applied sciences

MDPI

Article

TWEEF: Trustworthiness Estimation and Enhancement Framework for Machine Learning Models

Jonathan Ugalde [1,2,*], Rodrigo Salas [2,3,4,*], Romina Torres [5,6], Daira Velandia [7,8], Aurelio F. Bariviera [9], Pablo A. Estevez [2,10] and Maria Paz Godoy [11]

1 Informatics Engineering School, Faculty of Engineering, Universidad de Valparaíso, Valparaíso 2340000, Chile
2 Millennium Institute for Intelligent Healthcare Engineering (iHealth), Santiago 7820436, Chile
3 Biomedical Engineering School, Faculty of Engineering, Universidad de Valparaíso, Valparaíso 2362905, Chile
4 Center of Interdisciplinary Biomedical and Engineering Research for Health-MEDING, Universidad de Valparaíso, Valparaíso 2540064, Chile
5 Faculty of Engineering, Universidad Adolfo Ibáñez, Viña del Mar 2520000, Chile
6 National Center of Artificial Intelligence, Santiago 7820605, Chile
7 Statistical Institute, Faculty of Science, Universidad de Valparaíso, Valparaíso 2340000, Chile
8 Center for Atmospheric Studies and Climate Change (CEACC), Universidad de Valparaíso, Valparaíso 2540064, Chile
9 Department of Business, ECO-SOS, Universitat Rovira i Virgili, 43204 Reus, Spain
10 Department of Electrical Engineering, Faculty of Physical and Mathematical Sciences, Universidad de Chile, Santiago 8370458, Chile
11 School of Information Engineering and Management Control, Faculty of Economic and Administrative Sciences, Universidad de Valparaíso, Valparaíso 2362735, Chile
* Correspondence: jonathan.ugalde@postgrado.uv.cl (J.U.); rodrigo.salas@uv.cl (R.S.)

Abstract

The rapid adoption of Machine Learning (ML) in high-impact domains has intensified the need for systematic tools to assess and improve the trustworthiness of predictive models beyond conventional performance metrics. This paper presents TWEEF (Trustworthiness Estimation and Enhancement Framework), a modular and extensible framework that operationalizes trustworthiness through the joint evaluation of performance, fairness, and interpretability. TWEEF integrates intuitionistic fuzzy logic and subjective logic to transform quantitative trust-related metrics into linguistic assessments, which are subsequently aggregated using operators such as the Linguistic Weighted Average (LWA), Gaussian Weighted Aggregation (GWA), and Subjective Logic (SL). The framework extends the scikit-learn ecosystem through a meta-estimator, the TrustworthyClassifier, which orchestrates metric computation, bias-mitigation procedures, surrogate-model generation, and trust aggregation within a unified, pipeline-compatible workflow. The framework is empirically evaluated through four experiments on widely used benchmark datasets (German Credit, COMPAS, and Adult) in binary classification settings. Results show that TWEEF consistently reveals fairness and interpretability limitations that may remain hidden when relying solely on predictive performance, and that the resulting trust scores respond coherently to different metric configurations and weighting schemes. These findings indicate that TWEEF provides a structured mechanism for trust assessment and enhancement, while also offering a flexible foundation for future extensions to additional learning tasks and evaluation dimensions.

Keywords: trustworthiness; fairness; interpretability; fuzzy logic; subjective logic

Academic Editors: Raheleh Jafari, Alexander Gegov and Farzad Arabikhan

Received: 19 December 2025
Revised: 13 January 2026
Accepted: 16 January 2026
Published: 21 January 2026

1. Introduction

The widespread adoption of Machine Learning systems in decision-making processes across critical and high-impact domains (such as healthcare, finance, education, and cybersecurity) has raised fundamental concerns regarding the reliability, transparency, and ethical implications of algorithmic predictions. The concept of trustworthy AI has emerged as a critical research priority, emphasizing that AI systems must exhibit key properties including reliability, explainability, privacy, fairness, and accountability to be considered trustworthy [1]. Despite their predictive power, many of these systems operate as opaque "black boxes," whose decisions cannot easily be interpreted, audited, or explained. Traditional performance indicators, such as Accuracy or F1-Score, have historically dominated the evaluation of model quality. However, these metrics alone are insufficient to represent the broader concept of trustworthiness, which encompasses not only predictive competence but also fairness and interpretability, and may further include robustness and consistency under varying operational conditions [2–6]. Addressing this multidimensional challenge requires moving beyond performance-oriented validation toward frameworks that integrate diverse dimensions and provide interpretable, uncertainty-aware assessments for both technical and non-technical stakeholders.

Building on these theoretical and methodological foundations, this work introduces TWEEF, an extensible and modular framework designed to quantify, interpret, and improve the trustworthiness of Machine Learning models. In its current implementation, TWEEF focuses on three core trustworthiness dimensions (performance, fairness, and interpretability), which are explicitly operationalized and empirically evaluated in this study. Unlike existing frameworks that focus primarily on evaluation (e.g., CERTIFAI [7]) or single dimensions of trust (e.g., AIF360 for fairness [8]), TWEEF provides a unified, pipeline-oriented approach that integrates these three dimensions into a single interpretable trustworthiness score while being explicitly designed to support future extensions toward additional dimensions such as robustness, privacy, and accountability. Accordingly, TWEEF operationalizes trustworthiness by decomposing it into measurable components (performance, fairness, and interpretability) each represented by quantitative metrics that can be customized to the application domain. The framework integrates fuzzy and probabilistic reasoning within a unified computational architecture: the `TrustworthinessCalculator` module performs the fuzzy-to-linguistic transformation described in [9] through triangular membership functions, while the higher-level `TrustworthyClassifier` aggregates these linguistic values using Linguistic Weighted Average (LWA), Gaussian Weighted Aggregation (GWA), and Subjective Logic (SL) operators, enabling interpretable and uncertainty-aware trust estimation.

This hybridization of intuitionistic fuzzy aggregation and subjective logic fusion situates TWEEF within the paradigm of Trustworthy AI by Design, where trust is not an afterthought but an intrinsic property of the Machine Learning lifecycle. By embedding these mechanisms into a `scikit-learn`-compatible pipeline, TWEEF provides a reproducible, model-agnostic, and interpretable infrastructure for trustworthiness assessment. Furthermore, the framework incorporates explicit feedback mechanisms that enable the enhancement of trust-related properties through metric reweighting, fairness-aware processing, surrogate-based interpretability analysis, and uncertainty-aware aggregation. This dual capability (both estimating and enhancing trustworthiness) distinguishes TWEEF from purely diagnostic tools and positions it as a framework for trust-aware Machine Learning development grounded in transparent and configurable design choices.

TWEEF's implementation was guided by three foundational design principles: (i) flexibility, allowing the configuration of custom metric sets and aggregation strategies according to the evaluation context; (ii) extensibility, enabling researchers to incorporate

new metrics, operators, and processing components; and (iii) interpretability, ensuring that trustworthiness assessments remain transparent and explainable to human evaluators. Under these principles, the framework not only quantifies the current trust level of Machine Learning models but also supports structured enhancement procedures that remain explicitly traceable and reproducible.

Experimental validation is conducted in the context of binary classification tasks, where TWEEF is applied to representative benchmark datasets to assess the stability, coherence, and sensitivity of trustworthiness scores under different metric configurations and aggregation strategies. The comparative analysis of aggregation mechanisms reveals distinct behavioral profiles: while LWA tends to allow compensatory effects among metrics, GWA provides smoother integration under heterogeneous evidence, and SL yields more conservative assessments by explicitly modeling epistemic uncertainty. These findings suggest that the selection of aggregation mechanisms constitutes a normative design choice that should align with the intended evaluation context and risk sensitivity of the application domain.

The remainder of this paper is structured as follows. Section 2 reviews related works in trust assessment. Section 3 presents the architecture and main components of the TWEEF framework. Section 4 details the experimental setup and comparative results. Finally, Section 5 summarizes the main contributions and outlines limitations and potential extensions to multi-class, regression, and real-time trust-monitoring contexts.

2. Related Works

The pursuit of trustworthy AI has gained significant momentum recently, driven by both academic research and regulatory initiatives. Wing [1] established a foundational vision of trustworthy AI, emphasizing that systems must be reliable, explainable, private, fair, and accountable. Thiebes et al. [10] synthesized principles and requirements for trustworthy AI aligned with international guidelines, discussing ethical principles such as beneficence, non-maleficence, autonomy, justice, and explainability. More recently, Kowald et al. [11] provided a comprehensive overview identifying six key requirements for trustworthy AI: human oversight, fairness and non-discrimination, transparency and explainability, robustness and accuracy, privacy and security, and accountability. Together, these contributions establish a normative and conceptual foundation for multidimensional trustworthiness, which motivates (but does not uniquely determine) its operationalization in concrete Machine Learning systems.

Early studies on perceived trust connected user confidence either to the predictive performance of algorithms or to their interpretability in isolation [12,13]. Empirical results demonstrated that greater Accuracy often leads to increased trust in algorithmic decisions [12], while visual or linguistic explanations enhance the perceived transparency of black-box models [13]. However, treating these dimensions independently overlooks their interdependence: performance, fairness, and interpretability jointly shape both user trust and model reliability. This limitation motivated the emergence of integrated approaches capable of diagnosing (and in some cases mitigating) unfairness, opacity, or overconfidence in predictive systems.

Among the most widely recognized frameworks, CERTIFAI [7] is a black-box approach that employs counterfactual explanations to jointly assess interpretability and fairness according to the counterfactual definition of fairness [14]. CERTIFAI was one of the first unified frameworks to simultaneously analyze robustness, fairness, and explainability of black-box models using genetic algorithms for counterfactual generation. Although pioneering, CERTIFAI operates primarily as an analytical and diagnostic tool, providing assessments without enabling systematic, iterative enhancement or explicit control over

trade-offs among trust dimensions. Furthermore, CERTIFAI does not produce a unified trustworthiness score aggregating multiple dimensions, nor does it explicitly address accountability as an operational component.

Other contributions, such as Aequitas [15], provide auditing capabilities that generate detailed reports of bias and disparity across protected groups, serving as decision-support tools for data scientists. Likewise, IBM's AI Fairness 360 (AIF360) [8] constitutes one of the most comprehensive open-source toolkits for fairness assessment, offering libraries for detecting and mitigating bias before, during, and after model training. AIF360 provides an extensive catalog of fairness metrics and mitigation algorithms, but focuses exclusively on fairness and does not address interpretability or aggregation of trust dimensions. Complementary to AIF360, IBM's AI Explainability 360 (AIX360) [16] toolkit focuses on explainability methods, providing algorithms for both local (e.g., LIME, SHAP) and global explanations, along with metrics to evaluate explanation quality. However, AIX360 does not integrate fairness assessment and operates as a specialized toolkit rather than a unified trustworthiness framework. Similarly, the Adversarial Robustness Toolbox (ART) [17] provides comprehensive evaluation of adversarial robustness and defenses for Machine Learning models but focuses solely on robustness without addressing fairness or interpretability. While these frameworks have significantly advanced bias auditing, explainability, and robustness analysis, they typically address trust dimensions in isolation and provide limited support for aggregating heterogeneous evidence or linking evaluation to systematic model enhancement.

Fairness in Machine Learning has received increasing attention over the past decade, leading to a proliferation of definitions [2,18] and bias mitigation techniques [19–22]. These methods can be grouped according to the stage of the learning process in which they operate [2,8]. Pre-processing methods act on the dataset before training to remove or compensate for bias, including suppression of sensitive attributes, dataset massaging via minimal label reassignment using Naïve Bayes, reweighing of samples to counterbalance biased distributions [23], and sampling-based approaches that combine under- and over-sampling [24]. While attribute suppression alone does not always reduce bias due to indirect correlations between features [25], massaging and sampling techniques have shown promising results in achieving fairness with limited performance degradation. In contrast, in-processing methods modify the learning algorithm itself by incorporating fairness constraints or regularization terms into the optimization objective [26,27]. Finally, post-processing approaches adjust model predictions after training by modifying decision boundaries or probabilities to equalize outcomes across groups, including Equalized Odds post-processing [28], its calibrated variant [29], and Reject Option Classification [27].

In parallel with fairness research, interpretability has become a central topic in the pursuit of trustworthy artificial intelligence [3,4,6]. The AIX360 toolkit [16] provides a comprehensive taxonomy of explainability techniques and implements methods for both model-level (global) and instance-level (local) explanations. Global approaches aim to describe overall model behavior through surrogate representations, such as Partial Dependence Plots (PDP) [30], while local approaches focus on individual predictions using counterfactual explanations [31], SHAP [32], decision tree-based surrogates [33], or LIME [34]. These techniques have become standard tools in explainable AI research and practice.

The evaluation of interpretability can be broadly categorized into human-centered and functionality-based approaches. Human-centered evaluation relies on user studies assessing perceived interpretability and utility [35–37], whereas functionality-based evaluation employs quantitative metrics derived from model structure or behavior [4]. Such metrics include model size, interaction strength [38], operation counts [39], disagreement

between explanation methods [40], as well as fidelity- and stability-oriented measures such as monotonicity, effective complexity, sensitivity, and implementation invariance [41]. These metrics enable systematic comparison of explanation quality across models without requiring extensive human evaluation.

Beyond fairness and interpretability, fuzzy logic and aggregation-based approaches have emerged as effective mechanisms for integrating heterogeneous evaluation criteria into composite indicators. Deep Fuzzy Systems combine neural architectures with fuzzy inference mechanisms to improve interpretability without sacrificing predictive accuracy [42,43]. More generally, fuzzy aggregation operators enable the transformation of quantitative metrics into linguistic values and their combination through non-linear operators, enhancing robustness to noise and improving the interpretability of aggregated scores [43–45]. Such approaches have been successfully applied in information fusion tasks [46] and ensemble learning, where aggregation functions are optimized to balance performance and bias [47].

Taken together, the existing literature reveals a mature ecosystem of metrics and toolkits (such as CERTIFAI, Aequitas, AIF360, AIX360, and ART) that support the evaluation of individual trust dimensions. However, challenges remain in holistically integrating performance, fairness, and interpretability under uncertainty, aggregating heterogeneous evidence in an interpretable manner, and linking trust evaluation with structured enhancement mechanisms. TWEEF is positioned within this landscape as an integrative, pipeline-oriented framework that focuses on these three dimensions, explicitly addressing aggregation and enhancement while remaining compatible with established fairness and explainability toolkits.

3. Framework Design and Development

The TWEEF (Trustworthiness Estimation and Enhancement Framework) follows a modular and extensible architecture within the `scikit-learn` ecosystem, as illustrated in Figure 1. A central TrustworthyClassifier orchestrates estimators, processors, metrics, and utility modules, enabling pipeline-compatible evaluation and enhancement of model performance, fairness, and interpretability. The framework was designed to be compatible with Python 3.9 and later versions (Windows, Linux, or macOS), and integrates naturally with the `scikit-learn` API.

TWEEF layered structure enables interoperability, reusability, and transparency across the different stages of model trustworthiness estimation and enhancement, from data ingestion and metric computation to linguistic transformation and user interaction. The framework is publicly available as an open-source project under the MIT License in a GitHub repository. TWEEF is publicly available in https://github.com/ugald80/TWEEF (accessed on 13 January 2026). The repository provides the complete implementation of the framework, including the trustworthiness classifier, metric computation modules, aggregation mechanisms, versioning information, dependency specifications, and example configuration files to support reproducibility and facilitate reuse and extension by the research community.

3.1. Architecture

TWEEF is organized around a central component, the `TrustworthyClassifier`, which acts as a meta-estimator built on top of the `scikit-learn` API. This classifier coordinates the interaction among the main architectural modules: Estimators, Processors, Metrics, and Utils. Each of these modules encapsulates specific functionalities and communicates through well-defined interfaces, ensuring modularity, reproducibility, and ease of extension.

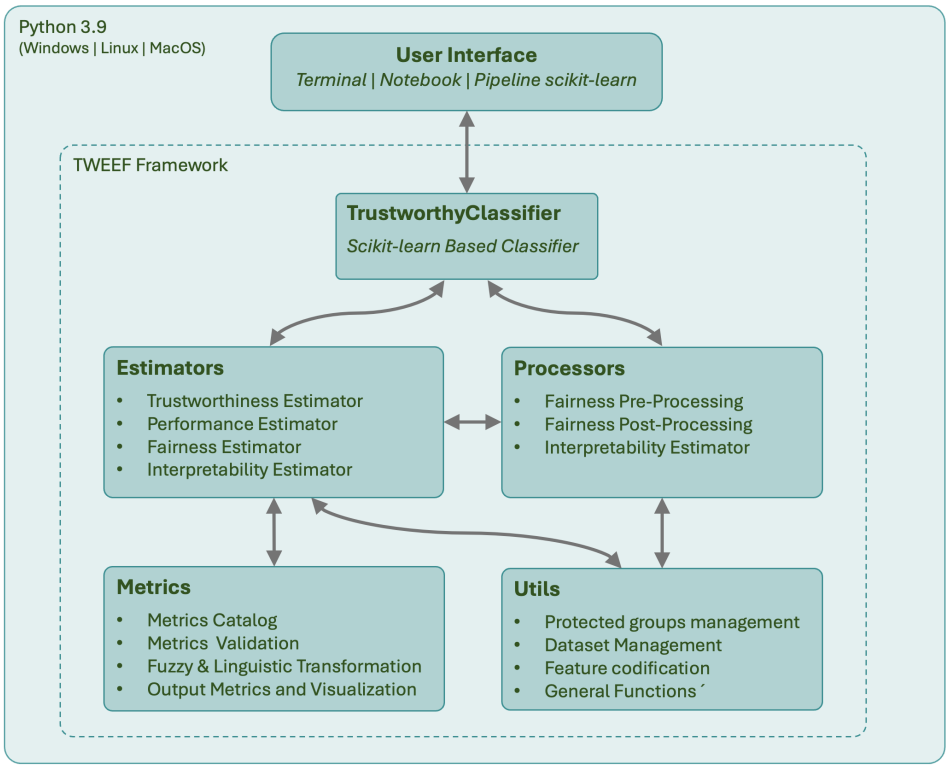


Figure 1. Software architecture of the TWEEF framework. The `TrustworthyClassifier` orchestrates estimators, processors, and metric modules integrates with `scikit-learn` pipelines, and provides interfaces for model evaluation and trustworthiness enhancement.

TWEEF offers a multi-channel user interface that allows interaction through command-line tools, Jupyter notebooks, or direct integration into `scikit-learn` pipelines. Users can fit, evaluate, and visualize model trustworthiness through a unified API, ensuring accessibility for both researchers and practitioners. Trustworthiness results are visualized as bar charts, reporting the overall trustworthiness indicator together with its three operationalized components: performance, fairness, and interpretability. These visualizations can be generated in a post-hoc manner after trustworthy model training and post-processing fairness adjustment.

3.1.1. Core Component: TrustworthyClassifier

At the core of TWEEF lies the `TrustworthyClassifier`, a meta-estimator that inherits from `scikit-learn`'s `BaseEstimator` and `ClassifierMixin`. This class serves as the central orchestrator of the framework, managing the data flow between trustworthiness evaluation components and the user interface. It extends the standard training and evaluation routines of `scikit-learn` to incorporate the computation of trust-related metrics, the application of fairness and interpretability processors, and the aggregation of results through fuzzy and probabilistic reasoning mechanisms.

The `TrustworthyClassifier` is explicitly designed to support a two-stage trustworthiness assessment workflow. First, an initial diagnostic evaluation of trustworthiness and its three components (performance, fairness, and interpretability) is performed on a given base model and the original input dataset. This initial diagnosis provides a baseline characterization of the model's trust profile before any enhancement or mitigation procedures.

Subsequently, after the application of fairness pre-processing and post-processing techniques and the training of an interpretable surrogate model, a second trustworthiness evaluation is conducted on the resulting model and the potentially modified dataset. By comparing the initial and final trustworthiness assessments, TWEEF enables explicit

analysis of trade-offs between trust dimensions, such as fairness gains versus performance or interpretability variations.

Internally, the `TrustworthyClassifier` supports three explicitly separated modes of operation: training, testing, and trust estimation. Training and testing follow standard train-test splits consistent with the underlying `scikit-learn` estimator, while trust estimation operates on held-out evaluation data using parameters calibrated exclusively on the training set, thereby preventing information leakage.

3.1.2. Estimator Layer

The Estimator module provides dedicated components for computing the operationalized dimensions of model trustworthiness. It includes:

- Trustworthiness Estimator: Integrates the outputs of all selected metric dimensions to produce a unified trustworthiness score, reflecting the model's overall assessment under one of the *LWA*, *GWA* or *SL* aggregation mechanisms described in Section 3.2.
- Performance Estimator: Computes classical performance metrics such as Accuracy, Precision, Recall, and F1-Score.
- Fairness Estimator: Calculates group-based fairness indicators incorporated in TWEEF, including Statistical Parity, Treatment Equality, Equalized Odds, and Equal Opportunity, which are widely used in the fairness literature [2,18].
- Interpretability Estimator: Computes quantitative interpretability metrics based on a feature-attribution approach (by using the Shapley values [48] method) including Monotonicity, Non-Sensitivity, and Effective Complexity [49]. Monotonicity evaluates the ordinal consistency between feature attributions and their expected predictive impact using Spearman's rank correlation coefficient, and is scaled to the $[0, 1]$ interval by taking its absolute value. Non-Sensitivity assesses whether only functionally irrelevant features are assigned zero attribution, and is computed as the normalized proportion of attributes whose zero-attribution assignments disagree with their expected predictive relevance. Effective Complexity measures the minimum number of dominant features (with higher absolute attributions) required to preserve predictive performance within a predefined tolerance (set up to 0.95% of total attributions). Then, Effective Complexity is normalized $[0, 1]$ by dividing the number of dominant features by the total number of input features. Interpretability metric normalization ensures comparability and consistent aggregation with performance and fairness dimensions.

Each estimator is implemented as a standalone class that can be invoked independently or as part of the trustworthiness pipeline, enabling flexible integration with external analytical workflows and supporting reproducible experimentation.

3.1.3. Processor Layer

The Processor module implements the enhancement capabilities of TWEEF by applying reproducible fairness and interpretability interventions around the standard model-training workflow. Fairness enhancement is implemented through AIF360 [8] and is decomposed into two complementary stages: first, a pre-processing bias mitigation based on remove_disparate_impact adjusting the training data distribution to reduce group-level disparities before fitting the predictive model. Second, a post-processing step based on Equalized Odds [28] that adjusts predictions to satisfy a user-defined fairness threshold after model training, following established post-processing principles [27].

In parallel, the module provides an Interpretability Processor that trains an interpretable surrogate model using the Explainable Boosting Machine (EBM) [50]. EBM is selected due to its inherently interpretable additive structure and its strong empirical fi-

delity when approximating complex classifiers, enabling interpretability assessment in a transparent and reproducible manner.

Processors interact iteratively with the estimator layer: after each fairness intervention (pre- or post-processing) and after surrogate training, TWEEF recomputes performance, fairness, and interpretability metrics to quantify the effect of each adjustment. More details on Fairness or Interpretability processor operation description are described in Section 3.3.

3.1.4. Metrics Layer

The Metrics module serves as the quantitative foundation of the framework. It maintains a comprehensive metrics catalog containing all supported performance, fairness, and interpretability measures. The module also implements a metrics validation process that checks for data consistency, missing values, and the validity of protected attribute definitions before metric computation. If missing values are detected in protected attributes or target labels, affected samples are excluded to prevent invalid fairness estimates. Also, if protected attribute specifications are inconsistent (e.g., undefined protected values or invalid group encoding), or out-of-range values or degenerate cases arise during metric computation (e.g., divisions by zero or undefined rates), TWEEF computation is aborted and an explicit validation error is returned.

Once validated, numerical metric values are transformed into linguistic variables through fuzzy and intuitionistic logic operators [9,51,52]. This transformation bridges numerical evaluation and human interpretability and forms the basis for subsequent aggregation into trustworthiness scores.

3.1.5. Utility Layer

The Utils module provides auxiliary functionalities that support all other components of the framework. It includes tools for managing protected group definitions, dataset partitioning, and general-purpose preprocessing functions. These utilities ensure that data handling and metric computation remain consistent across multiple executions and datasets. By decoupling generic operations from the core logic, this layer enhances the framework's maintainability, reproducibility, and ease of extension.

3.2. Aggregation Mechanisms

The aggregation layer of TWEEF combines heterogeneous trust-related evidence into a unified trustworthiness score. Before aggregation, TWEEF performs a calibration step that defines the fuzzy representation of each metric. This separation ensures that aggregation operators operate on a consistent linguistic or probabilistic input space, independently of how the calibration is obtained.

TWEEF supports two complementary calibration strategies for defining the triangular membership functions used in trustworthiness estimation. In a data-driven configuration, the framework receives a collection of metric values obtained from previously trained (and optionally diagnosed) models. For each metric, empirical quartiles are estimated from its statistical distribution, typically using the percentiles $\{20, 40, 60, 80\}$. These quartiles define the intersection points between adjacent triangular membership functions along the metric domain, yielding five ordered linguistic terms ranging from low to high trust evidence. Alternatively, TWEEF can be calibrated through user-provided quartiles. In this case, the quartile values are supplied as inputs (such as the default configuration used in this paper, $\{0.2, 0.4, 0.6, 0.8\}$) and the corresponding triangular membership functions are constructed deterministically.

For each triangular membership function, a corresponding non-membership function is derived as its complement, yielding inverse degrees of evidence that quantify opposition to the associated linguistic term. Together, membership and non-membership functions

provide the fuzzy evidence required by intuitionistic and uncertainty-aware aggregation operators. These triangular functions constitute the common representational layer across all aggregation mechanisms. Each operator subsequently consumes the outputs of these functions differently, reflecting distinct assumptions about compensability, uncertainty, and interpretability.

3.2.1. Linguistic Weighted Average (LWA)

The Linguistic Weighted Average (LWA) [9] aggregates trust evidence expressed directly in the linguistic domain. For each metric, the calibrated triangular membership functions yield a membership vector $z(x) = [\mu_0(x), \ldots, \mu_4(x)]$, from which a linguistic index q is obtained by projection onto the ordered scale $[-2, -1, 0, 1, 2]$. This index represents the dominant linguistic assessment induced by the triangular functions.

Given a set of linguistic values $S = \{s_q^1, \ldots, s_q^K\}$ and relevance weights $W = \{w_1, \ldots, w_K\}$ with $\sum_k w_k = 1$, LWA computes the aggregated linguistic trust assessment as

$$LWA(S) = \bigoplus_{k=1}^{K} w_k s_q^k, \tag{1}$$

where $\oplus$ denotes linguistic addition, preserving the ordinal semantics defined by the triangular membership structure.

The aggregated linguistic value lies in $[-2, 2]$, corresponding to trust levels from Very Untrustworthy, Untrustworthy, Neutral, Trustworthy to Very Trustworthy and is mapped to a normalized scalar score for numerical analysis:

$$Tw_{LWA} = \frac{LWA(S) + 2}{4}. \tag{2}$$

In the experiments reported in this paper, one representative metric per trustworthiness dimension is used, yielding $S = \{s_P, s_F, s_I\}$ with corresponding weights $\{w_P, w_F, w_I\}$.

3.2.2. Gaussian Weighted Aggregation (GWA)

The Gaussian Weighted Aggregation (GWA) operator [51] combines trust evidence through a non-linear fusion mechanism that limits compensability between criteria. In contrast to LWA, GWA does not aggregate linguistic indices directly. Instead, it consumes the same triangular membership activations to construct intuitionistic fuzzy values (IFVs). Specifically, for each metric, favorable and unfavorable evidence degrees μ_i and ν_i are derived from the calibrated triangular membership and non-membership functions. Where $\psi_i = \langle \mu_i, \nu_i \rangle$ with $0 \leq \mu_i + \nu_i \leq 1$. Given weights ω_i such that $\sum_{i=1}^{n} \omega_i = 1$, the Gaussian aggregation combines these IFVs via the error function $\mathrm{erf}(\cdot)$ and its inverse, producing an aggregated pair $\langle \mu_{\mathrm{agg}}, \nu_{\mathrm{agg}} \rangle$:

$$\mathrm{GWA}(\psi_1, \ldots, \psi_n) = \left\langle 1 - \mathrm{erf}^{-1}\left(\prod_{i=1}^{n} \frac{\mathrm{erf}(1-\mu_i)^{\omega_i}}{\mathrm{erf}(1)^{\omega_i - 1/n}} \right), \quad \mathrm{erf}^{-1}\left(\prod_{i=1}^{n} \frac{\mathrm{erf}(\nu_i)^{\omega_i}}{\mathrm{erf}(1)^{\omega_i - 1/n}} \right) \right\rangle. \tag{3}$$

The resulting trustworthiness score based on GWA is obtained by defuzzification:

$$Tw_{\mathrm{GWA}} = \frac{\mu_{\mathrm{agg}}}{\mu_{\mathrm{agg}} + \nu_{\mathrm{agg}}}. \tag{4}$$

By operating directly on the degrees induced by the triangular functions, GWA preserves the fuzzy semantics of the calibration step while introducing non-linear smoothing that reduces overcompensation.

3.2.3. Subjective Logic (SL)-Based Aggregation

Subjective Logic (SL) provides an uncertainty-explicit, probabilistic reasoning layer for trust aggregation under incomplete or imprecise evidence [52,53]. Unlike fuzzy operators that encode uncertainty implicitly through membership structure, SL represents each trust dimension as an opinion $\omega = \langle b, d, u \rangle$ composed of belief b, disbelief d, and uncertainty u, constrained by $b + d + u = 1$. This representation is well aligned with trust assessment scenarios, where metrics may be reliable in some regions of the feature space yet uncertain or unstable in others, and where it is useful to preserve an explicit uncertainty term rather than collapsing it into a single scalar.

In TWEEF, each performance, fairness, and interpretability metric value $x_i \in [0, 1]$ serves as evidence to construct opinions by introducing belief and disbelief components derived from the triangular membership and non-membership functions, respectively.

$$b_i = x_i - \frac{\pi_i}{2}, \qquad d_i = 1 - x_i - \frac{\pi_i}{2}, \qquad u_i = \pi_i = \frac{Q_{0.75}(x_i) - Q_{0.25}(x_i)}{\max(x_i) - \min(x_i)}. \tag{5}$$

where π_i is a metric-specific hesitation (uncertainty) parameter computed as the normalized interquartile. Then, these metrics are fused using the SL consensus operator, which balances supportive and contradictory evidence while propagating uncertainty through the fusion process:

$$K = u_1 + u_2 - u_1 u_2, \quad b = \frac{b_1 u_2 + b_2 u_1}{K}, \quad d = \frac{d_1 u_2 + d_2 u_1}{K}, \quad u = \frac{u_1 u_2}{K}. \tag{6}$$

The final trustworthiness estimate based on SL is obtained via the expected probability projection:

$$TW_{\mathrm{SL}} = b + \alpha u, \tag{7}$$

where $\alpha \in [0, 1]$ denotes the base rate (commonly $\alpha = 0.5$ when no prior preference is assumed). In practice, SL tends to be more conservative than purely fuzzy aggregation because high uncertainty directly reduces the effective confidence in the final score, making it particularly appropriate for high-stakes settings where uncertainty must be surfaced rather than averaged away.

3.3. Framework Operation Flow

The execution of the TWEEF framework follows a coherent, modular, and fully traceable sequence of operations in which each component contributes to the diagnosis and potential enhancement of a Machine Learning model regarding performance, fairness, and interpretability. The process is orchestrated by the `TrustworthyClassifier`, which acts as the central controller coordinating data preparation, metric computation, bias mitigation procedures, surrogate model training, and trustworthiness estimation. The overall operation flow of TWEEF is illustrated in Figure 2.

The primary objective of this workflow is to enable a structured comparison between the trustworthiness profile of an input predictive model and that of a final interpretable surrogate model, thereby making explicit the trade-offs introduced by fairness and interpretability enhancement procedures. TWEEF is therefore designed around a two-stage trustworthiness assessment paradigm: an initial diagnostic evaluation performed on the user-provided model and a final evaluation performed on the surrogate model obtained after mitigation and explanation steps.

The process begins with the identification of protected groups based on user-defined protected attribute specifications (including the protected values associated with each protected attribute), which constitute the basis for all subsequent fairness evaluation and

mitigation procedures. These definitions are provided explicitly as configuration parameters. Once protected groups are defined, the user configures the set of performance, fairness, and interpretability metrics to be evaluated, together with the aggregation operator used to compute trustworthiness scores. These choices determine the normative structure of the assessment and reflect application-specific priorities. The user then selects one or more Machine Learning algorithms compatible with the `scikit-learn` API, which are incorporated into TWEEF without requiring any modification to the original model implementation.

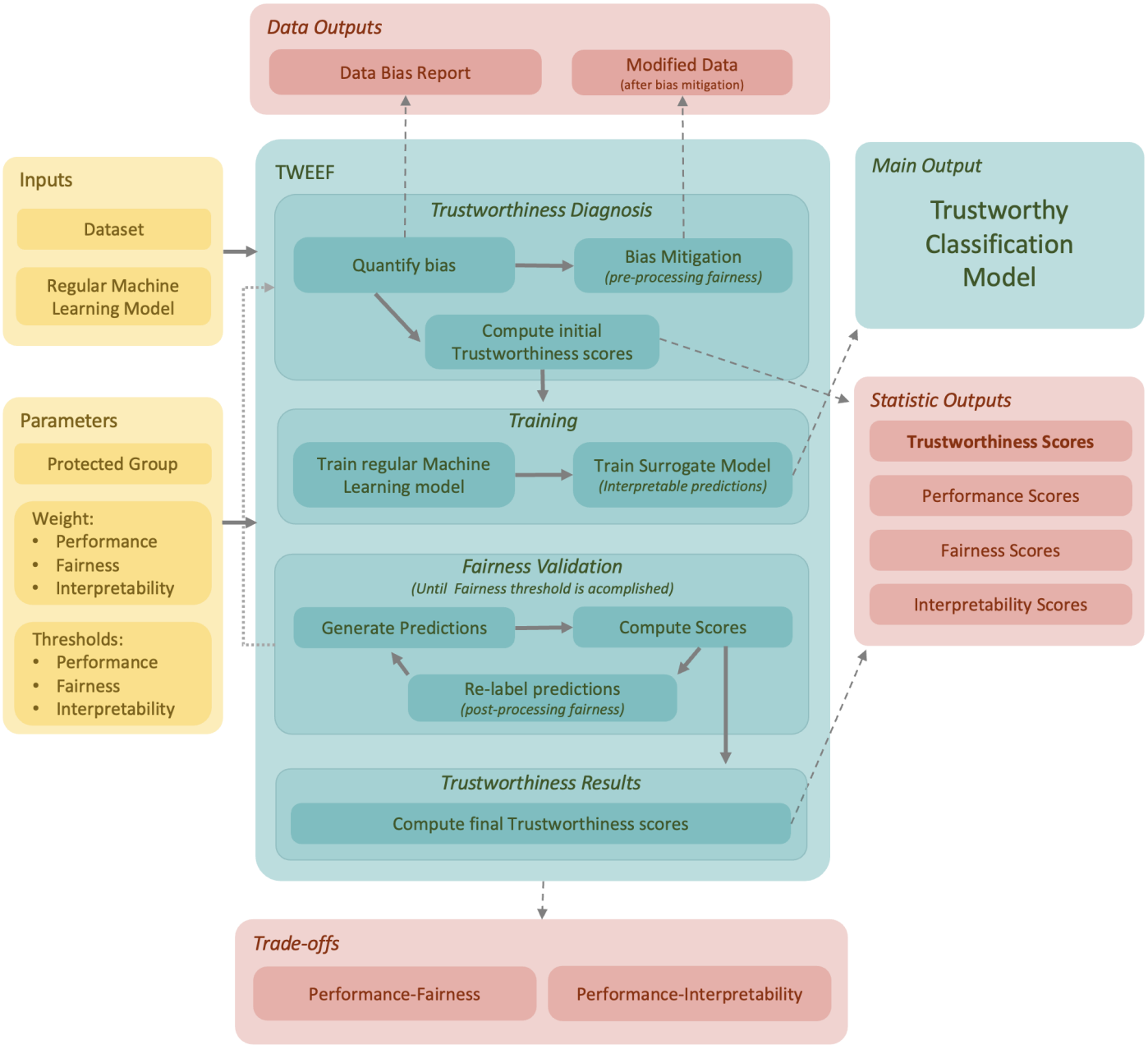


Figure 2. TWEEF's operation flow, including inputs, configuration parameters, internal processing stages, and outputs. Solid arrows describe operation flow, and dashed arrows indicate an output generation.

After setting the initial configuration, TWEEF performs an initial diagnostic trustworthiness assessment on the input model. In this stage, the model is trained using a standard train-test split, and performance, fairness, and interpretability metrics are computed on held-out evaluation data. This initial assessment yields a baseline trustworthiness

profile that characterizes the behavior of the original model before any enhancement or mitigation procedures.

After the trustworthiness diagnostic assessment, TWEEF activates its bias mitigation mechanisms. As a first corrective step, fairness pre-processing is applied to the training data using the remove_disparate_impact method implemented in the AIF360 framework, which performs dataset-level massaging to reduce group-level disparities before model retraining.

After this adjustment, TWEEF uses this adjusted data to train both the primary predictive model and an interpretable surrogate model. The surrogate model is constructed following a teacher-student strategy, where the original predictive model acts as a teacher whose predictions are used as targets for training an inherently interpretable student model. This surrogate training is performed iteratively until one of two conditions is met: (i) the predictive agreement between the surrogate model and the original model, quantified using the coefficient of determination (R^2) on held-out data, reaches or exceeds a user-defined minimum threshold ($R^2 \geq 0.8$); or (ii) a maximum number of surrogate training iterations is reached (set to 100 by default). In the current implementation, the Explainable Boosting Machine (EBM) is employed as a surrogate due to its transparent structure and strong empirical fidelity [50].

Once a satisfactory surrogate is obtained, feature attributions and expectation values are extracted exclusively from the surrogate model and used as the basis for interpretability metric computation. In this way, interpretability and final trustworthiness estimates are grounded on a model that is both faithful to the original predictor and inherently interpretable.

Following surrogate training, TWEEF applies post-processing fairness mitigation to the model's predictions. Specifically, the framework employs the Equalized Odds post-processing technique provided by AIF360, which adjusts predicted labels to reduce disparities across protected groups while preserving the underlying decision structure. This post-processing stage is governed by a user-configurable fairness threshold that defines the minimum acceptable level of fairness for the selected metric(s). The procedure operates iteratively, recalculating fairness metrics after each adjustment, until the threshold is met or a predefined iteration limit is reached. To ensure termination and computational tractability, the maximum number of post-processing iterations is capped at 100 by default, although this value can be configured by the user.

If the fairness threshold is satisfied within the allowed number of iterations, the resulting surrogate model is accepted as the final model. Otherwise, the procedure terminates after reaching the iteration limit, and the best-performing configuration observed during the process is retained.

Finally, TWEEF performs a second trustworthiness evaluation on the surrogate model and the potentially modified dataset. The same set of performance, fairness, and interpretability metrics used in the initial diagnostic stage is recomputed, and the results are aggregated using the selected fuzzy or probabilistic operator. By explicitly comparing the initial trustworthiness profile of the input model with the final trustworthiness profile of the surrogate model, TWEEF enables a systematic analysis of trade-offs between trust dimensions, such as fairness improvements relative to changes in predictive performance or interpretability.

The outputs of the framework include aggregated trustworthiness scores, component-level metric values for both evaluation stages, and post-hoc visualizations based on bar charts summarizing the evolution of performance, fairness, interpretability, and overall trustworthiness. Together, these outputs provide a transparent and reproducible basis for diagnosing, enhancing, and reasoning about trustworthiness trade-offs in Machine Learning models.

4. Framework Validation

This section presents the experimental evaluation conducted to validate the applicability of the proposed TWEEF framework. The experiments were designed to assess (i) the capability of TWEEF to produce consistent trustworthiness estimations across heterogeneous datasets, (ii) its effectiveness in diagnosing and improving fairness and interpretability limitations of predictive models, and (iii) its flexibility for exploring normative trade-offs through metric weighting. All experiments reported in this section are conducted using Support Vector Machine (SVM) classifiers.

Three publicly available datasets were used to evaluate TWEEF, representing diverse application domains and data characteristics. Table 1 summarizes the main characteristics of the datasets used in the experimental evaluation, including final sample size after preprocessing, class balance, and presence of missing values. Categorical attributes were encoded using one-hot encoding, substantially increasing the feature space, particularly for COMPAS. Class balance is reported as the proportion of positive instances in the final processed datasets.

Table 1. Descriptive statistics of the datasets used for framework validation.

Dataset	Original Samples	Final Samples	Final Features	Class Balance
German Credit	1000	1000	48	30.0%
COMPAS	7214	6172	401	45.3%
Adult	48,842	45,222	104	24.1%

As the framework is compatible with Python 3.9 and later versions, all experiments were conducted using Python 3.11, `scikit-learn 1.5.1`, `NumPy 2.3.1`, `Pandas 2.3.1`, and `Matplotlib 3.9.2`. In all experiments, the predictive model corresponds to a Support Vector Machine (SVM) classifier with a radial basis function (RBF) kernel. SVM hyperparameters were fixed throughout all evaluations, with $C = 1.0$, $\gamma =$ scale, class weights set to `balanced` and a maximum of 10,000 training iterations. All reported trustworthiness scores represent the mean value obtained over 4-fold cross-validation repeated 3 times. The multi-dimensional evaluation framework considers three key trustworthiness aspects (performance, fairness, and interpretability), each operationalized through one or more quantitative metrics, as summarized in Table 2.

Table 2. Trustworthiness dimension metrics integrated in TWEEF.

Dimension	Metric(s)
Performance	Accuracy, Precision, Recall, F1-Score
Fairness	Statistical Parity, Treatment Equality, Equalized Odds, Equal Opportunity
Interpretability	Monotonicity, Non-Sensitivity, Effective Complexity

In general, the weights associated with each trustworthiness dimension reflect normative design choices regarding what aspects of model behavior are considered more critical in a given application domain. However, in this study all trustworthiness dimensions are assigned equal weights by design. This choice is intentional and aims to avoid embedding implicit value judgments (such as systematically privileging predictive performance over fairness or interpretability) into the aggregation process. By adopting equal weights, the resulting trustworthiness scores are intended to reflect the structural behavior of the aggregation mechanisms themselves, rather than domain-specific or stakeholder-driven prioritization assumptions. The linguistic aggregation operator used to compute the trustworthiness score was LWA, as the baseline operator.

The experiments provide empirical evidence of how the framework captures imbalances, highlights unjustified disparities, and exposes interpretability limitations of widely used Machine Learning models.

4.1. Experiment 1: German Credit Dataset

The first experiment employed the German Credit dataset, which assesses creditworthiness based on demographic and financial features. Bias in this dataset, particularly against women and specific marital status categories, has been documented extensively in prior research. For this experiment, the protected attributes were gender and civil status, and the protected group was defined as female. The selected metrics were Precision (performance), Treatment Equality (fairness), and Effective Complexity (interpretability). The coefficient of determination of the surrogate model for this experiment reached $R^2 = 0.79$.

As shown in Figure 3, the diagnostic phase of TWEEF revealed that while the baseline SVM model achieved satisfactory Precision, it exhibited notable deficiencies in fairness and interpretability. In particular, the Treatment Equality metric highlighted substantial disparities in misclassification costs between protected and non-protected groups, with higher error rates affecting female applicants, consistent with known biases in the German Credit dataset. In addition, the model presented a relatively high Effective Complexity, indicating limited interpretability of its decision boundary. After applying TWEEF, improvements are observed across all trust dimensions, including Precision, fairness, and interpretability, leading to a higher overall trustworthiness score (Tw_{LWA}). These results demonstrate TWEEF's capacity to systematically identify and mitigate trust-related issues that are not captured by performance metrics alone.

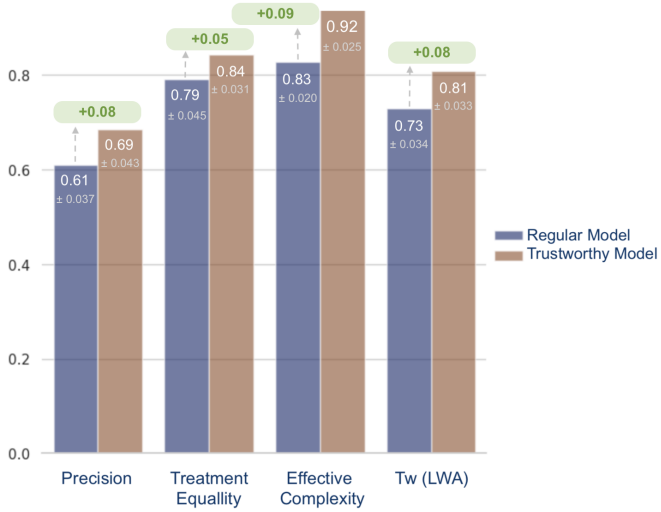


Figure 3. Comparison of trust-dimension metrics and aggregated trustworthiness for the German Credit dataset. Each pair of bars contrasts the initial diagnostic evaluation of the baseline SVM model (Regular Model) with the post-enhancement results obtained after applying TWEEF, computed on the corresponding surrogate-based Trustworthy Model. The arrows quantify the improvement achieved by the Trustworthy Model over the Regular Model for each metric.

4.2. Experiment 2: COMPAS Dataset

The second experiment used the COMPAS dataset, widely recognized for its historical concerns regarding racial bias in recidivism prediction systems. The protected attribute selected for this experiment was race, with the protected group defined as African Americans. The metrics considered were Recall (performance), Statistical Parity (fairness), and Effective Complexity (interpretability). Surrogate model coefficient of determination obtained was $R^2 = 0.85$.

As shown in Figure 4, TWEEF's analysis revealed that the SVM model achieved reasonable Recall but exhibited strong group-level disparities. Statistical Parity indicated systematic differences in predicted recidivism rates between African American individuals and other groups, consistent with findings reported in the literature. The interpretability metric again showed a relatively complex decision boundary, making the underlying logic difficult to capture with a rule-based or inherently interpretable model. TWEEF's fairness and interpretability diagnostics demonstrated that, even with decent performance, the SVM solution remains unreliable for decision-making contexts where racial fairness is required. This experiment validates TWEEF's capacity to surface fairness concerns aligned with widely publicized COMPAS critiques.

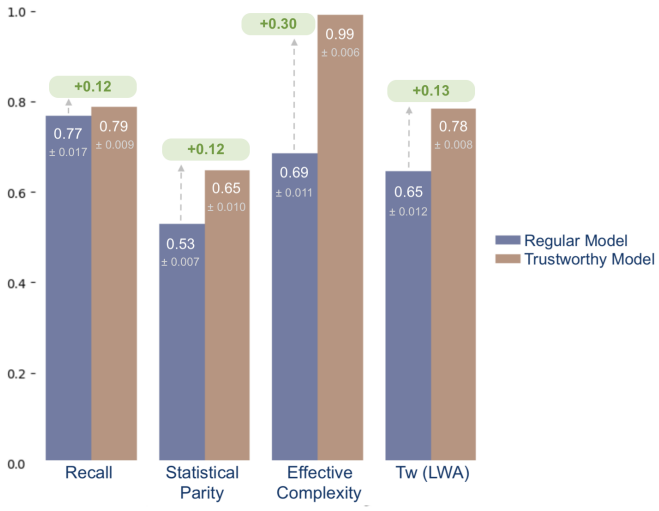


Figure 4. Trust-dimension metrics and overall trustworthiness comparison for the COMPAS dataset. For each metric, paired bars represent the baseline diagnostic assessment of the original SVM classifier (Regular Model) and the corresponding values obtained after the application of TWEEF, evaluated on the surrogate-based Trustworthy Model. The arrows quantify the improvement achieved by the Trustworthy Model over the Regular Model for each metric.

4.3. Experiment 3: Adult Income Dataset

The third experiment was carried out using the Adult dataset from the UCI repository, commonly used to study gender disparities in income prediction. The protected attribute defined for this study was sex, and the protected group was female. The chosen metrics were F1-Score (performance), Equal Opportunity (fairness), and Monotonicity (interpretability), and surrogate model coefficient of determination for this experiment reached $R^2 = 0.81$.

Results illustrated in Figure 5 showed that the SVM model performed well on F1-Score; however, the Equal Opportunity metric revealed meaningful differences in true positive rates between men and women. Female individuals were less likely to be correctly classified as high-income, indicating fairness issues consistent with the historical biases in the Adult dataset. The Monotonicity metric also indicated that the model's local feature influence patterns were difficult to reconcile with intuitive human reasoning, highlighting the need for a transparent surrogate. As in the previous experiments, TWEEF integrated these findings into an aggregated trustworthiness score, indicating that performance alone was insufficient to assess the model's reliability.

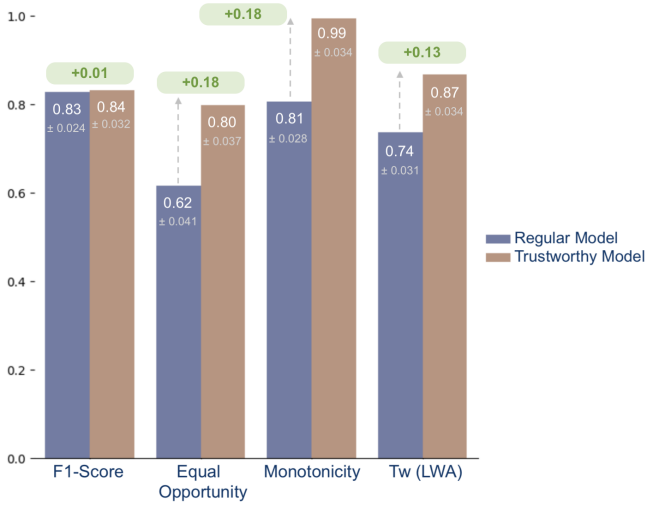


Figure 5. Evaluation of trust-related metrics and aggregated trustworthiness on the Adult dataset. Each bar pair contrasts the initial diagnostic results of the baseline SVM model (Regular Model) with the post-TWEEF outcomes computed on the enhanced surrogate Trustworthy Model. The arrows quantify the improvement achieved by the Trustworthy Model over the Regular Model for each metric.

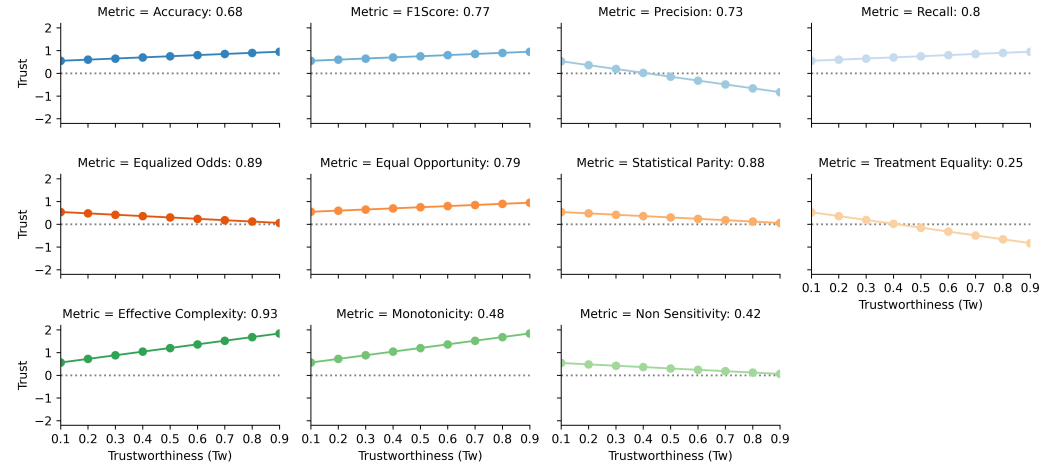
4.4. Experiment 4: Exploring Metric Weights

This experiment aims to demonstrate the flexibility of TWEEF beyond merely assigning weights to each of the performance, fairness, or interpretability metrics. Specifically, the framework allows the integration of a varying number of metrics, ranging from 0 to N, for each trustworthiness component. This flexibility means that the mechanism can accommodate a single metric per component, as in Experiment 1, multiple metrics, or even no metrics for a particular component if required.

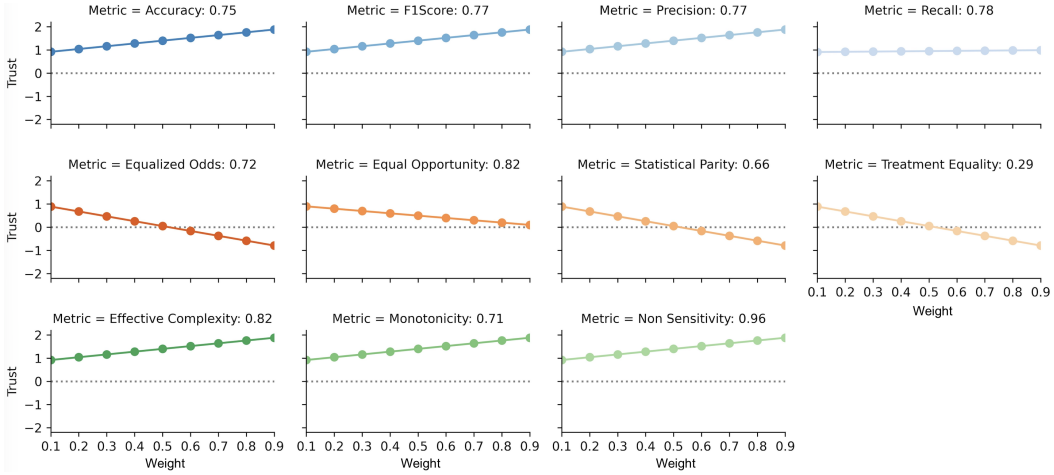
In this experiment, all performance, fairness, and interpretability metrics are used to compute the trustworthiness score Tw_{LWA} for the LWA operator. Then, the weights assigned to each of these metrics are varied, and their effects on the Tw_{LWA} value are observed for an SVM model. Figure 6 shows the variation of each metric for every dataset. For each metric, its assigned weight varies from 0.1 to 0.9, with the remaining weight distributed equally among the other metrics. In this way, Tw_{LWA} values vary according to the weight assigned to each metric can be observed.

The results indicate that performance indicators generally have a positive effect on the Tw_{LWA} value. In the German dataset, most performance indicators have a positive effect on Tw_{LWA}, except for Precision, which has a marked negative effect. The positive effect of performance indicators on the Tw_{LWA} trustworthiness value is consistent with evidence from the literature [12,54]. However, in the Adult dataset, the F1-Score and Recall metrics have a slightly negative effect on the Tw_{LWA} value. This behavior is explained by the relatively low performance values obtained by the SVM model on this dataset; as more weight is assigned to these metrics, the aggregated Tw_{LWA} score decreases accordingly. Therefore, as more weight is assigned to these low values, Tw_{LWA} is negatively affected.

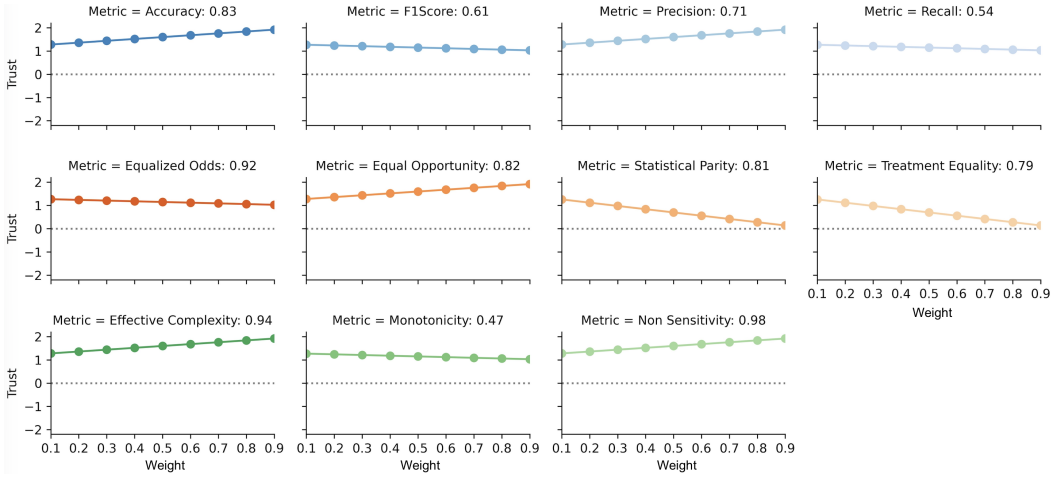
The fairness metrics generally negatively affect the Tw_{LWA} value. However, the relationship between the Equal Opportunity metric and the Tw_{LWA} value varies across the analyzed datasets. This is because this metric yields more extreme values than the other fairness metrics, leading to greater variation in the overall trustworthiness metric. Nevertheless, the Statistical Parity and Treatment Equality metrics show a very similar relationship across the analyzed datasets. This is because these metrics yielded similar values across datasets, so their contribution to the overall trustworthiness metric estimation is very similar across datasets.



(**a**) German dataset.



(**b**) COMPAS dataset.



(**c**) Adult dataset.

Figure 6. Trustworthiness score (Tw_{LWA}) variation for the SVM model on the (**a**) German dataset, (**b**) COMPAS dataset, and (**c**) Adult dataset. Each subplot illustrates the impact of each metric on Tw_{LWA} when varying the weight of the metric from 0.1 to 0.9, while the remaining weight is distributed equitably among the other metrics.

The interpretability metrics show different effects on the Tw_{LWA} value, but similar effects across the different datasets. In general, the Effective Complexity metric yielded very high values, so as its assigned weight increases, the Tw_{LWA} value also increases. Regarding the Monotonicity and Non-Sensitivity metrics, they exhibit less stable relationships with the Tw_{LWA} value across the three analyzed datasets, with broader ranges of values ([0.47, 0.71] and [0.42, 0.98], respectively). In the case of Monotonicity, it has a positive impact on the Tw_{LWA} value in both the German and COMPAS datasets, while in the Adult dataset, it has a very slight negative impact, which could even be considered neutral.

The Non-Sensitivity metric has a negative impact on the German dataset, whereas it has a positive impact on the Tw_{LWA} value in the COMPAS and Adult datasets. As observed in the three analyzed datasets, the nature of a metric's impact on the Tw_{LWA} value is generally related to the metric's value. The lower the metric's value, the more Tw_{LWA} decreases, as more weight is assigned to the metric. The same effect is observed at the upper extreme values of the metrics, where a clear positive impact Tw_{LWA} is generally observed.

In addition to observing how the weight assigned to a metric can affect the Tw_{LWA} value obtained by a Machine Learning model, this experiment highlights the flexibility of the Tw_{LWA} metric. It demonstrates that different combinations of metrics can be used as input variables depending on the user's (data scientist's) needs. Thus, the data scientist can model the internal structure of the Tw_{LWA} metric as deemed appropriate. Additionally, the data scientist can vary the weights associated with each variable, prioritizing the value of a particular metric or set of metrics over others. In this sense, it is possible to assign greater weight to fairness metrics than to performance or interpretability metrics when training certain types of models in an application domain where fairness plays a prominent role (e.g., the COMPAS criminal recidivism prediction system) over performance and interpretability.

Figure 7 offers an aggregated view of how metric weighting influences the overall trustworthiness score across datasets, making the role of weights explicit. In contrast to the sensitivity curves, which analyze metrics individually, the heatmap enables a comparative assessment of dominance, robustness, and interaction effects among performance, fairness, and interpretability dimensions. This visualization facilitates a more holistic understanding of how trustworthiness emerges from the weighted aggregation process. A key observation is that the trustworthiness score is considerably more sensitive to the weighting of fairness- and interpretability-related metrics than to performance metrics. Accuracy, Recall, and F1-Score contributions remain relatively stable across weight configurations, whereas fairness and interpretability metrics induce larger variations in the aggregated score. In particular, Equal Opportunity, Non-Sensitivity, and Monotonicity show consistent contributions, suggesting robust behavior and limited interaction effects with other trust dimensions.

In contrast, Treatment Equality, Effective Complexity, and Precision introduce non-linear effects and stronger interactions, as reflected by sharper gradients in the heatmap. These patterns expose latent trade-offs between performance, fairness, and interpretability, indicating that emphasizing certain metrics may partially offset gains in others. Effective Complexity, in particular, exhibits a high-impact but sensitive contribution, underscoring the delicate balance between model simplicity and expressive capacity. Also, Figure 7 reveals an intermediate weight region, approximately between 0.3 and 0.6, where trustworthiness scores remain more balanced across most metrics. This range can be interpreted as a stable operating region for the TWEEF framework, in which no single trust dimension disproportionately dominates the assessment. More broadly, these results highlight that metric weighting constitutes a normative design choice rather than a purely technical hyperparameter, as weight selection directly shapes the resulting trust profile of the model.

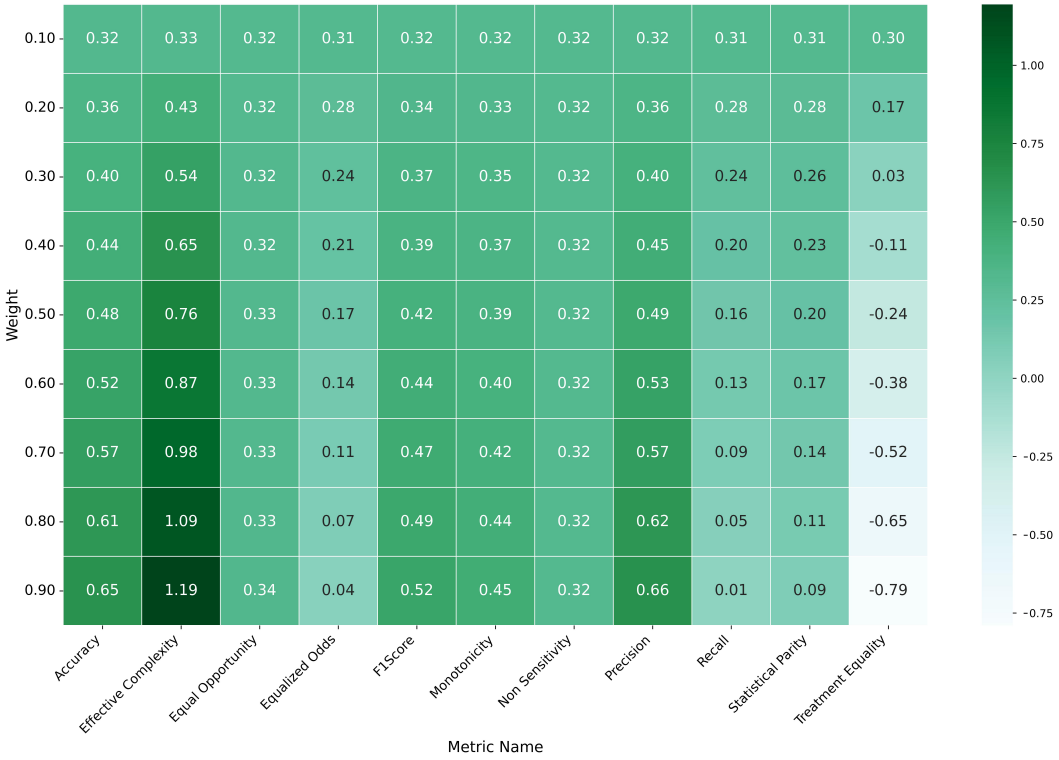


Figure 7. Aggregated Trustworthiness Score Heatmap by Metric Weight across Datasets. This heatmap visualizes the mean trustworthiness score as the weight of individual metrics (columns) is varied (rows) across all analyzed datasets. It highlights how different metrics and their assigned weights collectively influence the overall trustworthiness of trained models, revealing patterns of sensitivity and potential trade-offs between performance, fairness, and interpretability.

Additionally, weight assignment can serve as a configuration guide for applying TWEEF under different operational objectives, rather than as evidence of a single trustworthiness profile. By explicitly exposing how the aggregated trust score evolves as metric weights change, the framework enables practitioners to tailor trustworthiness evaluation to the goals and constraints of specific application contexts. When the objective is to maximize predictive reliability—such as in clinical decision-support or fault detection systems—configurations that privilege performance metrics (e.g., accuracy, F1-score, precision) combined with low effective complexity and monotonic behavior are appropriate, as they balance strong predictive power with interpretable and stable decision logic. In contrast, in regulatory-sensitive contexts such as credit approval, hiring, or social service allocation, TWEEF can be configured to emphasize fairness metrics (e.g., equalized odds, statistical parity, treatment equality), even when this leads to lower aggregated trust scores, thereby making explicit the cost of fairness enforcement in terms of predictive performance or interpretability. Finally, in settings where transparency and explainability are primary objectives—for example, public-sector analytics, policy modeling, or auditing of automated decisions—configurations that assign higher weights to interpretability metrics (effective complexity, monotonicity, non-sensitivity) produce more conservative but stable trustworthiness assessments, favoring models that are easier to inspect, justify, and communicate to non-technical stakeholders. Overall, the heatmap in Figure 7 demonstrates that TWEEF can be used as a decision-support tool for trust configuration, allowing users to explore and justify trade-offs between performance, fairness, and interpretability in alignment with domain-specific objectives rather than enforcing an integrated notion of trustworthiness.

5. Conclusions

This paper introduced TWEEF, a modular and extensible framework for the assessment and enhancement of Machine Learning model trustworthiness through the joint analysis of performance, fairness, and interpretability. TWEEF operationalizes trustworthiness by transforming quantitative metrics into linguistic representations using fuzzy membership functions and aggregating them through complementary operators, including Linguistic Weighted Average (LWA), Gaussian Weighted Aggregation (GWA), and Subjective Logic (SL).

The proposed framework was empirically validated using three widely adopted benchmark datasets (German Credit, COMPAS, and Adult) covering diverse application domains with documented fairness concerns. Experimental results demonstrate that TWEEF consistently identifies trust-related deficiencies that remain hidden when relying exclusively on predictive performance metrics, particularly in terms of group-level fairness disparities and interpretability limitations.

Across all experiments, TWEEF produced stable and interpretable trustworthiness estimations and enabled systematic comparison between initial diagnostic assessments and post-mitigation evaluations. The results confirm that bias mitigation and surrogate-based interpretability improvements can be explicitly reflected in the aggregated trustworthiness score, allowing users to reason about trade-offs between trust dimensions in a transparent and reproducible manner.

The analysis of metric weighting further demonstrated that trustworthiness aggregation is inherently sensitive to normative design choices. Variations in metric weights led to coherent and explainable changes in the aggregated trust score, highlighting that weight selection reflects contextual priorities rather than purely technical optimization. Together, these findings position TWEEF as a practical and theoretically grounded framework for trustworthiness assessment, supporting informed and accountable model evaluation in binary classification settings.

6. Limitations and Future Work

While TWEEF provides a coherent and extensible framework for trustworthiness assessment, several limitations must be acknowledged in its current implementation. First, the framework explicitly focuses on three trust dimensions (performance, fairness, and interpretability) excluding other dimensions such as privacy, security, accountability, or adversarial robustness to maintain conceptual clarity and methodological focus.

From a technical perspective, the current version of TWEEF is limited to binary classification tasks and to a predefined set of metrics for each trust dimension. Although the framework supports flexible metric selection and aggregation, the experimental validation presented in this work relies on a specific subset of performance, fairness, and interpretability indicators and on Support Vector Machine classifiers. As a result, conclusions should be interpreted within the context of these modeling and metric choices.

In addition, the aggregation mechanisms implemented in TWEEF assume independence between trust dimensions and metrics. While this assumption simplifies aggregation and improves interpretability, it does not capture potential dependencies or causal relationships between metrics, such as the interaction between fairness constraints and predictive performance.

Future work will address these limitations along several complementary directions. Planned extensions include support for multi-class classification and regression problems, the integration of additional trust dimensions such as privacy-preserving learning and accountability indicators, the incorporation of robustness metrics under adversarial or distribution-shift scenarios, and empirical validation with additional model families (e.g.,

tree ensembles or neural networks). Further research will also explore dependency-aware aggregation strategies and causal modeling approaches to better capture interactions between trust dimensions.

Finally, extending TWEEF with human-centered trust assessments and longitudinal monitoring capabilities represents a promising avenue for deployment in real-world, high-stakes environments and direct empirical comparisons with already established toolkits (e.g., CERTIFAI, Aequitas, AIF360, AIX360, ART). The modular architecture of the framework has been explicitly designed to accommodate these extensions, enabling TWEEF to evolve toward a more comprehensive and context-aware trust management ecosystem.

Author Contributions: Conceptualization, J.U. and R.S.; methodology, J.U., R.S. and R.T.; validation, R.S., A.F.B. and P.A.E.; investigation, J.U.; writing—original draft preparation, J.U.; writing—review and editing, J.U., R.S. and D.V.; visualization, J.U., D.V. and M.P.G.; supervision, R.S. and P.A.E.; funding acquisition, R.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by ANID Millennium Science Initiative Program ICN2021_004 and FONDECYT 1221938.

Data Availability Statement: No new data were created or analyzed in this study.

Acknowledgments: During the preparation of this manuscript, the authors used ChatGPT 5.1 for Spanish-to-English translation and for assistance with language editing. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wing, J.M. Trustworthy AI. *Commun. ACM* **2021**, *64*, 64–71. [CrossRef]
2. Mehrabi, N.; Morstatter, F.; Saxena, N.; Lerman, K.; Galstyan, A. A Survey on Bias and Fairness in Machine Learning. *ACM Comput. Surv.* **2021**, *54*, 115. [CrossRef]
3. Zhou, J.; Gandomi, A.H.; Chen, F.; Holzinger, A. Evaluating the quality of machine learning explanations: A survey on methods and metrics. *Electronics* **2021**, *10*, 593. [CrossRef]
4. Markus, A.F.; Kors, J.A.; Rijnbeek, P.R. The role of explainability in creating trustworthy artificial intelligence for health care: A comprehensive survey of the terminology, design choices, and evaluation strategies. *J. Biomed. Inform.* **2021**, *113*, 103655. [CrossRef] [PubMed]
5. Barocas, S.; Hardt, M.; Narayanan, A. *Fairness and Machine Learning: Limitations and Opportunities*; MIT Press: Cambridge, MA, USA, 2023.
6. Burkart, N.; Huber, M.F. A Survey on the Explainability of Supervised Machine Learning. *J. Artif. Intell. Res.* **2021**, *70*, 245–317. [CrossRef]
7. Sharma, S.; Henderson, J.; Ghosh, J. Certifai: A common framework to provide explanations and analyse the fairness and robustness of black-box models. In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, New York, NY, USA, 7–9 February 2020; pp. 166–172.
8. Bellamy, R.K.; Dey, K.; Hind, M.; Hoffman, S.C.; Houde, S.; Kannan, K.; Lohia, P.; Martino, J.; Mehta, S.; Mojsilović, A.; et al. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM J. Res. Dev.* **2019**, *63*, 1–4. [CrossRef]
9. Xu, Z. *Linguistic Decision Making*, 1st ed.; Springer: Berlin/Heidelberg, Germany, 2013. [CrossRef]
10. Thiebes, S.; Lins, S.; Sunyaev, A. Trustworthy Artificial Intelligence. *Electron. Mark.* **2021**, *31*, 447–464. [CrossRef]
11. Kowald, D.; Schedl, M.; Lex, E. Establishing and evaluating trustworthy AI: Overview and research challenges. *arXiv* **2024**, arXiv:2406.03012. [CrossRef]
12. Yin, M.; Vaughan, J.W.; Wallach, H. Understanding the Effect of Accuracy on Trust in Machine Learning Models. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, Glasgow, UK, 4–9 May 2019; ACM: New York, NY, USA, 2019. [CrossRef]
13. Chatzimparmpas, A.; Martins, R.M.; Jusufi, I.; Kucher, K.; Rossi, F.; Kerren, A. The State of the Art in Enhancing Trust in Machine Learning Models with the Use of Visualizations. *Comput. Graph. Forum* **2020**, *39*, 713–756. [CrossRef]
14. Kusner, M.J.; Loftus, J.; Russell, C.; Silva, R. Counterfactual fairness. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. [CrossRef]
15. Saleiro, P.; Kuester, B.; Hinkson, L.; London, J.; Stevens, A.; Anisfeld, A.; Rodolfa, K.T.; Ghani, R. Aequitas: A bias and fairness audit toolkit. *arXiv* **2018**, arXiv:1811.05577.

16. Arya, V.; Bellamy, R.K.E.; Chen, P.Y.; Dhurandhar, A.; Hind, M.; Hoffman, S.C.; Houde, S.; Liao, Q.V.; Luss, R.; Mojsilović, A.; et al. AI Explainability 360: An Extensible Toolkit for Understanding Data and Machine Learning Models. *J. Mach. Learn. Res.* **2020**, *21*, 1–6.
17. Nicolae, M.I.; Sinn, M.; Tran, M.N.; Buesser, B.; Rawat, A.; Wistuba, M.; Zantedeschi, V.; Baracaldo, N.; Chen, B.; Ludwig, H.; et al. Adversarial Robustness Toolbox v1.0.0. *arXiv* **2019**, arXiv:1807.01069. [CrossRef]
18. Verma, S.; Rubin, J. Fairness definitions explained. In Proceedings of the FairWare '18 International Workshop on Software Fairness, Gothenburg, Sweden, 29 May 2018; pp. 1–7. [CrossRef]
19. Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; Zemel, R. Fairness through awareness. In Proceedings of the ITCS '12 3rd Innovations in Theoretical Computer Science Conference, Cambridge, MA, USA, 8–10 January 2012; pp. 214–226. [CrossRef]
20. Grgic-Hlaca, N.; Zafar, M.B.; Gummadi, K.P.; Weller, A. The Case for Process Fairness in Learning: Feature Selection for Fair Decision making. In Proceedings of the NIPS Symposium on Machine Learning and the Law, Barcelona, Spain, 9 December 2016; Volume 1, p. 11.
21. Berk, R.; Heidari, H.; Jabbari, S.; Kearns, M.; Roth, A. Fairness in Criminal Justice Risk Assessments: The State of the Art. *Sociol. Methods Res.* **2021**, *50*, 3–44. [CrossRef]
22. Feldman, M.; Friedler, S.A.; Moeller, J.; Scheidegger, C.; Venkatasubramanian, S. Certifying and Removing Disparate Impact. In Proceedings of the KDD '15 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, Australia, 10–13 August 2015; pp. 259–268. [CrossRef]
23. Kamiran, F.; Calders, T. Classifying without discriminating. In Proceedings of the 2009 2nd International Conference on Computer, Control and Communication, Karachi, Pakistan, 17–18 February 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 1–6.
24. Kamiran, F.; Calders, T. Classification with no discrimination by preferential sampling. In Proceedings of the Machine Learning confer-ence of Belgium and The Netherlands, Leuven, Belgium, 27–28 May 2010; Citeseer: Princeton, NJ, USA, 2010; pp. 1–6.
25. Kamiran, F.; Calders, T. Data preprocessing techniques for classification without discrimination. *Knowl. Inf. Syst.* **2012**, *33*, 1–33. [CrossRef]
26. Zemel, R.; Wu, Y.; Swersky, K.; Pitassi, T.; Dwork, C. Learning fair representations. In Proceedings of the International Conference on Machine Learning, PMLR, Atlanta, GA, USA, 17–19 June 2013; pp. 325–333.
27. Kamishima, T.; Akaho, S.; Asoh, H.; Sakuma, J. Fairness-aware classifier with prejudice remover regularizer. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Bristol, UK, 24–28 September 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 35–50.
28. Hardt, M.; Price, E.; Srebro, N. Equality of Opportunity in Supervised Learning. In *Proceedings of the Advances in Neural Information Processing Systems*; Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2016; Volume 29.
29. Pleiss, G.; Raghavan, M.; Wu, F.; Kleinberg, J.; Weinberger, K.Q. On fairness and calibration. *arXiv* **2017**, arXiv:1709.02012. [CrossRef]
30. Goldstein, A.; Kapelner, A.; Bleich, J.; Pitkin, E. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *J. Comput. Graph. Stat.* **2015**, *24*, 44–65. [CrossRef]
31. Van Looveren, A.; Klaise, J. Interpretable counterfactual explanations guided by prototypes. *arXiv* **2019**, arXiv:1907.02584.
32. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
33. Freitas, A.A. Comprehensible classification models: A position paper. *ACM SIGKDD Explor. Newsl.* **2014**, *15*, 1–10. [CrossRef]
34. Ribeiro, M.T.; Singh, S.; Guestrin, C. "Why should i trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1135–1144.
35. Huysmans, J.; Dejaeger, K.; Mues, C.; Vanthienen, J.; Baesens, B. An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models. *Decis. Support Syst.* **2011**, *51*, 141–154. [CrossRef]
36. Zhou, J.; Li, Z.; Hu, H.; Yu, K.; Chen, F.; Li, Z.; Wang, Y. Effects of influence on user trust in predictive decision making. In Proceedings of the Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems, Glasgow, UK, 4–9 May 2019; pp. 1–6.
37. Zhou, J.; Arshad, S.Z.; Yu, K.; Chen, F. Correlation for user confidence in predictive decision making. In Proceedings of the 28th Australian Conference on Computer-Human Interaction, Launceston, Australia, 29 November–2 December 2016; pp. 252–256.
38. Molnar, C.; Casalicchio, G.; Bischl, B. Interpretable machine learning–a brief history, state-of-the-art and challenges. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Ghent, Belgium, 14–18 September 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 417–431.
39. Slack, D.; Friedler, S.A.; Scheidegger, C.; Roy, C.D. Assessing the local interpretability of machine learning models. *arXiv* **2019**, arXiv:1902.03501. [CrossRef]

40. Lakkaraju, H.; Kamar, E.; Caruana, R.; Leskovec, J. Interpretable & explorable approximations of black box models. *arXiv* **2017**, arXiv:1707.01154. [CrossRef]
41. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic attribution for deep networks. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; pp. 3319–3328.
42. AL-Sukeinee, R.J.; Khudeyer, R.S. Review: Deep Learning and Fuzzy Logic Applications. *Eng. Technol. J.* **2024**, *9*, 4231–4240. [CrossRef]
43. SS Júnior, J.; Mendes, J.; Souza, F.; Premebida, C. Survey on deep fuzzy systems in regression applications: A view on interpretability. *Int. J. Fuzzy Syst.* **2023**, *25*, 2568–2589. [CrossRef]
44. Fernandez-Peralta, R. A Comprehensive Survey of Fuzzy Implication Functions. *arXiv* **2025**, arXiv:2503.05702. [CrossRef]
45. Ouifak, H.; Idri, A. A comprehensive review of fuzzy logic based interpretability and explainability of machine learning techniques across domains. *Neurocomputing* **2025**, *647*, 130602. [CrossRef]
46. Cheng, D.; Zhou, X.; Xu, Y. Fuzzy Fusion of Wireless Sensor Network Data for Environmental Monitoring. *Expert Syst. Appl.* **2012**, *39*, 11756–11765.
47. Das, A.K.; Chakraborty, B.; Goswami, S.; Chakrabarti, A. A fuzzy set based approach for effective feature selection. *Fuzzy Sets Syst.* **2022**, *449*, 187–206. [CrossRef]
48. Cohen, S.; Ruppin, E.; Dror, G. Feature selection based on the Shapley value. In Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI-05), Edinburgh, UK, 30 July–5 August 2005; pp. 665–670.
49. Nguyen, A.; Martínez, M.R. On quantitative aspects of model interpretability. *arXiv* **2020**, arXiv:2007.07584. [CrossRef]
50. Nori, H.; Jenkins, S.; Koch, P.; Caruana, R. InterpretML: A Unified Framework for Machine Learning Interpretability. *arXiv* **2019**, arXiv:1909.09223. [CrossRef]
51. Ünver, M. Gaussian Aggregation Operators and Applications to Intuitionistic Fuzzy Classification. *J. Classif.* **2025**, *42*, 596–623. [CrossRef]
52. Jøsang, A. *Subjective Logic: A Formalism for Reasoning Under Uncertainty*; Springer International Publishing: Cham, Switzerland, 2016. [CrossRef]
53. Esposito, C.; Galli, A.; Moscato, V.; Sperlí, G. Multi-criteria assessment of user trust in Social Reviewing Systems with subjective logic fusion. *Inf. Fusion* **2022**, *77*, 1–18. [CrossRef]
54. Papenmeier, A.; Englebienne, G.; Seifert, C. How model accuracy and explanation fidelity influence user trust. *arXiv* **2019**, arXiv:1907.12652. [CrossRef]