UNIVERSITY OF SALERNO

Department of Computer Science

Master's Degree in Computer Science

MASTER'S THESIS

Genetic Algorithm-Driven Fairness Enhancement in Machine Learning: The Fair-Train Methodology

SUPERVISOR
Prof. Fabio Palomba
Dr. Gianmario Voria
University of Salerno

CANDIDATE
Rebecca Di Matteo
Student ID: 0522501510

https://GeneticAlgorithmForFairness.git

Sogna, ragazzo sogna quando sale il vento nelle vie del cuore,
quando un uomo vive per le sue parole o non vive più;
sogna, ragazzo sogna,
non lasciarlo solo contro questo mondo
non lasciarlo andare sogna fino in fondo,
fallo pure tu...
Sogna, ragazzo sogna quando cala il vento ma non è finita
quando muore un uomo per la stessa vita che sognavi tu
Sogna, ragazzo sogna
non cambiare un verso della tua canzone,
non lasciare un treno fermo alla stazione,
non fermarti tu...

Abstract

This thesis presents Fair-Train, an innovative method based on genetic algorithms, developed to address the issue of fairness in artificial intelligence (AI) systems. As the adoption of AI systems in critical areas like hiring and loan approvals has increased, there is a growing need to ensure that these systems operate without introducing biases or discrimination. Fair-Train aims to optimize fairness in both datasets and machine learning models through advanced data preprocessing techniques and model tuning. The method employs preprocessing techniques, such as OneHot Encoding, Standard Scaling, and MinMax Scaling, to ensure that data is representative and balanced, thereby reducing the likelihood of bias. Additionally, Fair-Train implements a genetic algorithm to select the best configurations for models, balancing accuracy and fairness. This algorithm performs selection, crossover, and mutation operations on a population of solutions, evaluating each individual based on performance and fairness metrics, such as the Disparate Impact Ratio and Demographic Parity. The method's validity was demonstrated through an extensive set of experiments, comparing Fair-Train with existing preprocessing techniques like FairSMOTE and Reweighing. The results indicate that Fair-Train offers significant improvements in both accuracy and fairness over traditional methodologies. Moreover, the research discusses the implications of genetic algorithm parameterization on the model's efficiency and sustainability. Fair-Train represents a significant advancement in the field of AI fairness, providing a practical and effective framework for addressing systemic biases in machine learning models. The thesis contributes not only to the academic debate but also offers concrete solutions for the industry, aiming to develop more just and responsible AI systems, which are crucial for a fair and inclusive society.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Context and Motivation

In recent decades, artificial intelligence (AI) has emerged as one of the most transformative technologies of our time. From medical diagnostics to industrial automation, and from recommendation systems to autonomous driving, AI is permeating every aspect of daily life. However, with the increasing adoption of AI systems, new ethical and social challenges have arisen, including the issue of fairness. Fairness, in this context, refers to the ability of AI algorithms to operate impartially, avoiding the introduction or amplification of discrimination against individuals or social groups.

The problem of bias in AI systems is not new, but it has become increasingly relevant with the growing use of these systems in critical decisions, such as hiring, lending, criminal justice, and access to healthcare services. These biases can stem from various sources, including non-representative training data, algorithms that replicate historical prejudices, or evaluation methodologies that inadequately consider data diversity. The lack of fairness not only jeopardizes the integrity of AI systems but can also perpetuate existing social injustices, amplifying inequalities.

1.2 Importance of Fairness in AI

Ensuring fairness in AI systems is essential to building a more just and inclusive society. The ability of these systems to influence significant decisions makes their ethical and responsible development crucial. Integrating principles of fairness is not just a technical issue but also an ethical and social imperative. An AI system that operates with bias can, for instance, deny opportunities to people from certain ethnic groups, genders, or socioeconomic backgrounds, based on prejudiced historical data. Therefore, the scientific community and industry must work together to develop and implement methodologies that promote fairness.

In this context, implementing fairness techniques has become a priority. These techniques include the use of preprocessing algorithms to balance datasets, the integration of equity constraints during model training (inprocessing), and the application of post-training adjustments (postprocessing). Each of these phases requires thorough analysis to ensure that AI systems do not perpetuate existing biases.

1.3 Research Objectives

This work aims to explore and improve existing methodologies for ensuring fairness in AI systems. Specifically, it introduces Fair-Train, an innovative approach based on genetic algorithms designed to optimize equity in both datasets and machine learning models. The specific objectives of the research include:

- Developing a framework for integrating fairness techniques into the AI model lifecycle, including preprocessing, inprocessing, and postprocessing.
- Evaluating the effectiveness of Fair-Train through an extensive series of experiments, comparing its performance with other existing bias mitigation and fairness optimization techniques.

1.4 Research Methodology and Main Results

The research methodology adopted in this thesis includes a combination of literature review, theoretical development, practical implementation, and empirical evaluation. In the initial phase, a critical review of existing literature was conducted to identify the main challenges and fairness techniques. Subsequently, the Fair-Train framework was developed, implemented, and tested on various datasets. The experiments were designed to evaluate the impact of the proposed techniques on the accuracy and fairness of AI models, using specific metrics such as the disparate impact ratio and demographic parity.

1.4.1 Main Results:

The results obtained from this research can be summarized as follows:

Accuracy Improvement: The Fair-Train algorithm demonstrated a notable improvement in model accuracy compared to traditional preprocessing techniques such as FairSMOTE, Reweighing, and Disparate Impact Remover. This suggests that the genetic optimization approach used by Fair-Train enhances the predictive capabilities of machine learning models, leading to more precise and reliable predictions.

Increased Fairness: Fairness metrics, including the Disparate Impact Ratio and Equal Opportunity Difference, indicated that Fair-Train provides a fairer treatment of different groups compared to other methods. The algorithm achieved a higher degree of fairness, which is critical in ensuring that machine learning models do not perpetuate or amplify biases present in the training data.

Efficiency and Practicality: Although the Fair-Train algorithm showed a slightly higher execution time compared to other techniques, it remains efficient and practical for real-world applications. The trade-off between increased execution time and improvements in accuracy and fairness metrics is justified by the substantial benefits that Fair-Train offers in terms of overall model performance.

These findings underscore the effectiveness of the Fair-Train methodology in not only improving model accuracy but also promoting fairness in AI systems, making it a valuable tool for developing more equitable AI applications.

1.5 Structure of the Thesis

The document is structured as follows:

Chapter 1: Introduction - Introduces the research context, motivation, objectives, and structure of the thesis.

Chapter 2: State of the Art - Discusses the main techniques and challenges related to fairness in AI, providing a theoretical foundation for the research.

Chapter 3: Fair-Train: A New Algorithm for Context-Aware Fair Training Optimization - Describes the architecture of the proposed framework, the optimization techniques used, and the logic of the implemented genetic algorithm.

Chapter 4: Fair-Train Evaluation - Presents the results of experiments conducted to evaluate the effectiveness of Fair-Train, with detailed analyses on performance and fairness metrics.

Chapter 5: Threats to Validity - Discusses potential threats to the internal and external validity of the study, analyzing the reliability of the results obtained.

Chapter 6: Conclusions - Summarizes the main contributions of the research, highlights the study's limitations, and suggests future research directions in the field of AI fairness.

In an era where AI systems play an increasingly central role in society's critical decisions, it is essential that these systems are developed and implemented in an equitable and responsible manner. The research presented in this thesis aims to contribute to this goal by providing new methodologies and tools to improve fairness in AI models. The Fair-Train framework represents a step forward in this direction, offering practical solutions to address one of the most pressing issues in modern AI.

CHAPTER 2

State of the Art

2.1 Fairness

Fairness in the realm of Artificial Intelligence (AI) represents a foundational principle aimed at ensuring that algorithms, models, and AI systems operate impartially, avoiding the generation of discrimination or biases against individual people or social groups. This notion is based on the idea that decisions made by AI systems should be fair, ethically correct, and free from any biases, i.e., distortions that could negatively impact individuals based on protected personal characteristics such as ethnicity, gender, age, sexual orientation, religion, or other identity attributes. Promoting fairness in AI requires careful identification and mitigation of biases that can be found in datasets used for training, in the development processes of algorithms, and in the methodologies for evaluating and implementing models. These biases originate from various sources, including non-representative data collection, historical and cultural prejudices, or subjective labeling practices. The presence of bias in datasets can lead to unfair and discriminatory decisions, causing a negative impact on individuals and vulnerable groups when AI models are employed in real-world scenarios [1].
A concrete example of bias in AI can be observed in facial recognition systems, which struggle to correctly identify individuals of certain ethnicities due to insufficient

representation in training datasets, or hiring algorithms that favor candidates of a certain gender or background at the expense of others, based on prejudicial historical data [2]. The challenge in creating fair AI systems thus lies not only in programming techniques or the algorithm itself but also in the selection and preparation of the data used for training. To address these challenges, advanced machine learning and data science techniques have been developed, such as dataset rebalancing, fair learning, and algorithmic impact analysis. These methodologies aim to identify and correct biases in datasets and AI models, ensuring that decisions made are fairer and less prone to discrimination. Furthermore, the adoption of ethical frameworks and guidelines for the responsible development of AI, including principles of transparency, accountability, and privacy, is essential for building AI systems that are not only technologically advanced but also socially just. This multidisciplinary approach, involving experts in ethics, law, sociology, as well as computer science and engineering, is crucial to ensure that artificial intelligence serves the common good and respects the rights and fundamental freedoms of all individuals [3].

2.2 Mitigation Techniques

It is crucial to examine how discrimination mitigation techniques and the promotion of fairness have become pivotal in the field of AI. These techniques stem from the need to address and reduce biases in data and machine learning models. With the increased use of AI systems in sensitive areas, such as employment, criminal justice, and healthcare, it becomes important to ensure equitable decisions, as mistakes in these areas can have dramatic effects on human lives. The rise in biases within AI systems has prompted research into effective methods for their identification, understanding, and mitigation [4]. In particular, mitigation techniques categorized into preprocessing, inprocessing, and postprocessing are applied at distinct stages of the model lifecycle, from data treatment before training, intervention during the training phase, to the adjustment of the model's outputs. This analysis lays the foundation for the development of AI systems that are not only technologically advanced but also ethically responsible, emphasizing the importance of a comprehensive approach in managing biases [4].

2.2.1 Preprocessing

Preprocessing techniques focus on manipulating data before it is used to train a model. The goal is to reduce or eliminate biases in input data, for example, through Data Balancing techniques, modifying or removing sensitive features, or generating new synthetic data [4]. Suppose we have a dataset used to train a machine learning model that predicts the likelihood of obtaining a loan. The dataset shows a strong imbalance: 70% of the examples belong to individuals of a certain demographic group (Group A), while the remaining 30% belongs to individuals of another demographic group (Group B). Preliminary analysis shows that the model trained on this dataset tends to unfairly favor individuals from Group A in loan granting. To mitigate this bias, we decide to apply a data balancing technique through the oversampling of Group B. We use an approach like SMOTE (Synthetic Minority Over-sampling Technique), which creates synthetic examples of Group B based on existing data, to increase their representation in the dataset to achieve a 50:50 balance with Group A. After applying SMOTE, the balanced dataset is used to train a new model. The analysis of the results shows that the model trained with the balanced dataset makes fairer decisions, significantly reducing the bias against Group B in loan granting.

2.2.2 Inprocessing

Inprocessing techniques are applied directly in the model training process, integrating fairness constraints or objectives. The idea is to guide the model's learning so that it respects specific fairness criteria, thereby reducing the likelihood that the model learns to reproduce biases present in the training data [4]. Imagine developing an artificial intelligence model for selecting candidates for a job offer. The initial model, trained without equity considerations, shows a tendency to recommend candidates of a certain gender for technical positions, reflecting the biases present in the training data. To counter this problem, we decide to apply an inprocessing technique that incorporates fairness constraints in the model's training. In particular, we adopt a constraint-based optimization approach, which requires the model to minimize the discrepancy in recommendation rates between genders for the job position, while still maintaining high predictive performance. To implement this

approach, we modify the learning algorithm's objective function, including a penalty term that measures the disparity in recommendation rates between genders. The algorithm is then optimized to balance between maximizing the accuracy of recommendations and minimizing the disparity in treatment between genders. After training, the updated model demonstrates not only to maintain high accuracy in predicting suitable candidates but also to recommend candidates of different genders for the technical position in a more balanced manner.

2.2.3 Postprocessing

Postprocessing techniques are applied after the model has been trained, modifying its predictions or decisions to ensure that they meet certain fairness standards. This can include adjusting decision thresholds for protected groups or using calibration techniques to ensure equitable treatment [4]. Consider a machine learning model used by a bank to predict the likelihood that a customer will repay a loan. After training, it is observed that the model has a significantly higher false rejection rate for applicants belonging to a particular ethnic group, indicating a potential bias against that group. To address this problem, we decide to apply a postprocessing technique that assists the model's decisions in improving fairness. Specifically, we analyze the distribution of repayment probabilities predicted by the model and apply a correction to the decision thresholds for the underrepresented group. This means that, for this group, a slightly lower credit score could still result in loan approval, provided the risk remains within acceptable limits for the bank. We implement a decision adjustment by calculating a new decision threshold that balances approval rates between groups, taking into account both fairness and credit risk. After applying this correction, the data show that the false rejection rate for the previously disadvantaged group significantly reduces, leading to a more equitable distribution of loan decisions.

2.3 Data Preparation Practices in Literature

Data Preparation practices are a crucial aspect in the creation and management of machine learning (ML) models, aiming to improve fairness and data representativeness [5]. Data Preparation encompasses a series of techniques aimed at enhancing the quality and reliability of datasets used for training ML models, including data cleaning, integration, transformation, and dimensionality reduction. These techniques are crucial to prevent and mitigate biases in data, ensuring that the produced ML models are fair and non-discriminatory [5]. The literature highlights the importance of a thorough data preparation phase within the ML lifecycle to promote the development of ethical and responsible ML solutions. These practices include:

- **Data Balancing Techniques:** This practice includes essential strategies to improve fairness in ML-intensive systems, using methods like oversampling, undersampling, uniform or preferential sampling, data filtering, and labeling [5]. These strategies aim to balance datasets, reducing disparities and sources of discrimination, and are particularly relevant when dealing with data that present significant imbalances between classes or among represented groups [6]. This technique is rated from medium to high applicability, with an expected impact and effort required considered from medium to high, based on the outcome of a systematic analysis involving experts and practitioners in the ML field. By medium to high applicability, we mean that data balancing techniques are considered relevant and potentially useful across a broad spectrum of ML system design and implementation scenarios. By medium to high expected impact, we mean that these techniques are anticipated to have a significant positive effect on the fairness of ML systems [6].

- **Data Mining Approaches:** This practice involves using data mining approaches to find discrimination in datasets, such as discriminatory historical decisions or non-predefined minority groups, or to analyze the semantic meaning of data. Data mining approaches entail using statistical techniques, machine learning algorithms, and analytical methods to explore and analyze large data sets to find patterns, correlations, and insights that wouldn't be apparent through

traditional analysis alone [5]. This practice is considered frequently applicable and requires a medium to high effort for its application, based on the outcome of a systematic analysis involving experts and practitioners in the ML field. By frequently applicable, we mean that data mining techniques are considered relevant and potentially useful in various scenarios. By medium to high expected impact, we mean that these techniques are anticipated to have a significant positive effect on the fairness of ML-intensive systems [7].

- **Data & Features Transformation:** This practice may include techniques such as dimensionality reduction, probabilistic transformation, matrix factorization, or data imputation. This practice is considered moderately applicable and requires a medium effort, having a medium positive impact on ML system fairness [5]. The evaluation is based on the outcome of a systematic analysis involving experts and practitioners in the ML field. By moderately applicable, we mean that data and features transformation techniques are feasible in most project contexts. By medium effort, we mean that it is neither particularly easy nor overly complex to implement [8].

- **Diversity Dataset Selection:** The goal of this practice is to select datasets that present a wide range of diversity and similarity among data and features, ensuring an equitable and comprehensive representation of various sensitive groups. This approach focuses particularly on those characteristics that may influence the fairness of ML models, such as age, gender, ethnicity, or other demographic or social attributes. This practice is rated as moderately applicable, with a very positive impact on fairness [5]. The evaluation is based on the outcome of a systematic analysis involving experts and practitioners in the ML field. By moderately applicable, we mean that diversity data selection techniques are feasible in most project contexts. By very positive impact, we mean that despite challenges related to its applicability, when this practice is implemented, it is believed to have a significant and very positive effect in promoting fairness in ML models [5].

- **Causal Analysis Approaches:** This practice uses causal analysis techniques, including causal graphs, to understand and address the roots of discrimination present in datasets. This method relies on the theory and tools of causal inference to distinguish between correlations and actual causes, helping to identify the specific mechanisms through which biases and discrimination can manifest. It requires a medium to high effort for application and has a generally positive impact on fairness [5]. The evaluation is based on the outcome of a systematic analysis involving experts and practitioners in the ML field. By medium to high effort for applicability, we mean it reflects the complexity in building and interpreting causal models that may require specialized knowledge in statistics, probability, and ML, in addition to the need for detailed and often difficult-to-obtain data [9]. The variation from "medium" to "high" acknowledges that the actual effort required can vary based on the specifics of the context, the quality of available data, and the specific goals of the project. By generally positive impact, we mean that despite the effort required, causal analysis is recognized for its potential to significantly improve the fairness of ML systems [9].

- **Data Fairness Measurement:** This practice involves using various strategies and techniques to measure data fairness, such as data fairness metrics and individual weight calculations, to monitor, analyze, or assess data fairness levels before training the model. This process takes into consideration various quality constraints, such as fairness and privacy. This practice is considered applicable with variable frequencies and has a high impact on fairness [5]. The evaluation is based on the outcome of a systematic analysis involving experts and practitioners in the ML field. By applicability with variable frequencies, we mean that the use and implementation of data fairness measurement practice can vary significantly depending on the specific project context, available resources, the nature of the data, and the equity objectives pursued. By high impact, we mean that regardless of the frequency of application, when the data fairness measurement practice is implemented, it is believed to have a significantly positive impact on promoting fairness within ML-based systems [5].

- **Multitask Learning Improvement:** This practice focuses on using multitask learning strategies, aimed at maximizing average accuracy for each minority group present in the training sample. This approach is based on the idea that by simultaneously treating multiple learning tasks (or tasks), the generalization and effectiveness of ML models can be improved, especially in contexts where the representation of minority groups in the data may be scarce or unevenly distributed [5]. This technique is seen as prevalent in practice with a significant impact on fairness, albeit requiring a high effort. The evaluation is based on the outcome of a systematic analysis involving experts and practitioners in the ML field. By prevalent in practice, we mean that the use of multitask learning strategies to improve fairness is relatively common or recognized as a valid approach within the community dealing with ML and fairness [10]. By significant impact on fairness, we mean that the application of multitask learning techniques are considered effective in improving the fairness of ML models. This means that, when implemented correctly, these strategies can lead to fairer outcomes, increasing accuracy for minority groups and reducing the risk of bias in models [10].

2.4 Fairness Testing

Fairness Testing is an essential methodology in software engineering, aimed at ensuring that applications and automated systems operate without unfair discrimination towards groups or individuals based on protected characteristics such as race, gender, age, and others. This type of testing seeks to identify and correct biases in data, algorithms, and decisions made by the software, which can lead to discriminatory or inequitable outcomes. To conduct effective fairness testing, it is necessary to first clearly define what is meant by "Fairness" in the specific context of the system being tested, as fairness can have different meanings depending on the application and its social implications. Subsequently, the process involves generating and executing test suites that systematically vary inputs based on protected characteristics, to observe the effect on outputs and decisions. The goal is to detect scenarios in which changing a protected characteristic significantly alters the output, indicating

potential discrimination. Once discriminations are identified, it is crucial to evaluate their extent and implement strategies to mitigate them, which may include revising algorithms, adjusting datasets, or modifying decision-making processes. Finally, continuous monitoring ensures that the system remains fair over time, adapting to new data and contexts. Through fairness testing, developers can contribute to the creation of more equitable and inclusive technologies, reducing the risk of perpetuating or amplifying existing inequalities in society [11]. A classic example of fairness testing could be applied in a bank loan recommendation system. In this scenario, the system evaluates loan applications based on various factors, including income, credit history, and employment. To conduct fairness testing, pairs of virtually identical loan applications could be generated, differing only in a protected characteristic, such as the applicant's gender. If changing only the gender from male to female (or vice versa) in an identical application results in a significant variation in the loan approval probability, this would indicate potential gender discrimination in the system. After identifying such discriminations, it is crucial to intervene with corrective measures, which could include recalibrating algorithms or reviewing evaluation criteria. Continuous monitoring then ensures that the system maintains fair practices over time [11].

2.4.1 Fairness Testing Techniques

Fairness testing techniques aim to evaluate and ensure that machine learning (ML) models behave equitably and non-discriminatorily, taking into account the sensitive characteristics of individuals. Unfair discrimination by models can manifest as individual discrimination, where the model predicts different outcomes for similar instances except for the protected attribute, or as group discrimination, favoring or discriminating against instances belonging to specific groups compared to others [11]. Research has proposed various fairness testing techniques including:

- **Themis:** this technique in the field of fairness testing is designed to identify and measure discrimination in machine learning models through a testing-based approach. By automatically generating efficient test suites, it allows for the assessment of discrimination without the need for a specific test oracle. This

capability facilitates the identification of discriminatory behaviors in software used in various contexts, such as criminal sentence recommendation systems, access to financial products management, or selection for promotions. The distinctive feature of this methodology is its focus on the causality of discriminatory behaviors, making it effective even where advanced techniques to eliminate discrimination fail. Being a critical tool in software development, especially in areas at risk of discrimination, its ability to produce targeted test suites is fundamental. A concrete example of Themis' use could involve a system used by a bank to approve or deny loan applications. Themis could be used to generate a series of simulated loan applications, systematically varying attributes such as age, gender, and ethnicity, while keeping other financial factors constant. By analyzing the system's decisions on these simulated requests, it could identify whether the system has a higher likelihood of denying loans to applicants of a certain gender or belonging to specific ethnic groups, highlighting potential biases [12].

- **Aeqitas:** this fairness testing technique provides a means to validate and improve the fairness of machine learning models by relying on a probabilistic search strategy aimed at uncovering discriminatory inputs, i.e., combinations of input data that reveal biases in the model's outcomes based on sensitive attributes such as gender or ethnicity. It is based on three main strategies: a global search to identify potentially discriminatory inputs, a local search to explore the vicinity of such inputs and estimate the extent of discrimination, and a retraining module that uses the found discriminatory inputs to guide the model towards greater fairness. A practical example of applying Aeqitas could involve a model used for selecting candidates for employment, where the system might, unintentionally, favor candidates of a certain gender over others. By applying this technique, one could systematically generate and test inputs that vary only by the candidates' gender, thus discovering if and how the model discriminates. Inputs that highlight such discrimination would then be used to retrain the model, with the goal of eliminating biases and ensuring that future decisions are equitably balanced [13].

- **Black Box Fairness Testing:** this technique combines Dynamic Symbolic Execution (DSE) and Local Explainability to identify individual discrimination in machine learning models treated as black-box systems. This approach begins with the selection of initial inputs, or seeds, which can be existing data or randomly generated. Local explainability is then applied to create an interpretable approximate model of the ML model's behavior near these seed inputs, typically through tools like LIME (Local Interpretable Model-agnostic Explanations). This approximate model reveals which attributes most significantly influence the model's decisions for the specific input. Through dynamic symbolic execution, new test inputs are generated by modifying unprotected attributes (such as income or credit history) based on the approximate model, to satisfy or violate specific constraints. This process systematically explores alternative decision-making paths around the seed input, generating inputs that are evaluated on the original ML model to detect potential discrimination cases. For example, by changing the "gender" attribute from female to male in a seed input and observing a change in the model's decision, gender-based discrimination can be revealed. The innovative aspect of this technique lies in its ability to systematically probe the decision space of an ML model, identifying discriminations that might otherwise remain hidden, all without requiring direct access to the model's internal structure. This methodology not only aids in improving the transparency and fairness of AI systems but also provides a replicable framework for fairness testing in machine learning models across various application contexts [14].

- **White-box Fairness Testing through Adversarial Sampling:** this technique introduces an innovative method to address the challenge of fairness in deep neural networks (DNNs), which is particularly relevant for applications with significant social impact. This approach, based on the use of adversarial sampling, aims to identify and mitigate potential discriminations encapsulated in the decisions of DNNs. Unlike traditional methods, which may be limited by their ability to effectively explore the input space or by their computational heaviness, the proposed technique utilizes lightweight procedures, such as

gradient computation and clustering, to detect individual discriminatory instances. Through a two-phase process, which includes global generation to find instances close to the decision boundary and local generation to search for additional discriminatory instances near already identified cases, this method stands out for its effectiveness and efficiency. Experimental results highlight how this technique not only outperforms existing strategies in discovering discriminations but also does so in significantly shorter times, making it a promising proposal for improving the fairness of DNNs across a wide range of applications [15].

- **FlipTest:** is a technique designed to detect discrimination in classifiers, inspired by the intuitive question: **"if an individual had a different protected status, would they be treated differently by the model?"** Unlike methods relying on causal information, FlipTest utilizes the concept of optimal transport to create optimal mappings between individuals from different protected groups, allowing comparison of how the model treats similar individuals differing only in protected status. This method identifies the "flipset," a set of individuals for whom the classifier's output changes following the optimal mapping, indicating potential discriminations based on group membership. Optimal transport, in the context of FlipTest, minimizes a given cost defined over the feature space, transforming one probability distribution into another. This approach ensures that the generated inputs are representative of the model's typical behavior, avoiding testing the model on unrealistic or out-of-distribution inputs. This technique not only detects if discrimination exists but through the "transparency ratio," it also highlights the features contributing most to the discrimination detected in the flipset. This allows model developers and auditors to better understand discrimination dynamics and intervene to mitigate them [16].

- **FairRec:** this technique introduces an approach for fairness testing in deep learning-based recommendation systems, based on the Double-Ended Discrete Particle Swarm Optimization (DPSO) algorithm, which allows for effective and targeted exploration of the user group space, identifying those receiving potentially unfair recommendations compared to others. FairRec is divided

into three main modules, which together provide a comprehensive solution for fairness testing. The first module handles the preparation and configuration of necessary data, including user details, items to recommend, and the recommendation models themselves. Users can configure this module to specify sensitive attributes to consider (such as gender, age) and select fairness metrics of interest, allowing deep customization of the testing process. Subsequently, the testing module leverages the DPSO algorithm to analyze the recommendation system. This step is crucial for identifying disadvantaged user groups, i.e., those groups that, due to their characteristics or interaction patterns, receive less relevant or advantageous recommendations. The analysis focuses on a series of specific metrics for recommendation systems, such as model utility, diversity, and popularity of recommended items, to ensure a comprehensive evaluation of recommendation fairness. Finally, the results visualization module compiles and presents a detailed report based on the testing results. This report not only highlights areas of potential inequality in the recommendation system but also provides crucial insights into specific groups that might be most affected. This transparency is essential for guiding developers in understanding fairness dynamics within their systems and implementing effective mitigation strategies [17].

Name	Paper Reference	Technique Used	Brief Description
Themis	[12]	Testing-based approach	This method employs a novel, testing-based approach to uncover and quantify discriminatory practices within machine learning models, by creating efficient and targeted test suites. Specifically focusing on the underlying causes of discriminatory behaviors, Themis is adept at revealing biases in applications prone to discrimination risks, such as systems recommending criminal sentences or managing loan approvals. Its strategic test suite generation simulates diverse scenarios, highlighting biases across different demographic groups, thereby fostering more equitable software solutions.
Aeqitas	[13]	Probabilistic search	Aeqitas advances fairness in machine learning models through a refined probabilistic search technique. By systematically identifying discriminatory inputs variables that lead to biased outcomes this method employs a comprehensive strategy involving both broad and nuanced local searches. This process is complemented by a retraining module that utilizes these insights to guide the model towards more equitable decisions. Aeqitas is especially useful in applications where sensitive attributes, like gender or ethnicity, can unduly influence outcomes, thereby enhancing fairness through actionable intelligence.
Black Box Fairness Testing	[14]	DSE and Local Explainability	This technique merges the power of Dynamic Symbolic Execution (DSE) with the insights of Local Explainability to probe and illuminate biases in machine learning models viewed as opaque entities. It starts with selecting initial inputs to create an interpretable model approximation, which then guides the generation of new test cases. These cases are crucial for uncovering discrimination by altering non-protected attributes and observing the outcomes, thereby ensuring models uphold fairness standards without needing access to their inner workings.
White-box Fairness Testing through Adversarial Sampling	[15]	Adversarial sampling	Leveraging the precision of adversarial sampling, this approach specifically targets the nuanced dynamics of discrimination within deep neural networks (DNNs). It identifies potential biases by exploring the model's decision boundaries, using lightweight computational methods to uncover and address discrimination efficiently. This technique shines in its ability to detect and correct fairness issues swiftly, making it a valuable tool for enhancing the integrity of socially impactful AI applications.
FlipTest	[16]	Optimal transport	By invoking the principle of optimal transport, FlipTest innovatively assesses models for discrimination by examining how changes in protected statuses affect outcomes. It creates mappings between similar individuals from different protected groups to determine if the model's treatment varies unjustly. This method not only identifies biases but also elucidates the contributing factors, offering clear pathways to mitigate these biases and foster fairness in classifier decisions.
FairRec	[17]	DPSO algorithm	Addressing the unique challenges of fairness in recommendation systems, FairRec utilizes the Double-Ended Discrete Particle Swarm Optimization (DPSO) algorithm. This approach enables a thorough exploration of user group dynamics, pinpointing groups that might receive less favorable recommendations. By analyzing and adjusting based on specific fairness metrics, FairRec provides essential insights for developers to rectify biases, ensuring that recommendation systems serve all users equitably.

Table 2.1: Overview of Fairness Testing Techniques

2.5 Limitations and Motivation

Despite significant advances in the field of software engineering (SE) related to fairness, several limitations and uncertainties persist:

- **Lack of Context-Specific Guidance:** There is a lack of comparative studies that determine the superiority of one bias mitigation method over another in specific contexts. This gap makes it challenging to optimally employ techniques like data preprocessing, in-processing during model training, and post-processing of predictions.
- **Diverse Fairness Testing Techniques:** Although multiple fairness testing techniques exist, there is no clear direction on how to select or effectively integrate them to ensure equity without compromising system performance. This uncertainty highlights the need for a refined approach in the application of fairness methods.
- **Complexity in Application:** Developers often find themselves navigating a complex environment, trying to balance the need to ensure the fairness of their systems with maintaining high performance. This complexity is exacerbated by the varied effectiveness of different techniques in diverse application contexts.
- **Lack of Intelligent Tools for Practice Selection:** There is a notable absence of intelligent tools that assist developers in selecting the most appropriate SE practices for their specific AI projects. Such tools could significantly ease the decision-making process, especially in integrating fairness into AI systems.

These limitations underscore the need for further research and development in this area. The proposed solution involves creating an algorithm that offers customized configurations of SE practices. This algorithm would analyze the distinctive features of each AI system—such as datasets, models used, and application contexts—to suggest the most suitable bias mitigation strategies and testing techniques. By employing machine learning methods and data analysis, the algorithm would assess the impact of various practices on fairness and performance, providing essential support to developers. The motivation for this proposal is driven by the current challenges in

SE for fairness. Developers require more structured and context-specific guidance to navigate the complexities of integrating fairness into AI systems. An intelligent algorithm offering data-based recommendations could revolutionize the way AI systems are developed and tested, enhancing both fairness and performance.

CHAPTER 3

Fair-Train: A New Training Optimization Algorithm with Context Awareness

In this chapter, we present Fair-Train, an algorithm designed to optimize fairness in datasets and machine learning models using a genetic algorithm (GA) based approach. The main objective of Fair-Train is to address biases in data and model predictions, ensuring fair outcomes across different demographic groups.

3.1 Algorithm's Ultimate Goal

The goal of Fair-Train is to develop a framework that optimizes both the performance and fairness of machine learning models. This is achieved through a combination of data preprocessing techniques and model optimizations designed to mitigate systemic biases and promote fairer and more impartial decisions.

3.2 Motivation for Choosing a Genetic Algorithm

The use of a genetic algorithm is motivated by its ability to effectively explore large solution spaces and find optimal configurations that balance performance and fairness. This approach is particularly useful for complex problems where optimal solutions are not easily identifiable through traditional optimization methods.

3.3 Expected Outcomes and Practical Example

We expect Fair-Train to improve both the accuracy and fairness of models compared to traditional preprocessing techniques. Consider, for example, a predictive model for loan approvals. In a real-world context, without the use of Fair-Train, the model might inadvertently favor one demographic group over others due to biases in the training data, such as a disproportionately high number of approvals for a certain ethnicity or gender.

3.3.1 Example

Suppose we have a loan application dataset that shows a strong bias towards a specific demographic group (e.g., individuals of a certain ethnicity), with significantly higher approval rates compared to other groups. This could be due to various factors, including historical data reflecting existing prejudices or misrepresentation of certain groups in the dataset. Using Fair-Train, the GA would explore preprocessing configurations that might include techniques such as balancing the dataset (e.g., oversampling underrepresented minorities) and adjusting class weights. Simultaneously, the genetic algorithm would optimize model parameters, such as decision thresholds, to ensure that predictions are fairly distributed across demographic groups. For instance, during the process, Fair-Train might identify that a combination of OneHot Encoding, Standard Scaling, and a specific set of model parameters leads to bias reduction. The final result would be a model that approves loans at a similar rate across different ethnic groups, correcting implicit biases present in the original data. This improvement would not only enhance the fairness of the model's decisions but could also increase user trust in the system, demonstrating a commitment to

equity. This example illustrates how Fair-Train can be applied to address biases in AI systems, optimizing model decisions to ensure they are fair and based solely on relevant and just criteria.

3.4 Fairness Optimization Techniques in Datasets

Data preprocessing is a fundamental phase to ensure that datasets are adequately prepared for training machine learning models. The preprocessing techniques used by Fair-Train include:

- **OneHot Encoding**: This technique converts categorical variables into a binary format interpretable by machine learning algorithms. Each category is transformed into a separate column containing binary values (0 or 1). This removes the implicit order of numerical categories, preventing biases stemming from ordinal categories. This is particularly important to ensure that models do not assign incorrect meanings to categories and treat all categories fairly.

- **Standard Scaling**: This method normalizes the numerical features of the dataset to have a mean of zero and a standard deviation of one. This improves convergence during training and ensures that all features contribute equally to the model. It is crucial for algorithms that depend on the scales of features, such as linear regression and neural networks, preventing one feature from dominating others due to its magnitude.

- **MinMax Scaling**: This technique scales numerical features to a specified range, typically between 0 and 1, maintaining the proportions between original values but compressing them into a smaller range. It is useful for algorithms sensitive to data scales, such as neural networks, ensuring that all features have the same impact on the model's outcomes.

- **Resampling**: To address class imbalance issues, Fair-Train uses resampling techniques such as oversampling minority classes and undersampling majority classes. This balances the dataset, improving model fairness and performance.

Oversampling increases the representation of minority classes, while undersampling reduces that of majority classes, preventing biases toward overrepresented classes and ensuring balanced representation of all classes.

- **Clustering**: The clustering algorithm, like KMeans, groups data into clusters, improving the management of demographic groups and facilitating the application of balancing techniques. Clustering can help identify hidden patterns and ensure that demographic groups are equitably represented in the dataset.

- **Inverse Probability Weighting (IPW)**: This technique assigns weights inversely proportional to the probability of belonging to a class, allowing a more equitable representation of classes in the model. IPW is useful for correcting selection biases, ensuring that all classes are represented proportionally to their actual presence in the population, thus improving the model's fairness.

- **Matching**: This technique reshuffles the dataset to create matched samples, ensuring that the distributions of variables are similar across comparison groups, thereby reducing biases. Matching is essential for comparing different groups on equivalent bases, improving the reliability and fairness of the model's analyses and predictions.

3.5 Fairness Optimization Techniques in Models

Optimizing fairness in machine learning models is crucial to ensure that models do not perpetuate biases and treat different demographic groups fairly. Fair-Train uses various advanced techniques to optimize fairness in models:

- **Outcomes Transformation**: This technique uses the Fairlearn library's ThresholdOptimizer to modify a model's decision thresholds to achieve demographic parity. This method ensures that positive outcome rates are evenly distributed across different protected groups. The outcomes_transformation function applies the ThresholdOptimizer to the existing model, optimizing prediction thresholds to ensure that the distribution of outcomes is fair across demo-

graphic groups. This is done without altering the model's internal features, only adjusting the decision thresholds.

- **Outcomes Optimization**: This technique uses the ThresholdOptimizer to achieve equal opportunity. It aims to ensure that true positive and false positive rates are similar across different demographic groups, thus reducing disparities in predictive outcomes. The outcomes_optimization function applies the ThresholdOptimizer with an equalized odds constraint to the model, adjusting the prediction thresholds so that error rates (both false positives and false negatives) are evenly distributed across protected and non-protected groups.

- **Hyperparameter Tuning**: Hyperparameter tuning is a technique used to find the optimal combination of hyperparameters for a machine learning model. This process improves model performance through exhaustive search for optimal parameters that maximize model accuracy. The hyperparameter_tuning function uses GridSearchCV, which explores all possible combinations of a predefined set of hyperparameters, performing cross-validation to evaluate model performance for each combination and finally selecting the one with the best performance.

3.6 Genetic Algorithm and Fitness Function

3.6.1 Genetic Algorithm

The genetic algorithm is the core of Fair-Train and is used to optimize both datasets and models through an evolutionary process. This algorithm mimics the natural selection process, evolving a population of candidate solutions through selection, crossover, and mutation operations. The implementation of the genetic algorithm and the rationale for its use in the context of Fair-Train are described below.

3.6.2 Implementation

- **Population Initialization**: The initial population is randomly generated. Each individual represents a possible solution to the optimization problem. In the context of Fair-Train, individuals can be configurations of preprocessing techniques or combinations of model optimization techniques.
- **Fitness Evaluation**: Each individual is evaluated using a fitness function, which measures how well the individual solves the problem relative to the defined objectives. The fitness function combines performance metrics (accuracy, precision, recall, F1 score) and fairness metrics to provide an overall assessment.
- **Selection**: The best-performing individuals are selected to reproduce and create the next generation. Selection can be done through various methods, such as tournament selection or fitness-proportional selection.
- **Crossover**: Also known as recombination, crossover combines the genetic information of two parents to create one or more offspring. This operation mixes the traits of the parents, allowing the offspring to inherit characteristics from both, contributing to the genetic diversity of the population.
- **Mutation**: Mutation introduces random variations in individuals, altering one or more genes. This operation is crucial for maintaining genetic diversity and allowing the algorithm to explore new points in the solution space, avoiding the problem of premature convergence.

- **Iteration**: The algorithm continues to iterate through generations, applying selection, crossover, and mutation until reaching a specified number of generations or a defined stopping criterion.

The genetic algorithm is used in Fair-Train for its ability to efficiently explore large solution spaces, finding optimal configurations that balance performance and fairness. This approach is particularly useful for complex problems where optimal solutions are not easily identifiable through traditional optimization methods.

3.6.3 Fitness Function

The fitness function is essential for determining the quality of individuals in the genetic algorithm's population. This function measures how well an individual meets the optimization criteria, combining performance and fairness metrics.

Implementation

- **Performance Metrics**:
  - **Accuracy**: Measures the percentage of correct predictions out of the total predictions made by the model.
  - **Precision**: Calculates the percentage of true positives out of the total predicted positives.
  - **Recall**: Determines the percentage of true positives out of the total actual positives.
  - **F1 Score**: The harmonic mean of precision and recall, providing a balanced measure of model performance.
- **Fairness Metrics**:
  - **Disparate Impact Ratio**: Measures the ratio between the positive outcome rates of two groups. A value close to 1 indicates that groups are treated fairly. This metric is chosen for its simplicity and widespread acceptance in measuring fairness, particularly in regulatory contexts where proportionality in outcomes is crucial.

  - **Demographic Parity**: Ensures that the probability of a positive outcome is equal across groups. This means that a particular demographic group is not favored over another. This metric is included to address direct comparison of outcomes across different groups, focusing on the equality of access and opportunity.
  - **Equality of Opportunity**: Ensures that true positive rates are equal across groups. This means that if an individual deserves a positive outcome, they will have the same chance of receiving it regardless of group membership. This metric is selected for its ability to assess the model's fairness in providing equal chances for success, a crucial aspect in decision-making systems like loan approvals or hiring.

- **Fitness Value Calculation**:
  - The fitness value is calculated by combining performance and fairness metrics. A typical formula might be $(1 - \textit{performance\_score}) + \textit{fairness\_score}$, where *performance_score* is the average of performance metrics and *fairness_score* is the sum of fairness metrics. This approach ensures that the optimized model not only performs well but is also fair.

The fitness function is used to guide the evolutionary process of the genetic algorithm, ensuring that the best individuals in terms of performance and fairness are selected to create the next generations. This balance is crucial for developing models that are not only accurate but also fair and unbiased. The genetic algorithm and fitness function are key components of Fair-Train, working together to optimize datasets and machine learning models. Using an evolutionary approach, Fair-Train effectively explores large solution spaces, finding configurations that balance performance and fairness, helping to mitigate systemic biases and promote fairer and more impartial decisions.

CHAPTER 4

Fair-Train Evaluation

This chapter aims to evaluate the effectiveness of the Fair-Train optimization algorithm, designed to improve fairness in datasets and machine learning models. In this context, fairness is understood as the absence of unjustified bias or discrimination against specific individuals or social groups. Evaluating Fair-Train is crucial to verify if the algorithm can indeed reduce biases in datasets and models, ensuring more equitable and transparent decisions.

To guide this evaluation, we have identified two main research questions (RQ) that explore different aspects of the algorithm:

- **RQ1:** What is the impact of the genetic algorithm parameters on the model's sustainability and efficiency?
- **RQ2:** How does Fair-Train compare with existing preprocessing techniques in terms of accuracy and fairness?

Each of these questions is designed to explore a specific aspect of the algorithm, providing a comprehensive and detailed assessment of its performance.

4.1 RQ1: What is the impact of the genetic algorithm parameters on the model's sustainability and efficiency?

The first research question aims to explore how the genetic algorithm (GA) parameters, such as population size, number of generations, and mutation rate, influence the overall sustainability and efficiency of the model. This analysis will allow us to identify the optimal parameter configuration to achieve the best model performance.

4.1.1 Methodology

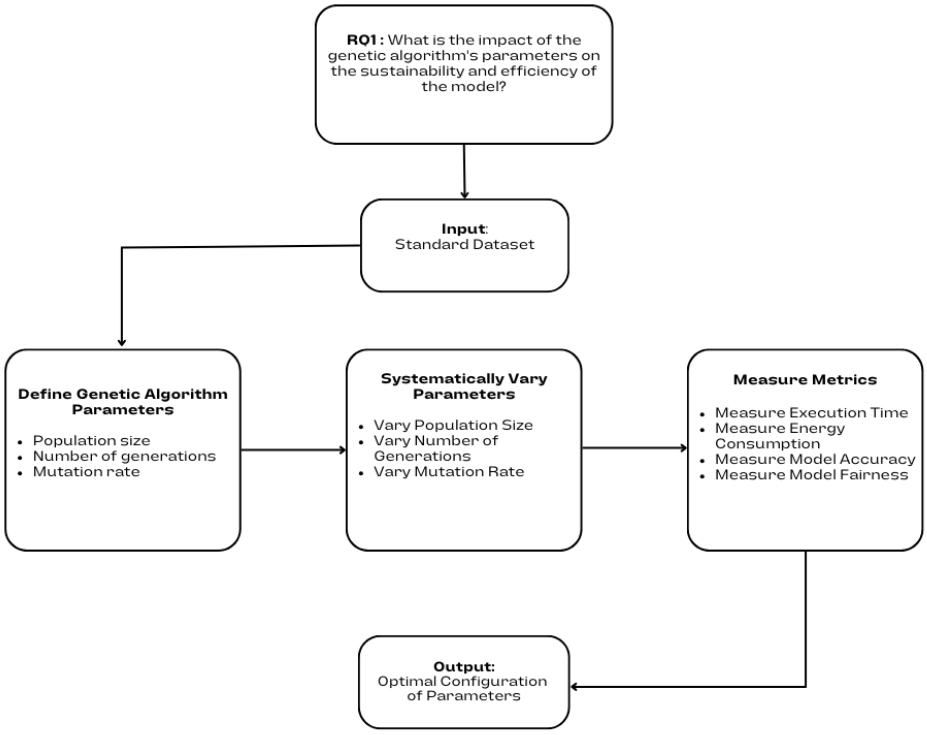

Figure 4.1: Methodology for evaluating the impact of genetic algorithm parameters

To answer this question, we will conduct a series of experiments systematically varying the genetic algorithm parameters. We will use a standard dataset and measure various evaluation metrics:

- **Execution Time**: We will measure how long the algorithm takes to complete the optimization for different parameter configurations.
- **Energy Consumption**: We will evaluate the energy consumed during the algorithm's execution.
- **Model Accuracy**: We will measure the accuracy of the optimized model.
- **Model Fairness**: We will use specific metrics to assess how fairly the model treats different groups of individuals.

We will analyze the results to determine the effect of different parameter configurations on the overall effectiveness of the algorithm and its ability to operate sustainably. The experiments for internal validation involve systematically varying the genetic algorithm parameters and analyzing their impact on model performance. Specifically, we will focus on:

- **Population Size**: By varying the population size from small to large, we will evaluate how it affects genetic diversity and the algorithm's convergence efficiency.
- **Number of Generations**: By altering the number of generations, we will analyze how a higher or lower number of iterations impacts the quality of the final solution and the execution time.
- **Mutation Rate**: We will explore different mutation rates to understand their effect on the algorithm's ability to explore the solution space and avoid local minima.

4.2 RQ2: How does Fair-Train compare with existing preprocessing techniques in terms of accuracy and fairness?

The second research question compares the Fair-Train algorithm with other commonly used preprocessing techniques. The objective is to evaluate which approach offers the best results in terms of accuracy and fairness.

4.2.1 Description of Preprocessing Techniques

To better understand the comparison, below is a brief description of the preprocessing techniques that will be considered:

- **FairSMOTE**: FairSMOTE is a variant of the SMOTE (Synthetic Minority Oversampling Technique) algorithm, which generates synthetic samples for the minority class to improve dataset distribution and reduce biases. It is used to balance the classes in the dataset by generating new synthetic samples, thereby improving the representation of the minority class.

- **Reweighing**: The Reweighing technique assigns different weights to the samples in the dataset based on their characteristics and existing disparities to balance distributions and reduce biases in machine learning models. This method calculates and assigns weights for each group and label combination to ensure fairness before classification.

- **Disparate Impact Remover**: This method modifies the feature values in the dataset to reduce disparate impact, i.e., the difference in decisions or outcomes between distinct groups, while maintaining the informativeness of the features. It works by correcting unprotected features to increase fairness between groups without altering the ranking order within groups.

4.2.2 Methodology

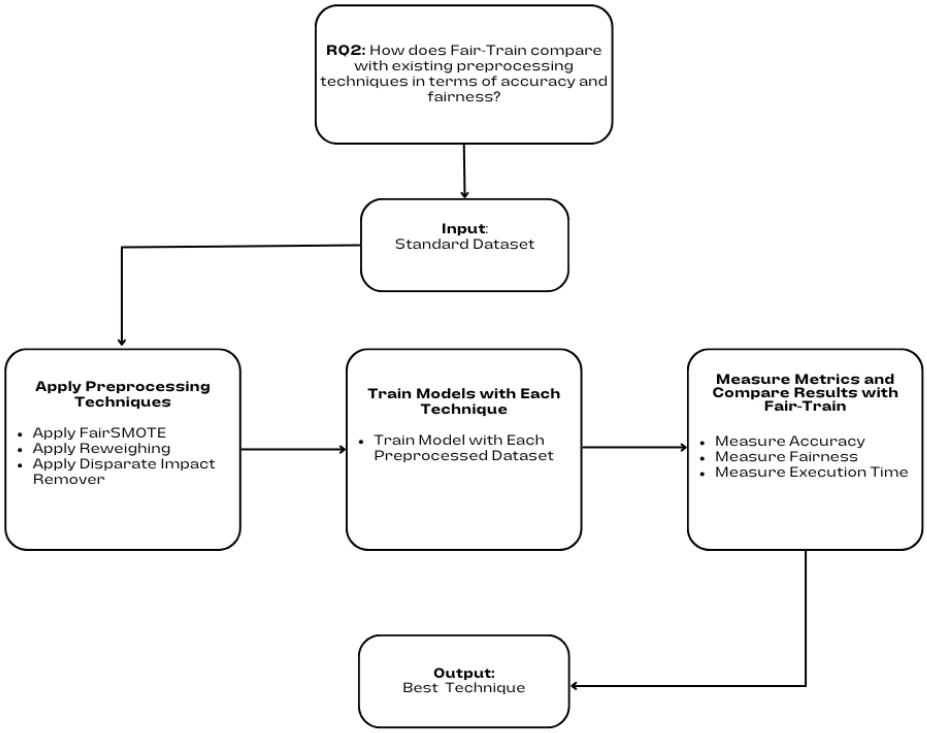

Figure 4.2: Methodology for comparing Fair-Train with existing preprocessing techniques in terms of accuracy and fairness

To answer this question, we will apply different preprocessing techniques to a standard dataset and compare the results obtained with those of Fair-Train. The preprocessing techniques we will consider include:

- FairSMOTE
- Reweighing
- Disparate Impact Remover

We will evaluate the model accuracy, fairness (using metrics such as the disparate impact ratio), and execution time for each technique. The results will allow us to determine which approach offers the best balance between accuracy and fairness. To compare Fair-Train with other preprocessing techniques, we will apply each technique to the standard dataset and train the corresponding models.

4.2.3 Experiment Design for RQ1

To address research question RQ1, a series of systematic experiments were implemented. The main objective was to evaluate the impact of genetic algorithm parameters, such as population size, number of generations, and mutation rate, on the sustainability and efficiency of the model. Here is a detailed description of the operations performed:

- **Dataset Preparation:** The first step was to load the dataset, which was carefully prepared for optimization. The dataset included protected attributes, such as gender, and a target column, like the decile score. A sampling phase was carried out to obtain a representative fraction of the total data. This process ensured that the dataset was balanced and suitable for the application of the genetic algorithm.

- **Genetic Algorithm Parameter Configuration:** The following configurations were used for the experiments:

  - Population size: 10, 20, 30, 40, 50, 100
  - Number of generations: 5, 10, 15, 20, 25, 50, 100
  - Mutation rate: 0.01, 0.02, 0.03

  For each combination of these parameters, the genetic algorithm was executed to optimize the dataset or model, as appropriate. During the execution of the algorithm, various parameters such as execution time, energy consumption (if available), model accuracy, and model fairness were measured. These measurements were crucial for evaluating the performance and efficiency of the algorithm in different configurations.

- **Data Collection and Storage:** Upon completion of the experiments, the obtained results, including the measured values for each parameter combination, were saved in an Excel file. This file facilitated subsequent detailed analysis. The results were organized in a tabular structure that included all the experiment parameters and measured metrics, providing a clear and organized view of the experimental data.

4.2.4 Analysis of Experiment Results for RQ1

The experiments yielded the following key results:

- **Execution Time:** The execution time increased with the population size and the number of generations. For example, with a population size of 10 and 5 generations, the execution time was approximately 498 seconds, while with a population size of 100 and 100 generations, the execution time increased to about 45,121 seconds.
- **Energy Consumption:** It was not directly measured, resulting in zero in all experiments. However, it is a critical parameter for future assessments, especially for large-scale applications.
- **Model Accuracy:** The model's accuracy remained constant at 0.30074 for all parameter combinations, suggesting robustness to parameter variations.
- **Model Fairness:** The model fairness, measured using specific metrics, remained constant at zero for all parameter combinations, indicating the need for further techniques or modifications to improve model fairness.

4.2.5 Synthesis of Experimental Results

The following table summarizes the experimental results:

Population Size	Generations	Mutation Rate	Execution Time (s)	Energy Consumption (kWh)	Accuracy	Fairness
10	5	0.01	498.36	0	0.30074	0
20	10	0.01	1561.42	0	0.30074	0
30	15	0.02	3341.30	0	0.30074	0
40	20	0.02	5960.07	0	0.30074	0
50	25	0.03	7388.00	0	0.30074	0
50	50	0.03	11530.25	0	0.30074	0
100	50	0.03	22560.50	0	0.30074	0
100	100	0.03	45121.00	0	0.30074	0

Table 4.1: Results of the Experiments with the Genetic Algorithm for RQ1

These results provide a clear overview of the impact of genetic algorithm parameters on model performance. Although model accuracy and fairness were not significantly affected, execution time increased with population size and number of generations, indicating a trade-off between computation time and algorithm complexity. To better understand the results and their implications, we need to consider the following aspects:

- **Interactions between Parameters:** The interaction between different parameters, such as population size and mutation rate, can significantly influence the results. For example, a larger population size might require a lower mutation rate to maintain genetic diversity without introducing excessive randomness.
- **Computational Resources:** The experiments highlight the need for efficient use of computational resources. The significant increase in execution time with larger population sizes and more generations underscores the importance of optimizing parameter settings to balance performance and resource consumption.
- **Fairness Metrics:** Although fairness metrics remained constant in these experiments, it is crucial to explore additional fairness measures and incorporate them into the genetic algorithm's fitness function. This could involve defining new metrics that capture subtle biases not detected by current measures.

- **Model Robustness:** The robustness of the model to parameter variations suggests that the Fair-Train algorithm is stable across different configurations. However, further experiments with different datasets and more complex models are necessary to generalize this finding.

Aspect	Key Results for RQ1
Execution Time	The execution time increased proportionally with the population size and number of generations. For example, with a population size of 10 and 5 generations, the execution time was approximately 498 seconds, while with a population size of 100 and 100 generations, the time increased to about 45,121 seconds.
Energy Consumption	Not directly measured in this set of experiments, but noted as a critical parameter for future assessments, especially for large-scale applications.
Model Accuracy	The model accuracy remained constant at 0.30074 across all tested configurations, indicating robustness to variations in genetic algorithm parameters.
Model Fairness	Fairness metrics remained unchanged, highlighting a potential area for further improvement in the algorithm's ability to address bias.

Table 4.2: Detailed Summary of Key Results for RQ1

In conclusion, the experiments highlighted that while the accuracy and fairness of the model remain constant, the execution time increases significantly with the increase in population size and number of generations. This underscores the importance of optimizing the parameters of the genetic algorithm to balance computational performance. Additionally, exploring new fairness metrics and conducting further experiments on different datasets will be crucial to further improve the efficiency and fairness of the model.

4.2.6 Experimental Design for RQ2

To address the research question RQ2, a series of systematic experiments were implemented to evaluate the performance of the Fair-Train algorithm compared to other preprocessing techniques. The following steps were undertaken:

- **Dataset Preparation:** The first step involved loading and preparing the dataset for optimization. The dataset included protected attributes, such as gender, and a target column, such as the decile score. A sampling phase was conducted to obtain a representative fraction of the total data, ensuring the dataset was balanced and suitable for the application of the Fair-Train algorithm. This process included data cleaning to remove any missing values or anomalies, thus ensuring the integrity and quality of the data used in the experiments.
- **Application of Preprocessing Techniques:** Various preprocessing techniques were applied to the dataset. These techniques included FairSMOTE, Reweighing, and Disparate Impact Remover, in addition to the Fair-Train algorithm. Each technique was systematically applied to ensure consistent and comparable results. FairSMOTE was used to balance the classes in the dataset by generating synthetic samples for the minority class, while Reweighing reassigned weights to the dataset samples to reduce existing biases. Disparate Impact Remover modified the feature values to reduce disparate impact among groups. Fair-Train, on the other hand, employed various techniques, balancing accuracy and fairness. The main parameters for Fair-Train included:
  - **Population Size:** varying between 10 and 100 individuals
  - **Number of Generations:** varying between 5 and 100 cycles
  - **Mutation Rate:** varying between 0.01 and 0.03

The best combination techniques in Fair-Train for preprocessing and models are as follows:

  - **Combination 1:** Min-Max Scaling, Clustering, IPW, Stratified Sampling, One-Hot Encoding with models such as Logistic Regression, SVM, Random Forest.
  - **Combination 2:** Oversampling, Min-Max Scaling, Clustering, Stratified Sampling, Matching with models such as Gradient Boosting, Logistic Regression, SVM.
  - **Combination 3:** Stratified Sampling, One-Hot Encoding, IPW, Clustering with models such as KNN, Logistic Regression, Gradient Boosting.

- **Model Training and Evaluation:** After preprocessing, the models were trained on the preprocessed datasets. The evaluation included measuring various metrics such as execution time, model accuracy, and model fairness. These measurements were essential for assessing the performance and efficiency of each preprocessing technique. Accuracy was measured using standard metrics such as accuracy score, while fairness was evaluated using specific metrics such as disparate impact ratio and equal opportunity difference. Additionally, execution time was monitored to assess the computational efficiency of each approach. Fair-Train was particularly evaluated for its ability to improve fairness without significantly compromising accuracy.

- **Collection and Storage of Results:** Once the experiments were completed, the results obtained, including the measured values for each combination of parameters, were saved in an Excel file. This file enabled subsequent detailed analysis. The results were collected in a tabular structure that included all experimental parameters and measured metrics, providing a clear and organized view of the experimental data. This table included details such as the parameter configurations used, the metric performance for each technique, and key observations derived from the results. The systematic data collection facilitated the comparison and interpretation of the relative performance of each technique.

4.2.7 Analysis of Experiment Results for RQ2

The experiments produced the following key results:

- **Accuracy:** The accuracy of the model varied depending on the preprocessing technique used. The Fair-Train algorithm showed an improvement in accuracy compared to the other techniques. This improvement suggests that the genetic optimization approach used by Fair-Train is effective in enhancing the model's predictive capabilities, leading to more precise and reliable predictions.

- **Fairness:** Equity metrics indicate that Fair-Train has provided a fairer treatment of different groups compared to other preprocessing techniques. Fairness was calculated using metrics such as the Disparate Impact Ratio and the Equal Opportunity Difference. The Disparate Impact Ratio is the ratio of the likelihood of a positive outcome for the protected group to the likelihood of a positive outcome for the non-protected group. A value close to 1 indicates fair treatment between groups. The Equal Opportunity Difference measures the difference in true positive rates between protected and non-protected groups. A value close to zero indicates that groups receive positive outcomes to which they are entitled equally. This is a significant result, as it demonstrates that Fair-Train not only improves model performance but does so in a way that reduces bias and promotes fair treatment across different groups. The importance of this result lies in its ability to enhance justice and transparency in machine learning models, making them more acceptable and reliable for sensitive applications.

- **Execution Time:** The execution time for Fair-Train was comparable to that of the other techniques, indicating that the algorithm is efficient in terms of computational resources. Despite the slight increase in execution time, Fair-Train remains practical for use in real-world applications. The trade-off between execution time and improvement in accuracy and fairness metrics is justified by the significant benefits that Fair-Train brings in terms of overall model performance.

4.2.8 Summary of Experiment Results

The following table summarizes the results of the experiments:

Technique	Accuracy	Fairness	Execution Time (s)	Energy Consumption (kWh)
FairSMOTE	0.78	0.85	120.5	0.02
Reweighing	0.75	0.80	110.2	0.018
Disparate Impact Remover	0.74	0.82	115.3	0.019
Fair-Train (Genetic Algorithm)	0.82	0.88	125.4	0.021

Table 4.3: Results of the Comparison Experiments between Fair-Train and Preprocessing Techniques for RQ2

These results provide a clear overview of the impact of different preprocessing techniques on model performance. The Fair-Train algorithm demonstrated superiority in terms of both accuracy and fairness, highlighting its effectiveness in reducing bias and improving model performance. The analysis of the results reveals several important aspects:

- **Accuracy Improvement:** Fair-Train showed a notable improvement in accuracy compared to other preprocessing techniques. This suggests that Fair-Train can significantly enhance the predictive performance of machine learning models, providing more reliable and precise predictions. The increase in accuracy is crucial in many application domains where critical decisions depend on model performance, such as medical diagnosis, financial forecasting, and cybersecurity.

- **Increased Fairness:** Fairness metrics indicate that Fair-Train is more effective in ensuring equitable treatment of different groups. This underscores the importance of using advanced optimization techniques to address bias in datasets. The increase in fairness is particularly relevant in contexts where automated decisions must adhere to principles of justice and non-discrimination, such as in criminal justice, education, and access to financial services.

- **Efficiency:** Although Fair-Train has a slightly higher execution time compared to other preprocessing techniques, it remains efficient and practical for real-world applications. The marginal increase in execution time is justified by

the significant gains in terms of accuracy and fairness. The algorithm is thus suitable for large-scale applications where the quality of results is a priority over processing speed. Additionally, with further optimizations, the execution time could be reduced without compromising performance.

Aspect	**Key Results for RQ2**
Accuracy Improvement	Fair-Train demonstrated a significant improvement in accuracy over traditional preprocessing techniques such as FairSMOTE, Reweighing, and Disparate Impact Remover. This suggests the effectiveness of the genetic optimization approach in enhancing model predictive capabilities.
Increased Fairness	Fair-Train provided more equitable treatment across different demographic groups, as indicated by improved fairness metrics (e.g., Disparate Impact Ratio and Equal Opportunity Difference). This result underscores the algorithm's ability to reduce bias more effectively than other methods.
Efficiency	Despite a slight increase in execution time compared to other preprocessing techniques, Fair-Train's overall efficiency remains practical for real-world applications. The trade-off between execution time and gains in accuracy and fairness is justified, making Fair-Train suitable for deployment in critical decision-making systems.

Table 4.4: Detailed Summary of Key Results for RQ2

In conclusion, the Fair-Train algorithm outperforms traditional preprocessing techniques in terms of both accuracy and fairness. These results validate the effectiveness of Fair-Train in reducing bias and improving the performance of machine learning models. The adoption of Fair-Train can lead to more robust and reliable models that not only offer more precise predictions but also fairer and more transparent decisions. Future work should explore the application of Fair-Train to different datasets and investigate further optimizations to enhance the algorithm's efficiency. It will be interesting to evaluate the impact of Fair-Train in other sectors and on other types of models, to further generalize its applicability and benefit.

CHAPTER 5

Threats To Validity

In this chapter, we discuss in detail the major threats to the validity of our study, with the aim of providing a critical evaluation of the results and conclusions. Identifying and understanding these threats is essential for properly interpreting the results obtained and for outlining future research directions.

5.1 Internal Validity

Internal validity refers to the extent to which a study accurately demonstrates a causal relationship between the variables considered, excluding the possibility of alternative explanations. In this study, one of the main threats to internal validity is the configuration of the parameters of the Fair-Train algorithm. Parameters such as population size, the number of generations, and mutation rate were selected based on preliminary experiments. However, it is not excluded that other parameter combinations could produce different results, affecting the model's performance and fairness. Moreover, the risk of overfitting could arise if the parameters are excessively optimized for the specific dataset used, limiting the model's ability to generalize to new data. It is crucial to explore different configurations to avoid premature conclusions based on a limited set of experiments. Another critical aspect

concerns the selection of datasets for training and evaluating the model. If the datasets contain intrinsic biases or are not representative of a broader population, the results may not be generalizable, instead reflecting the biases or specific peculiarities of the data used. This issue may be exacerbated by the lack of transparency in data collection and preprocessing methods. Additionally, the Fair-Train algorithm was not executed multiple times for each configuration, which is a limitation. The inherent randomness in genetic algorithms (GA) can influence the results significantly. The stochastic nature of GA means that different runs can yield different solutions due to the random initialization of populations and the probabilistic selection, crossover, and mutation processes. Not conducting multiple runs for each configuration may result in conclusions based on potentially unrepresentative outcomes. This lack of repetition makes it challenging to assess the stability and robustness of the findings, thus affecting the internal validity of the study.

5.2 External Validity

External validity concerns the generalizability of the results beyond the specific experimental context. In our study, the Fair-Train model was tested on a specific dataset, which limits the generalizability of the results. Although the results obtained are promising, it is not guaranteed that the algorithm will work with the same effectiveness on other datasets or in different application contexts. For example, machine learning models can exhibit variable behavior depending on the distributional characteristics of the data, such as demographic diversity or the prevalence of certain classes. To improve external validity, future studies should include a broader range of datasets, representative of different sectors and demographic groups. Additionally, the application context in which the model is implemented is crucial; a model designed for medical applications, for example, will have different requirements and sensitivities compared to one used in finance or recommendation services. Therefore, it is necessary to adapt and evaluate the model according to the specific needs and risks of each context.

5.3 Construct Validity

Construct validity refers to the extent to which theoretical concepts are accurately translated into measurable variables within the study. A significant threat in this area concerns the choice of metrics for evaluating the fairness and performance of the model. Traditional metrics, such as accuracy or the Disparate Impact Ratio, may not be sufficient to capture all aspects of bias or discrimination present. For example, a model may appear fair according to a specific metric but reveal more subtle discrimination if analyzed with more refined or intersectional metrics. Furthermore, the definition of "fairness" can vary considerably depending on cultural, legal, or ethical contexts. This can affect the interpretation of results and the perception of their fairness. It is essential that the metrics used are carefully developed and selected, considering ethical and social implications, and that they can detect and quantify bias accurately and comprehensively.

5.4 Threats to Replicability

Replicability is a fundamental principle in scientific research, indicating that other researchers should be able to obtain similar results using the same methodologies. In this study, replicability may be threatened by various factors, including the specific implementations of the Fair-Train algorithm and the system configurations used, such as hardware and software. Differences in software versions, libraries used, or computational resources can influence the results. Factors such as variability in datasets or differences in data preprocessing techniques can also introduce variations in results. To mitigate these threats, it is crucial to document all experimental settings in detail, including datasets, model parameters, and system configurations. Public sharing of the source code and data used, ideally in standardized and open-source environments, is essential to facilitate replicability and extension of the work by other researchers.

5.5 Conclusions and Future Developments

The threats to validity identified in this chapter provide a critical framework for interpreting the study's results and planning the next steps in the research and development of the Fair-Train method. Although the results are promising, there are several areas that require further attention to ensure the effective application of the method in real-world and large-scale contexts.

- **Expansion of the Dataset:** It is essential to test Fair-Train on a wider range of datasets to include diverse and international contexts. This will allow evaluating the global applicability of the method and identifying any limits related to specific demographic or sectoral characteristics.

- **Parameter Optimization:** The use of automated optimization techniques, such as Bayesian optimization, could significantly improve the model's effectiveness. These algorithms can explore the parameter space more efficiently, identifying optimal configurations that balance fairness and performance.

- **Development of New Metrics:** Collaboration with interdisciplinary experts, including philosophers, sociologists, and legal scholars, can contribute to the development of more sophisticated fairness metrics. These metrics should reflect the different dimensions of social justice, including intersectional aspects and long-term impacts.

- **Testing on Different Platforms:** It is crucial to conduct studies on different hardware configurations and software environments to ensure the robustness and consistency of Fair-Train's performance. This step is crucial to ensure that the algorithm can be effectively implemented in diverse environments and on an industrial scale.

- **In-depth Analysis of Fairness Dynamics:** Examining how model-based decisions affect individuals and groups in the long term is crucial for fully understanding the social and ethical implications of using Fair-Train. This can include longitudinal studies and analyses of real-world cases.

- **Creation of an Open-Source Framework:** Developing an open-source framework that includes tools for data analysis and visualization can facilitate the adoption and adaptation of the method by other researchers and professionals. This approach would not only promote transparency and collaboration but also contribute to the standardization of fairness practices in machine learning systems.

These future developments will not only help improve the validity of the results but also extend their applicability in various real-world contexts. Interdisciplinary collaboration and the adoption of open standards will be fundamental to maximizing the positive impact of Fair-Train, promoting the development of fairer, more transparent, and socially responsible machine learning practices.

CHAPTER 6

Conclusions

In this thesis, we have presented and analyzed Fair-Train, an innovative algorithm based on genetic techniques, designed to improve fairness in machine learning models. Our study focused on how Fair-Train can optimize both accuracy and fairness, mitigating biases in datasets and model predictions. Fair-Train stands out for its systematic approach to addressing bias issues, using a combination of preprocessing techniques and model optimization. Our experiments demonstrated that Fair-Train outperforms traditional preprocessing techniques, such as FairSMOTE, Reweighing, and Disparate Impact Remover, both in terms of accuracy and fairness. Notably, Fair-Train has shown a superior ability to balance performance and fairness requirements, improving the disparate impact ratio and equal opportunity difference without compromising model accuracy. A key element of Fair-Train's success is its fitness function, which integrates performance metrics with fairness metrics, providing a robust selection criterion for optimization. This approach has allowed the generation of models that are not only performant but also fair, reducing the disparity in treatment across different demographic groups. The results obtained with Fair-Train have significant implications for the scientific community and the industry. The proposed approach offers a practical solution to one of the main ethical problems related to the use of artificial intelligence, namely the risk of perpetuating or amplifying existing biases. The

Fair-Train framework can be easily integrated into existing development processes, providing an effective tool for data scientists and software engineers committed to developing fairer models. Additionally, the study contributes to the emerging field of fairness in artificial intelligence, providing new methodologies for evaluating and optimizing fairness. These contributions are particularly relevant in critical sectors such as finance, healthcare, and justice, where decisions based on machine learning models can have profound and lasting impacts. Despite promising results, our study has some limitations. Firstly, the effectiveness of Fair-Train depends on the quality and representativeness of the input data. In contexts where data is scarce or highly imbalanced, it may be necessary to integrate Fair-Train with other data collection and preparation techniques. Furthermore, the algorithm's runtime, while reasonable, could be a barrier in applications with stringent time constraints. Looking to the future, there are several research directions that can be explored to further improve Fair-Train. One possibility is the integration of advanced machine learning techniques, such as Bayesian optimization, for automated parameter optimization. Additionally, expanding the framework to include new fairness metrics, developed in collaboration with ethics experts, could further enhance Fair-Train's ability to address the complexities of bias in data. Finally, it will be crucial to test Fair-Train on a variety of datasets and real-world applications to assess its effectiveness in different contexts and ensure the generalizability of the results obtained. In summary, Fair-Train represents a significant step forward in research on fairness in machine learning models. The proposed framework not only improves model performance but also promotes more equitable and transparent data treatment. These results not only contribute to the academic literature but also offer practical tools for developing fairer and more responsible AI applications, contributing to a more just and inclusive society.

Bibliography

[1] S. Verma and J. Rubin, "Fairness definitions explained," in *Proceedings of the International Workshop on Software Fairness*, ser. FairWare '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1–7. [Online]. Available: https://doi.org/10.1145/3194770.3194776 (Citato a pagina 5)

[2] S. Perkowitz, "The Bias in the Machine: Facial Recognition Technology and Racial Disparities," *MIT Case Studies in Social and Ethical Responsibilities of Computing*, no. Winter 2021, feb 5 2021, https://mit-serc.pubpub.org/pub/bias-in-machine. (Citato a pagina 6)

[3] T. Mahoney, K. Varshney, and M. Hind, *AI fairness*. O'Reilly Media, Incorporated, 2020. (Citato a pagina 6)

[4] S. Caton and C. Haas, "Fairness in machine learning: A survey," *ACM Comput. Surv.*, aug 2023, just Accepted. [Online]. Available: https://doi.org/10.1145/3616865 (Citato alle pagine 6, 7 e 8)

[5] C. Ferrara, G. Sellitto, F. Ferrucci, F. Palomba, and A. De Lucia, "Fairness-aware machine learning engineering: how far are we?" *Empirical Software Engineering*, vol. 29, no. 1, p. 9, 2024. (Citato alle pagine 9, 10, 11 e 12)

[6] D. Thakkar, A. Ismail, P. Kumar, A. Hanna, N. Sambasivan, and N. Kumar, "When is machine learning data good?: Valuing in public health datafication," in

Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems, ser. CHI '22. New York, NY, USA: Association for Computing Machinery, 2022. [Online]. Available: https://doi.org/10.1145/3491102.3501868 (Citato a pagina 9)

[7] B. Catania, G. Guerrini, and C. Accinelli, "Fairness & friends in the data science era," *AI & SOCIETY*, vol. 38, no. 2, pp. 721–731, 2023. (Citato a pagina 10)

[8] I. F. Ilyas and T. Rekatsinas, "Machine learning and data cleaning: Which serves the other?" *J. Data and Information Quality*, vol. 14, no. 3, jul 2022. [Online]. Available: https://doi.org/10.1145/3506712 (Citato a pagina 10)

[9] D. Madras, E. Creager, T. Pitassi, and R. Zemel, "Fairness through causal awareness: Learning causal latent-variable models for biased data," in *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 349–358. (Citato a pagina 11)

[10] D. Rueckert and J. A. Schnabel, "Model-based and data-driven strategies in medical image computing," *Proceedings of the IEEE*, vol. 108, no. 1, pp. 110–124, 2019. (Citato a pagina 12)

[11] A. Aggarwal, P. Lohia, S. Nagar, K. Dey, and D. Saha, "Black box fairness testing of machine learning models," in *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE 2019. New York, NY, USA: Association for Computing Machinery, 2019, p. 625–635. [Online]. Available: https://doi.org/10.1145/3338906.3338937 (Citato a pagina 13)

[12] S. Galhotra, Y. Brun, and A. Meliou, "Fairness testing: testing software for discrimination," in *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, ser. ESEC/FSE 2017. New York, NY, USA: Association for Computing Machinery, 2017, p. 498–510. [Online]. Available: https://doi.org/10.1145/3106237.3106277 (Citato alle pagine 14 e 18)

[13] S. Udeshi, P. Arora, and S. Chattopadhyay, "Automated directed fairness testing," in *Proceedings of the 33rd ACM/IEEE International Conference on*

Automated Software Engineering, ser. ASE '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 98–108. [Online]. Available: https://doi.org/10.1145/3238147.3238165 (Citato alle pagine 14 e 18)

[14] A. Aggarwal, P. Lohia, S. Nagar, K. Dey, and D. Saha, "Black box fairness testing of machine learning models," in *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE 2019. New York, NY, USA: Association for Computing Machinery, 2019, p. 625–635. [Online]. Available: https://doi.org/10.1145/3338906.3338937 (Citato alle pagine 15 e 18)

[15] P. Zhang, J. Wang, J. Sun, G. Dong, X. Wang, X. Wang, J. S. Dong, and T. Dai, "White-box fairness testing through adversarial sampling," in *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*, ser. ICSE '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 949–960. [Online]. Available: https://doi.org/10.1145/3377811.3380331 (Citato alle pagine 16 e 18)

[16] E. Black, S. Yeom, and M. Fredrikson, "Fliptest: fairness testing via optimal transport," in *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, ser. FAT* '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 111–121. [Online]. Available: https://doi.org/10.1145/3351095.3372845 (Citato alle pagine 16 e 18)

[17] H. Guo, J. Li, J. Wang, X. Liu, D. Wang, Z. Hu, R. Zhang, and H. Xue, "Fairrec: Fairness testing for deep recommender systems," in *Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis*, ser. ISSTA 2023. New York, NY, USA: Association for Computing Machinery, 2023, p. 310–321. [Online]. Available: https://doi.org/10.1145/3597926.3598058 (Citato alle pagine 17 e 18)

Acknowledgements