Fair and Square? Evaluating Fairness of LLM-Generated Synthetic Datasets

Gianmario Voria, Benedetto Scala, Leopoldo Todisco, Carlo Venditto, Giammaria Giordano, Gemma Catolino, Fabio Palomba

Software Engineering (SeSa) Lab - Department of Computer Science, University of Salerno, Italy

Abstract

Context. Machine Learning (ML) is driving advancements across various industries, including healthcare, finance, and entertainment, but it also raises significant ethical concerns, particularly regarding fairness. Biases in training data can lead to unfair outcomes, perpetuating or even amplifying existing disparities. Prior research in the Software Engineering (SE) and ML communities has developed numerous bias mitigation techniques, yet two key limitations persist: (1) most approaches intervene at later stages of development, such as after data collection or model training, rather than addressing fairness from the outset; and (2) these methods often mitigate bias without fully eliminating it, since the root issue frequently lies in the data itself. **Objective.** In this paper, we explore an alternative approach to mitigate unfairness: *synthetic data generation*, which involves creating artificial datasets that mimic the statistical properties of real-world data. We aim to assess how this approach can contribute to generating data that positively impacts the trade-off between performance and fairness by creating datasets that reduce the influence of real-world biases through synthetic feature generation. **Methods.** To this end, we conducted an empirical study comparing ML models trained on synthetic datasets generated by large language models to ML models trained on real-world data, evaluating performance and fairness indicators. **Results.** Our results demonstrate that models trained with synthetic data, particularly those generated using simpler prompts, can achieve competitive performance while enhancing fairness. **Conclusion.** Our work suggests that synthetic data generation may be a viable approach to addressing fairness requirements in ML systems.

Keywords: Synthetic Dataset Generation; Machine Learning Fairness; Software Engineering for Artificial Intelligence; Empirical Software Engineering.

1. Introduction

Machine Learning (ML) has been an hot topic, rapidly expanding across industries and everyday applications [97]. ML projects—projects powered by ML algorithms [57]—are integral to

Email addresses: gvoria@unisa.it (Gianmario Voria), b.scala1@studenti.unisa.it (Benedetto Scala), l.todisco4@studenti.unisa.it (Leopoldo Todisco), c.venditto@studenti.unisa.it (Carlo Venditto), giagiordano@unisa.it (Giammaria Giordano), gcatolino@unisa.it (Gemma Catolino), fpalomba@unisa.it (Fabio Palomba)

modern business operations and technology interactions, transforming industries like healthcare and entertainment by enhancing efficiency, decision-making, and innovation [62, 67, 68, 90].

The widespread adoption of ML has raised significant ethical concerns, particularly regarding *fairness* [61], i.e., the principle that models should make impartial decisions without favoring or discriminating against groups. Unfairness arises when models perpetuate biases in training data [70, 74], leading to decisions that erode trust and create ethical and legal challenges [63].

Recognizing the importance of fairness, the Software Engineering (SE) research community—particularly within the field of Software Engineering for Artificial Intelligence (SE4AI)—has made significant advances in developing bias mitigation techniques [44]. These efforts can be broadly classified into three categories: *pre-processing*, *in-processing*, and *post-processing techniques*. Pre-processing techniques aim to reduce bias before the data is fed into the model. An example is Fair-SMOTE [14], which removes biased labels and rebalances internal distributions to ensure equal representation of different sensitive attribute groups. In-processing techniques modify the learning algorithm itself to minimize bias during model training. For instance, Li et al. [53] proposed fairness-aware training methods, while Chakraborty et al. [16] explored the impact of hyper-parameter optimization. Finally, post-processing techniques adjust the outputs of ML models to ensure fairness after model training. As an example, Galhotra et al. [36] developed automated methods for testing software fairness properties, which may be used to evaluate and correct unfair outcomes.

While these bias mitigation techniques have proven valuable, two considerations should be highlighted. First, these techniques can be applied at later stages of the development process, i.e., after data has already been collected or even when models have been trained. As such, these techniques address bias *reactively* rather than *proactively*. Perhaps more importantly, these techniques often only mitigate bias rather than fully eliminate it, as the fundamental issue originates from the data used to train machine learning models. Indeed, if the training data is biased, even advanced mitigation strategies may only achieve *limited success* in correcting unfairness [21].

Stemming from these considerations, this paper proposes to investigate the problem from a different, orthogonal perspective by exploring the concept of *synthetic data generation* [55], i.e., the creation of artificial datasets that mimic the statistical properties of real-world data. We see the application of this concept as potentially breakthrough. Unfairness typically arises from the data itself and how sensitive features are distributed [61]: a synthetic generation may inherently control for feature distribution, possibly leading to the generation of data that fairly represents demographic groups, thereby reducing the likelihood of biased outcomes. This approach would contribute to *fairness-by-design*, embedding fairness into the system from the very beginning. At the same time, however, synthetic datasets must also preserve predictive performance to be useful in practice. For this reason, our study explicitly evaluates both dimensions: fairness, as a critical non-functional requirement, and predictive performance, measured through standard metrics such as accuracy and F1-score. The trade-off between these two dimensions is central to determining whether synthetic data generation can provide practical value.

The generation of synthetic instances becomes particularly crucial in scenarios where real-world data is scarce or inaccessible. We have seen, and continue to see, concrete examples where synthetic data would have been or would be invaluable, such as rare surgical events in medical domains [64], where ethical considerations and data rarity make collection extremely challenging, or the COVID-19 pandemic, during which urgent data needs exacerbated discrimination against underrepresented groups due to limited data availability [92]. Besides these documented examples, the need for fair synthetic data generation extends far beyond emergency or rare-event scenarios. As machine learning systems increasingly permeate domains where data collection

is costly, ethically sensitive, or prone to historical biases, the ability to generate *fair* synthetic datasets becomes essential not just for improving performance metrics, but for mitigating real-world harms—such as perpetuating systemic discrimination in healthcare access, criminal justice, or employment decisions.

In this broader context, synthetic data generation emerges as a proactive strategy that not only addresses data scarcity but also embeds fairness requirements directly into machine learning pipelines—marking a paradigm shift from reactive bias mitigation to fairness-by-design in data-centric AI development. From a software engineering perspective, fairness has been recently acknowledged as a critical non-functional requirement that belongs to the broader spectrum of software quality [34]. Ensuring such requirements is central to the field of SE4AI, which aims at providing methods and practices for building trustworthy AI-enabled systems. Within this view, synthetic data generation can be understood as a software engineering technique for data preparation, verification, and quality assurance. Its integration has direct implications for requirements engineering (e.g., specifying and validating fairness requirements early on [33]), quality assurance and testing (e.g., using synthetic data in fairness testing pipelines alongside other quality assurance methods [3]), and MLOps (e.g., monitoring fairness regressions across model updates using synthetic data in CI/CD pipelines [80]).

The concept of synthetic data generation is gaining popularity in AI research, where deep learning (DL), Generative Adversarial Networks (GANs), and more recently, Large Language Models (LLMs), are being employed to generate high-quality synthetic data [7, 26]. Our work aims to extend this concept to the domain of SE4AI, examining how synthetic data generated using LLMs affects the trade-off between performance and fairness, and ultimately assessing its potential usefulness in fairness engineering.

◎ Main Objective

*We aim to explore the **potential** of mitigating unfairness at its source, i.e., the data itself, by examining the capabilities of LLMs to generate synthetic datasets that match real-world datasets in terms of predictive performance while avoiding the perpetuation of bias. Our study uncovers the inherent ability of LLMs to produce fully synthetic tabular datasets, offering practical solutions for practitioners who must collect data in unseen contexts, urgent scenarios, or situations where high-quality real-world data is scarce or inaccessible.*

To this aim, we design an empirical study where we first leverage LLMs to generate synthetic datasets resembling the attributes of three well-known datasets: *German Credit* [42], *Heart Disease* [47], and *Student Performance* [22].[1] We then train machine learning models on both synthetic and real datasets and evaluate them using (1) traditional performance metrics, such as accuracy and F1-score, to compare effectiveness; and (2) fairness metrics, such as statistical parity, equal opportunity, average odds, to assess fairness improvements.

The main findings indicate that models trained on synthetic datasets typically perform worse during cross-validation than those trained on real data; however, they generalize well to unseen real-world data, particularly with simpler prompts, such as 0-shot learning. Synthetic datasets

[1]For the purpose of this study, we specifically focus on datasets that include sensitive information, referred to as *sensitive datasets*, to assess how well synthetic data can replicate the characteristics of real-world data while mitigating biases.

generated with 1-shot prompts consistently improve fairness metrics without significantly compromising performance. In contrast, increasing prompt complexity (e.g., 2-shot) tends to degrade both fairness and performance, indicating that simpler prompts are more effective for balanced outcomes. Overall, we conclude that synthetic data generation is a potentially promising alternative approach for enhancing machine learning fairness.

To summarize, the main contributions of our research are:

1. An empirical evaluation of the capabilities of different LLMs and prompting strategies in generating fully synthetic datasets based solely on feature descriptions;
2. A comprehensive empirical analysis of the predictive performance and fairness characteristics of LLM-generated synthetic datasets;
3. An openly available appendix containing all datasets, scripts, and replication materials to support transparency and reproducibility of our study [87].

2. Background and Related Work

This section introduces two subsections: background, which outlines the key concepts underpinning our study, and related work, which reviews prior research on synthetic data generation and positions our contributions within the broader landscape.

2.1. Background

Approaches to fairness in machine learning can be categorized as either *reactive* or *proactive* strategies. Reactive strategies operate at the pre-processing, in-processing, or post-processing stages, adjusting existing data, modifying learning algorithms, or correcting model outputs after training [19, 36, 44]. By contrast, proactive strategies aim to prevent unfairness from arising in the first place—for example, by adjusting data acquisition with fairness objectives [41] or performing sensitive attribute-aware data collection [85]. While our study does not propose a new proactive technique, we position LLM-based synthetic dataset generation within this conceptual space, as it potentially addresses fairness at its source rather than through later adjustments.

Synthetic data is artificially generated data that simulates the properties of real-world data [23, 29] and is useful in ML, especially when access to real data is limited due to privacy, scarcity, or ethical concerns [6]. There are two primary methods for generating synthetic data: *fully synthetic* and *partially synthetic*. Fully synthetic datasets are created entirely from scratch and aim to preserve aggregate properties of real data without including any real records, thereby minimizing privacy risks. Partially synthetic datasets, in contrast, replace selected attributes (often sensitive or high-risk ones) while retaining the remaining features, but require access to original real-world data.

2.2. Related Works — Synthetic Data Generation for Tabular ML

This section introduces methods and findings from prior work that directly motivate our study's contribution, as they represent the main alternatives and historical baselines for generating synthetic tabular data. Traditional methods such as Bayesian Networks [94] and Copulas [54] have long been used to model tabular distributions but tend to struggle as dimensionality and feature heterogeneity increase. More recent generative models—such as GANs and Variational Autoencoders (VAEs)—offer more expressive solutions [39, 51], yet still face well-known challenges in producing statistically coherent heterogeneous tabular data. Although our study

does not employ these techniques, they remain the dominant paradigms in synthetic tabular data generation, and their limitations are precisely what has motivated recent investigations into LLM-based generation [32].

Building on this trajectory, recent studies show that LLMs can generate structured synthetic datasets with promising levels of fidelity [10, 72], although maintaining complex statistical relationships across variables remains an open challenge. Ongoing research explores complementary techniques such as fine-tuning to further improve generative performance [32, 82]. Synthetic data generators are typically evaluated by training predictive models on the synthetic datasets and comparing their performance with models trained on real data; close performance between the two is widely considered a strong indicator of generator quality [9, 20].

In this paper, we concentrate on tabular datasets for several reasons: (1) they are a common format used to investigate bias and fairness concerns in software engineering [31]; (2) they offer a structured, interpretable format that is widely applicable across various industries, making the findings of our study more broadly relevant and aligned with previous research in the field [18, 19, 73]; and (3) they allow for a more straightforward evaluation of the structural similarity between synthetic and real-world data, as the relationships between variables in tabular data are often well-defined and measurable [9].

As for the focus on fully synthetic data, this approach allows a broader and deeper investigation into the capabilities of LLMs. Specifically, it covers two key scenarios that can arise in real-world machine learning engineering contexts. First, the case where data is unavailable due to privacy concerns, legal restrictions, or simply because the data does not yet exist. By generating fully synthetic datasets, engineers can bypass these barriers and still develop, test, and refine machine learning models. Second, the case where data is available but suboptimal (e.g., incomplete, imbalanced, or biased). In this scenario, fully synthetic data helps engineers create tailored datasets to meet specific needs. This capability is useful for testing models under ideal conditions, exploring edge cases, or generating more data for tasks like deep learning, where the quantity and quality of available data can significantly impact model performance [10].

2.3. Related Works — Fairness Evaluation in SE

Fairness in ML projects has become a critical research area within the SE community, with a growing body of literature addressing the issue from various angles [19, 44, 61, 74, 81]. Researchers have been exploring fairness from multiple perspectives, proposing empirical investigations and defining bias mitigation techniques.

Rakova et al. [75] emphasized the role of organizational culture in embedding fairness into AI systems. Ferrara et al. [33] emphasized the importance of context-aware fairness requirements engineering, which aligns with the need for early intervention in the development process. Ferrara et al. [34] also explored how practitioners handle fairness in real-world scenarios, suggesting the need for a software-defined approach that integrates fairness considerations throughout the software development lifecycle. Further expanding this view, recent works have proposed catalogs of fairness-aware practices in ML engineering [88] and conducted large-scale surveys to understand practitioners' perspectives on fairness requirements and adoption [89]. Related empirical studies have also evaluated how fairness toolkits are adopted in practice [86] and how fairness-aware practices can be integrated cost-effectively into ML workflows [71]. Beyond engineering practices, recent investigations have analyzed the broader impact of bias mitigation algorithms on the sustainability of ML systems [24], further highlighting the need to balance fairness interventions with system robustness and maintainability. Building on these insights,

our work explores an additional dimension that enhances the design of comprehensive software-defined approaches to fairness, further advancing the concept of fairness-by-design.

Sesari et al. [78] noted that unfairness may extend beyond protected attributes. This observation further motivates our work, as it reinforces the need for investigations that extend fairness considerations to the entire data collection phase.

Another line of research is based on the hypothesis that discrimination often stems from training on biased or unbalanced datasets [84]. Zhang and Harman [95] argued that increasing the number of features in a dataset does not necessarily reduce discrimination, while Chakraborty et al. [14] demonstrated the importance of selecting relevant features. These findings highlight the complexity of the data selection process, suggesting the need for further investigation into strategies that can manage data selection and extraction—such as the one explored in our study.

To address the challenges associated with managing training data, pre-processing techniques like those proposed by Sharma et al. [79] and Calmon et al. [13] used probabilistic methods to reduce discrimination, while Chakraborty et al. [14] introduced Fair-SMOTE, a synthetic augmentation technique that maintains learning performance. In-processing techniques aim to modify the learning algorithm itself to minimize bias during training. Zhang et al. [93] employed an adversarial approach to reduce bias, and Kamishima et al. [50] used regularization methods to promote equal treatment. Reweighting methods like those by Kamiran and Calders [49] adjust instance weights to enhance fairness, while Chakraborty et al. [15] and Johnson et al. [48] focused on balancing fairness and performance through multi-objective optimization. Finally, post-processing techniques adjust model outputs after training to ensure fairness. Galhotra et al. [36] introduced Themis, which identifies bias through input perturbations, and Udeshi et al. [83] developed Aequitas, which improves bias detection efficiency. Aggarwal et al. [3] proposed a black-box fairness testing method, while Zhang et al. [96] developed a white-box approach using adversarial sampling to reveal and mitigate biases.

2.4. Motivation and Contributions

Our study is motivated by the need to understand whether LLMs can generate tabular datasets that are not only structurally consistent and predictive, but also fair. Existing approaches to bias mitigation are inherently *reactive*: they adjust existing datasets, modify algorithms during training, or adapt outputs after training [19, 36, 44]. By contrast, our work investigates whether fairness can be addressed at its very source by verifying if synthetic data is less biased from the outset. This *proactive*[41, 85] framing adds a new dimension to fairness research, expanding the set of techniques available to engineers and researchers.

Traditional synthetic data generation techniques—such as Bayesian Networks, Copulas, GANs, and VAEs—have long served as the main paradigms for generating tabular data, yet each suffers from well-documented limitations in handling high-dimensionality, heterogeneous features, or complex inter-variable dependencies [39, 51, 54, 94]. Recent work on LLM-based synthetic data generation has demonstrated promising improvements in producing realistic, high-quality tabular datasets [10, 32, 72], but these studies focus primarily on structural fidelity or predictive utility, leaving open questions about the fairness properties of the produced data. Our study is therefore situated at the intersection of these two lines of research: we build upon prior advances in LLM-based data generation while addressing a dimension—fairness—that the existing literature has not yet systematically investigated.

Our contributions are twofold. First, we extend the body of knowledge on synthetic data generation by benchmarking the current capabilities of frontier LLMs in producing fully synthetic tabular datasets. Unlike prior works that have primarily focused on structural fidelity or

predictive performance, we evaluate these datasets along three dimensions simultaneously: performance, structural quality, and fairness. By grounding our benchmark in the limitations identified in earlier generative approaches and the gaps highlighted by recent LLM-based studies, we position our work as a comprehensive analysis of the potential and boundaries of LLMs for synthetic tabular data generation.

Second, we contribute to the literature on fairness in software engineering. By treating fairness as a non-functional requirement within AI-enabled systems, our study positions synthetic data generation as a software engineering practice that can be integrated into existing processes, particularly within quality assurance and testing activities, where synthetic data can extend test suites to include fairness checks alongside functional and performance evaluations. In this way, our study not only explores the empirical capabilities of LLMs, but also provides actionable insights for integrating synthetic data generation into SE4AI practices.

Contribution to Machine Learning Fairness Engineering Research.

Prior works on synthetic data generation have largely focused on structural fidelity and performance, with limited attention to fairness and little connection to Software Engineering practices. Furthermore, bias mitigation strategies in Software Engineering often operate after model training, requiring significant resource efforts. This study addresses these gaps by empirically benchmarking frontier LLMs for fully synthetic tabular data generation, jointly assessing performance, structural quality, and fairness. Our findings contribute to the Software Engineering community by framing fairness as a non-functional requirement and positioning synthetic data generation as a practical tool for non-functional quality assurance activities.

3. Research Design

The *goal* of the study was to assess the extent to which the generation of sensitive tabular datasets through LLMs can contribute to addressing fairness requirements while maintaining performance in ML models, with the *purpose* of understanding the current state of this technique in terms of the overall level of fairness of the generated datasets. The *perspective* is of both researchers and practitioners. The former are interested in assessing both the functional and non-functional implications of synthetically generating sensitive datasets to better assist practitioners during the data collection and preparation phases. The latter are interested in determining whether LLMs can provide unbiased and valuable data sources that are applicable in real-world scenarios where lack of data hinders the development of ML solutions.

3.1. Research Questions

Our empirical study was structured around two main research questions. First, we aimed to assess the usefulness and practical value of the synthetic datasets, following the design of previous studies [9, 20]. This involved an empirical evaluation to determine whether synthetic datasets can serve as viable substitutes for real datasets. A key element of this assessment was the comparison of *predictive performance*, aiming to establish if and how closely synthetic data was able to match the predictive accuracy and reliability of real-world data. Hence, we asked:

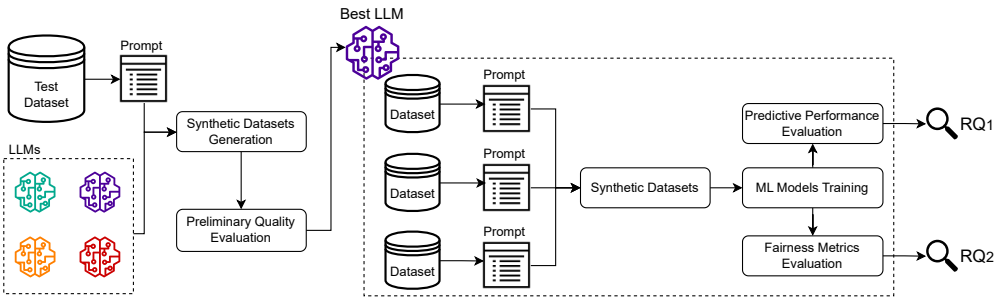

Figure 1: Overview of the research method proposed for our study.

> **RQ$_1$. Predictive Performance Evaluation.**
>
> *How does the predictive performance of machine learning models trained on synthetic datasets compare to those trained on real-world datasets?*

The second objective, which aligns with the overarching goal of this research, involved a comprehensive analysis of the *fairness* of these synthetic datasets in comparison to their original counterparts. This aspect is crucial, as the ethical implications of deploying machine learning models trained on synthetic data hinge on whether these datasets perpetuate, mitigate, or exacerbate biases present in the original data. As such, we formulated our second research question:

> **RQ$_2$. Fairness Evaluation.**
>
> *How does the fairness of machine learning models trained on synthetic datasets compare to those trained on real-world datasets?*

By investigating both the predictive capabilities and the fairness of synthetic datasets, our study sought to contribute to the broader discourse on the viability and ethical considerations of using synthetic data in machine learning applications. Figure 1 provides an overview of the research methods used to achieve our research questions, with the subsequent sections detailing the specific methods applied to each of them. In terms of reporting, we adhered to the *ACM/SIGSOFT Empirical Standards*.[2] Based on the nature of our study, we followed the *"General Standard"* guidelines.

3.2. Seed Datasets Selection

To conduct our study, we required reference datasets that could simulate realistic machine learning scenarios. These seed datasets served a dual purpose: first, they defined the objectives for synthetic data generation by describing the structure, features, and predictive tasks that synthetic instances were expected to replicate; second, they provided an oracle for evaluating the resulting models, enabling a direct comparison of performance and fairness trade-offs between models trained on synthetic versus real data. In this way, the seed datasets allowed us

[2] Available at: `https://github.com/acmsigsoft/EmpiricalStandards`.

to simulate practical conditions where synthetic data might be needed, while also establishing a baseline for rigorous assessment. In this respect, we intentionally selected *sensitive* datasets, i.e., datasets containing protected attributes, as the presence of sensitive features was a fundamental requirement for our study. Without such features, it would not have been possible to assess how effectively synthetic data generation mitigates bias across different demographic groups. As such, we drew on the work by Fabris et al. [31], who provided a comprehensive survey of the ML fairness literature, offering standardized documentation of datasets used to address, measure, or identify fairness-related issues in existing research.

Three considerations informed the selection process. First, the datasets needed to be of a *size manageable by a freely available LLM,*[3] ensuring that the model could generate synthetic data with a sufficient number of rows—on the order of thousands—to allow for meaningful analysis. This consideration was required to balance the capabilities of the LLM with the need for datasets large enough to produce statistically significant results. Second, the datasets had to include *more than one sensitive feature, e.g., sex*. This requirement allowed us to mitigate bias in our findings, as it ensured the LLM did not demonstrate better performance in some contexts over others. Lastly, the datasets were chosen from *various application domains*. This diversity was important to prevent skewed results due to the LLM's varying degrees of comprehension across different contexts. Based on these considerations, we selected three datasets previously used in the literature [19, 74] and that span various domains. The *German Credit* dataset [42] includes data on approximately 1,000 loan applicants, featuring sensitive attributes such as *'age'* and *'sex'*. The *Heart Disease* dataset [47] comprises around 1,000 patient records aimed at predicting coronary artery disease, also incorporating *'age'* and *'sex'*. Finally, the *Student* tracks the performance of approximately 600 students, alongside demographic and socioeconomic data, again considering *'age'* and *'sex'*.

Seed Dataset Selection

We selected three datasets for our study: German Credit [42], Heart Disease [47], and Student Performance [22].

3.3. Definition of the LLM-based Synthetic Dataset Generators

As a second step, we had to define the *subjects* of the study, namely the LLM-based synthetic dataset generators we would experiment with. This procedure involved (1) designing the prompts, which are the inputs that guide the LLM in generating synthetic datasets; and (2) selecting the most suitable LLM for our study. The following sections describe these steps.

3.3.1. Prompt Selection and Design.

The choice of prompting technique is critical in leveraging the full potential of LLMs [58]. In our work, we aimed to optimize model performance while minimizing inherent biases by selecting a technique that balances the need for contextual guidance with the flexibility to generalize across diverse inputs. We used the few-shot prompting technique [11], which incorporates examples within the prompt to guide the model toward better performance. We chose few-shot learning over other prompt engineering techniques (e.g., Chain-of-Thought) based on findings

[3] At least during the execution of the experiments.

by Wang et al. [32], which indicate that few-shot performs better for tabular data. Their survey highlights that manually selected examples in prompts improve performance, with a 14.7 F1-score advantage over random examples [65]. In addition, while moving from 1-shot to 2-shot improves performance, further increases offer minimal gains [17]. Based on these findings, we implemented 0-shot, 1-shot, and 2-shot prompts to generate synthetic datasets using an LLM.

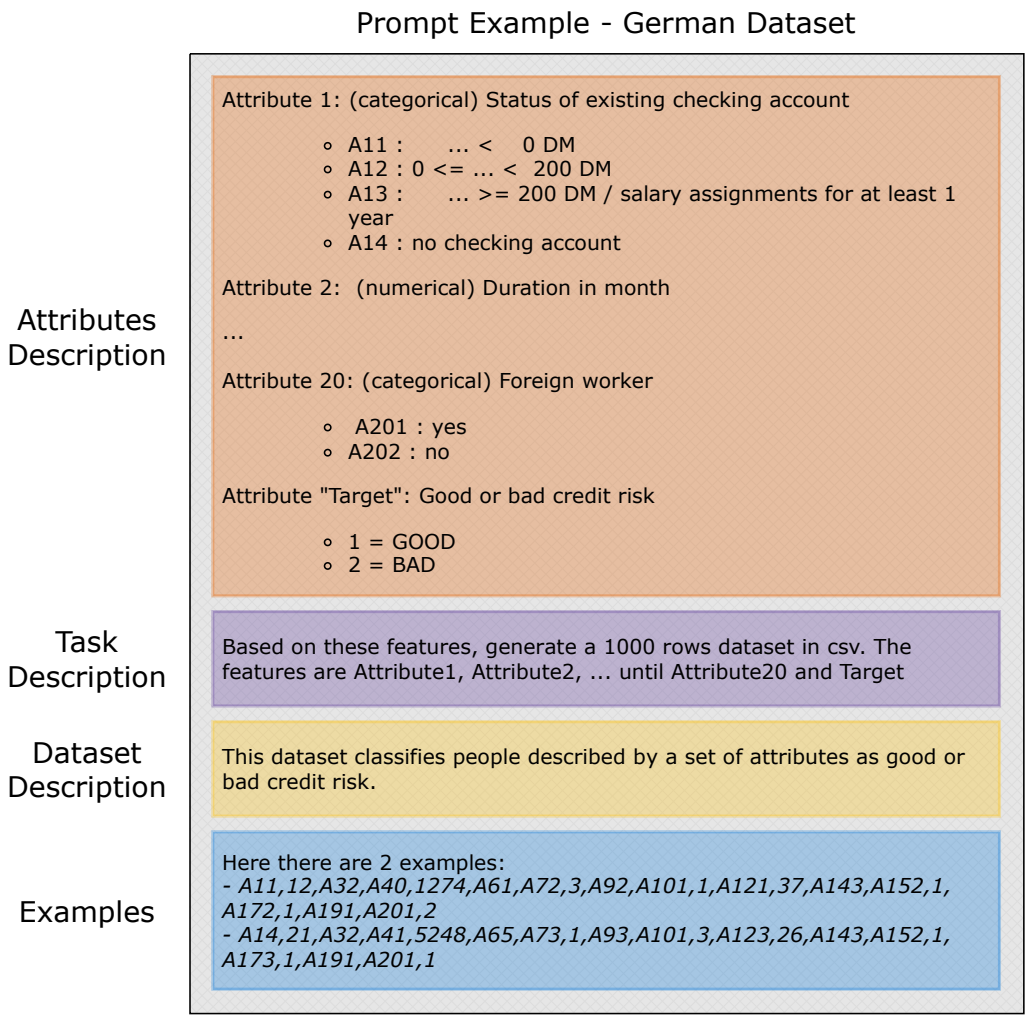

Figure 2: Summarized example of a 2-shot prompt constructed to generate the German Credit dataset. The values A11–A14 or A201–A202 represent categorical encodings from the original dataset (e.g., types of checking accounts or employment status). Sensitive features such as *age* and *sex* are also part of the dataset and are included as encoded attributes in the example provided.

Some considerations guided our approach to prompt design. First, we focused on generating datasets with sensitive features, aiming to evaluate the *generator's inherent ability to avoid bias* rather than *enforcing fairness* in the output: this approach aligned with our goal of assessing the current capabilities of LLMs, rather than developing new fairness-oriented prompt engineering

techniques. To this end, we provided only a factual and neutral description of the dataset (according to its original source) and each feature's possibel values (e.g., categorical options such as A11 = checking account < 0 DM, A12 = 0 ≤ balance < 200 DM, etc.), without including any additional instructions that could bias the generation toward fairness. Second, we instructed the LLM to generate datasets with the same number of rows (N) as the original dataset, with each row containing all features and the target variable, so as to enable a direct comparison between synthetic and real datasets. Third, we described the dataset's intended use, specifically the classification task it was created for. Finally, for the 1-shot and 2-shot prompts, we manually selected examples from the original datasets. For the 2-shot prompt, we chose examples with varying target and sensitive feature values, while for the 1-shot prompt, we selected samples representing privileged groups to avoid forcing the LLM toward equity.

It is important to clarify that our goal was not to optimize prompt engineering techniques, but to assess whether different levels of prompt complexity (0-shot, 1-shot, 2-shot) influence the quality and fairness of the generated datasets. In other words, prompt variation was treated as an exploratory factor to control for potential artifacts, rather than as an independent research focus.

As part of our online appendix [87], we provide detailed examples of the prompts used in our study, including 0-shot, 1-shot, and 2-shot variations. For the sake of understandability, Figure 2 shows an example of a 2-shot prompt crafted to generate the *German Credit* dataset [42].

> **Prompt Selection and Design**
>
> *The prompting technique selected was the* few-shot *prompting technique. In particular, we experimented with 0-shot, 1-shot, and 2-shot prompts.*

3.3.2. LLM Selection.

In contrast to prompt design, for LLM selection, we performed a preliminary benchmark to identify the most reliable model and then fixed this choice, so as to reduce confounding effects in the main study. We experimented with four widely used and freely available models: GPT-4o [69], Claude Sonnet 3.5 [5], LLAMA3 8B [4], and Phi3 mini 3B [1]. GPT-4o and Claude Sonnet 3.5 were accessed via their respective online platforms, while LLAMA3 8B and Phi3 mini 3B were run locally.

We focused on assessing each model's ability to generate *structurally similar* datasets, with the goal of identifying the model that best replicates the key statistical properties of the original data, ensuring structural fidelity. Structural fidelity is a precondition to synthetic dataset generation [27]; without it, the data may not accurately reflect the original patterns, hence indicating that an LLM would be unable to produce reliable synthetic datasets for further analysis.

The preliminary study targeted the *German Credit* dataset [42], chosen as a representative seed dataset due to its extensive use in research [31]. Conducting a complete preliminary assessment on all datasets would have been both costly, due to the need to experiment with multiple LLMs and prompts, and potentially redundant. Indeed, the preliminary evaluation is independent from the dataset domain [27, 91], as the structural characteristics and generation capabilities of the LLMs are not inherently tied to the dataset's content. As such, the use of a single dataset for this preliminary assessment does not compromise the generalizability of our conclusions.

From a technical standpoint, we tasked each LLM with generating 250 new entries (25% of the original dataset) using the specific prompts previously designed, similarly to the example in Figure 2. Firstly, we observed that while GPT4o and Claude Sonnet 3.5 successfully generated

the synthetic dataset, LLAMA3 8B and Phi3 mini 3B failed to do so. This inability could be attributed to limitations in processing capacity, prompt comprehension, or model constraints that affect their performance in handling complex data generation tasks. In any case, LLAMA3 8B and Phi3 mini 3B could not be assessed and were therefore excluded from our analysis.

The generation process resulted in the collection of six synthetic datasets, three from each of two remaining LLMs—GPT4o and Claude Sonnet 3.5—with three different prompts. We then evaluated the structural quality of these datasets using metrics established by Elouataoui et al. [28], which have been previously applied in assessing dataset quality [76]. Specifically, we computed four metrics: (1) *Completeness*, defined as the proportion of non-missing values across all rows and features; (2) *Uniqueness*, defined as the ratio of distinct rows over the total number of rows, thereby identifying redundant or repeated entries; (3) *Consistency*, defined as the proportion of values adhering to the expected schema (e.g., categorical codes restricted to valid options); and (4) *Readability*, defined as the proportion of entries free from malformed tokens, typographical errors, or nonsensical values, ensuring syntactic validity of the dataset. In other words, readability does not simply refer to schema replication, but to whether each entry is interpretable and correctly formatted. We computed these metrics for every feature in both the synthetic and real datasets, finally assessing their fidelity. Due to space limitations, a detailed analysis of the preliminary results, including datasets, code, and plots, is provided in our online appendix [87], while the main insights are discussed in the following.

The results of this analysis highlight clear differences between the two models. Both GPT-4o and Claude 3.5 achieved 100% completeness and consistency across all prompt types, confirming that the generated datasets contained no missing values and adhered to the expected schema. In terms of readability, the synthetic datasets were comparable to the originals: all values were correctly formatted and interpretable, with no malformed or incoherent entries. The main difference emerged in uniqueness. GPT-4o consistently generated more diverse datasets across all prompts, while Claude 3.5 showed noticeable redundancy, particularly under the 2-shot configuration, where duplicate rows became more frequent. Overall, these results suggest that GPT-4o is more robust in producing structurally sound and varied datasets, whereas Claude 3.5 tends to repeat values as prompt complexity increases.

To provide a qualitative view of how LLMs generate synthetic data based on the descriptions of the attributes in the original dataset, Figure 3 compares the *"Age"* distribution between the original dataset (left), GPT-4o (center), and Claude 3.5 (right). Starting with GPT-4o, we observed that the synthetic dataset mirrors the age range of the original dataset, spanning from approximately 20 to 70 years, hence indicating that GPT-4o can capture the age boundaries appropriately. Both distributions exhibit a decline in frequency as age increases, though the decline is higher in the original dataset. This suggests that while GPT-4o successfully replicated the overall trend, it did so with less precision than real data—which is, however, expected since the generation is not aware of the real distribution. Both the original and the GPT-4o synthetic datasets lack significant outliers or extreme age values, making GPT-4o able to avoid the introduction of unrealistic data. However, a key difference emerges in the distribution peaks: the original dataset has a peak in the 25–30 age range, while the synthetic dataset generated by GPT-4o peaks at 40–45, indicating a shift in the age distribution that reveals differences between the two datasets. Turning to Claude 3.5, the synthetic dataset exhibits more significant deviations from the original. The age range in the synthetic dataset is narrower, spanning from 25 to 60 years, compared to the original dataset's range of 20 to 70 years. Additionally, the synthetic dataset generated by Claude 3.5 displays a pronounced peak in the 50–55-year range, which is absent in the original dataset. Given these discrepancies, particularly in the structural and manual

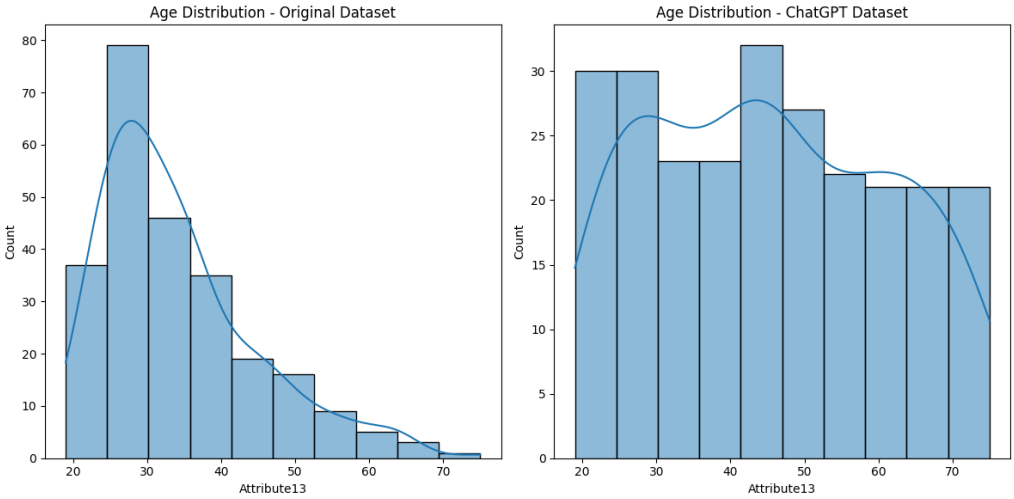

Figure 3: Distribution of the *"Age"* attribute within the original and the synthetic datasets.

investigations, we decided to select GPT-4o for further experiments because it generated more realistic representations of the dataset. Moreover, this decision is supported by the survey by Fang et al. [32] on LLMs for tabular data, which found the GPT model to be competent in performing complex table reasoning tasks, particularly when paired with reasoned prompting techniques.

Interpreting these observations, the synthetic data generated by both LLMs tends to follow a more pronounced normal distribution, with some differences in dispersion. While the general trend resembles the original dataset, the synthetic distributions are more spread out. This could affect model performance, as these differences may lead to divergence between models trained on synthetic versus original data. At the same time, broader or more balanced distributions may positively impact fairness by reducing the dominance of majority groups and better representing minority ones. It is important to emphasize that these properties should not be interpreted as universally beneficial or detrimental. Their value depends on the Software Engineering context in which synthetic data is applied. For instance, in quality assurance and regression testing, engineers may prefer distribution-preserving synthetic datasets, since these mirror real-world data more closely and thus allow models to be validated under realistic conditions without exposing sensitive records. In contrast, during requirements engineering or fairness-by-design activities, distribution-balancing synthetic datasets may be more useful, as they enable practitioners to explore how models behave when demographic groups are equitably represented, helping to uncover potential sources of unfairness early in the process. Finally, in operational MLOps pipelines, both variants have a role: distribution-preserving synthetic data can be used to monitor fairness regressions under realistic workloads, while distribution-balancing data provides a way to stress-test models against underrepresented groups. These preliminary insights indicate that synthetic data can support different engineering objectives depending on whether similarity or divergence with the original distribution is prioritized. This dual role is further explored in $\mathbf{RQ}_1$ and $\mathbf{RQ}_2$.

LLM Selection

GPT-4o was selected for our empirical study due to its ability to generate diverse and structurally consistent datasets. This decision aligns with findings from previous research [32], which highlighted GPT's effectiveness in complex table reasoning tasks, especially when supported by prompt engineering techniques.

3.4. Synthetic Datasets Generation Process

As a final step, we detail the procedure employed to define the *objects* of the study. These pertain to the real and synthetic datasets that will be assessed to address the two research questions of the study. As for the former, we considered the raw data of the three datasets considered in the study, i.e., *German Credit* [42], *Heart Disease* [47], and *Student Performance* [22]. As for the latter, we applied the designed few-shot prompts (Section 3.3.1) to GPT4o (Section 3.3.2), generating synthetic datasets for each of the three real datasets we aimed to replicate.

From an operational perspective, we executed the 0-shot, 1-shot, and 2-shot prompts for each dataset, incorporating descriptions relevant to the dataset features. These prompts were run on ChatGPT [69], using GPT4o with default settings. Each prompt execution was repeated *three times* to mitigate the inherent variability in LLM outputs [45], which arises due to stochastic processes within the model. By repeating the executions, we increased the reliability of our analysis, ensuring that any observed patterns or anomalies were not the result of a single, potentially atypical generation, but rather reflective of consistent behavior across multiple runs.

Overall, this process resulted in the generation of three synthetic versions of each dataset per prompt, totaling nine synthetic datasets per original dataset and 27 synthetic datasets. Combined with the three original datasets, *our study encompasses 30 sensitive tabular datasets*. All the prompts used and the collected datasets are available in our online appendix [87].

3.5. Research Method for RQ_1 — Performance Analysis

To address $\mathbf{RQ_1}$, we leveraged the 30 sensitive tabular datasets in our study, experimenting with 27 machine learning models provided by the *Lazy Predict* library.[4] This Python library facilitates model comparison by automating the process and removing the need for manual parameter tuning. We employed ten-fold cross-validation [52] to partition the datasets and evaluate the performance of each model. The models trained on both synthetic and original datasets were then assessed using standard performance metrics, including *F1-score* and *accuracy* [25].

However, this analysis alone would not be sufficient to fully determine the practical applicability of synthetic datasets. In fact, while the initial analysis offers valuable insights into the model's performance in terms of accuracy and consistency when compared to models trained on real datasets, it does not address the generalization capabilities of these models when applied to real-world scenarios. In other terms, to fully understand the impact of synthetic datasets, it is also key to evaluate how well models trained on synthetic data perform when predicting outcomes on actual, real-world data. This additional step ensures that *the synthetic datasets not only support robust model training but also translate effectively into practical applications*. For this reason, we extended our analysis to evaluate the predictive performance of models trained on synthetic datasets and tested on real-world data, specifically using the original dataset samples.

[4]The *Lazy Predict* library: `https://github.com/shankarpandala/lazypredict`.

This evaluation followed the same design as the first analysis, with a key difference aiming at simulating a real-world scenario: instead of using cross-validation, we trained each of the 27 models provided by *Lazy Predict* on the entire synthetic dataset. The trained models were then evaluated on the full original dataset, and their performance was assessed using standard metrics, such as the *F1-score* and *accuracy*.

To verify the significance of our findings, we finally applied statistical tests. We first assessed the normality of our data to determine the appropriate statistical methods. Using the Shapiro-Wilk test [38] with a significance level of $\alpha = 0.05$, we found that the studied datasets did not always conform to a normal distribution, suggesting the use of non-parametric tests. As such, we applied the Friedman test [43], a method for detecting differences across multiple groups: it allowed us to identify statistically significant differences among models trained on real and synthetic data by testing the null hypothesis that no significant differences exist. However, while the Friedman test can indicate the presence of differences, it does not specify which groups differ. Therefore, when the Friedman test showed a significant result (ρ-value < 0.05), we conducted a post-hoc analysis using the Nemenyi test [43, 66] to pinpoint specific pairwise differences between the models, allowing us to identify which ML models exhibited statistically significant variations. We performed multiple statistical evaluations across the datasets generated at each prompt execution: as such, we applied the Bonferroni correction to control for the increased risk of Type I errors that arises when conducting multiple comparisons [12].

3.6. Research Method for RQ_2 — Fairness Analysis

To address $\mathbf{RQ_2}$, we evaluated the 27 models trained with *Lazy Predict* using three key fairness metrics: *statistical parity difference* (SPD) [2], *equal opportunity difference* (EOD) [40], and *average odds difference* (AOD) [2]. These metrics, widely recognized as standard tools for assessing the fairness of machine learning models [34, 35, 70], allowed us to determine how equitably the models perform across different demographic groups in the data. Specifically, SPD measures the disparity in favorable outcomes between privileged and unprivileged groups, highlighting potential biases in the model's predictions. EOD assesses the maximum difference in true positive rates between subgroups, revealing if one group is unfairly advantaged in receiving favorable outcomes. Lastly, AOD combines the differences in false and true positive rates between privileged and non-privileged groups, offering an overall assessment of models' fairness. For all these metrics, a value of zero indicates perfect fairness, with both groups receiving equal treatment. Deviations from zero signal bias, where positive values favor the privileged group and negative values favor the non-privileged group. A value of 1 reflects a strong bias toward the privileged group, while -1 indicates a strong bias toward the non-privileged group.

Note that, unlike in $\mathbf{RQ_1}$, where we examined models trained on synthetic data and tested on real data, for $\mathbf{RQ_2}$ we could only target models trained using either synthetic or real datasets. This is because fairness metrics are designed to assess how the models perform across different demographic groups based on the data they were trained on, rather than evaluating their generalization to new, unseen data [37]. In other words, the fairness evaluation is intrinsically linked to the training data's distribution, as it measures the model's ability to treat various groups equitably within that specific context. Through our evaluation, we assessed the practical applicability of the models from a social sustainability perspective [59]. By focusing on fairness metrics that directly measure the models' treatment of different demographic groups based on their training data, we evaluated the extent to which these models may perpetuate or exacerbate biases in their decision-making processes.

We computed these fairness metrics using AI Fairness 360 [8], focusing on the sensitive features available in the datasets (see Section 3.2). As recommended in the literature [19], the analysis was conducted under the *multiple-sensitive* attribute setting, which accounts for combinations of sensitive attributes, hence capturing the potential intersectional biases that may arise when considering multiple attributes simultaneously.

We collected and finally compared these metrics for both the synthetic and original datasets. Additionally, we performed statistical analyses using the same methods applied in $\mathbf{RQ}_1$, i.e., the Friedman test to detect significant differences between the models, followed by the Nemenyi post-hoc test to identify which specific model pairs exhibited statistically significant differences.

4. Analysis of the Results

In the following sections, we present and discuss the results of the empirical study. First, we provide preliminary insights into the structure of the generated datasets. Afterward, the discussion is organized according to the corresponding **RQ**.

4.1. Synthetic Datasets' Structure

Before discussing our results, we first examine the synthetic datasets from a structural perspective, focusing on their sensitive features. For the sake of space, being the results for each datasets very similar, we present only the analysis of sensitive features derived from the synthetic *German Credit* dataset. Comprehensive analyses of the other datasets are available in our online appendix [87]. Table 1 and Table 2 provide a statistical description for the variables age and sex, respectively. As shown in Table 1, all the datasets exhibit similar values for the minimum, first quartile, and median. However, the first generated dataset (row eight) with the 2-shot prompt stands out due to the presence of implausible outliers, such as a maximum age of 460 years, which significantly skews the other statistical metrics. While the implausible maximum in one synthetic run clearly inflates dispersion, such anomalies were relatively rare; nevertheless, even in their absence, differences in maximum and standard deviation persist for several synthetic datasets, reinforcing that distributional shifts are present and should be interpreted with care.

The original dataset has a standard deviation of 11.37, while the synthetic datasets exhibit much lower values, indicating less variation in this variable compared to the original data. These differences drove us to two considerations. First, while the presence of implausible outliers (e.g., a maximum age of 460 years) highlights that LLM-based synthetic data generation may occasionally struggle with ensuring fully realistic value distributions, such anomalies were relatively rare, and the overall statistical consistency observed in most datasets indicates only a partial tendency of LLMs to approximate structural patterns in tabular data.

Second, and more importantly, the lower standard deviation observed in the synthetic datasets compared to the original one may in some cases suggest a potential advantage regarding fairness: the distribution of age values appears more balanced, with a more equitable representation between privileged (age above the mean) and unprivileged (age below the mean) groups. However, this interpretation is context-dependent. In scenarios where imbalances reflect sampling biases or underrepresentation, a reduced variance can indeed improve fairness. Conversely, in domains where skewed age distributions mirror real demographic characteristics, artificially reducing variance may distort reality and not correspond to genuine fairness gains.

This initial evidence suggests that synthetic data generation may, in certain contexts, contribute not only to maintaining predictive structure but also to mitigating biases related to sensitive attributes. However, this effect is conditional: when imbalances result from sampling biases

or underrepresentation, synthetic generation can help rebalance the data. Conversely, when disparities reflect genuine demographic characteristics, such adjustments may not correspond to real fairness improvements.

Dataset	Min	First Qu.	Median	Mean	Third Qu.	Max	Std. Dev.
German Credit	19	27	33	35.546	42	75	11.3755
Synthetic German Credit 0-shot A	19	25.5	30	30.9804	38	50	8.2112
Synthetic German Credit 0-shot B	20	25	27	28.9145	33	50	6.9621
Synthetic German Credit 0-shot C	25	32	38	37.3407	41	50	5.4590
Synthetic German Credit 1-shot A	23	30	33	33.3101	36	54	4.0469
Synthetic German Credit 1-shot B	14	23.5	30	32.7168	46	52	12.0919
Synthetic German Credit 1-shot C	22	29.75	31	32.5010	35	50	4.4998
Synthetic German Credit 2-shot A	20	31	38	117.237	164.25	460	133.2761
Synthetic German Credit 2-shot B	20	27	30	30.206	33	47	2.9550
Synthetic German Credit 2-shot C	20	25	33	32.054	37	50	7.8138

Table 1: Statistical summary of age for the German Credit Dataset and Synthetic Variants

Table 2 presents the distribution of the sex attribute in both the original *German Credit* dataset and the synthetic versions. As observed, the previously under-represented categories (e.g., Male Divorced/Separated, Male Married/Widowed, and Female Single) generally see an increase in representation within the synthetic datasets. However, there are notable exceptions, such as the Male Divorced/Separated category in the "Synthetic German Credit 1-shot B" dataset and the Female Single category in the "Synthetic German Credit 0-shot B" and "Synthetic German Credit 0-shot C" datasets (Table 2). These observations suggest that LLM-generated datasets can enhance the representation of minority groups, potentially addressing imbalances present in the original data. Although not uniformly consistent across all synthetic datasets, the general trend toward increased diversity indicates that synthetic generation could serve as a tool for promoting fairness at the data level. However, the occasional persistence of under-representation in some categories also highlights the need for careful prompt design and validation to ensure the desired fairness improvements are systematically achieved.

All in all, this preliminary structural analysis of the synthetic datasets highlights both the potential and the challenges of using LLMs for synthetic data generation. While occasional anomalies suggest the need for careful quality control, the overall preservation of key statistical properties and the improvement in the representation of underprivileged groups indicate that LLMs can serve as promising tools for promoting fairness at the data generation stage. These observations further motivate our research questions.

*4.2. **RQ$_1$** - Performance Analysis*

As discussed in Section 3.5, we performed two performance analyses: (1) the first aimed at validating the predictive performance of models trained and tested on synthetic datasets compared to those trained and tested on real datasets; and (2) the second assessing the generalizability of models trained with synthetic datasets to real data.

Cross-Validation Results. Table 3 reports the average and standard deviation of accuracy and F1-score for models trained on both real datasets (grey rows) and synthetic datasets (white rows). Results are shown for each prompt configuration used to generate the synthetic datasets (0-shot, 1-shot, and 2-shot). All values are averaged over 27 machine learning models, since our

Dataset	M Divorced/Separated	F Divorced/Separated/Married	M Single	M Married/Widowed	F Single
German Credit	50	310	548	92	0
Synthetic German Credit 0-shot A	212	210	253	305	43
Synthetic German Credit 0-shot B	205	200	203	197	201
Synthetic German Credit 0-shot C	248	252	249	3	249
Synthetic German Credit 1-shot A	241	248	121	256	140
Synthetic German Credit 1-shot B	14	178	346	362	110
Synthetic German Credit 1-shot C	250	60	249	251	198
Synthetic German Credit 2-shot A	286	10	288	289	127
Synthetic German Credit 2-shot B	248	246	248	250	8
Synthetic German Credit 2-shot C	397	200	196	204	3

Table 2: Summary of marital status based on sex distribution for the German Credit Dataset (M = Male, F = Female)

	Accuracy		F1-Score		Test Outcome
Dataset Name	**Avg.**	**Std Dev.**	**Avg.**	**Std Dev.**	
German Credit	0.725	0.038	0.711	0.043	–
Synthetic German Credit 0-shot	0.908	0.095	0.902	0.122	↑
Synthetic German Credit 1-shot	0.838	0.079	0.831	0.104	↑
Synthetic German Credit 2-shot	0.924	0.105	0.917	0.134	↑
Heart Disease	0.878	0.112	0.871	0.135	–
Synthetic Heart Disease 0-shot	0.498	0.016	0.488	0.038	↓
Synthetic Heart Disease 1-shot	0.499	0.015	0.485	0.038	↓
Synthetic Heart Disease 2-shot	0.502	0.019	0.494	0.043	↓
Student Performance	0.839	0.070	0.832	0.082	–
Synthetic Student Perf. 0-shot	0.722	0.074	0.715	0.095	–
Synthetic Student Perf. 1-shot	0.518	0.019	0.506	0.043	↓
Synthetic Student Perf. 2-shot	0.513	0.019	0.498	0.036	↓

Table 3: Performance comparison (Accuracy and F1-score) between original datasets and their synthetic counterparts. Each cell reports the average and standard deviation across 27 ML models. The **Test Outcome** column highlights statistically significant differences from the original dataset, as determined by Friedman and Nemenyi tests. Light purple cells indicate significance; ↑ marks a statistically significant increase, while ↓ marks a significant decrease. Representative confusion matrices are provided in the online appendix [87].

goal is to capture the overall effect of synthetic datasets on performance rather than the behavior of specific algorithms. A breakdown by individual model is provided in the online appendix [87] and further discussed in Section 6. In addition, for each prompt setting we generated three synthetic datasets and averaged results across them. For instance, "Synthetic German Credit 0-shot" refers to the aggregated performance of models trained on the three 0-shot synthetic instances of the German Credit dataset.

The *Test Outcome* column highlights statistically significant differences from the original dataset, as determined by Friedman and Nemenyi tests. Light purple cells indicate significance; an upward arrow (↑) marks a significant improvement, while a downward arrow (↓) marks a significant decline. Representative confusion matrices are provided in the online appendix [87].

This table should therefore be interpreted as follows: the averages and standard deviations provide a high-level view of model performance across multiple synthetic datasets, while the statistical outcomes ensure that observed differences are robust across multiple runs and models.

Looking at the results on the *German Credit* dataset (Table 3), we observed statistically

significant improvements in both accuracy and F1-score when using synthetic data generated with prompts, especially with the 2-shot prompt. This indicates that the synthetic data generation process effectively captured the patterns of the original dataset, with prompt complexity playing a key role. Interestingly, the 0-shot prompt outperformed the 1-shot prompt, challenging the assumption that more examples lead to better performance. This suggests that a well-structured prompt with minimal examples may suffice for this dataset, potentially due to its categorical nature and limited value ranges. Additionally, the models trained on synthetic data exhibited a slightly higher variability, as reflected in the standard deviations, suggesting that synthetic data can result in less consistent performance across different model instances.

In contrast, the results on the *Heart Disease* and *Student Performance* datasets (Table 3) were opposite. The models trained with real datasets consistently outperformed their synthetic counterparts across all prompting strategies. The statistical analysis confirmed significant decreases in performance with synthetic data. These findings may highlight the complexity and domain specificity of data synthesis, particularly for medical datasets like *Heart Disease*, where interdependent features may be harder to replicate accurately with synthetic data. Similarly, while the *Student Performance* dataset showed a less pronounced performance gap, it still indicates that educational data may pose challenges for synthetic data generation, though not as severe as in the medical domain. In addition to context, the shape and structure of the datasets may have played a role in our results. Features in the *German Credit* dataset were mostly categorical, and they could assume a few different possible values, making it easier for the LLM to generate plausible and complete samples. In contrast, more features in the other datasets required interpretation by the LLM during the generation, as they were numerical or could assume a wide range of values.

The number of prompt examples (0-shot, 1-shot, 2-shot) did not consistently correlate with improved performance. In fact, the 0-shot prompt often delivered the best results, particularly in the case of the *Student Performance* dataset, where the model's predictive performance was closest to that of the original data. This challenges the typical assumption in few-shot learning that more examples lead to better outcomes, suggesting that simpler prompts may sometimes be more effective for synthetic data generation.

In conclusion, while synthetic datasets can achieve competitive performance in certain cases, especially with minimal prompting, they generally exhibit lower predictive performance compared to real datasets. In addition, the variability in performance across domains highlights the need for domain-specific considerations in synthetic data generation.

Dataset Name	**Model**	**Accuracy**	**F1-Score**
Synthetic German 0-shot C	KNeighborsClassifier	0.679	0.604
Synthetic German 1-shot A	CalibratedClassifierCV	0.679	0.687
Synthetic German 2-shot C	NearestCentroid	0.641	0.644
Synthetic Heart 0-shot B	BernoulliNB	0.701	0.685
Synthetic Heart 1-shot A	AdaBoostClassifier	0.692	0.688
Synthetic Heart 2-shot A	DecisionTreeClassifier	0.583	0.57
Synthetic Student 0-shot A	BernoulliNB	0.88	0.883
Synthetic Student 1-shot C	NearestCentroid	0.689	0.642
Synthetic Student 2-shot B	LGBMClassifier	0.683	0.685

Table 4: Comparison of accuracy and F1-score of the best models in the real-world evaluation.

Real-world Evaluation. In this analysis, we aimed to assess the generalizability of mod-

els trained on synthetic datasets by testing them on real-world data. The underlying assumption is that if models trained on synthetic data can perform well when applied to real datasets, it would suggest that synthetic data can serve as a viable alternative to real data in practice, even when initial model validation shows lower performance. Table 4 presents the results of this evaluation, reporting the best-performing models for each synthetic dataset across different prompt configurations. These models were automatically selected from the 27 classifiers trained and evaluated using LazyClassifier, which systematically explores a wide pool of algorithms. To aid interpretation, we briefly note the type of each model: `KNeighborsClassifier` and `NearestCentroid` are distance-based classifiers, `BernoulliNB` is a probabilistic Naïve Bayes variant, `DecisionTreeClassifier` and `AdaBoostClassifier` are tree-based ensemble methods, and `LGBMClassifier` is a gradient boosting algorithm. The appearance of these diverse classifiers reflects the variation in how different modeling approaches interact with synthetic data. Due to space limitations, we focus on the top model for each prompt engineering technique, with full results available in our online appendix [87].

From the table, various considerations can be drawn. For the *German Credit* dataset (Table 4), the highest performance was achieved with the 1-shot prompting configuration, which resulted in an accuracy of 0.679 and an F1-score of 0.687. These values are relatively close to the performance obtained by models trained on real datasets (the performance reported in Table 3 may offer a good basis for comparison), with a loss of approximately 0.04 in accuracy and 0.02 in F1-score. This slight performance drop suggests that, despite the use of synthetic data, the models trained on synthetic datasets can still generalize well when tested on real data. Such small differences indicate that synthetic data could serve as a viable alternative in practical applications.

When evaluating the *Heart Disease* dataset (Table 4), the best-performing model was trained using the 0-shot prompting configuration, achieving an accuracy of 0.701 and an F1-score of 0.685. Although these results are significantly lower than those achieved by models trained on real data (accuracy of 0.878 and F1-score of 0.871), they still demonstrate valuable generalization potential. When compared to the average accuracy and F1-score from the model evaluation phase (Table 3), the improvements are substantial. More generally, the performance drop, while notable, may be justified if it leads to gains in fairness, as explored in $\mathbf{RQ}_2$, making the trade-off between performance and fairness acceptable in domains such as healthcare, where mitigating bias is critical to ensure equal treatment.

Moving to *Student Performance* (Table 4), the highest performance was observed with the 0-shot prompting configuration, yielding an accuracy of 0.88 and an F1-score of 0.883. Interestingly, these values exceed the performance of models trained on real data (accuracy of 0.839 and F1-score of 0.832), highlighting a case where synthetic data may even lead to better-performing models. This suggests that synthetic data can sometimes outperform models trained on real datasets, likely due to the effectiveness of the 0-shot prompt and the characteristics of the dataset.

Per-algorithm Analysis. An inspection of the per-algorithm results (raw data are in our online appendix[87]) provides further nuance to these findings. For the *German Credit* dataset, the outcomes across 22 classifiers were highly dispersed, with accuracy ranging around 0.55 on average (SD $\approx$ 0.11) and F1-scores showing even greater variability (mean $\approx$ 0.53, SD $\approx$ 0.13). This confirms that while aggregated averages capture the overall behavior, individual models react very differently to synthetic data. In contrast, the *Heart Disease* dataset yielded more stable results across the models (accuracy mean $\approx$ 0.60, SD $\approx$ 0.06; F1 mean $\approx$ 0.54, SD $\approx$ 0.09), sug-

gesting that here the aggregated results are more representative of per-algorithm trends. Finally, the *Student Performance* dataset displayed the largest variance across models, with accuracy and F1 averaging around 0.66 but with a high dispersion (SD $\approx$ 0.14). Some classifiers achieved very high scores, while others degraded substantially, indicating that synthetic data interacts unevenly with different modeling assumptions.

Summary. In summary, two key conclusions emerge from this analysis. First, while models trained on synthetic datasets tend to perform worse than those trained on real datasets, the performance gap is often not dramatic. In some cases, models trained on synthetic data perform comparably or even better than those trained on real data, indicating that synthetic data can be a practical alternative in various scenarios. Second, simpler prompting strategies, particularly 0-shot learning, tend to yield the best results across various datasets. This finding reinforces the notion that prompt complexity does not always translate to better performance, and that simplicity in prompt design can be highly effective when generating synthetic datasets for real-world applications.

Key findings of $\mathbf{RQ}_1$.

While models trained on synthetic datasets exhibit lower performance during cross-validation, they generalize relatively well when applied to unseen real data. The performance gap between synthetic and real data-trained models is often small, with simpler prompting strategies, such as 0-shot learning, yielding the best results. Our findings suggest that synthetic data may be a viable alternative in practice, particularly if it should accompanied by improvements in fairness.

4.3. $\boldsymbol{RQ_2}$ - *Fairness Analysis*

Table 5 reports the average and standard deviation of the three fairness metrics (SPD, EOD, AOD) for models trained on real datasets (grey rows) and their synthetic counterparts (white rows). For each dataset and prompt configuration (0-shot, 1-shot, and 2-shot), results were averaged across 27 machine learning models and three independent runs of synthetic dataset generation. Each metric is accompanied by a *Test* column, which indicates whether the difference with respect to the original dataset is statistically significant according to Friedman and Nemenyi tests. Light purple cells highlight significance: an upward arrow (↑) marks a statistically significant improvement in fairness, while a downward arrow (↓) indicates a significant decline.

From the Table (5), several observations can be made. First, while the absolute differences in average fairness values may appear small (often below 0.1), our statistical analysis (Friedman and Nemenyi tests across 27 models and three generation runs) confirmed that many of these differences are systematic and statistically significant. In particular, synthetic datasets generated using 1-shot learning tend to reduce unfairness in terms of AOD across the *German Credit*, *Heart Disease*, and *Student Performance* datasets. We note, however, that in some cases (e.g., the Student dataset), the 0-shot configuration yields values closer to zero, showing that improvements are not universal and must be interpreted with care. Overall, these results suggest that 1-shot prompts can find a balance between simplicity of the prompt and quality of the guidance provided to the LLM, leading to fairer outcomes. While such improvements are sometimes accompanied by performance declines, as observed in $\mathbf{RQ}_1$, the overall performance of the models—particularly in terms of F1-score—remains reasonably competitive.

As a second observation, we noticed that increasing the complexity of prompts does not always lead to better results. The synthetic datasets built using 2-shot learning on the *German*

	SPD			EOD			AOD		
Dataset Name	**Avg.**	**Std.**	**Test**	**Avg.**	**Std.**	**Test**	**Avg.**	**Std.**	**Test**
German Credit	0.083	0.076	–	0.009	0.108	–	0.036	0.080	–
Synthetic German Credit 0-shot	0.173	0.043	–	0.044	0.052	–	0.048	0.067	–
Synthetic German Credit 1-shot	0.055	0.050	↑	-0.036	0.086	↓	-0.003	0.062	↑
Synthetic German Credit 2-shot	-0.441	0.067	↓	-0.425	0.212	↓	-0.250	0.103	↓
Heart Disease	-0.016	0.014	–	0.002	0.017	–	0.093	0.009	–
Synthetic Heart Disease 0-shot	-0.026	0.046	–	-0.031	0.061	↓	-0.028	0.045	↑
Synthetic Heart Disease 1-shot	-0.004	0.033	–	-0.085	0.060	↓	-0.011	0.033	↑
Synthetic Heart Disease 2-shot	-0.038	0.041	–	-0.038	0.054	–	-0.038	0.040	↑
Student Performance	0.186	0.066	–	0.079	0.063	–	0.073	0.071	–
Synthetic Student Perf. 0-shot	-0.006	0.063	–	0.030	0.093	–	-0.023	0.063	–
Synthetic Student Perf. 1-shot	-0.031	0.136	↑	-0.039	0.173	–	-0.032	0.138	↑
Synthetic Student Perf. 2-shot	-0.044	0.106	↑	-0.011	0.143	↑	-0.038	0.109	↑

Table 5: Fairness metrics (SPD, EOD, AOD) for models trained on real datasets (grey rows) and synthetic counterparts (white rows). Each value is averaged across 27 machine learning models and three independently generated synthetic datasets. For each metric, the **Test** column reports whether differences compared to the original dataset are statistically significant, as determined by Friedman and Nemenyi tests. Light purple cells indicate significance; ↑ marks a statistically significant improvement in fairness, while ↓ marks a significant decrease. Detailed per-model fairness values are available in the online appendix [87].

Credit dataset, for instance, deteriorated fairness across all metrics. This confirms findings from previous studies [32], which observed that adding complexity in table reasoning tasks does not necessarily yield better results and can, in some cases, negatively affect the outcomes. In contrast, 0-shot and 1-shot prompts yield better or at least stable fairness metrics, further supporting the idea that simplicity in prompting can be more effective in this context.

Thirdly, Table 5 also shows that the results for EOD exhibit a different pattern compared to AOD and SPD. In several instances, e.g., the *Heart Disease* dataset under 0-shot and 1-shot prompts, the synthetic datasets negatively impact EOD. This suggests that while the synthetic data improves the overall fairness of outcomes (as indicated by SPD and AOD), it may still struggle with equalizing true positive rates across different subgroups. EOD measures the ability of the model to correctly identify favorable outcomes for different groups, and a decrease in this metric points to disparities in this regard. This highlights the need for further refinement in synthetic data generation to ensure that improvements in one fairness metric (e.g., AOD) do not come at the expense of others (e.g., EOD). In this sense, it seems that our research calls for novel, explicit fair prompt engineering approaches that may further improve our results.

Finally, looking at SPD, the results generally suggest that synthetic datasets perform well in terms of overall fairness. Most of the models trained on synthetic data do not worsen fairness, and in some cases, the use of synthetic data even leads to statistically fairer outcomes. This trend is consistent across the 0-shot and 1-shot prompts for all datasets, reinforcing the idea that synthetic data can be a viable tool for mitigating bias. The exception, again, comes from the 2-shot learning on the *German Credit* dataset, where fairness deteriorates across all metrics. This further underscores the limitations of increasing prompt complexity.

Per-algorithm Analysis. Looking at the per-algorithm fairness outcomes, we observed different behaviors across datasets. For the *German Credit* dataset, results were highly dispersed across 27 classifiers, with overall mean differences averaging around -0.06 but with a large vari-

ance (SD ≈ 0.36). This indicates that while some models improved fairness, others amplified disparities, leading to inconsistent outcomes. By contrast, the *Heart Disease* dataset showed much greater stability across 26 classifiers, with all three fairness metrics clustering tightly around the mean (overall mean difference ≈ -0.02, SD ≈ 0.04). This suggests that here aggregated averages are more representative of model-level behavior. Finally, the *Student Performance* dataset exhibited intermediate behavior: averages were close to zero (overall mean difference ≈ -0.01), yet standard deviations were large (up to ≈ 0.19 for equal opportunity difference), showing that certain classifiers benefited from synthetic data while others did not. Taken together, these results confirm that fairness outcomes are not only dataset-dependent but also model-dependent, reinforcing the need to consider per-algorithm variance in addition to aggregated trends.

Summary. In conclusion, the findings suggest that using synthetic datasets for fairness in ML is promising, particularly with 1-shot learning. Increasing prompt complexity beyond one example, however, does not provide additional benefits and may even degrade fairness, as seen with 2-shot learning. The results show that carefully constructed synthetic datasets can lead to fairer ML models, though further research is needed to address multiple fairness metrics simultaneously.

Key findings of RQ_2.

Synthetic datasets generated with 1-shot prompts consistently improve fairness across most metrics without severely compromising model performance, whereas increasing prompt complexity (2-shot) often degrades fairness outcomes. These results suggest that synthetic dataset generation is a promising research direction, but further research is needed to address multiple fairness metrics at the same time.

5. Discussion and Take-Away Messages

The findings of our study provide a number of discussion points, which we elaborate in the following to provide implications and take-away messages for practitioners and researchers.

On the Trade-off Between Performance and Fairness. One of the key findings of this study lies in the trade-off between model performance and fairness when using synthetic datasets. While models trained and tested on synthetic data generally exhibit lower performance compared to those trained on real datasets, we observe notable improvements when these models are tested on real data, demonstrating the potential for synthetic data to generalize effectively in real-world scenarios. For instance, in cases like 0-shot learning on the *Student Performance* dataset, the accuracy of models trained on synthetic data was even better than those trained on real data.

More importantly, synthetic datasets, especially those generated using 1-shot prompts, consistently lead to significant reductions in bias, as demonstrated by improvements in fairness metrics such as AOD and SPD. As such, the overall takeaway of our study is that *the use of synthetic datasets often results in fairer models without dramatically compromising performance*.

These findings have important implications for *practitioners*: although models trained on synthetic data may experience slight performance declines compared to those trained on real data, the improvements in fairness make this trade-off justifiable in contexts where fairness is a primary concern. Practitioners can therefore harness synthetic data generation, especially with simpler prompts, to produce models that reduce bias without sacrificing too much accuracy or

F1-score. This opens up practical opportunities to deploy fairer models in real-world machine learning systems, particularly in sectors where ethical considerations are paramount.

> 💼 **Take-Away Message #1.** Synthetic datasets may offer a valuable trade-off between performance and fairness, making them a potentially practical tool for building fairer machine learning models in ethically sensitive domains.

Fairness Gains vs. Performance in Prompt Design. Another significant takeaway from our study is the role of prompt design in optimizing the balance between fairness and performance. The results indicate that 0-shot learning tends to optimize performance at the expense of fairness, whereas 1-shot learning offers fairness improvements, albeit with a slight decline in performance. In addition, an increasing prompt complexity (e.g., moving to 2-shot learning) is generally not beneficial and may, in fact, degrade both fairness and performance. From a practical perspective, while our study does not identify a definitive "best" prompt design, it informs future research: *prompt simplicity seems to be the best approach to synthetic data generation*.

In this regard, *researchers* can build upon these findings by further optimizing 0-shot and 1-shot learning techniques or by exploring novel fair prompt engineering approaches. We see these new research avenues as crucial steps toward enhancing the social awareness of LLMs in synthetic dataset generation, which may lead to ML systems which are fairer by-design. Likewise, *practitioners* can apply our findings based on the specific demands of their use cases. If fairness is the primary concern, 1-shot learning offers a viable approach for generating synthetic data with reduced bias. In contrast, 0-shot learning may be preferable in contexts where maximizing model performance is essential, even though fairness still remains a consideration: for instance, when limited or no real data is available, practitioners may prioritize 0-shot learning to achieve a well-performing model that incorporates some degree of fairness. Through these insights, practitioners may have an actionable knowledge base to handle fairness requirements.

> 💼 **Take-Away Message #2.** Simplicity in prompt design serves as the foundation to optimize prompt strategies and advance fair prompt design research. Our findings lay the groundwork for future studies on leveraging synthetic data generation to address fairness, while offering practitioners practical insights to manage fairness requirements effectively.

Integrating Synthetic Dataset Generation within Quality Assurance Processes. As a last point of discussion, we discuss how our study may inform quality assurance (QA) processes of ML systems. At first, *synthetic data can be seamlessly incorporated into existing bias mitigation techniques across the ML pipeline*. For example, synthetic data generation can act as an additional step in pre-processing, ensuring that training data is more equitably distributed. This approach may complement traditional pre-processing techniques by directly generating balanced datasets rather than merely adjusting real data. Furthermore, in-processing methods, which aim to mitigate bias during the training phase, can benefit from synthetic datasets to improve fairness-aware algorithms, ensuring models are trained on less biased, yet representative data. Finally, post-processing approaches, which adjust model outputs for fairness, can leverage synthetic datasets to fine-tune results by comparing outcomes with models trained on synthetic

data, thus providing another layer of fairness verification. The integration of synthetic data generation and its potential to enhance the effectiveness and reduce the cost of existing bias mitigation algorithms present valuable opportunities for further exploration by *researchers*.

Beyond fairness, our study demonstrates that LLMs can generate synthetic datasets with high structural fidelity. This is an interesting finding per se, as it may have broader implications for the SE4AI research field. The structural integrity of these datasets means they are potentially able to replicate the characteristics of real data, making them a valuable tool for various QA processes. While synthetic data has already been experimented with in areas like vulnerability prediction [30, 46] and functional software testing [60, 77], our findings suggest additional opportunities for software quality and verification tasks. For instance, synthetic datasets could be used to improve stress-testing processes, test system scalability, and evaluate other non-functional requirements. From a software engineering perspective, this positions synthetic dataset generation as a quality assurance instrument comparable to other established techniques for testing and verification. In particular, fairness is a non-functional requirement to be validated during development, and synthetic datasets provide a means to specify, test, and monitor this requirement alongside traditional ones such as reliability or performance. This aligns with SE4AI practices that increasingly embed data quality and fairness assurance into software processes. Synthetic datasets could thus be integrated into (i) *requirements engineering*, by validating fairness requirements early with controlled synthetic inputs, (ii) *quality assurance and testing*, by expanding the test suite with fairness-specific checks, and (iii) *MLOps pipelines*, by enabling regression testing of fairness across continuous updates. These represent actionable insights not only for *researchers*, but also for *practitioners* aiming to embed fairness and other NFRs into their engineering workflows.

Take-Away Message #3. Synthetic data generation can be combined with existing bias mitigation approaches, potentially enhancing their effectiveness while reducing their cost. Beyond AI-specific techniques, the structural fidelity of LLM-generated data positions synthetic datasets as a candidate tool for software engineering quality assurance, from requirements validation to fairness testing in MLOps pipelines.

Implications for the SE4AI domain. These insights also have practical implications for both researchers and practitioners. For researchers, our findings open avenues for investigating synthetic data generation as a reusable instrument for fairness testing, complementing existing research on bias mitigation, dataset curation, and fairness-aware ML pipelines. In particular, future work could study how synthetic datasets interact with different fairness metrics, or how they may serve as controlled environments for reproducible experiments. For practitioners, the results suggest concrete ways to embed synthetic dataset generation into daily workflows. For example, engineers specifying fairness requirements during the design phase can use synthetic datasets to validate assumptions before real data is collected; QA teams can integrate synthetic data into test suites to systematically evaluate fairness alongside functional correctness; and DevOps/MLOps teams can incorporate synthetic datasets into CI/CD pipelines to detect fairness regressions when retraining or deploying updated models. By bridging these two perspectives, our study highlights synthetic data generation not only as a conceptual contribution but also as a practical tool for engineering and maintaining fair AI-enabled systems.

6. Threats to Validity

This section discusses the potential threats to the validity of our empirical study, along with the mitigation strategies employed in terms of the research methods applied.

Construct Validity. These threats involve discrepancies between the theoretical framework and observations. One concern is dataset selection, as inappropriate datasets could undermine our findings. To mitigate this, we selected three well-known datasets [19]: *German Credit*, *Heart Disease*, and *Student Performance*, following the ontology proposed by Fabris et al. [31]. These datasets are sensitive, aligning with our goal of evaluating the trade-off between performance and fairness. Another threat relates to our selected fairness metrics (SPD, EOD, AOD), which may not capture all dimensions of fairness. We chose these metrics based on literature [19, 56], which identified them as effective measures for assessing fairness. Lastly, the models used could influence results. However, using the *Lazy Predict* library, we were able to experiment with various ML algorithms and their hyperparameters.

Finally, our prompting strategy was purposefully *naïve*, meaning that we did not include any explicit instructions aimed at ensuring fairness in the synthetic data generation process. While we acknowledge that this could have influenced our results, it was an intended choice: our objective was indeed to *evaluate* the extent to which LLMs, when given only basic, task-oriented prompts, can autonomously produce ethically aligned synthetic datasets. This setup mirrors real-world scenarios where practitioners, without specialized knowledge of fairness issues, may use LLMs in a straightforward manner to generate data. By assessing the default behavior of LLMs under naïve prompting, we establish a necessary baseline that future research can build upon to design more effective, fairness-aware synthetic data generation strategies.

Internal Validity. Internal validity concerns how well results are attributed to studied variables. Our choice of 0-shot, 1-shot, and 2-shot approaches was guided by prior research identifying these techniques as suitable for tabular data generation [10, 72]. We did not go beyond 2-shot, as studies show minimal gains with increased prompt complexity [32]. We also conducted a preliminary evaluation to select the best LLM (GPT-4o), focusing on structural fidelity. Additionally, we purposely selected relatively small datasets to reduce the risk of hallucinations: smaller datasets indeed enable a more controlled generation, ensuring that the synthetic data better aligns with the intended structure and content of the original datasets. Finally, LLMs were used with default settings to maintain consistency across experiments and minimize the introduction of confounding variables, reducing the likelihood of model misconfigurations or unforeseen biases affecting the results. Furthermore, we executed each prompt three times to mitigate the inherent variability of LLM outputs [45], increasing the soundness of our research methods. Finally, we acknowledge that variance across classifiers may influence the interpretation of aggregate results. To mitigate this, we complemented averages with standard deviations, applied non-parametric statistical tests that explicitly account for variability, and reported representative per-algorithm outcomes for transparency. These measures reduce the risk that mean values obscure substantial dispersion across models.

External Validity. External validity refers to the extent to which our findings can be generalized beyond the specific research context. While we acknowledge that the results may not be directly transferable to all domains, we selected datasets from diverse domains and tasks. Nonetheless, further studies are needed to confirm and extend our findings. To encourage replication and additional research, we have made all data and scripts publicly available in our appendix [87]. A significant threat lies in the use of well-known datasets [31], since they could

have been part of the training set of the LLMs employed. To mitigate this, we made sure to provide a prompt that was as generic as possible, without referring to the specific dataset name and using hand-crafted descriptions rather than existing ones. Furthermore, our analysis of the datasets showed that there are significant differences in feature distribution between the synthetic datasets and the original ones, partially mitigating this threat.

Conclusion Validity. Conclusion validity relates to the reliability of our conclusions. We conducted the study under a multiple-sensitive attribute setting, focusing on the combination of *'age'* and *'sex'*. We acknowledge that conclusions may differ under a *single-sensitive* attribute setting, where fairness outcomes may not account for intersectional biases. To overcome this limitation, we conducted extra experiments using a *single-sensitive* attribute setting. The results, which can be found in our online appendix [87], align with those in Section 4, confirming that generating synthetic data is a promising approach to meet fairness requirements.

Additionally, we employed non-parametric statistical tests (Friedman and Nemenyi) to confirm the significance of performance and fairness differences. Lastly, aggregating results across 27 models could introduce variability. While this introduces the risk of masking individual model differences, the relatively low standard deviation suggests that performance metrics and fairness outcomes were consistent across the models. Therefore, we argue that the average values reported are representative of the general behavior of the models. In addition, our online appendix [87] provides results grouped by individual machine learning algorithms. These results confirm our primary observations, demonstrating that the conclusions drawn from the aggregated data are reliable and applicable across different models.

7. Conclusion

This paper explored the potential of synthetic data generation to mitigate unfairness in ML models while maintaining competitive performance. We employed GPT-4o to generate synthetic datasets that resemble the attributes of three well-known datasets, such as *German Credit*, *Heart Disease*, and *Student Performance*, and compared ML models trained with synthetic datasets against those trained with real datasets.

We found that although models trained on synthetic datasets typically show lower performance during cross-validation, they generalize relatively well when applied to unseen real data. The observed performance gap is often small, particularly when simpler prompting strategies are used. This suggests that synthetic data can be a viable alternative in practice when accompanied by fairness improvements. Furthermore, our results indicate that 1-shot prompting consistently improves fairness across most metrics without severely compromising performance. By contrast, increasing prompt complexity to 2-shot often degrades fairness outcomes. These findings highlight that synthetic data generation is promising for fairness-by-design, but further work is needed to develop techniques that jointly optimize multiple fairness metrics.

The observations and take-away messages of our study represent the input for our future research agenda. We will explore prompt optimization and fair prompt engineering, as well as investigate the impact of synthetic dataset generation on bias mitigation algorithms and other quality assurance processes in software engineering.

Credits

Gianmario Voria: Formal analysis, Investigation, Data Curation, Validation, Writing - Original Draft, Visualization. **Benedetto Scala**: Formal analysis, Investigation, Data Curation, Val-

idation. **Leopoldo Todisco**: Formal analysis, Investigation, Data Curation, Validation. **Carlo Venditto**: Formal analysis, Investigation, Data Curation, Validation. **Giammaria Giordano**: Supervision, Validation, Writing - Review & Editing. **Gemma Catolino**: Supervision, Validation, Writing - Review & Editing. **Fabio Palomba**: Supervision, Validation, Writing - Review & Editing.

Conflict of interest

The authors declare that they have no conflict of interest.

Data Availability

The data collected in the context of this research, along with the scripts and the results of the experiments, are publicly available in our online appendix [87].

Acknowledgement

We acknowledge the use of ChatGPT-4 to ensure linguistic accuracy and enhance the readability of this article. We acknowledge the support of the European Union - NextGenerationEU through the Italian Ministry of University and Research, Project PRIN 2022 PNRR "FRINGE: context-aware FaiRness engineerING in complex software systEms" (grant n. P2022553SL, CUP: D53D23017340001) and Project PRIN 2022 "QualAI: Continuous Quality Improvement of AI-based Systems" (grant n. 2022B3BP5S, CUP: H53D23003510006)..

References

[1] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. 2024. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone. arXiv:2404.14219 [cs.CL] `https://arxiv.org/abs/2404.14219`

[2] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. In *International conference on machine learning*. PMLR, 60–69.

[3] Aniya Aggarwal, Pranay Lohia, Seema Nagar, Kuntal Dey, and Diptikalyan Saha. 2019. 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*. 625–635.

[4] AI@Meta. 2024. Llama 3 Model Card. `https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md`

[5] Anthropic. 2024. Claude 3.5 Sonnet. `https://www.anthropic.com/news/claude-3-5-sonnet`

[6] Enrico Barbierato, Marco L Della Vedova, Daniele Tessera, Daniele Toti, and Nicola Vanoli. 2022. A methodology for controlling bias and fairness in synthetic data generation. *Applied Sciences* 12, 9 (2022), 4619.

[7] André Bauer, Simon Trapp, Michael Stenger, Robert Leppich, Samuel Kounev, Mark Leznik, Kyle Chard, and Ian Foster. 2024. Comprehensive exploration of synthetic data generation: A survey. *arXiv preprint arXiv:2401.02524* (2024).

[8] R. K. E. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilović, S. Nagar, K. Natesan Ramamurthy, J. Richards, D. Saha, P. Sattigeri, M. Singh, K. R. Varshney, and Y. Zhang. 2019. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development* 63, 4/5 (2019), 4:1–4:15. `https://doi.org/10.1147/JRD.2019.2942287`

[9] Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. 2021. Deep Neural Networks and Tabular Data: A Survey. *CoRR* abs/2110.01889 (2021). arXiv:2110.01889 `https://arxiv.org/abs/2110.01889`

[10] Vadim Borisov, Kathrin Seßler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. 2023. Language Models are Realistic Tabular Data Generators. arXiv:2210.06280 [cs.LG] `https://arxiv.org/abs/2210.06280`

[11] Tom B. Brown and et al. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165 [cs.CL]

[12] Robert J Cabin and Randall J Mitchell. 2000. To Bonferroni or not to Bonferroni: when and how are the questions. *Bulletin of the ecological society of America* 81, 3 (2000), 246–248.

[13] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. 2017. Optimized pre-processing for discrimination prevention. *Advances in neural information processing systems* 30 (2017).

[14] Joymallya Chakraborty, Suvodeep Majumder, and Tim Menzies. 2021. Bias in machine learning software: why? how? what to do?. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 429–440.

[15] Joymallya Chakraborty, Suvodeep Majumder, Zhe Yu, and Tim Menzies. 2020. Fairway: a way to build fair ML software. In *Proceedings of the 28th ACM joint meeting on European software engineering conference and symposium on the foundations of software engineering*. 654–665.

[16] Joymallya Chakraborty, Tianpei Xia, Fahmid M Fahid, and Tim Menzies. 2019. Software engineering for fairness: A case study with hyperparameter optimization. *arXiv preprint arXiv:1905.05786* (2019).

[17] Wenhu Chen. 2023. Large Language Models are few(1)-shot Table Reasoners. In *Findings of the Association for Computational Linguistics: EACL 2023*, Andreas Vlachos and Isabelle Augenstein (Eds.). Association for Computational Linguistics, Dubrovnik, Croatia, 1120–1130. `https://doi.org/10.18653/v1/2023.findings-eacl.83`

[18] Zhenpeng Chen, Jie M. Zhang, Federica Sarro, and Mark Harman. 2022. MAAT: a novel ensemble approach to addressing fairness and performance bugs for machine learning software. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering* (Singapore, Singapore) *(ESEC/FSE 2022)*. Association for Computing Machinery, New York, NY, USA, 1122–1134. `https://doi.org/10.1145/3540250.3549093`

[19] Zhenpeng Chen, Jie M Zhang, Federica Sarro, and Mark Harman. 2024. Fairness Improvement with Multiple Protected Attributes: How Far Are We?. In *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*. 1–13.

[20] Edward Choi, Siddharth Biswal, Bradley Malin, Jon Duke, Walter F. Stewart, and Jimeng Sun. 2018. Generating Multi-label Discrete Patient Records using Generative Adversarial Networks. arXiv:1703.06490 [cs.LG] `https://arxiv.org/abs/1703.06490`

[21] Sam Corbett-Davies, Johann D Gaebler, Hamed Nilforoshan, Ravi Shroff, and Sharad Goel. 2023. The measure and mismeasure of fairness. *The Journal of Machine Learning Research* 24, 1 (2023), 14730–14846.

[22] Paulo Cortez. 2014. Student Performance. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5TG7T.

[23] Ashish Dandekar, Remmy A. M. Zen, and Stéphane Bressan. 2018. A Comparative Study of Synthetic Dataset Generation Techniques. In *Database and Expert Systems Applications*, Sven Hartmann, Hui Ma, Abdelkader Hameurlain, Günther Pernul, and Roland R. Wagner (Eds.). Springer International Publishing, Cham, 387–395.

[24] Vincenzo De Martino, Gianmario Voria, Ciro Troiano, Gemma Catolino, and Fabio Palomba. 2025. Examining the Impact of Bias Mitigation Algorithms on the Sustainability of Ml-Enabled Systems: A Benchmark Study. *Journal of Systems and Software* (2025).

[25] Pedro Domingos. 2012. A few useful things to know about machine learning. *Commun. ACM* 55, 10 (oct 2012), 78–87. `https://doi.org/10.1145/2347736.2347755`

[26] Peter Eigenschink, Thomas Reutterer, Stefan Vamosi, Ralf Vamosi, Chang Sun, and Klaudius Kalcher. 2023. Deep generative models for synthetic data: A survey. *IEEE Access* 11 (2023), 47304–47320.

[27] Khaled El Emam, Lucy Mosquera, and Richard Hoptroff. 2020. *Practical synthetic data generation: balancing privacy and the broad availability of data*. O'Reilly Media.

[28] Widad Elouataoui, Imane El Alaoui, Saida El Mendili, and Youssef Gahi. 2022. An advanced big data quality framework based on weighted metrics. *Big Data and Cognitive Computing* 6, 4 (2022), 153.

[29] Markus Endres, Asha Mannarapotta Venugopal, and Tung Son Tran. 2022. Synthetic Data Generation: A Comparative Study. In *Proceedings of the 26th International Database Engineered Applications Symposium* (Budapest, Hungary) *(IDEAS '22)*. Association for Computing Machinery, New York, NY, USA, 94–102. `https://doi.org/10.1145/3548785.3548793`

[30] Matteo Esposito and Davide Falessi. 2024. VALIDATE: A deep dive into vulnerability prediction datasets. *Information and Software Technology* (2024), 107448.

[31] Alessandro Fabris, Stefano Messina, Gianmaria Silvello, and Gian Antonio Susto. 2022. Algorithmic fairness datasets: the story so far. *Data Mining and Knowledge Discovery* 36, 6 (Sept. 2022), 2074–2152. `https://doi.org/10.1007/s10618-022-00854-z`

[32] Xi Fang, Weijie Xu, Fiona Anting Tan, Jiani Zhang, Ziqing Hu, Yanjun Jane Qi, Scott Nickleach, Diego Socolinsky, Srinivasan Sengamedu, Christos Faloutsos, et al. 2024. Large language models (LLMs) on tabular data: Prediction, generation, and understanding-a survey. (2024).

[33] Carmine Ferrara, Francesco Casillo, Carmine Gravino, Andrea De Lucia, and Fabio Palomba. 2024. ReFAIR: Toward a Context-Aware Recommender for Fairness Requirements Engineering. In *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*. 1–12.

[34] Carmine Ferrara, Giulia Sellitto, Filomena Ferrucci, Fabio Palomba, and Andrea De Lucia. 2024. Fairness-aware machine learning engineering: how far are we? *Empirical Software Engineering* 29, 1 (2024), 9.

[35] Jade S Franklin, Karan Bhanot, Mohamed Ghalwash, Kristin P Bennett, Jamie McCusker, and Deborah L McGuinness. 2022. An Ontology for Fairness Metrics. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*. 265–275.

[36] Sainyam Galhotra, Yuriy Brun, and Alexandra Meliou. 2017. Fairness testing: testing software for discrimination. In *Proceedings of the 2017 11th Joint meeting on foundations of software engineering*. 498–510.

[37] Pratyush Garg, John Villasenor, and Virginia Foggo. 2020. Fairness metrics: A comparative analysis. In *2020 IEEE international conference on big data (Big Data)*. IEEE, 3662–3666.

[38] Elizabeth González-Estrada and Waldenia Cosmes. 2019. Shapiro–Wilk test for skew normal distributions based on data transformations. *Journal of Statistical Computation and Simulation* 89, 17 (2019), 3258–3272.

[39] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Networks. arXiv:1406.2661 [stat.ML] `https://arxiv.org/abs/1406.2661`

[40] Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. *Advances in neural information processing systems* 29 (2016).

[41] Jahid Hasan and Romila Pradhan. 2024. Data Acquisition for Improving Model Fairness using Reinforcement Learning. arXiv:2412.03009 [cs.LG] `https://arxiv.org/abs/2412.03009`

[42] Hans Hofmann. 1994. Statlog (German Credit Data). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5NC77.

[43] Myles Hollander, Douglas A Wolfe, and Eric Chicken. 2013. *Nonparametric statistical methods*. John Wiley & Sons.

[44] Max Hort, Zhenpeng Chen, Jie M Zhang, Mark Harman, and Federica Sarro. 2024. Bias mitigation for machine learning classifiers: A comprehensive survey. *ACM Journal on Responsible Computing* 1, 2 (2024), 1–52.

[45] Xinyi Hou, Yanjie Zhao, Yue Liu, Zhou Yang, Kailong Wang, Li Li, Xiapu Luo, David Lo, John Grundy, and Haoyu Wang. 2023. Large language models for software engineering: A systematic literature review. *arXiv preprint arXiv:2308.10620* (2023).

[46] Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and J Doug Tygar. 2011. Adversarial machine learning. In *Proceedings of the 4th ACM workshop on Security and artificial intelligence*. 43–58.

[47] Andras Janosi, William Steinbrunn, Matthias Pfisterer, and Robert Detrano. 1988. Heart Disease. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C52P4X.

[48] Brittany Johnson, Jesse Bartola, Rico Angell, Katherine Keith, Sam Witty, Stephen J Giguere, and Yuriy Brun. 2020. Fairkit, fairkit, on the wall, who's the fairest of them all? Supporting data scientists in training fair models. *arXiv preprint arXiv:2012.09951* (2020).

[49] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and information systems* 33, 1 (2012), 1–33.

[50] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-aware classifier with prejudice remover regularizer. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II 23*. Springer, 35–50.

[51] Diederik P Kingma and Max Welling. 2022. Auto-Encoding Variational Bayes. arXiv:1312.6114 [stat.ML] `https://arxiv.org/abs/1312.6114`

[52] Ron Kohavi et al. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, Vol. 14. Montreal, Canada, 1137–1145.

[53] Yanhui Li, Linghan Meng, Lin Chen, Li Yu, Di Wu, Yuming Zhou, and Baowen Xu. 2022. Training data debugging for the fairness of machine learning software. In *2022 IEEE/ACM 44th International Conference on Software Engineering (ICSE)*. IEEE, 2215–2227.

[54] Zheng Li, Yue Zhao, and Jialin Fu. 2020. SYNC: A Copula based Framework for Generating Synthetic Data from Aggregated Sources. arXiv:2009.09471 [stat.AP] `https://arxiv.org/abs/2009.09471`

[55] Fernando Lucini. 2021. The real deal about synthetic data. *MIT Sloan Management Review* 63, 1 (2021), 1–4.

[56] Suvodeep Majumder, Joymallya Chakraborty, Gina R Bai, Kathryn T Stolee, and Tim Menzies. 2023. Fair enough: Searching for sufficient measures of fairness. *ACM Transactions on Software Engineering and Methodology* 32, 6 (2023), 1–22.

[57] Silverio Martínez-Fernández, Justus Bogner, Xavier Franch, Marc Oriol, Julien Siebert, Adam Trendowicz, Anna Maria Vollmer, and Stefan Wagner. 2022. Software Engineering for AI-Based Systems: A Survey. *ACM Transactions on Software Engineering and Methodology* 31, 2 (2022). `https://doi.org/10.1145/3487043`

[58] Ggaliwango Marvin, Nakayiza Hellen, Daudi Jjingo, and Joyce Nakatumba-Nabende. 2023. Prompt engineering

in large language models. In *International conference on data intelligence and cognitive informatics*. Springer, 387–402.

[59] Sean McGuire, Erin Schultz, Bimpe Ayoola, and Paul Ralph. 2023. Sustainability is stratified: Toward a better theory of sustainable software engineering. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. IEEE, 1996–2008.

[60] Phil McMinn. 2004. Search-based software test data generation: a survey. *Software testing, Verification and reliability* 14, 2 (2004), 105–156.

[61] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)* 54, 6 (2021), 1–35.

[62] Claire Cain Miller. 2015. Can an algorithm hire better than a human. *The New York Times* 25 (2015).

[63] Seumas Miller. 2019. Machine learning, ethics and law. *Australasian Journal of Information Systems* 23 (2019), 1–13.

[64] Hossein Mohamadipanah, LaDonna Kearse, Brett Wise, Leah Backhus, and Carla Pugh. 2023. Generating Rare Surgical Events Using CycleGAN: Addressing Lack of Data for Artificial Intelligence Event Recognition. *Journal of Surgical Research* 283 (2023), 594–605. `https://doi.org/10.1016/j.jss.2022.11.008`

[65] Avanika Narayan, Ines Chami, Laurel Orr, Simran Arora, and Christopher Ré. 2022. Can Foundation Models Wrangle Your Data? arXiv:2205.09911 [cs.LG] `https://arxiv.org/abs/2205.09911`

[66] Peter Bjorn Nemenyi. 1963. *Distribution-free multiple comparisons.* Princeton University.

[67] Jianjun Ni, Yinan Chen, Yan Chen, Jinxiu Zhu, Deena Ali, and Weidong Cao. 2020. A survey on theories and applications for self-driving cars based on deep learning methods. *Applied Sciences* 10, 8 (2020), 2749.

[68] Parmy Olson. 2011. The algorithm that beats your bank manager. *CNN Money March* 15 (2011).

[69] OpenAI. 2024. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL] `https://arxiv.org/abs/2303.08774`

[70] Tiago P Pagano, Rafael B Loureiro, Fernanda VN Lisboa, Rodrigo M Peixoto, Guilherme AS Guimarães, Gustavo OR Cruz, Maira M Araujo, Lucas L Santos, Marco AS Cruz, Ewerton LS Oliveira, et al. 2023. Bias and unfairness in machine learning models: a systematic review on datasets, tools, fairness metrics, and identification and mitigation methods. *Big data and cognitive computing* 7, 1 (2023), 15.

[71] Alessandra Parziale, Gianmario Voria, Giammaria Giordano, Gemma Catolino, Gregorio Robles, and Fabio Palomba. 2025. Contextual Fairness-Aware Practices in ML: A Cost-Effective Empirical Evaluation. In *Proceedings of the 1st st International Workshop on Fairness in Software Systems*.

[72] Premraj Pawade, Mohit Kulkarni, Shreya Naik, Aditya Raut, and K.S. Wagh. 2024. Efficiency Comparison of Dataset Generated by LLMs using Machine Learning Algorithms. In *2024 International Conference on Emerging Smart Computing and Informatics (ESCI)*. 1–6. `https://doi.org/10.1109/ESCI59607.2024.10497340`

[73] Kewen Peng, Joymallya Chakraborty, and Tim Menzies. 2022. FairMask: Better Fairness via Model-based Rebalancing of Protected Attributes. arXiv:2110.01109 [cs.LG] `https://arxiv.org/abs/2110.01109`

[74] Dana Pessach and Erez Shmueli. 2022. A review on fairness in machine learning. *ACM Computing Surveys (CSUR)* 55, 3 (2022), 1–44.

[75] Bogdana Rakova, Jingying Yang, Henriette Cramer, and Rumman Chowdhury. 2021. Where responsible AI meets reality: Practitioner perspectives on enablers for shifting organizational practices. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (2021), 1–23.

[76] Gilberto Recupito, Raimondo Rapacciuolo, Dario Di Nucci, and Fabio Palomba. 2024. Unmasking Data Secrets: An Empirical Investigation into Data Smells and Their Impact on Data Quality. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering-Software Engineering for AI*. 53–63.

[77] Vincenzo Riccio, Gunel Jahangirova, Andrea Stocco, Nargiz Humbatova, Michael Weiss, and Paolo Tonella. 2020. Testing machine learning based systems: a systematic mapping. *Empirical Software Engineering* 25 (2020), 5193–5254.

[78] Emeralda Sesari, Federica Sarro, and Ayushi Rastogi. 2024. Understanding Fairness in Software Engineering: Insights from Stack Exchange. *arXiv preprint arXiv:2402.19038* (2024).

[79] Shubham Sharma, Yunfeng Zhang, Jesús M Ríos Aliaga, Djallel Bouneffouf, Vinod Muthusamy, and Kush R Varshney. 2020. Data augmentation for discrimination prevention and bias disambiguation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 358–364.

[80] Yining She, Sumon Biswas, Christian Kästner, and Eunsuk Kang. 2025. FairSense: Long-Term Fairness Analysis of ML-Enabled Systems. *arXiv preprint arXiv:2501.01665* (2025).

[81] Christopher Starke, Janine Baleis, Birte Keller, and Frank Marcinkowski. 2021. Fairness Perceptions of Algorithmic Decision-Making: A Systematic Review of the Empirical Literature. arXiv:2103.12016 [cs.HC]

[82] Fahim Sufi. 2024. Generative Pre-Trained Transformer (GPT) in Research: A Systematic Review on Data Augmentation. *Information* 15, 2 (2024). `https://doi.org/10.3390/info15020099`

[83] Sakshi Udeshi, Pryanshu Arora, and Sudipta Chattopadhyay. 2018. Automated directed fairness testing. In *Proceedings of the 33rd ACM/IEEE international conference on automated software engineering*. 98–108.

[84] Sriram Vasudevan and Krishnaram Kenthapadi. 2020. Lift: A scalable framework for measuring fairness in ml

applications. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2773–2780.

[85] Michael Veale and Reuben Binns. 2017. Fairer machine learning in the real world: Mitigating discrimination without collecting sensitive data. *Big Data & Society* 4, 2 (2017), 2053951717743530.

[86] Gianmario Voria, Stefano Lambiase, Maria Concetta Schiavone, Gemma Catolino, and Fabio Palomba. 2025. From Expectation to Habit: Why Do Software Practitioners Adopt Fairness Toolkits?. In *Proceedings of the 50th IEEE/ACM International Conference on Software Engineering*.

[87] Gianmario Voria, Benedetto Scala, Leopoldo Todisco, Carlo Venditto, Giammaria Giordano, Gemma Catolino, and Fabio Palomba. 2025. Online Appendix. `https://figshare.com/s/fc0bdef65bef553d445a`

[88] Gianmario Voria, Giulia Sellitto, Carmine Ferrara, Francesco Abate, Andrea De Lucia, Filomena Ferrucci, Gemma Catolino, and Fabio Palomba. 2025. A catalog of fairness-aware practices in machine learning engineering. In *Proceedings of the 1st International Workshop on Evaluation of Qualitative Aspects of Intelligent Software Assistants*.

[89] Gianmario Voria, Giulia Sellitto, Carmine Ferrara, Francesco Abate, Andrea De Lucia, Filomena Ferrucci, Gemma Catolino, and Fabio Palomba. 2025. Fairness-aware practices from developers' perspective: A survey. *Information and Software Technology* 182 (2025), 107710.

[90] Pin Wang, En Fan, and Peng Wang. 2021. Comparative analysis of image classification algorithms based on traditional machine learning and deep learning. *Pattern recognition letters* 141 (2021), 61–67.

[91] Scott Cheng-Hsin Yang, Baxter Eaves, Michael Schmidt, Ken Swanson, and Patrick Shafto. 2024. Structured Evaluation of Synthetic Tabular Data. *arXiv preprint arXiv:2403.10424* (2024).

[92] Stella S. Yi, Lan N. Doan, Juliet K. Choi, Jennifer A. Wong, Rienna Russo, Matthew Chin, Nadia S. Islam, M. D. Taher, Laura Wyatt, Stella K. Chong, Chau Trinh-Shevrin, and Simona C. Kwon. 2021. With no data, there's no equity: addressing the lack of data on COVID-19 for asian american communities. *eClinicalMedicine* 41 (01 Nov 2021). `https://doi.org/10.1016/j.eclinm.2021.101165`

[93] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. 2018. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. 335–340.

[94] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. 2017. PrivBayes: Private Data Release via Bayesian Networks. *ACM Trans. Database Syst.* 42, 4, Article 25 (oct 2017), 41 pages. `https://doi.org/10.1145/3134428`

[95] Jie M Zhang and Mark Harman. 2021. "Ignorance and Prejudice" in Software Fairness. In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 1436–1447.

[96] Peixin Zhang, Jingyi Wang, Jun Sun, Guoliang Dong, Xinyu Wang, Xingen Wang, Jin Song Dong, and Ting Dai. 2020. White-box fairness testing through adversarial sampling. In *Proceedings of the ACM/IEEE 42nd international conference on software engineering*. 949–960.

[97] Jianlong Zhou and Fang Chen. 2018. *Human and Machine Learning*. Springer.