MMM-fair: An Interactive Toolkit for Exploring and Operationalizing Multi-Fairness Trade-offs

Swati Swati*
swati.swati@unibw.de
University of the Bundeswehr
Munich, Germany

Arjun Roy*
arjun.roy@unibw.de
University of the Bundeswehr
Munich, Germany
Free University Berlin
Berlin, Germany

Emmanouil Panagiotou
emmanouil.panagiotou@fu-berlin.de
Free University Berlin
Berlin, Germany
University of the Bundeswehr
Munich, Germany

Eirini Ntoutsi
eirini.ntoutsi@unibw.de
University of the Bundeswehr
Munich, Germany

Abstract

Fairness-aware classification requires balancing performance and fairness, often intensified by intersectional biases. Conflicting fairness definitions further complicate the task, making it difficult to identify universally fair solutions. Despite growing regulatory and societal demands for equitable AI, popular toolkits offer limited support for exploring multi-dimensional fairness and related trade-offs. To address this, we present *mmm-fair*, an open-source toolkit leveraging boosting-based ensemble approaches that dynamically optimizes model weights to jointly minimize classification errors and diverse fairness violations, enabling flexible multi-objective optimization. The system empowers users to deploy models that align with their context-specific needs while reliably uncovering intersectional biases often missed by state-of-the-art methods. In a nutshell, mmm-fair uniquely combines in-depth multi-attribute fairness, multi-objective optimization, a no-code, chat-based interface, LLM-powered explanations, interactive Pareto exploration for model selection, custom fairness constraint definition, and deployment-ready models in a single open-source toolkit, a combination rarely found in existing fairness tools. Demo walkthrough available at: https://youtu.be/_rcpjlXFqkw.

CCS Concepts

• **Human-centered computing** → *User interface toolkits*; • **Computing methodologies** → **Machine learning algorithms**; **Supervised learning**; **Ensemble methods**; *Model development and analysis*; **Bias, fairness and unawareness in machine learning**.

---

*Both authors contributed equally to this research.

Keywords

Bias, Fairness, Multi-attribute, Multi-objective, Multi-definition, Fairness-aware Classification

1 Introduction

Motivation. As machine learning (ML) systems are increasingly deployed across critical domains such as healthcare, finance, and hiring, concerns about biased or unfair outcomes have intensified [2, 4, 12]. These concerns arise from the growing recognition that algorithmic systems can inadvertently reproduce existing societal inequalities or introduce new forms of discrimination, often disproportionately affecting historically marginalized groups [1, 10]. Fairness-aware classification presents an inherently complex, multi-objective challenge: it requires balancing predictive performance with fairness constraints across multiple protected attributes, often under conditions of class or group imbalance [9]. This challenge is further compounded by multiple, often incompatible definitions of fairness [5], making it difficult to identify solutions that are both universally acceptable and practically effective.

Solution overview. To address the multifaceted challenges of fairness-aware classification, we introduce *mmm-fair*, a Python package designed to support flexible, interpretable, and context-sensitive model development. It provides comprehensive support for *multi-fairness* (the simultaneous consideration of *multiple protected attributes*, *fairness definitions*, and *optimization objectives*) within a unified framework. The system features an interactive, modular workflow that operationalizes fairness as a practical, end-to-end process, guiding users through data profiling, subgroup analysis, model training, trade-off exploration, LLM-based explanation, and deployment. Among its key components is the integrated

Table 1: Feature-based comparison of *mmm-fair* and related popular fairness-aware toolkits.

Feature	mmm-fair (Ours)	IBM AIF360 [2]	MS Fairlearn [3]	Google What-If [14]	Snowflake-TruEra [13]	WhyLabs [15]	Fiddler [6]	FairBench [8]
In-depth multi-attribute fairness	✓	✓	✓	○	○	○	○	○
Multi-objective optimization	✓	–	–	○	–	–	–	○
No-code user interface	✓	–	✓	✓	✓	✓	–	✓
Chat-based interaction	✓	–	–	–	–	✓	○	–
Pareto trade-off explorer	✓	–	–	△	–	–	–	△
Custom constraints (DP, EO, etc.)	✓	✓	✓	○	○	○	○	○
Fairness-aware training, not just auditing	✓	✓	✓	–	–	–	–	–
Deployment-ready models	✓	–	–	–	–	–	–	–
Open source	✓	✓	✓	✓	–	△	△	✓
LLM-based explanations	✓	–	–	–	✓	✓	△	–

Note: ✓ Fully Supported; △ Partial/Basic Support; ○ Reporting Only; – Not Supported.

Pareto front explorer, which enables users to evaluate trade-offs across multiple fairness and performance metrics and select models aligned with user-defined goals. A chat-based interface further facilitates explanation and supports iterative refinement throughout the workflow. *Why mmm-fair? Because "What's fair?" should be the user's choice, not the algorithm's.*

Comparison with existing works. Although existing fairness toolkits have contributed significantly to the field, they often fall short in their ability to provide comprehensive multi-fairness support within a single, integrated framework. As shown in Table 1, *mmm-fair* stands out from popular toolkits by offering a broader and more versatile set of capabilities. Empirical results on benchmark datasets further demonstrate that *mmm-fair* reliably uncovers intersectional biases and reduces group disparities without sacrificing accuracy or increasing overfitting. It maintains robust performance even in imbalanced and complex real-world datasets, establishing itself as a uniquely effective and versatile solution.

System Availability. *mmm-fair* is publicly available: as a package on **PyPI** (https://pypi.org/project/mmm-fair/), with source code on **GitHub** (https://github.com/arjunroyihrpa/MMM_fair), and a demonstration video at **YouTube** (https://youtu.be/_rcpjlXFqkw).

2 mmm-fair: Package Overview

Built on the foundational work of Roy et al. [11], *mmm-fair* is a Python toolkit that generalizes the "Multi-fairness Under Class-Imbalance" approach to support a broader set of fairness definitions, optimization objectives, and experimental configurations within a unified framework. While the original work focused on Disparate Mistreatment, it extends its support to widely used fairness criteria such as Demographic Parity, Equalized Odds, and others, offering enhanced flexibility and customization during model training. At its core is a boosting-based ensemble method that adjusts model weights to jointly minimize classification error and fairness violations. This architecture enables efficient exploration of trade-offs across multi-fairness constraints and incorporates the following key functionalities:

- **Multi-attribute fairness:** allows fairness assessment across multiple protected attributes, such as age, race, and gender.
- **Multiple fairness definitions:** enables selection among demographic parity, equal opportunity, equalized odds, and others.

- **Fairness-integrated boosting:** leverages AdaBoost-style and gradient-boosted ensembles with a fairness-weighted objective, controlled by the hyper-parameter *gamma*, to jointly optimize predictive accuracy and fairness during training.
- **Pareto front-based model selection:** offers identification and visualization of optimal trade-offs between fairness and predictive performance across multiple objectives, fairness definitions, and protected attributes.
- **Adaptive handling of difficult instances:** dynamically adjusts emphasis on difficult samples to reduce over-correction and improve stability, particularly once fairness goals are partially met.
- **Extensible modular design:** allows easy integration of fairness metrics, definitions, and base learners through scikit-learn APIs and CLI/code, ensuring adaptability and maintainability.
- **Chat-based user interface:** provides a simple, no-code interface for interacting with the toolkit and receiving explanations and guidance.
- **Seamless model deployment:** enables export of trained models for downstream use and application integration.
- **User-friendly and open source:** offers intuitive interface, step-by-step guidance, and publicly available reproducible workflows.

Previous fairness-aware boosting approaches, such as fairness-aware AdaBoost variants [7] and MFBPP [11], have typically focused on a single fairness definition and relied primarily on reweighting strategies. In contrast, our package offers greater flexibility by allowing users to choose from multiple fairness definitions and directly incorporating the selected constraint across multiple protected attributes into the gradient boosting objective. To achieve joint optimization, we introduce a softmax-weighted aggregation of fairness gradients across attributes, enabling the model to balance predictive performance with fairness requirements across multiple attributes. Finally, we adopt a Pareto-front model selection strategy that explicitly balances accuracy, class imbalance, and fairness in a principled manner, moving beyond ad-hoc or purely post-hoc adjustments.

This multifaceted architecture enables mmm-fair to support a broad spectrum of real-world applications where fairness is essential. By decoupling the boosting strategy from any single definition,

the toolkit allows users to flexibly explore and prioritize fairness, performance, or a tailored balance of both.

3 Using mmm-fair

mmm-fair is a comprehensive toolkit for fairness-aware machine learning, designed to address real-world challenges across diverse application domains, with an interactive user interface and command-line options to suit varied user needs.

Application domains: *mmm-fair* applies to a wide range of domains, including credit scoring, healthcare, hiring, and any scenario where fairness across multiple protected groups and definitions is required [9]. Typical use cases include evaluating fairness-accuracy trade-offs in imbalanced datasets, mitigating algorithmic bias for regulatory compliance, and analyzing subgroup outcomes in high-stakes decision-making systems.

Installation. The toolkit offers both an intuitive web-based interface and a command-line interface (CLI), making it accessible to users with diverse technical backgrounds. To install, use: `pip install mmm-fair`. Once installed, the CLI is particularly suited for machine learning practitioners who prefer scripting and need precise control over model configurations and fairness constraints. The following script demonstrates a minimal working example for setting fairness objectives and training a model:

Quickstart: Defining Fairness Constraints and Training.

```
1   from mmm_fair import MMM_Fair
2   from sklearn.tree import DecisionTreeClassifier
3
4   mmm = MMM_Fair(
5   estimator=DecisionTreeClassifier(max_depth=5),
6       constraints="EO", # or "DP", "EP", etc.
7       saIndex=sa_index, # (n_samples, n_protected)
8       saValue=sa_value # dictionary or None
9       # other parameters, e.g. alpha, gamma, etc.
10  )
11
12  mmm.fit(X, y) # X: features, y: labels
13  y_pred = mmm.predict(X_test)
```

3.1 An Interactive Demonstration

A Real-World Scenario as an Example. Consider a cross-functional team of data scientists, policy analysts, compliance experts, software engineers, and other stakeholders at a financial institution. Their goal is to develop a system that maintains high predictive performance while satisfying fairness constraints and policy requirements. The team must ensure that the model behaves equitably across protected attributes and their intersections, such as age, income, and education, using multiple fairness definitions. These challenges intensify with imbalanced data, overlapping subgroups, and competing objectives. *mmm-fair* supports such teams in operationalizing multi-dimensional fairness through a seamless, interactive workflow. The following steps illustrate how mmm-fair can be used to systematically address these complex challenges in practice.

① **Dataset Selection and Profiling.** The workflow begins with users selecting a built-in benchmark, such as Adult Income or German Credit, or uploading a custom dataset. Once loaded, it generates an interactive, nested visualization that displays the distribution of key protected attributes and their intersections with the target variable; users can explore these profiles to immediately identify how different subgroups are represented, revealing imbalances or underrepresented groups that may raise fairness concerns. This initial exploration provides a clearer understanding of the dataset's protected structure before moving on to fairness configuration and model development.

② **Attribute Specification and Metric Selection.** This step empowers users to define protected attributes and subgroups relevant to their context. The interface enables selection of a diverse set of features, such as gender, age, race, or education, to capture nuanced sources of bias. Users can also define fairness and performance metrics to guide evaluation, choosing from a diverse set of options like demographic parity, equalized odds, and balanced accuracy, among others. Metrics are prioritized as constraints, ensuring alignment with the goals of the cross-functional team.

③ **Multi-Objective Training and Trade-Off Visualization.** Once objectives are set, the workflow advances to model training, producing a diverse set of candidate models in a single run. *mmm-fair* then generates the Pareto front, clearly visualizing trade-offs between fairness and accuracy (see Figure 1). The interface enables users to inspect metric values, compare alternatives, and immediately see how different configurations affect competing objectives. As users explore these plots, they often discover solutions or trade-offs not anticipated during setup. This interactive process supports transparent, interpretable, and robust model selection, ensuring alignment with institutional policies and stakeholder values.

④ **Interactive Chat-Based Explanations with LLMs.** After model training and visualization, users can request explanations through the integrated chat interface; the system prompts for a provider, such as OpenAI or ChatGPT, and generates clear, context-aware natural language summaries. Users may ask follow-up questions, making it easy to interpret results and communicate with both technical and non-technical audiences.

Importantly, the LLM used here does not compute statistical outputs; instead, it functions solely as a narrative layer that renders structured results into natural language. To minimize the potential for hallucinations, the module combines constrained prompting, structured response formats, and an algorithm-first pipeline with precomputed summaries. Explanations are shown alongside raw plots and metrics, allowing users to verify results directly. Together, these measures strengthen reliability and transparency while maintaining usability.

⑤ **Model Selection and Deployment.** Once users identify the most suitable solution, usually by selecting a trade-off value (theta) along the Pareto front, they can save the model directly from the interface. This ensures the final configuration matches the defined fairness and performance priorities. Users can also export all relevant plots and charts for documentation and future audits.

In this credit scoring scenario, *mmm-fair* enables data scientists to systematically explore model trade-offs and identify solutions that
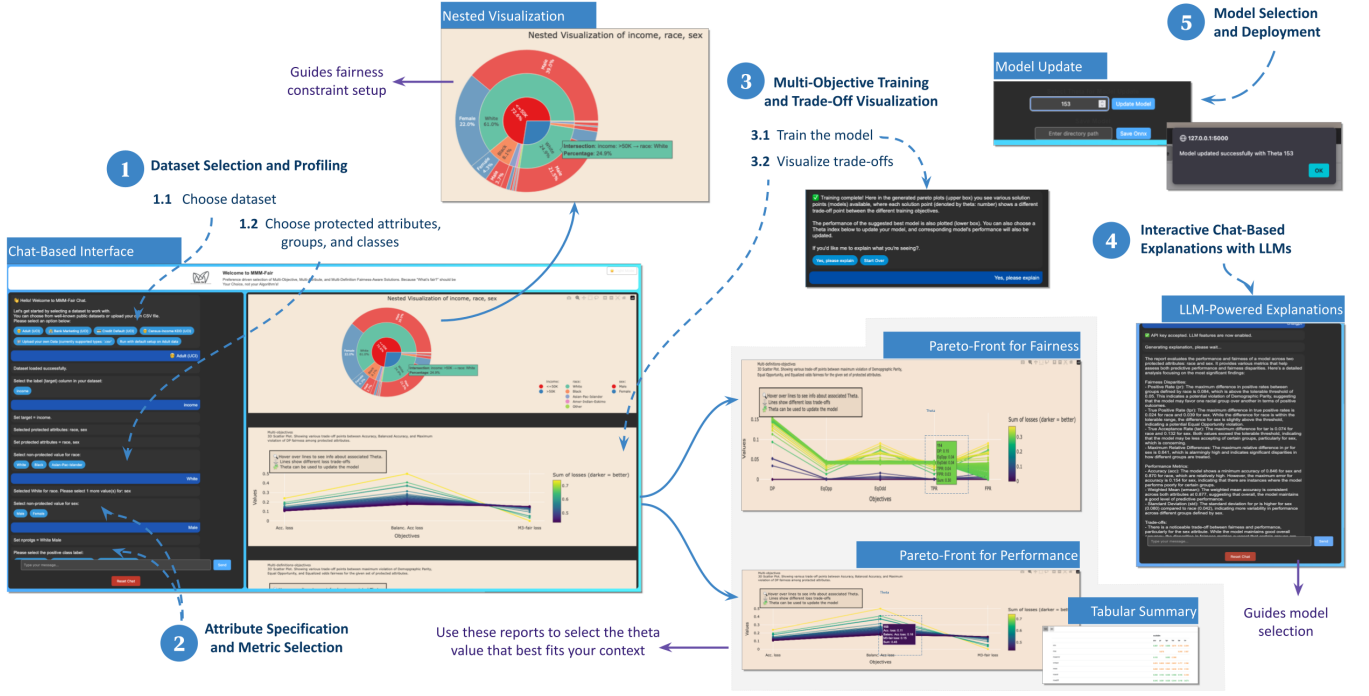
Figure 1: End-to-end pipeline for exploring and operationalizing multi-fairness trade-offs in *mmm-fair*.

best balance accuracy and fairness across different demographic groups. Policy analysts can assess how various fairness definitions apply to intersecting attributes, ensuring that institutional priorities and regulatory guidelines are reflected in the final model. Compliance experts benefit from transparent reporting and intuitive explanations, which support audit readiness and strengthen accountability. Non-technical stakeholders, including managers and external partners, can understand the impact of each modeling decision through clear visualizations and natural language summaries.

This collaborative and interpretable approach is not limited to credit scoring. Demonstrated through a realistic end-to-end scenario, *mmm-fair*'s unified workflow and rich feature set guide users from dataset selection to model deployment. By combining multi-attribute fairness analysis, multi-objective optimization, and interactive exploration in a single interface, the toolkit empowers users to surface hidden biases, understand fairness-performance trade-offs, and select models that reflect their institutional goals. Rather than offering yet another fairness metric calculator, *mmm-fair* equips practitioners with a hands-on, decision-support system for building fair, auditable, and deployment-ready models. It turns fairness from an abstract ideal into a concrete, actionable outcome. Preliminary internal evaluations indicate that the workflow facilitates adoption across diverse user groups, with usability studies planned to assess its effectiveness.

4 Conclusion and Future Work

In this work, we introduce *mmm-fair*, an interactive Python toolkit for exploring and operationalizing multi-fairness trade-offs in fairness-aware machine learning. It combines a boosting-based ensemble

approach with a no-code, chat-driven interface, integrated LLM explanations, interactive Pareto front visualizations, and deployment-ready models, enabling users to define fairness constraints, examine subgroup outcomes, and address intersectional bias while preserving predictive accuracy.

Limitations and Future Work. The current implementation supports only tabular data with manually specified protected attributes; the explanation module relies on external LLMs with user-provided API keys. At present, it is best suited for small to medium-scale analyses and does not yet support causal inference, longitudinal modeling, or large-scale deployment. Planned extensions include desktop and cloud-based interfaces, support for local models and hybrid explanation methods, and broader applicability to vision, text, and other non-tabular modalities.

Ethics statement.

mmm-fair promotes transparency and aims to reduce algorithmic bias; however, users must choose fairness definitions and interpret results with care, as societal contexts vary. While it does not guarantee universal fairness, the toolkit empowers practitioners to explore trade-offs and make informed decisions.

Acknowledgments

GenAI Usage Disclosure

Generative AI tools (e.g., ChatGPT and GPT-4) were employed to assist with the preparation of this manuscript. Specifically, these tools were utilized for text refinement, editing for clarity and readability, and generating illustrative examples. No original intellectual contributions or novel results were generated by these tools. The authors confirm that all AI-generated content has been reviewed, verified for accuracy, and integrated thoughtfully into the manuscript, ensuring full compliance with ACM's Policy on Authorship, available at: https://www.acm.org/publications/policies/new-acm-policy-on-authorship.

References

[1] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2023. *Fairness and machine learning: Limitations and opportunities.* MIT press.

[2] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilović, et al. 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.

[3] Sarah Bird, Miro Dudík, Richard Edgar, Brandon Horn, Roman Lutz, Vanessa Milan, Mehrnoosh Sameki, Hanna Wallach, and Kathleen Walker. 2020. Fairlearn: A toolkit for assessing and improving fairness in AI. *Microsoft, Tech. Rep. MSR-TR-2020-32* (2020).

[4] Zhisheng Chen. 2023. Ethics and discrimination in artificial intelligence-enabled recruitment practices. *Humanities and Social Sciences Communications* 10, 1 (2023), 1–12.

[5] Sorelle A. Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. 2021. The (Im)possibility of fairness: different value systems require different mechanisms for fair decision making. *Commun. ACM* 64, 4 (March 2021), 136–143. doi:10.1145/3433949

[6] Krishna Gade, Sahin Geyik, Krishnaram Kenthapadi, Varun Mithal, and Ankur Taly. 2020. Explainable AI in industry: Practical challenges and lessons learned. In *Companion proceedings of the web conference 2020.* 303–304.

[7] Vasileios Iosifidis, Arjun Roy, and Eirini Ntoutsi. 2022. Parity-based cumulative fairness-aware boosting. *Knowledge and Information Systems* 64, 10 (2022), 2737–2770.

[8] Emmanouil Krasanakis and Symeon Papadopoulos. 2024. Towards Standardizing AI Bias Exploration. (2024). arXiv:2405.19022 [cs.LG]

[9] Tai Le Quy, Arjun Roy, Vasileios Iosifidis, Wenbin Zhang, and Eirini Ntoutsi. 2022. A survey on datasets for fairness-aware machine learning. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 12, 3 (2022), e1452.

[10] Arjun Roy, Jan Horstmann, and Eirini Ntoutsi. 2023. Multi-dimensional discrimination in law and machine learning-A comparative overview. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency.* 89–100.

[11] Arjun Roy, Vasileios Iosifidis, and Eirini Ntoutsi. 2022. Multi-fairness under class-imbalance. In *International Conference on Discovery Science.* Springer, 286–301.

[12] Swati Swati, Arjun Roy, and Eirini Ntoutsi. 2024. Exploring Fusion Techniques in Multimodal AI-Based Recruitment: Insights from FairCVdb. *arXiv preprint arXiv:2407.16892* (2024).

[13] TruEra. [n. d.]. AI Quality Education. https://truera.com/ai-quality-education/. Accessed: 2025-06-18.

[14] James Wexler, Mahima Pushkarna, Tolga Bolukbasi, Martin Wattenberg, Fernanda Viégas, and Jimbo Wilson. 2019. The what-if tool: Interactive probing of machine learning models. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 56–65.

[15] WhyLabs. 2025. WhyLabs – AI Observability and Data Monitoring Platform. https://whylabs.ai/ Accessed: 2025-06-18.