik visualisiert, die die

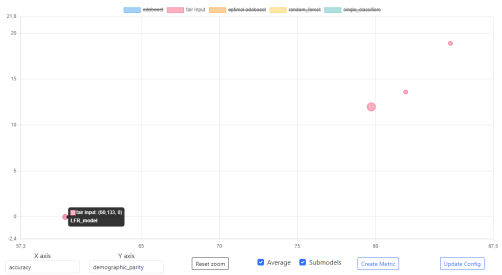


Abbildung 5.2: Genauigkeit und Fairness der Fair-Input-Strategie

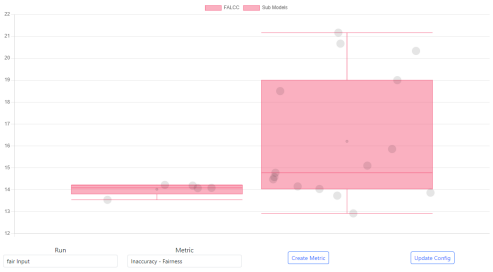


Abbildung 5.3: Ungenauigkeits-Fairness-Metrik der Fair-Input-Strategie

lokale und globale Fairness und die Ungenauigkeit berücksichtigt, sieht für die FALCC-Performanz deutlich vielversprechender aus. Der Durchschnittswert ist zwar nur geringfügig besser, aber das obere Quartal ist deutlich geringer für den FALCC-Algorithmus.

Falls die gewünschte Fairness noch nicht erreicht wurde, könnte man die gleichen Experimente für niedrigere Werte für unseren Gewichtungs λ (siehe Tabelle 2.1) durchführen. Dies könnte allerdings einen negativen Einfluss auf die Genauigkeit haben.

5.2 Auswertung der Proxy-Minderung-Techniken

In der Studie [LH24] wurden die in Abschnitt 2.2.1 beschriebenen Proxy-Minderung-Techniken ausgewertet. Dabei wurden die Techniken auf synthetische Datensätze mit impliziten Bias angewandt. Impliziter Bias bedeutet, dass das sensitive Attribut keinen direkten Einfluss auf die Vorhersagen hat, aber es einige Proxy Attribute gibt, die einen starken Einfluss auf die Vorhersagen haben. Da die Proxy Attribute stark mit den sensitiven Attributen korrelieren, dienen sie als indirekte Indikatoren für die sensitiven Attribute. In diesen Abschnitt werden einige Ergebnisse der Studie durch die Benutzung des interaktiven Systems bestätigt und verdeutlicht.

5.2.1 Eingabe

Wir benutzten das gleiche Setup und geben dementsprechend drei verschiedene Konfigurationen ein. Jeweils eine Konfiguration für die Remove- und Reweigh-Option und eine Konfiguration, die keine Proxy-Minderung-Strategie verwendet. Als Datensatz benutzten wir den *implicit50*-Datensatz, dieser hat starken impliziten Bias. Die bevorzugte sensitive Gruppe hat eine 75% Wahrscheinlichkeit ein positves Label zu erhalten und die benachteiligte sensitive Gruppe nur eine 25% Wahrscheinlichkeit. Es gibt also eine 50% Differenz in der Wahrscheinlichkeit ein positives Label zu erhalten. Wir verwenden wieder die Demographic Parity Metrik und die restlichen Parameter sind auf die Standardwerte gestellt. Für alle Konfigurationen werden fünf Runs durchgeführt.

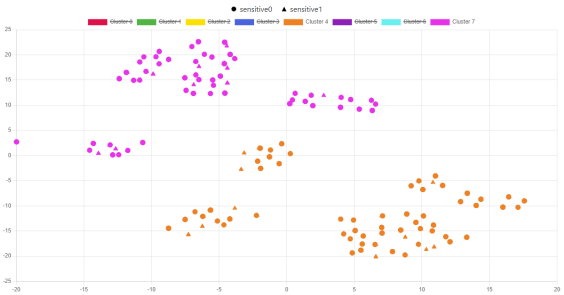
5.2.2 Ergebnisse

Die Autoren der Studie erwarten, dass die Proxy-Minderung-Techniken zu Clustern führen, die hinsichtlich der sensitiven Gruppen ausgewogener sind. Das bedeutet, dass die Anzahl der Punkte pro sensitiver Gruppe in jedem Cluster der Gesamtverteilung der Daten ähnlicher ist, als es ohne Anwendung von Proxy-Minderung-Techniken der Fall wäre. Um diese Aussage zu überprüfen, kann das Datensatzstreudiagramm verwendet werden. Wir wählen als Visualisierungsoption aus, dass sowohl die Cluster als auch die sensitiven Attribute visualisiert werden sollen. Für eine bessere Übersicht werden in den folgenden Abbildungen jeweils nur zwei Cluster eingeblendet. Abbildung 5.4c visualisiert die Cluster, die ohne eine Proxy-Minderung-Technik entstanden sind. Beide Cluster sind dominiert von einer sensitiven Gruppe. Das grüne Cluster enthält zu einem großen Teil Punkte, die zu der sensitiven Gruppe 0 gehören. Das rote Cluster enthält beinahe nur Punkte, die zu der sensitiven Gruppe 1 gehören. Wohingegen die Abbildung 5.4a, die den Datensatz nach der Remove Technik visualisiert und die Abbildung 5.4b die den Datensatz nach der Reweigh Technik visualisiert, in den beiden angezeigten Clustern ausgewogen viele Punkte von jeder sensitiven Gruppe enthalten. Die Proxy-Minderung-Strategie führt also dazu, dass die Cluster weniger von den sensitiven Attributen beeinflusst werden.

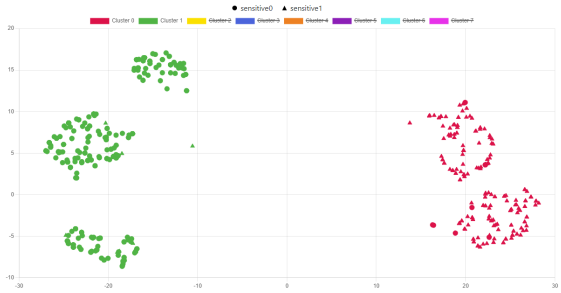
Diese Beobachtung kann verstärkt werden, wenn wir die sensitiven Gruppen visualisieren (siehe Abbildung 5.5). In den oberen beiden Streudiagrammen wurde die Remove Strategie angewendet. In Abbildung 5.5a werden beide sensitiven Gruppen gleichzeitig eingeblendet und für eine bessere Übersicht wird in Abbildung 5.5b nur die sensitive Gruppe 1 dargestellt. Wir sehen, dass die t-SNE-Koordinaten größtenteils unabhängig von den sensitiven Gruppe sind. Gleiche Resultate erhalten wir für die Reweigh Strategie. Wohingegen die Abbildungen 5.5c und 5.5d das gleiche für die Konfiguration, die keine Proxy-Minderung-Technik verwenden anzeigen. Man sieht, dass die t-SNE-Koordinaten stark von den sensitiven Gruppen abhängen. Die Punkte links gehören meistens zur Gruppe 0 und die Punkte rechts zur Gruppe 1. Da die t-SNE-Darstellung und die Cluster zu ähnlichen Regionen führen, ist das auch ein Hinweis darauf, dass die im FALCC-Algorithmus gebildeten Cluster stark von den sensitiven Attributen abhängen, falls keine Proxy-Minderung-Technik angewandt wird. Somit, konnte durch das interaktive System gezeigt werden, wie sich die Proxy-Minderung-Strategien, auf die im FALCC-Algorithmus gebildeten Cluster auswirkt.

Weiterführend kann überprüft werden, ob die gezeigten Auswirkungen der Proxy-Minderung-Strategie positive oder negative Folgen auf die Ungenauigkeit, die globale Fairness und die lokale Fairness haben. Das Balkendiagramm in Abbildung 5.6 zeigt die globale Demographic Partiy, die lokale Demographic Parity und die Ungenauigkeit. Wir sehen, dass die globale Demographic Parity durch die Anwendung einer Proxy-Minderung-Technik deutlich geringer wird. Allerdings wird die lokale Demographic Parity und die Ungenauigkeit größer, wobei die Verbesserung der globalen Fairness, die Verschlechterung der lokalen Fairness und der Genauigkeit überwiegt. Diese Erkenntnisse wurden auch in der Studie festgestellt.

(a) Remove Strategie



(b) Reweigh Strategie



(c) Keine Strategie

Abbildung 5.4: Darstellung der Cluster und der sensitiven Gruppen :
● sensitive Gruppe 0 ▲ sensitive Gruppe 1

5.3 Weitere Anwendungsfälle

In den bisherigen Beispielen wurde veranschaulicht, wie bestimmte Werkzeuge von FALCC-IS praktisch genutzt werden können. Es ist jedoch anzumerken, dass das gesamte Spektrum der verfügbaren Instrumente noch nicht zur Anwendung kam. Die nachstehenden Anwendungsfälle verdeutlichen die Möglichkeiten der bisher ungenutzten Werkzeuge:

- **Manuelle Untersuchung der Fairness-Metriken:** Das in Kapitel 2.2.1 vorgestellte Beispiel könnte durch eine detailliertere Analyse der Fairness weiter vertieft werden. Es besteht die Möglichkeit, im Datensatz-Streudiagramm eine noch nicht genutzte Visualisierungsoptionen auszuwählen, wie etwa die Darstellung der vorhergesagten oder korrekt vorhergesagten Labels. Dies ermöglicht es für jede sensitive Gruppe in jeder beliebigen Region die Anzahl an positiven beziehungsweise korrekten Vorhersagen zu ermitteln. Dadurch können Aussagen über lokale Fairness gemacht werden, die nicht in den anderen Evaluationswerkzeugen angezeigt werden.

- **Modellkombinationen untersuchen:** Im Kapitel 5.1 wurde erwähnt, dass das LFR-Modell in wenigen Clustern angewendet wird. Diese Aussage konnte mittels der Cluster-Information-Komponente überprüft werden. Ein weiterer Einsatz dieses Werkzeugs besteht darin, die Auswirkungen hypothetischer Änderungen im Gewichtungsparameter λ und in der Metrik zu untersuchen, ohne dafür ein neues Modell trainieren zu müssen. Der Benutzer kann so, bevor neue Runs mit angepassten Parametern trainiert werden, beobachten, welche Werte zum gewünschten Verhalten führen.

(a) Remove Strategie - Beide sensitiven Gruppen

(b) Remove Strategie - sensitive Gruppe 1

(c) Keine Strategie - Beide sensitiven Gruppen

(d) Keine Strategie - sensitive Gruppe 1

Abbildung 5.5: Darstellung der sensitiven Gruppen:
● sensitive Gruppe 0 ● sensitive Gruppe 1

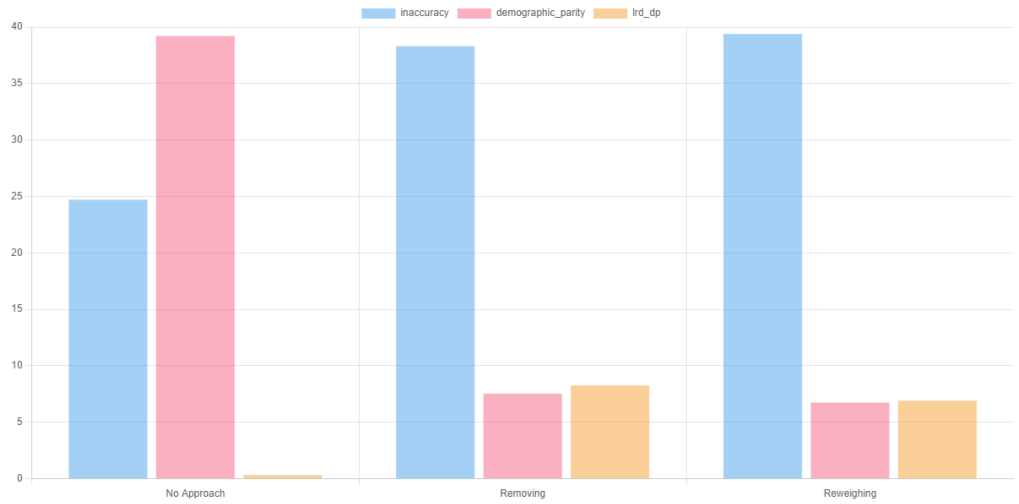


Abbildung 5.6: Vergleich der Proxy-Minderung-Techniken

- **Online Phase starten:** Angenommen, ein Benutzer hat ein FALCC-Modell trainiert und möchte diesen nun tatsächlich einsetzen, dann kann die gewünschte Vorhersage über die Datenpunkt-Tabelle in FALCC-IS initialisiert werden.
- **Vorhersage erklären:** Der Benutzer kann mithilfe des Datensatz-Streudiagramms und der Datenpunkt-Tabelle, What-if-Analysen und kontrafaktischen Untersuchungen durchführen.

6 Zusammenfassung und Ausblick

In dieser Arbeit wurde FALCC-IS, ein interaktives System für das FALCC-Framework vorgestellt. Das FALCC-Framework wird für faire, binäre Klassifikationsprobleme verwendet, weshalb zunächst die verschiedenenen Fairness-Klasssen eingeführt wurden. Danach wurde der FALCC-Algorithmus vorgestellt, mit einem besonderen Fokus auf seine Parameter. Das Konzept von interaktiven Systemen ist weit verbreitet und es gibt eine Vielzahl von interaktiven Systemen für Algorithmen im Bereich der Künstlichen Intelligenz. Relevante ähnliche Systeme wurden ermittelt und es wurde eine Gruppierung der Werkzeuge vorgenommen. Danach ging es um die Implementierung von FALCC-IS. Dabei wurden einige Konzepte aus den anderen Systemen übernommen, sowie auch neue Werkzeuge entwickelt. Unter anderen wurde ein FALCC-spezifisches Eingabefeld, eine zweidimensionale Darstellung des Datensatzes, inklusive erhaltener Vorhersagen und erstellten Clustern und einige Evaluationskomponenten vorgestellt. Schließlich wurde im letzten Kapitel anhand von Anwendungsfällen demonstriert, wie eine geschickte Verwendung des interaktiven Systems aussehen kann.

6.1 Ausblick

Das Ziel dieser Arbeit war die Entwicklung eines interaktiven Systems, dass die Untersuchung der Ergebnisse des FALCC-Algorithmus ermöglicht. FALCC-IS sollte im Besonderen Werkzeuge zur Fairness-Analyse und Erklärbarkeit enthalten. Es wurden bereits umfangreiche Möglichkeiten implementiert, aber in den folgenden Bereichen könnten in zukünftigen Arbeiten noch weitere Komponenten ergänzt werden:

Lokale Fairness

FALCC-IS bietet mit dem Datensatzstreudiagramm eine Möglichkeit die Fairness zu untersuchen. Das ist nicht nur für die im FALCC-Algorithmus erstellten Regionen möglich, sondern auch für beliebige Regionen. Allerdings ist das Auswerten der Fairness in beliebigen Regionen momentan nur sehr umständlich und manuell möglich. Ein sinnvolles Werkzeug wäre daher, dass im Graph eine beliebige Fläche ausgewählt werden kann und die lokale Gruppenfairness-Metrik auf alle Punkte angewandt wird, die in dieser Fläche liegen. Dadurch erhielte FALCC-IS ein Werkzeug um den FALCC-Algorithmus für beliebige Cluster auszuwerten.

Deskriptive Statistik

Der Datensatz kann durch einen Dimensionreduktions-Algorithmus betrachtet werden. Um noch mehr Überblick und Informationen über die Daten zu erhalten, wäre es sinnvoll, weitere Kenngrößen des Datensatzes zu visualisieren. Interessant wären beispielsweise die Anzahl an positiven Labels und der Median eines Attributes. Ein weiteres nützliches Konzept,wäre die Möglichkeit, den Datensatz beliebig zu gruppieren. Es ist Hilfreich sich einige Werte nur für Punkte anzuschauen, die in einer ausgewählten Gruppe liegen. Zum Beispiel würde eine Untersuchung der Anzahl an positiven Labels von Frauen und Männern eine entscheidende Rolle bei der Erkennung von Bias im Datensatz spielen.

Erklärbarkeit

Auch im Bereich der erklärbaren KI gibt es viele Forschungsgebiete, die in dem System noch nicht vertreten sind. LIME (Local Interpretable Model-agnostic Explanations) [RSG16] erklärt die Vorhersage, indem es einfache und erklärbare Modelle in der Nähe des zu vorhersagenden Punktes erstellt, die die Vorhersage approximieren. Da das neue Modell für den Menschen verständlich ist, werden die Gründe für die Vorhersage deutlich. Weitere Methoden wie die SHAP (SHapley Additive exPlanations) Werte [LL17] versuchen die Relevanz aller Attribute auf die Vorhersagen zu quantifizieren. Einen ähnlichen Verwendungszweck bieten die "Dependency Plots", die schon im Abschnitt über das What-If Tool eingeführt wurden und visualisieren, wie sich die Veränderung eines Attributes auf die Vorhersage auswirkt.

Effizienz

Einige Berechnungen des Systems können durch bessere Algorithmen beschleunigt werden. Auch könnte die Anwendung von paralleler Ausführung die für das Trainieren der einzelnen Runs benötigte Zeit verkürzen. Zusätzlich ist die Verwendung eines effizienteren Algorithmus zur Berechnung der kontrafaktischen Erklärungen essentiell, um diese auch auf großen Datensätzen durchzuführen.

Falls mehrere Runs mit einer ähnliche Konfigurationen trainiert werden sollen, die sich nur im Balancierungsparameter λ unterscheiden, könnten die gleichen Submodelle für alle Runs verwendet werden. Nur die Modellbewertungsphase des FALCC-Algorithmus müsste für jeden Run individuell ausgeführt werden

Interaktive Modell-Ensembles

Ähnlich wie in [TLKT09] ist es für das FALCC-Framework sinnvoll, eine interaktive und visuelle Möglichkeit zu haben, das Ensemble Modell zu erstellen. Dabei könnte der Benutzer beliebige, bereits trainierte und faire Modelle als Submodelle für den FALCC-Algorithmus auswählen. Da die Funktionsweise des FALCC-Algorithmus unabhängig von den Submodellen ist, wäre ein solcher Ansatz durchaus realisierbar. In Kapitel 5 haben wir bereits gesehen, dass die Ergebnisse durch faire Submodelle verbessert werden können, dementsprechend wäre es interessant, diese Beobachtungen interaktiv zu bestätigen.

Literaturverzeichnis

[ASS21] F. Anowar, S. Sadaoui, B. Selim. „Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kpca, lda, mds, svd, lle, isomap, le, ica, t-sne)“. In: *Computer Science Review* 40 (2021), S. 100378 (zitiert auf S. 23, 35).

[BCHC09] J. Benesty, J. Chen, Y. Huang, I. Cohen. *Noise reduction in speech processing*. Bd. 2. Springer Science & Business Media, 2009 (zitiert auf S. 20).

[BDE+20] S. Bird, M. Dudík, R. Edgar, B. Horn, R. Lutz, V. Milan, M. Sameki, H. Wallach, K. Walker. *Fairlearn: A toolkit for assessing and improving fairness in AI*. Techn. Ber. MSR-TR-2020-32. Microsoft, Mai 2020. URL: https://www.microsoft.com/en-us/research/publication/fairlearn-a-toolkit-for-assessing-and-improving-fairness-in-ai/ (zitiert auf S. 25, 27–29).

[BDH+18] R. K. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilovic et al. „AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias“. In: *arXiv preprint arXiv:1810.01943* (2018) (zitiert auf S. 25, 27–29).

[BHJ+21] R. Berk, H. Heidari, S. Jabbari, M. Kearns, A. Roth. „Fairness in criminal justice risk assessments: The state of the art“. In: *Sociological Methods & Research* 50.1 (2021), S. 3–44 (zitiert auf S. 17).

[CMJK20] A. Chatzimparmpas, R. M. Martins, I. Jusufi, A. Kerren. „A survey of surveys on the use of visualization for interpreting machine learning models“. In: *Information Visualization* 19.3 (2020), S. 207–233 (zitiert auf S. 13).

[CMM21] J. Chakraborty, S. Majumder, T. Menzies. „Bias in machine learning software: why? how? what to do?“ In: *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ESEC/FSE 2021. Athens, Greece: Association for Computing Machinery, 2021, S. 429–440. ISBN: 9781450385626. DOI: 10.1145/3468264.3468537. URL: https://doi.org/10.1145/3468264.3468537 (zitiert auf S. 19).

[DK17] F. Doshi-Velez, B. Kim. „Towards a rigorous science of interpretable machine learning“. In: *arXiv preprint arXiv:1702.08608* (2017) (zitiert auf S. 24).

[DK18] J. J. Dudley, P. O. Kristensson. „A Review of User Interface Design for Interactive Machine Learning“. In: *ACM Trans. Interact. Intell. Syst.* 8.2 (Juni 2018). ISSN: 2160-6455. DOI: 10.1145/3185517. URL: https://doi.org/10.1145/3185517 (zitiert auf S. 23).

[DNAA23] F. Di Carlo, N. Nezami, H. Anahideh, A. Asudeh. „FairPilot: An Explorative System for Hyperparameter Tuning through the Lens of Fairness“. In: *arXiv preprint arXiv:2304.04679* (2023) (zitiert auf S. 27–29).

[FBS20] M. Fritz, M. Behringer, H. Schwarz. „LOG-means: efficiently estimating the number of clusters in large datasets“. In: *Proceedings of the VLDB Endowment* 13.12 (2020), S. 2118–2131 (zitiert auf S. 20).

[GA19] D. Gunning, D. Aha. „DARPA's explainable artificial intelligence (XAI) program“. In: *AI magazine* 40.2 (2019), S. 44–58 (zitiert auf S. 12, 13).

[GPM22] P. A. Grabowicz, N. Perello, A. Mishra. „Marrying fairness and explainability in supervised learning“. In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. 2022, S. 1905–1916 (zitiert auf S. 19).

[HHC+19] F. Hohman, A. Head, R. Caruana, R. DeLIne, S. Drucker. „Gamut: A Design Probe to Understand How Data Scientists Understand Machine Learning Models“. In: *SIGCHI*. ACM. Mai 2019. URL: https://www.microsoft.com/en-us/research/publication/gamut-a-design-probe-to-understand-howdata-scientists-understand-machine-learning-models/ (zitiert auf S. 27, 29).

[KLTH10] A. Kapoor, B. Lee, D. Tan, E. Horvitz. „Interactive optimization for steering machine classification“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '10. Atlanta, Georgia, USA: Association for Computing Machinery, 2010, S. 1343–1352. ISBN: 9781605589299. DOI: 10.1145/1753326.1753529. URL: https://doi.org/10.1145/1753326.1753529 (zitiert auf S. 25).

[Läs23] N. Lässig. „Towards an AutoML System for Fair Classifications“. In: *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. 2023, S. 3913–3917. DOI: 10.1109/ICDE55515.2023.00380 (zitiert auf S. 3, 18).

[LH24] N. Lässig, M. Herschel. „FALCC: Efficiently performing locally fair and accurate“. In: (2024) (zitiert auf S. 11, 14, 15, 18, 19, 32, 47).

[LI16] K. Lum, W. Isaac. „To predict and serve?“ In: *Significance* 13.5 (2016), S. 14–19 (zitiert auf S. 11).

[LL17] S. M. Lundberg, S.-I. Lee. „A unified approach to interpreting model predictions“. In: *Advances in neural information processing systems* 30 (2017) (zitiert auf S. 52).

[LOH22] N. Lässig, S. Oppold, M. Herschel. „Metrics and algorithms for locally fair and accurate classifications using ensembles“. In: *Datenbank-Spektrum* 22.1 (2022), S. 23–43 (zitiert auf S. 17, 18).

[LT19] A. Lambrecht, C. Tucker. „Algorithmic bias? An empirical study of apparent gender-based discrimination in the display of STEM career ads“. In: *Management science* 65.7 (2019), S. 2966–2981 (zitiert auf S. 11).

[Mil15] C. C. Miller. „Can an algorithm hire better than a human“. In: *The New York Times* 25 (2015) (zitiert auf S. 12).

[MMS+21] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, A. Galstyan. „A Survey on Bias and Fairness in Machine Learning“. In: *ACM Comput. Surv.* 54.6 (Juli 2021). ISSN: 0360-0300. DOI: 10.1145/3457607. URL: https://doi.org/10.1145/3457607 (zitiert auf S. 15, 17).

[Nie23] O. Nies. „Maximierung der Modell-Diversität in Modell-Ensembles für lokal faire Klassifikationen“. B.S. thesis. 2023 (zitiert auf S. 19).

[PS22] D. Pessach, E. Shmueli. „A Review on Fairness in Machine Learning“. In: *ACM Comput. Surv.* 55.3 (Feb. 2022). ISSN: 0360-0300. DOI: 10.1145/3494672. URL: https://doi.org/10.1145/3494672 (zitiert auf S. 17, 39).

[Red09] M. Redmond. *Communities and Crime*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C53W3X. 2009 (zitiert auf S. 33).

[RSG16] M. T. Ribeiro, S. Singh, C. Guestrin. „"Why should i trust you?Ëxplaining the predictions of any classifier“. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, S. 1135–1144 (zitiert auf S. 52).

[Tho53] R. L. Thorndike. „Who belongs in the family?“ In: *Psychometrika* 18.4 (1953), S. 267–276 (zitiert auf S. 20).

[TLKT09] J. Talbot, B. Lee, A. Kapoor, D. S. Tan. „EnsembleMatrix: Interactive Visualization to Support Machine Learning with Multiple Classifiers“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’09. Boston, MA, USA: Association for Computing Machinery, 2009, S. 1283–1292. ISBN: 9781605582467. DOI: 10.1145/1518701.1518895. URL: https://doi.org/10.1145/1518701.1518895 (zitiert auf S. 24, 52).

[VV11] S. Van Den Elzen, J. J. Van Wijk. „Baobabview: Interactive construction and analysis of decision trees“. In: *2011 IEEE conference on visual analytics science and technology (VAST)*. IEEE. 2011, S. 151–160 (zitiert auf S. 24, 27, 29).

[WPB+19] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. Viégas, J. Wilson. „The what-if tool: Interactive probing of machine learning models“. In: *IEEE transactions on visualization and computer graphics* 26.1 (2019), S. 56–65 (zitiert auf S. 23–25, 27, 28).

[ZWLC19] X. Zhao, Y. Wu, D. L. Lee, W. Cui. „iForest: Interpreting Random Forests via Visual Analytics“. In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2019), S. 407–416. DOI: 10.1109/TVCG.2018.2864475 (zitiert auf S. 23, 27, 29).

[ZWS+13] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, C. Dwork. „Learning fair representations“. In: *International conference on machine learning*. PMLR. 2013, S. 325–333 (zitiert auf S. 17, 19).

Alle URLs wurden zuletzt am 19. 03. 2024 geprüft.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift