
Causal-Proof Guided Learning and Certification for Fairness in ML Classifiers

Journal Title

XX(X):1–60

©The Author(s) 2016

Reprints and permission:

sagepub.co.uk/journalsPermissions.nav

DOI: 10.1177/ToBeAssigned

www.sagepub.com/

SAGE

Leonardo Ceragioli¹ and Giuseppe Primiero¹

Abstract

This article presents a proof-guided neuro-symbolic framework for certifying, diagnosing, and repairing fairness in machine-learning systems. Causal structures learned from data support formal fairness derivations. Successful proofs yield machine-readable constraints for preserving certified properties, while diagnostic reports identify structural or statistical obstacles and guide causal model revisions or the construction of differentiable retraining losses. An illustrative evaluation on the German Credit dataset shows how proof-generated diagnoses guide targeted interventions and the assessment of their effectiveness. The framework connects causal discovery, formal verification, constraint synthesis, and model adaptation, extending fairness assessment beyond metric-based evaluation.

Keywords

NeSy AI; algorithmic fairness; causal reasoning; formal verification; proof-guided diagnosis; constraint synthesis; model repair; trustworthy AI.

Introduction

Machine Learning (ML) systems are increasingly deployed in socially sensitive domains including credit scoring, hiring, healthcare, insurance, education, and

¹Logic, Uncertainty, Computation and Information Lab, Department of Philosophy, University of Milan, Italy.

Corresponding author:

Giuseppe Primiero

Email: giuseppe.primiero@unimi.it

criminal justice. In such applications, automated decisions may have significant consequences for individuals and groups, making fairness not only an ethical objective but also an important legal and societal requirement.

A major challenge is that many modern ML models operate as highly complex statistical systems whose internal decision processes are difficult to interpret. An appropriate strategy is to evaluate fairness through post-hoc statistical analyses comparing the behavior of a classifier across individuals or demographic groups. Widely adopted notions include demographic parity, equality of opportunity, individual fairness, and counterfactual fairness, each capturing different aspects of algorithmic discrimination (1; 2; 3; 4). Although these approaches provide valuable quantitative indicators, they generally offer limited insight into the causal and inferential mechanisms responsible for unfair decisions.

Recent developments in causal inference have shown that causal models provide a principled framework for distinguishing genuine discriminatory effects from spurious statistical correlations (5). At the same time, the growing interest in neuro-symbolic artificial intelligence has highlighted the importance of integrating statistical learning with symbolic reasoning, allowing learned models and formal knowledge representations to interact within a unified framework (6). However, existing fairness pipelines largely remain sequential: predictive models are trained from data, fairness is evaluated afterwards, and the outcome of the verification process rarely influences subsequent learning.

In this paper, we consider fairness verification as an active component of the learning process. We introduce a neuro-symbolic framework, called *Causal Proof-Guided Learning and Certification* (CPGLC), in which causal structures are learned directly from observational data, translated into a symbolic representation suitable for formal reasoning, analysed through proof-theoretic verification, and subsequently transformed into optimization constraints that guide the retraining of the predictive model. In the present work, proof derivations become explanatory objects from which fairness constraints are systematically synthesized and incorporated into the optimization process, becoming an integral component of model adaptation rather than a purely post-hoc verification tool. Within this perspective, failed proof obligations identify the causal structures responsible for unfair behavior and generate the optimization objectives required to progressively improve the predictive model.

The remainder of the paper is organized as follows. We first review the most relevant literature at the intersection of fairness verification, causal fairness, neuro-symbolic learning, and proof-theoretic approaches. Next, we introduce the proposed Causal Proof-Guided Learning and Certification framework and present its five phases, from causal discovery to proof-guided retraining. We offer an exemplificative validation of the process. The final Section discusses the implications of the proposed methodology and outline directions for future research.

Related Work

The proposed framework combines ideas originating from three distinct research directions: causal reasoning for trustworthy machine learning, proof-theoretic verification of fairness and neuro-symbolic learning. Each of these areas has independently contributed important techniques for understanding, verifying, or mitigating algorithmic bias, but their integration into a unified learning-and-certification framework remains so far unexplored.

Causal Reasoning for Fair and Trustworthy Machine Learning

Structural causal models support trustworthy machine learning by representing data-generating mechanisms, distinguishing causal influences from statistical associations, and enabling interventions, counterfactual reasoning, and explanations (5; 7; 8). In algorithmic fairness, this perspective shifts attention from observational criteria to the mechanisms underlying discrimination. Kilbertus et al. (9) showed the limitations of purely statistical fairness criteria; Kusner et al. (3) introduced counterfactual fairness, requiring prediction invariance under interventions on protected attributes; and Nabi and Shpitser (10) proposed path-specific constraints distinguishing fair from unfair causal effects. Causal reasoning thus provides a framework for analysing and comparing fairness criteria (11).

Our approach develops this perspective in two respects: the causal representation is learned from observational data rather than assumed a priori, and it supplies the semantics for proof-theoretic verification. The resulting proof objects are compiled into symbolic constraints guiding fairness-aware retraining, integrating causal discovery, formal verification, and model adaptation within a neuro-symbolic learning loop.

Proof-Theoretic Verification of Fairness

Proof-theoretic methods seek mathematical guarantees for trustworthy machine learning by treating fairness as a property of the reasoning producing model outputs. They certify fairness through admissible inference rules and formal derivations.

Our framework builds on Typed Natural Deduction for Probabilistic Queries (TNDPQ), introduced for reasoning about probabilistic queries over machine-learning systems (12). Its foundations include proof-theoretic verification of trustworthy probabilistic computations (13), complete with respect to a possible-worlds semantics (14). Subsequent extensions introduced causal labels and structural reasoning to characterize individual, intersectional, and counterfactual fairness through derivation admissibility (15; 16). A complementary non-causal approach defines a correction measure for unfair outputs (17), with a sound game-theoretic semantics (18). These developments support proof-theoretic certification and correction of fairness. However, the causal framework assumed given causal structures and produced certificates after training, without feeding verification results back into learning.

The present work turns proof objects into learning signals by systematically synthesizing optimization constraints from their explanatory content. This connects formal verification with statistical learning, enabling proof-theoretic reasoning to guide model adaptation within a unified neuro-symbolic framework.

Neuro-Symbolic Fairness Learning

Neuro-symbolic artificial intelligence integrates statistical learning with symbolic reasoning, using formal knowledge to guide learning and learned representations to support inference and explanation (6; 19; 20; 21).

Neuro-symbolic approaches to fairness typically rely on explicitly supplied logical specifications. Logic Tensor Networks (LTNs) interpret first-order formulas through a continuous semantics optimized jointly with predictive objectives (22). Greco et al. (23) declaratively encoded fairness properties in LTNs independently of the learning algorithm, while Heilmann et al. (24) incorporated counterfactual fairness through joint optimization of predictive performance and logical constraints. Related approaches translate declarative requirements into differentiable objectives (25; 26), distinguish optimization objectives from constraints that must remain satisfied during training (27), and repair models in response to violated specifications while limiting changes to previously acquired behaviour (28; 29).

Our framework automatically derives fairness constraints from formal proofs over learned causal structures. Constraints compiled from admissible proofs define invariants to preserve during retraining, while obligations generated from failed proofs identify behaviour requiring repair. Preservation restricts admissible model updates, and repair directs optimization within that region or structural modifications to the causal model. Because satisfaction during optimization alone provides no formal preservation guarantee, the retrained model must undergo certification again.

The Causal-Proof-Guided Learning and Certification Protocol

The *Causal-Proof-Guided Learning and Certification* (CPGLC) protocol establishes a closed neuro-symbolic loop connecting causal discovery, proof-theoretic fairness verification, symbolic constraint generation, and model adaptation (Figure 1). The protocol consists of five phases:

1. Learning a causal graph from data;
2. Translating the learned graph into TNDPQ causal labels;
3. Executing proof-theoretic fairness verification;
4. Generating symbolic fairness constraints from proof violations;
5. Retraining the model under such constraints.

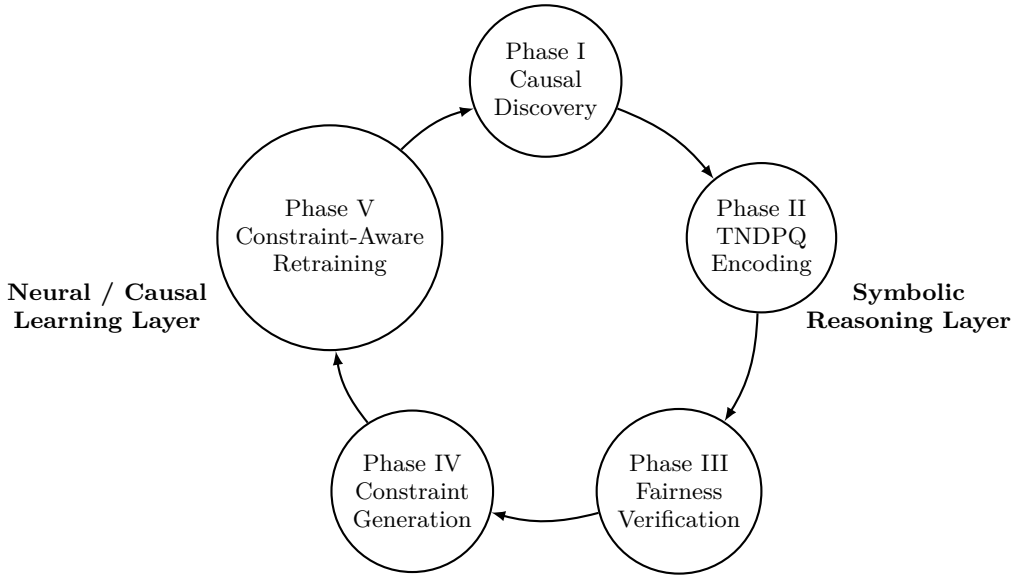


Figure 1. The Causal-Proof-Guided Learning and Certification (CPGLC) cycle. Causal structures are learned from data (Phase I), translated into TNDPQ causal labels (Phase II), analysed through proof-theoretic fairness verification (Phase III), transformed into symbolic fairness constraints (Phase IV), and incorporated into constraint-aware retraining (Phase V). The retrained model produces a revised causal structure, thereby closing the neuro-symbolic feedback loop.

Phase I: Learning and Refining Causal Structures from Data

The first phase constructs a weighted, confidence-annotated causal representation directly from observational data. This structural model connects statistical learning with symbolic reasoning, supporting formal fairness verification beyond analysis of the predictive model alone. All data, source code, and outputs of this phase are available at <https://github.com/gprimiero/CPGLC> (Release v.1.0).

Dataset

The evaluation uses the UCI *Statlog German Credit Data* dataset, a benchmark for explainability and fairness in credit-risk assessment (30). We adopt Strathclyde University’s numerical version of the original 20-attribute dataset, with several ordered categorical attributes encoded as integers. After preprocessing, it contains twenty-five variables covering financial history, employment, personal characteristics, housing, and the credit-risk target. Representative variables include

- *status*: status of the checking account;
- *duration*: duration of the requested loan;

- *credit_history*: previous repayment history;
- *credit_amount*: requested credit amount;
- *savings/bonds*: available savings;
- *employment_duration*: current employment duration;
- *property*: property ownership;
- *housing*: housing conditions;
- *job*: employment category;
- *age*;
- *gender*;
- *marital_status*;
- *predicted_label*: prediction produced by the trained classifier.

Bootstrap-based Augmentation

Constraint-based causal discovery algorithms generally require a sufficiently large number of observations to reliably estimate conditional independencies. Let

$$D_{\text{full}} = \{x_1, \dots, x_N\}, N = 1000,$$

denote the complete German Credit dataset used as the source of the experiment. For experimental control, and to reduce the sparsity of the initial dataset, we first perform a sample-size ablation and select a fixed base sample

$$D_{\text{base}} \subset D_{\text{full}}, |D_{\text{base}}| = n = 300.$$

The ablation was performed by random sampling without replacement, stratified with respect to the `credit_risk` target and using a fixed random seed of 42. Stratification preserves the original 70/30 class distribution: the resulting sample contains 210 observations from the majority class and 90 from the minority class. This fixed subset constitutes the effective observational basis of the subsequent causal-discovery experiment. Given D_{base} , we construct a bootstrap-augmented working dataset

$$D_{\text{aug}} = \{x_1^*, \dots, x_m^*\}, m = 1000,$$

by sampling complete observations independently and uniformly from D_{base} with replacement. Since complete observations are resampled, D_{aug} is drawn from the empirical joint distribution induced by D_{base} . Expected joint and marginal cell counts therefore reflect the corresponding empirical probabilities of the base sample. Realized marginal and conditional frequencies may nevertheless differ because of resampling variability. Thus, $D_{\text{full}} \rightarrow D_{\text{base}} \rightarrow D_{\text{aug}}$ separates sample selection from bootstrap augmentation. Resampling enlarges the dataset without reintroducing excluded observations or providing the information of 1000 independent observations. The following conditional-independence results must be interpreted relative to the empirical distribution of the fixed sample D_{base} .

Causal Structure Discovery

Causal-discovery frameworks include *CausalExplain* (31; 32), which combines predictive explainability with causal inference, and *causal-learn* (33), which implements constraint-based, score-based, and functional discovery algorithms.

Our implementation uses the PC (Peter–Clark) algorithm from *causal-learn* on $m = 1000$ bootstrap-resampled rows, with Fisher’s Z conditional-independence test (`indeptest=fisherz`) and significance level $\alpha = 0.05$. The test uses this nominal sample size without calibration to $n = 300$. Since resampling adds no independent evidence, this pseudo-replication can affect test statistics, p -values, and the decisions through which PC removes edges. We therefore interpret the resulting topology as the output of a heuristic resampling experiment grounded in 300 independent observations.

PC starts from a fully connected graph, removes edges through conditional-independence tests, and applies orientation rules to infer identifiable causal directions while preserving acyclicity. The resulting graph contains twenty-five variables, fifty-five directed relations, and two undirected dependencies, and provides the qualitative causal topology used throughout the framework.

Definition 0.1. Causal Partial DAG. Given the dataset D_{aug} , the causal discovery procedure produces a partially directed acyclic graph

$$G_{PC} = (V, E_{ref}, U_{ref}),$$

where

- $V = \{X_1, \dots, X_n\}$ denotes the set of observed variables,
- $E_{ref} \subseteq V \times V$ denotes the set of directed causal relations,
- $U_{ref} \subseteq V \times V$ denotes the set of undirected relations.

For every node $X_j \in V$, the algorithm determines a corresponding parent set $\text{Pa}(X_j) = \{X_{i_1}, \dots, X_{i_k}\}$, which represents the variables identified as direct causal predecessors of X_j . The PC procedure additionally identifies two unoriented dependencies, *gender* – *marital_status* and *housing* – *marital_status*. These relations are retained as part of the partially directed output of causal discovery but are not included in the subsequent parameter-learning and pruning stages, since their causal direction cannot be established from the observational data.

At this stage the graph captures exclusively the qualitative causal topology. Quantitative information concerning the relative strength and reliability of the discovered dependencies is introduced in the following stages through parameter estimation and bootstrap stability analysis. This analysis assesses the reproducibility of the resulting directed dependencies under resampling, but does not recalibrate or otherwise correct the conditional-independence tests used to construct the reference topology.

Parameter Learning and Confidence Estimation

To enrich the learned structure with quantitative information, we perform two complementary analyses:

1. estimation of *structural coefficients* to quantify the local causal influence associated with every directed edge;
2. estimation of *bootstrap confidence* to measure the stability of each discovered dependency under repeated resampling.

Definition 0.2. Causal DAG. Given G_{PC} , the weighted directed acyclic graph

$$G_{ref} = (V, E_{ref}, \beta, c),$$

includes

- V and E_{ref} are as in G_{PC} of Definition 0.1;
- $\beta : E \rightarrow \mathbb{R}$ associates each retained edge with a structural coefficient,
- $c : E \rightarrow [0, 1]$ measures the confidence assigned to that edge.

Definition 0.3. Edge Structural Coefficient. Given G_{ref} , let $\text{Pa}(X_j) = \{X_{i_1}, \dots, X_{i_k}\}$ be the parent set of a node $X_j \in V$ possessing at least one parent. Based on the bootstrap-augmented dataset D_{aug} , containing m observations of these variables, let $x_j^{(r)}$ denote the observed value of X_j in the r -th observation, and let $x_i^{(r)}$ denote the corresponding observed value of each parent $X_i \in \text{Pa}(X_j)$. The local linear model associated with X_j is

$$x_j^{(r)} = \beta_{0j} + \sum_{X_i \in \text{Pa}(X_j)} \beta_{ij} x_i^{(r)} + \epsilon_j^{(r)}, \quad r = 1, \dots, m,$$

where

- β_{0j} is the intercept associated with the local model for X_j ;
- β_{ij} is the coefficient associated with the directed edge (X_i, X_j) ;
- $\epsilon_j^{(r)}$ is the residual error associated with observation r .

Let now

- $\mathbf{y}_j = \begin{pmatrix} x_j^{(1)} \\ \vdots \\ x_j^{(m)} \end{pmatrix}$ be the response vector collecting the m observed values of the child variable X_j ;
- $\mathbf{Z}_j = \begin{pmatrix} x_{i_1}^{(1)} & \cdots & x_{i_k}^{(1)} \\ \vdots & \ddots & \vdots \\ x_{i_1}^{(m)} & \cdots & x_{i_k}^{(m)} \end{pmatrix}$ be the design matrix whose columns collect the observed values of the k parent variables of X_j ; and

Table 1. Estimated structural coefficients associated with the directed edges of the learned causal graph.

Parent	Child	β
credit_history	age	0.416
residence_duration	age	0.308
credit_risk	age	-1.180
age_group	age	10.658
employment_duration	age_group	0.187
people_liable	age_group	0.584
status	credit_risk	-0.113
duration	credit_risk	0.009
credit_history	credit_risk	-0.079
installment_rate	credit_risk	0.040
$\vdots$	$\vdots$	$\vdots$

- $\beta_j = (\beta_{i_{1j}} \ \cdots \ \beta_{i_{kj}})^T$ be the vector of structural coefficients associated with the directed edges from those parents to X_j .

The intercept β_{0j} is estimated separately and is therefore not included in Z_j or in β_j . In matrix form, the local linear model is

$$\mathbf{y}_j = \beta_{0j}\mathbf{1}_m + Z_j\beta_j + \epsilon_j,$$

where $\mathbf{1}_m$ is the m -dimensional vector of ones and ϵ_j is the corresponding residual vector. The Ordinary Least Squares estimates are defined by*

$$(\hat{\beta}_{0j}, \hat{\beta}_j) \in \arg \min_{\beta_{0j} \in \mathbb{R}, \beta_j \in \mathbb{R}^k} \|\mathbf{y}_j - \beta_{0j}\mathbf{1}_m - Z_j\beta_j\|_2^2.$$

For every edge $e = (X_i, X_j) \in E_{\text{ref}}$, its *edge structural coefficient* is the corresponding estimated regression coefficient:

$$\beta(e) := \hat{\beta}_{ij}.$$

Parameter learning preserves the discovered topology and assigns each directed edge $e : (X_i \rightarrow X_j)$ a structural coefficient $\beta(e)$. Its sign indicates positive or negative influence, while its absolute value measures effect magnitude. Table 1 reports selected estimates; the complete table contains fifty-five coefficients, one for each directed relation identified by PC.

Coefficient magnitude does not measure how consistently an edge is recovered. We therefore assess edge stability through a second bootstrap procedure, distinct

*This formulation corresponds to the implementation using `sklearn.linear_model.LinearRegression` with `fit_intercept=True`, where the intercept is estimated internally without explicitly adding a column of ones to Z_j .

from the augmentation used to construct D_{aug} and learn the reference graph. We independently generate $B = 20$ replicates $D_{\text{aug}}^{*(1)}, \dots, D_{\text{aug}}^{*(B)}$, each containing $m = 1000$ observations sampled with replacement from D_{aug} . Running PC separately on each replicate yields $G_b = (V, E_b)$, used to assess the recovery of edges in the fixed reference graph.

Definition 0.4. Edge confidence. Given $e \in E_{\text{ref}}$ one of the directed edges discovered during the original causal discovery phase, its confidence is defined as

$$c(e) = \frac{1}{B} \sum_{b=1}^B \mathbf{1}(e \in E_b),$$

where $\mathbf{1}(\cdot)$ denotes the indicator function. Consequently, $c(e) = 1$ indicates that the edge is recovered in every bootstrap iteration, whereas $c(e) = 0$ indicates that it never reappears after resampling.

Because the replicates resample D_{aug} , itself derived from the 300-observation base sample, they add no observational information. The scores measure edge stability conditional on the realized augmented dataset, not the probability of causal correctness in the population. With $B = 20$, scores have a resolution of 0.05 and serve as exploratory robustness indicators rather than precise confidence estimates. Table 2 reports representative values.

The graph G_{ref} , learned from D_{aug} , remains the fixed reference topology. For stability analysis, each iteration independently resamples D_{aug} with replacement and reruns PC. Each reference edge receives a confidence score equal to the proportion of runs recovering it with the same orientation. This score measures empirical stability rather than structural effect magnitude.[†]

Graph Pruning by Structural and Predictive Relevance

Not every discovered dependency contributes meaningfully to the subsequent neuro-symbolic analysis. Several relations exhibit negligible structural coefficients, indicating that their contribution to the local structural equations is practically insignificant despite their presence in the learned topology. The objective of the pruning phase is therefore to simplify the learned graph while preserving those causal dependencies that are either structurally relevant or demonstrably important for the predictive model.

The first pruning criterion is derived directly from the estimated structural coefficients.

[†]An alternative would combine all edges discovered across bootstrap runs and annotate each with its empirical frequency, including additional candidate relations. We restrict confidence estimation to reference edges to preserve a single topology.

Table 2. Representative bootstrap confidence values associated with directed causal relations.

Parent	Child	Confidence
credit_amount	installment_rate	1.00
duration	purpose	1.00
employment_duration	predicted_label	1.00
employment_duration	residence_duration	1.00
property	predicted_label	1.00
property	housing	1.00
credit_history	number_of_credits	0.95
savings/bonds	predicted_label	0.95
credit_history	predicted_label	0.90
installment_rate	job	0.90
property	job	0.90
⋮	⋮	⋮

Definition 0.5. Structural Pruning. Let $e = (X_i, X_j) \in E_{ref}$. e is initially classified as structurally negligible whenever $|\beta(e)| < \tau$, where $\tau = 0.05$ is the coefficient threshold adopted throughout the experiments.

This criterion identifies dependencies whose contribution to the corresponding structural equation is sufficiently small to be considered negligible.

However, coefficient magnitude alone is not sufficient for deciding whether an edge should be removed. A variable may exhibit a limited local causal influence while remaining highly relevant for the predictive behavior of the machine-learning model. Consequently, pruning solely on the basis of regression coefficients may inadvertently remove variables that play an important role in the decision process. To preserve variables that are operationally important for the classifier, the structural pruning procedure is complemented with an opacity analysis provided by the MIRAI Toolbox.[‡] The Opacity Module of the Toolbox analyses the trained prediction model through Local Interpretable Model-Agnostic Explanations (LIME), producing a ranking of feature relevance with respect to the generated predictions. The analysis identifies the following variables as the most relevant contributors to the prediction process:

$$V_{\text{BRIO}} = \{duration, employment_duration, savings/bonds, credit_history, status, age\}.$$

These variables correspond to the principal determinants of the classifier’s decision and therefore constitute explanatory evidence that complements the causal information extracted from observational data. Table 3 lists the results.

[‡]<https://brio-stg.mirai.systems/>

Table 3. Mean LIME feature importance values computed by the MIRAI Toolbox opacity analysis. Positive values indicate an increase in the probability of the predicted outcome, whereas negative values indicate an opposite contribution. The ranking is based on the absolute magnitude of the LIME scores.

Feature	Mean LIME Importance
duration	7.22×10^{-4}
age	-3.66×10^{-4}
status	-1.37×10^{-4}
credit_history	-5.64×10^{-5}
savings/bonds	-4.29×10^{-5}
employment_duration	-2.89×10^{-5}

The final pruning strategy combines structural relevance and predictive relevance.

Definition 0.6. For every directed edge $e = (X_i, X_j) \in E_{ref}$, e is retained whenever at least one of the following conditions holds:

1. its structural coefficient exceeds the pruning threshold, i.e $|\beta(e)| \geq \tau$;
2. its parent variable belongs to the BRIO relevance set, i.e $X_i \in V_{BRIO}$.

$$e \in E' \iff (|\beta(e)| \geq \tau) \vee (X_i \in V_{BRIO}),$$

where $E' \subseteq E_{ref}$ denotes the pruned edge set.

This hybrid criterion ensures that variables identified as highly influential by the predictive model are preserved even when their local structural coefficient is relatively small. Conversely, variables exhibiting both negligible structural influence and limited predictive relevance are removed from the graph. Applying the hybrid pruning criterion removes eleven dependencies, yielding a final graph containing forty-four directed edges. Table 4 summarizes the discarded relations.

Many pruned edges have near-zero structural coefficients, sometimes despite high bootstrap confidence. For example, *credit_amount* $\rightarrow$ *installment_rate* is recovered in every bootstrap run ($c = 1$) but has negligible estimated influence. Conversely, *credit_history* $\rightarrow$ *predicted_label* combines high confidence with substantial influence and is retained. Bootstrap confidence thus measures reproducibility under resampling, while structural coefficients quantify influence within local equations. Pruning prioritizes structural influence, while allowing predictive relevance to override coefficient-based decisions when supported by the opacity analysis.

Definition 0.7. Pruned Causal DAG. The output of the pruning phase is therefore a refined causal graph

$$G' = (V, E', \beta, c),$$

Table 4. Edges removed during graph pruning.

Parent	Child	β	c
credit_amount	duration	0.002	0.40
credit_amount	installment_rate	-0.000	1.00
credit_amount	property	0.000	0.50
credit_amount	job	0.000	0.20
credit_amount	telephone	0.000	0.90
credit_amount	foreign_worker	0.000	0.25
property	foreign_worker	-0.033	0.90
people_liable	foreign_worker	0.049	0.60
installment_rate	credit_risk	0.040	0.60
purpose	installment_plans	-0.040	0.80
purpose	predicted_label	0.013	0.60

Table 5. Retained causal relations associated with decision and financial variables after graph pruning.

Parent	Child	β	c
status	credit_risk	-0.113	0.85
duration	credit_risk	0.009	0.90
credit_history	credit_risk	-0.079	0.85
status	predicted_label	-0.140	0.75
duration	predicted_label	0.010	0.50
credit_history	predicted_label	-0.075	0.90
savings/bonds	predicted_label	-0.065	0.95
employment_duration	predicted_label	-0.059	1.00
property	predicted_label	0.086	1.00
duration	purpose	0.041	1.00
credit_history	purpose	-0.319	0.70
credit_history	number_of_credits	0.280	0.95
job	number_of_credits	-0.068	0.25
people_liable	number_of_credits	0.315	0.40

whose edges satisfy both structural and predictive relevance criteria.

Compared with the original causal topology, the resulting graph is substantially more compact while preserving the principal causal pathways leading to the predicted credit-risk decision. This refined weighted graph constitutes the final output of the data-driven learning phase and serves as the causal knowledge base subsequently translated into symbolic representations for fairness verification. The final graph is presented in Figure 2. Tables 5, 6, 7, 8 below offer a complete list of dependencies with coefficients split thematically by the child node.

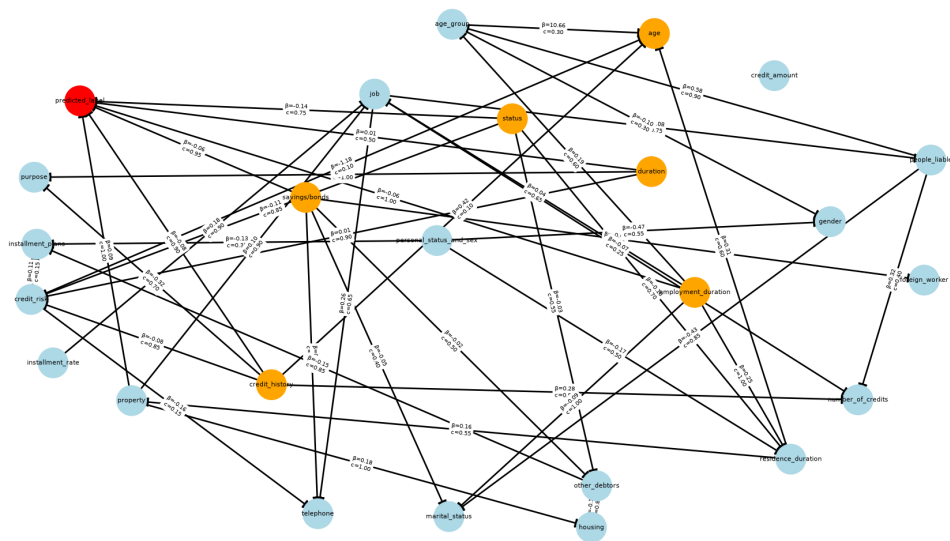


Figure 2. Causal graph learned from the bootstrap-augmented working dataset using the PC algorithm implemented in causal-learn, with structural coefficient estimated through OLS and pruning. The red node is the target variable, the orange nodes are those identified as relevant by the BRIO analysis.

Table 6. Retained causal relations associated with socioeconomic variables.

Parent	Child	β	c
status	residence_duration	-0.096	0.70
employment_duration	residence_duration	0.246	1.00
personal_status_and_sex	residence_duration	-0.172	0.50
residence_duration	property	0.161	0.55
other_debtors	housing	-0.116	0.80
property	housing	0.177	1.00
employment_duration	job	0.036	0.65
installment_rate	job	0.180	0.90
property	job	0.096	0.90
job	people_liable	-0.082	0.75
savings/bonds	telephone	0.042	0.75
job	telephone	0.265	0.65
credit_risk	telephone	-0.165	0.15
status	other_debtors	-0.031	0.55
savings/bonds	other_debtors	-0.023	0.50

Interpretation of the Final Causal Model

After pruning, the classifier-output node *predicted_label* has six direct parents:

Table 7. Retained causal relations associated with demographic variables.

Parent	Child	β	c
credit_history	age	0.416	0.10
residence_duration	age	0.308	0.60
credit_risk	age	-1.180	0.10
age_group	age	10.658	0.30
employment_duration	age_group	0.187	0.60
people_liable	age_group	0.584	0.90
personal_status_and_sex	gender	-0.465	0.55
age_group	gender	-0.101	0.30
savings/bonds	marital_status	-0.048	0.40
employment_duration	marital_status	-0.092	1.00
people_liable	marital_status	-0.431	0.85

Table 8. Retained causal relations involving auxiliary variables.

Parent	Child	β	c
personal_status_and_sex	installment_plans	-0.130	0.35
other_debtors	installment_plans	-0.150	0.85
credit_risk	installment_plans	0.115	0.15
savings/bonds	foreign_worker	-0.010	0.65

$\{status, duration, credit_history,$
 $savings/bonds, employment_duration, property\}.$

The edge from *purpose* is removed because its coefficient falls below the threshold ($\beta = 0.013 < 0.05$) and BRIO opacity analysis does not identify this feature as predictively relevant. The largest absolute coefficients belong to *status* ($\beta = -0.140$), *property* ($\beta = 0.086$), and *credit_history* ($\beta = -0.075$). The remaining parents are retained for their predictive relevance. Bootstrap recovery is also high: $c = 1$ for the edges from *employment_duration* and *property*, $c = 0.95$ for *savings/bonds*, and $c = 0.90$ for *credit_history*.

The graph also captures indirect pathways. For example, *employment_duration* influences the prediction through *residence_duration* and then *property*, while *status* affects both *other_debtors* and *residence_duration*. These pathways expose intermediate dependencies beyond those represented by feature-importance rankings.

Demographic variables form a largely separate subgraph. In the learned topology, the parents of *age* are *credit_history*, *residence_duration*, *credit_risk*, and *age_group*; those of *gender* are *personal_status_and_sex* and *age_group*; and those of *marital_status* are *savings/bonds*, *employment_duration*, and *people_liable*. None of these demographic variables directly parents the prediction,

so any influence within the graph must follow indirect pathways. This distinction informs the subsequent verification of direct and indirect discrimination.

Structural coefficients and bootstrap confidence provide complementary information. Under the stated causal assumptions, $\beta(e)$ estimates a local linear causal effect, whereas $c(e)$ measures edge recovery under the adopted resampling procedure. The pruned edge *credit_amount* $\rightarrow$ *installment_rate*, with $\beta \approx 0$ and $c = 1$, illustrates that high recovery stability can coexist with negligible estimated influence.

The resulting graph provides the causal knowledge base for the symbolic phase. Its topology defines logical dependencies, while coefficients and confidence values retain quantitative information about estimated influence and recovery stability. This representation supports subsequent causal explanation, opacity analysis, and formal fairness verification.

Limitations of Coefficient Estimation and Graph Pruning

OLS coefficients measure conditional linear associations under the adopted graph and model. Their interpretation as local causal effects requires correct edge orientations, adequate structural equations, and no relevant unobserved confounding. For numerically encoded categorical or ordinal variables, coefficient magnitudes also depend on encoding and scale, limiting comparisons across variables. A threshold such as $|\beta| < 0.05$ is therefore a heuristic pruning criterion, not an invariant measure of relevance.

Deleting edges can remove active paths and introduce d -separations absent from the original graph. Fairness certificates derived from the pruned graph are consequently conditional on the pruning procedure and its thresholds. We therefore treat this graph as an exploratory, model-dependent approximation rather than a uniquely identified causal DAG. Future work should assess certificate sensitivity to alternative encodings, estimation models, pruning thresholds, and unpruned graph specifications.

Phase II: Causal Knowledge Injection and the TNDPQ Calculus

Phase II translates the learned causal structure into a symbolic environment supplying the premises for the calculus TNDPQ, independently of the discovery algorithm used in Phase I.

Graph Injection

Definition 0.8. Graph Injection. Let $G = (V, E, \Theta)$ be the learned, pruned directed acyclic graph, where V contains its variables, $E \subseteq V \times V$ its directed edges, and $\Theta = \{\beta, c\}$ its structural coefficients and bootstrap confidence scores. The *graph injection*

$$\mathcal{J} : G \longrightarrow CGph$$

encodes the graph's structure and annotations as logical facts in the causal environment $CGph$.

The injection represents:

- vertices as atomic propositions from which logical formulas are built;
- directed edges as immediate causal relations;
- quantitative annotations as labels on causal judgments;
- structural configurations, including paths, forks, colliders, and conditional independencies, through the corresponding TNDPQ symbolic relations.

TNDPQ reasons exclusively over $CGph$, using the graph as its semantic source. The injected environment stores primitive facts supported by the learned graph; further relations are derived through the calculus. Its deductive closure comprises all causal knowledge derivable from these facts under the proof system.

TNDPQ Language and Causal Contexts

We first introduce the sets of atomic protected (AtA) and atomic unprotected variables (AtX), the set of atomic variables being defined as their union, $AtV = AtA \cup AtX$. Protected (A) and unprotected variables (X) are then defined inductively from their respective atomic variables, with the distinction that the former are constructed exclusively from protected components, whereas the latter variables may depend on both protected and unprotected ones. The set of all variables is finally defined as $V = A \cup X$. For each atomic variable, we define its corresponding domain of values, denoted by Val_{a_i} for protected variables and Val_{x_i} for unprotected ones. The sets of protected values (Val_A) and unprotected values (Val_X) are defined inductively over the unions of the corresponding atomic domains, with unprotected values allowed to incorporate both protected and unprotected atomic values. Finally, the set of all values is defined as $Val = Val_A \cup Val_X$.

Definition 0.9. Language.

$$AtA := a_1, \dots, a_n$$

$$AtX := x_1, \dots, x_n$$

$$AtV := AtA \mid AtX$$

$$A := AtA \mid \langle A, A \rangle$$

$$X := AtX \mid \langle A, X \rangle \mid \langle X, A \rangle \mid \langle X, X \rangle$$

$$V := A \mid X$$

$$Val_{a_i} := \alpha^{i_1}, \alpha^{i_2}, \dots, \alpha^{i_j} \mid Val_{a_i} + Val_{a_i} \mid (Val_{a_i})^\perp$$

$$Val_{x_i} := \beta^{i_1}, \beta^{i_2}, \dots, \beta^{i_j} \mid Val_{x_i} + Val_{x_i} \mid (Val_{x_i})^\perp$$

$$Val_A := Val_{a_1}, \dots, Val_{a_n} \mid Val_A \times Val_A \mid Val_A + Val_A \mid (Val_A)^\perp$$

$$Val_X := Val_{x_1}, \dots, Val_{x_n} \mid Val_X \times Val_X \mid Val_A \times Val_X \mid Val_X \times Val_A \mid Val_X + Val_X \mid (Val_X)^\perp$$

$$Val := Val_A \mid Val_X$$

Table 9. Rules for AtJ

$$\frac{a_i \in \text{AtA} \quad \alpha^{ij} \in \text{Val}_{a_i}}{a_i : \alpha^{ij} \in \text{AtJ}} \text{AtJ} \quad \frac{x_i \in \text{AtX} \quad \beta^{ij} \in \text{Val}_{x_i}}{x_i : \beta^{ij} \in \text{AtJ}} \text{AtJ}$$

Table 10. Rules for J

$$\frac{V : \text{Val}_1 \quad V : \text{Val}_2}{V : \text{Val}_1 + \text{Val}_2} \text{J+} \quad \frac{V_1 : \text{Val}_1 \quad V_2 : \text{Val}_2}{\langle V_1, V_2 \rangle : \text{Val}_1 \times \text{Val}_2} \text{Jx} \quad \frac{V : \text{Val}}{V : \text{Val}^\perp} \text{J}^\perp$$

Table 11. Rules for J^P

$$\frac{V : \text{Val} \in \text{J} \quad p \in [0, 1]}{V : \text{Val}_p \in \text{J}^P} \text{J}^P$$

When we want to refer to variables in general, regardless of their protected or unprotected status, we use $V_1, \dots, V_n$, and $v_1, \dots, v_n$ for the atoms in particular, using $\Delta^{1_1}, \dots, \Delta^{n_j}$ and $\delta^{1_1}, \dots, \delta^{n_j}$ respectively for their values. As usual, given $V = \langle V_1, V_2 \rangle$, we use $fst(V)$ and $snd(V)$ to refer to V_1 and V_2 respectively.

Notice that variables that are composed by both protected and unprotected variables are considered unprotected. This is due to the fact that protected variables are used to address fairness, which requires that the value of protected variables does not influence the value of the target feature. It would be an exorbitant demand to require that all variables composed by both protected and unprotected variables were irrelevant to the value of the target, and so these complex variables cannot be regarded as protected, under this interpretation.

Now we codify three components of sequents:

- information about values received by the variables used in the decisions of the classifier;
- information about the probabilistic decision of the classifier;
- information about the causal structure that lies beneath the decision of the classifier.

The first two components are expressed through Judgments, composed associating variables and values just defined. The third component is expressed by relational atoms connecting atomic variables.

Definition 0.10. Judgments. Atomic judgments AtJ are defined by the rules of Table 9. Judgments J are the closure of AtJ under the three rules in Table 10. Probabilistic judgments J^P are defined by the single rule of Table 11.

With (probabilistic) judgments we define the left-hand and right-hand components of the sequents as follows.

Table 12. Rules to derive paths from immediate causal relations in CGph. The rule of Transitivity has condition *: $i \in O \cup P$ and $j \in M \cup N$.

Reflexive cause $\vdash v_i \blacktriangleright^{\{i\}} v_i$	Transitive cause $v_i \blacktriangleright^N v_j, v_j \triangleright v_k \vdash v_i \blacktriangleright^{N \cup \{k\}} v_k$
Chain $v_i \triangleright v_j, v_j \triangleright v_k \vdash v_i \diamond^{\{j\}} v_k$	Fork $v_j \triangleright v_i, v_j \triangleright v_k \vdash v_i \diamond^{\{j\}} v_k$
Collider $v_i \triangleright v_j, v_k \triangleright v_j, v_j \blacktriangleright^N v_z \vdash v_i \diamond_{\{j,N\}} v_k$	Transitivity* $v_x \diamond_N^M v_i, v_j \diamond_P^O v_y \vdash v_x \diamond_{N \cup P}^{M \cup O} v_y$

Definition 0.11. Context. Contexts codify the known causal information in the left-hand side. The definition is inductive, starting from atomic contexts:

$$At\Sigma =_{df} \{at\sigma \subseteq AtJ \mid \forall i(v_i : \delta^{i_l} \in at\sigma \wedge v_i : \delta^{i_m} \in at\sigma \rightarrow l = m)\}$$

A Context is an Atomic Context, or the result of (possibly multiple) applications of the rules in Table 10 to an Atomic Context. Formally,

- $At\Sigma \subseteq \Sigma$
- If $\sigma \in \Sigma$, $V_1 : Val_1 \in \sigma$, $V_2 : Val_2 \in \sigma$, and $V_3 : Val_3$ is the conclusion of an application of a rule in Table 10 to $V_1 : Val_1$ and $V_2 : Val_2$, then $\sigma \setminus \{V_1 : Val_1, V_2 : Val_2\} \cup \{V_3 : Val_3\} \in \Sigma$.
- If $\sigma \in \Sigma$, $V_1 : Val_1 \in \sigma$, and $V_2 : Val_2$ is the conclusion of an application of a rule in Table 10 to $V_1 : Val_1$, then $\sigma \setminus \{V_1 : Val_1\} \cup \{V_2 : Val_2\} \in \Sigma$.

We now consider relational atoms and causal relations. To treat causal relations, we first add the binary relation $v_i \triangleright v_j$ between variables, with the intended meaning: v_i receiving a given value is a cause of v_j receiving a given value. Then, we compose causal graphs as sets of causal relations – imposing non-cyclic transitive closures and define causal sequents as sequents with causal graphs in their antecedent.

Definition 0.12. Immediate Causal Relations.

$$CRel := AtV \triangleright AtV$$

To compose causal graphs from immediate causal relations, we need to take into account the closure of subsets of CRel under reflexivity and transitivity, to reconstruct mediate causal relations.[§] Table 12 displays the rules to derive both mediate causal relations $\blacktriangleright$ and paths $\diamond$ (with intermediate nodes) from a set of relational atoms. Practically, we will consider the closure of these subsets under the $\blacktriangleright$ -rules (Reflexive Cause and Transitive Cause) of this table. We denote with $crel$ the subsets of CRel, and with $crel^\blacktriangleright$ their closure under these rules.

[§]Transitivity would be enough in this case, but for technical reasons related to the rule of Weakening it is more convenient to close the set of immediate causal relations also under reflexivity.

Definition 0.13. Causal Relations and Paths. Given a set of immediate causal relations $\text{crel} \subseteq \text{CRel}$, the set $\text{crel}^\blacktriangleright$ of (possibly mediated) causal relations is its closure under the rules **Reflexive Cause** and **Transitive Cause** of Table 12. Moreover, the set $\text{crel}^\diamond$ of causal paths is defined as its closure under all the rules of Table 12. Since $\text{cgph} \subseteq \text{CRel}$, this definition applies to $\text{cgph}^\blacktriangleright$ and $\text{cgph}^\diamond$ as well.

Then, we define causal environments and causal sequents as follows:

Definition 0.14. Causal Environment.

$$\text{CGph} =_{df} \{ \text{cgph} \subseteq \text{CRel} \mid \forall i, j (\exists M, N (v_i \blacktriangleright^M v_j \in \text{cgph}^\blacktriangleright \wedge v_j \blacktriangleright^N v_i \in \text{cgph}^\blacktriangleright) \rightarrow i = j) \}$$

Definition 0.15. Causal Sequents.

$$\text{CGph}, \Sigma \vdash \text{J}^P$$

Example 1. [¶] Let us consider the graph in Figure 2. As directed edges are acyclic, when translating them using relational atoms for immediate causal relations, we end up with a set cgph of $\triangleright$ -formulas. Let us consider a decision regarding a loan ($\text{target_label} : \delta$) to buy a radio or television ($\text{Purpose} : \beta^1$), and assume the person making the request is a single male ($\text{Personal_status_and_sex} : \beta^2$), with an existing checking account between 0 and 200 DM ($\text{Status} : \beta^3$), who always paid back the existing credits till now ($\text{Credit_history} : \beta^4$). A plausible decision would be the following:

$$\text{cgph}, \text{Purp} : \alpha^1, \text{Per_stat_sex} : \beta^2, \text{St} : \beta^3, \text{Cr_hist} : \beta^4 \vdash \text{t_label} : \delta_{13/20}$$

We add now weights for both causal strength (denoted here with w) and confidence (denoted here with c) in line with Phase I of the pipeline.

Definition 0.16. Weighted Immediate Causal Relation.

$$\text{WCRel} := \text{AtV} \triangleright_c^w \text{AtV}$$

The intended meaning of $v_i \triangleright_c^w v_j$ is : v_i receiving a given value is a cause with strength w and confidence c of v_j receiving a given value. Also in this case, we define (possibly mediated) weighted causal relations ($\triangleright_c^w$) in terms of immediate ones. Moreover, we take into account the existence of alternative paths connecting two variables, giving the weight of the resulting causal connection: $v_i \tilde{M}, \tilde{N}_c^w v_j$ intuitively means that there are two directed paths from v_i to v_j , crossing respectively the nodes in sets M and N , that the strength of such a causal

[¶]The Examples in this and the following sections do not necessarily rely only on the pruned final causal model obtained in the previous Phase, and are to be considered only illustrative of the type of decision-making process symbolically formalised in TNDPQ.

Table 13. Rules to derive weighted paths from weighted immediate causal relations in CGph_c^w . ‘Alt.’ stands for ‘alternative’, ‘con.’ for ‘connection’, ‘tr.’ for ‘transitive’, ‘di-path’ for ‘direction-ignoring-path’. The rule *Alt(ernative) paths* has condition *: $M \notin N$. The rule *Tr(ansitive) Conn(ection)* has condition **: $M \cap N = \emptyset$, $v_i \notin N$, and $v_k \notin M$.

Reflexive cause $\vdash v_i \succ_1^1 v_i$	Transitive cause $v_i M \succ_{c_1}^{w_1} v_j, v_j \succ_{c_2}^{w_2} v_k \vdash v_i M \cup \{v_y\} \succ_{\min(c_1, c_2)}^{w_1 w_2} v_k$
Path $v_i M \succ_c^w v_j \vdash v_i \overrightarrow{M}_c^w v_j$	Alt. Paths* $v_i M \succ_{c_1}^{w_1} v_j, v_i \overrightarrow{N}_{c_2}^{w_2} v_j \vdash v_i \overrightarrow{M, N}_{\max(c_1, c_2)}^{w_1 + w_2} v_j$
Con. $v_i M \succ_c^w v_j \vdash v_i M \dot{\vdash}_c v_j$	Connection $v_j M \succ_c^w v_i \vdash v_i M \dot{\vdash}_c v_j$
Tr. Conn.** $v_i M \dot{\vdash}_{c_1} v_j, v_j N \dot{\vdash}_{c_2} v_k \vdash v_i M \cup N \dot{\vdash}_{\min(c_1, c_2)} v_k$	
Di-path $v_i M \dot{\vdash}_c v_j \vdash v_i \overline{M}_c v_j$	Alt. di-paths $v_i M \dot{\vdash}_{c_1} v_j, v_i \overline{N}_{c_2} v_j \vdash v_i \overline{M, N}_{\max(c_1, c_2)} v_j$

relation is w , and that its confidence is c . The intended meaning of $v_i M \dot{\vdash}_c v_j$ is: v_i and v_j are connected on the causal graph, and the connection’s confidence is c . Notice that connection ignores the direction of the causal relations. Direction-ignoring paths (di-paths) are constructed relying on connections, and the existence of alternative di-paths connecting v_i and v_j is expressed by $v_i \overline{M, N}_c v_j$, where c specifies the confidence that v_i and v_j are connected. We call these paths ‘direction-ignoring’ rather than ‘undirected’ to stress that they are extracted from a directed graph, for which we choose to ignore the direction. Defining direction-ignoring paths allows, at a next stage, to check the confidence of all connections between two points: this, in turn, allows us to establish the likelihood that they will be disconnected in the graph when some rule is applied. In particular, this will be needed to evaluate failure of individual fairness under Weakening. Both mediated causal relations and alternative paths (with their respective weights) can be derived from immediate causes using the rules of Table 13.^{||}

Definition 0.17. Weighted Causal Relations and paths. The set $\text{wcrel}^\triangleright$ of (possibly mediated) weighted causal relations is the closure of a set of immediate weighted causal relations $\text{wcrel} \subseteq \text{WCREl}$ under the rules *Reflexive cause* and *Transitive cause* of Table 13. Moreover, wcrel determines the set $\text{wcrel}^\rightarrow$ of alternative paths connecting two variables, which is obtained closing wcrel under the rules *Reflexive cause*, *Transitive cause*, *Path*, and *Alternative Paths* of Table 13. The set $\text{wcrel}^{\dot{=}}$ of connections that ignore directions is obtained by the closure of wcrel under the rules *Reflexive cause*, *Transitive cause*, *Connection* and *Transitive connection* of Table 13. Moreover, the set wcrel^- of alternative direction-ignoring paths connecting two variables is obtained by the closure of wcrel under all rules of Table 13. Since $\text{wgcph} \subseteq \text{WCREl}$, this definition applies to $\text{wgcph}^\triangleright$, $\text{wgcph}^\rightarrow$, $\text{wgcph}^{\dot{=}}$, and wgcph^- as well.

^{||}The restriction on *Alt. Paths* avoids counting twice the same path. The same restriction is not needed for *Alt. di-Paths*, because di-paths are not labeled with a strength weight. The restriction on *Tr. Conn.* avoids the construction of paths in which there is a ‘detour’: a nonempty path connecting one point to itself. The same restriction is not needed for directed paths, since we know the causal graph is acyclic.

The justification of the rules in Table 13 is based on how the weights (causal strength and confidence) are defined in Subsection *Parameter Learning and Confidence Estimation*. The strength of the immediate causal relation is based on the structural coefficient obtained through a linear multiple regression on its parent nodes of the final causal model obtained by bootstrapping. Consider the variable v_z with parents $v_{y_1}, \dots, v_{y_m}$, and, of these variables, consider the variable v_{y_i} with parents $v_{x_1}, \dots, v_{x_n}$. For every parent v_{y_i} of v_z and every parent v_{x_j} of v_{y_i} , the regression gives weights $w_{y_i,z}$ and w_{x_j,y_i} , such that:

$$\begin{aligned} v_z &= \beta_{0_z} + \sum_{v_{y_i} \triangleright^{w_{y_i,z}} v_z} w_{y_i,z} v_{y_i} + \varepsilon_z \\ v_{y_i} &= \beta_{0_{y_i}} + \sum_{v_{x_j} \triangleright^{w_{x_j,y_i}} v_{y_i}} w_{x_j,y_i} v_{x_j} + \varepsilon_{y_i} \end{aligned}$$

To obtain the weight of the *mediated* causal relation connecting v_{x_j} with v_z , we compose the functions. In particular, we compose the function giving the value of v_{y_i} in terms of v_{x_j} with the function giving the value of v_z in terms of v_{y_i} . In this case, the composition of the previous equations gives:

$$v_z = \beta_{0_z} + \sum_{v_{y_i} \triangleright^{w_{y_i,z}} v_z} w_{y_i,z} \beta_{0_{y_i}} + \sum_{v_{y_i} \triangleright^{w_{y_i,z}} v_z} \sum_{v_{x_j} \triangleright^{w_{x_j,y_i}} v_{y_i}} w_{y_i,z} w_{x_j,y_i} v_{x_j} + \sum_{v_{y_i} \triangleright^{w_{y_i,z}} v_z} w_{y_i,z} \varepsilon_{y_i} + \varepsilon_z$$

We can divide this equation into three elements: an intercept (which is independent of the value of v_{x_j}), a part that depends on the value of v_{x_j} (and some structural coefficient), the remaining error. More in detail, the value of v_z is given in terms of:

1. the intercept, consisting in the sum of:
 - the intercept of the regression of v_z on its parents (β_{0_z});
 - the sum of the products of the intercept $\beta_{0_{y_i}}$ for the structural coefficient obtained by linear regression on the parents v_{y_i} of v_z .
2. the contribution of the parents (v_{x_j}) of the parents (v_{y_i}) of v_z , mediated by the respective structural coefficients, consisting in the sum of the products of:
 - the structural coefficient obtained by linear regression on the parents v_{y_i} of v_z ;
 - the structural coefficient obtained by linear regression on the parents v_{x_j} of v_{y_i} ;
 - the value of v_{x_j} .
3. the residual error, consisting of:

- the sum of the products of the structural coefficient obtained by linear regression on the parents v_{y_i} of v_z and the residual error of the regression for v_{y_i} ;
- the residual error for v_z .

Since the strength of the causal relation is defined as the structural coefficient of the regression, the strength of the mediate causal relation can be identified as the product of the strength of the immediate relations composing it, as specified in Rule *Transitive cause*. As remarked in point 2, the contribution of the value v_{x_j} to that of v_z is mediated by the product of the structural coefficients of the regression of v_z on its parents v_{y_i} and of that of v_{y_i} on its respective parents v_{x_j} . Moreover, when alternative causal paths connect two variables, the strength of the causal relation is the sum of the strength of the singular causal path, as specified in Rules *Path* and *Alternative paths* (we avoid counting the same path multiple times). This stipulation is made plausible by the following observation: when v_z has n parents $v_{y_1}, \dots, v_{y_n}$, and they all have the same single parent v_x , the middle term of the equation for the value of v_z in terms of v_x becomes:

$$\sum_{v_{y_i} \triangleright^{w_{y_i,z}} v_z} \sum_{v_x \triangleright^{w_{x,y_i}} v_{y_i}} w_{y_i,z} w_{x,y_i} v_x = \left(\sum_{v_{y_i} \triangleright^{w_{y_i,z}} v_z} \sum_{v_x \triangleright^{w_{x,y_i}} v_{y_i}} w_{y_i,z} w_{x,y_i} \right) v_x$$

As a consequence, v_x is multiplied for the sum of the strengths of the respective mediate causal relations connecting it with v_z . The same argument generalizes to causal relations requiring longer paths of immediate causes.**

Hence, fixing two nodes v_x and v_y , we define the weight measuring the strength of their causal connection as follows:

$$w(v_x, v_y) = \sum_{v_x \xrightarrow{M} v_y \in \text{cgph}} \prod_{v_i \triangleright^{w_{v_i,v_j}} v_j : \{i,j\} \subseteq \{x\} \cup M \cup \{y\}} w_{v_i,v_j}$$

Notice that in this way the strength of indirect causal connection is established relying on that of direct causal relations and the chains that can be individuated in the causal graph. Obviously, when v_x, v_y have an immediate connection, $w(v_x, v_y) = w_{v_x,v_y}$.

Since the function (obtained by linear regression on a variable and its parents) that determines the strength of the causal relation may not be invertible, there is no uniform way of calculating the weight of the (direction-ignoring) correlations

**Notice that the causal paths defined using $\diamond$ in Table 12 are different from the paths defined using $\rightarrow$ in Table 13. Indeed, in the first case, the paths do not necessarily follow the direction of causality, and contain the consequences of their colliders, while in the second case, the paths follow the direction of causality and multiple alternative paths connecting the same variables are considered. The reason for this difference is just that the paths relevant for conditional independence are not the ones relevant to estimate the weights of the causal relation.

$v_i \doteq v_j$. This is the reason why rules *Connection*, *transitive connection*, *di-path* and *alternative di-paths* of Table 13 drop strength.

As for the confidence, a path from variable v_{x_j} to variable v_z remains connected in a graph obtained by an application of PC algorithm iff *all* the immediate causal relations that make up the path between remain in place. Although a path from variable v_{x_j} to variable v_z is present only when all its constituent edges are present, the marginal bootstrap recovery frequencies of those edges do not determine their joint recovery frequency. The calculus therefore assigns a path the minimum edge-stability value as a graph-theoretic bottleneck weight by Rule *Transitive cause*. When alternative paths in $wcgp\vec{h}^{\rightarrow}$ exist from variable v_{x_j} to variable v_z , the maximum selects the path having the greatest bottleneck weight by Rules *Path* and *Alternative paths* since a single path is sufficient to connect the variables. These min-max operations define an ordering of paths in the fixed weighted graph and are not estimators of path-recovery probabilities:

$$c(v_{x_j}, v_z) = \max_{v_{x_j} M \succ_c^{w_c} v_z \in wcgp\vec{h}^{\rightarrow}} \min_{e \in (v_{x_j} M \succ_c^{w_c} v_z)} c(e).$$

This way of calculating the confidence of the paths connecting two points generalizes to the case of direction-ignoring paths defined in Table 13. This is the reason why weight for confidence is retained in the rules *Connection*, *transitive connection*, *di-path* and *alternative di-paths*, as opposed to what happens with strength.^{††} Although the calculation also applies to direction-ignoring paths, the confidence calculated over directed paths need not be the same as that calculated over direction-ignoring paths, since more direction-ignoring paths may connect the same nodes. The confidence calculated on direction-ignoring paths between nodes X, Y is denoted $c(X, Y)^-$.

On this basis we generalize environment and sequents to their weighted counterparts:

Definition 0.18. Weighted Causal Environment.

$$\begin{aligned} WCGph =_{df} \{ & wcgp\vec{h} \subseteq WCRel \mid \forall i, j (\exists M, N, w_1, w_2, c_1, c_2 (v_i M \succ_{c_1}^{w_1} v_j \in wcgp\vec{h}^{\rightarrow} \\ & \wedge v_j N \succ_{c_2}^{w_2} v_i \in wcgp\vec{h}^{\rightarrow}) \rightarrow i = j) \} \end{aligned}$$

Definition 0.19. Weighted Causal Sequents.

$$WCGph, \Sigma \sim J^P$$

^{††}Notice that the causal paths defined using $\diamond$ in Table 12 are different from the direction-ignoring paths defined using $-$ in Table 13. While in both cases the paths do not necessarily follow the direction of causality, only in the first case the consequences of the colliders are taken into account. For this reason, it is easier to define direction-ignoring paths anew instead of adding weights to Table 12. The ground reason is that the paths relevant for conditional independence are not the ones relevant to estimate the confidence of the connections between two nodes.

Example 2. Using data from Tables 5, 6 and 7, we have

$$\begin{aligned} \text{status} &\triangleright_{0.70}^{-0.096} \text{residence_duration} & \text{residence_duration} &\triangleright_{0.60}^{0.308} \text{age} \\ \text{status} &\triangleright_{0.85}^{-0.113} \text{credit_risk} & \text{credit_risk} &\triangleright_{0.10}^{-1.180} \text{age} \end{aligned}$$

Respectively from the first and the second pair of causal relations, we can derive two paths:

$$\begin{aligned} (\text{status}, \text{residence_duration}) &\triangleright_{0.60}^{-0.03} \text{age} \\ (\text{status}, \text{credit_risk}) &\triangleright_{0.10}^{0.13} \text{age} \end{aligned}$$

In conclusion, taking both paths together, we have:

$$(\text{status}, \overrightarrow{\text{residence_duration}, \text{credit_risk}}) \triangleright_{0.60}^{0.10} \text{age}$$

Phase III: TNDPQ Calculus and Formal Fairness Verification

In this section, we define an inference relation for reasoning about the probabilistic outputs of a classifier on the basis of the information encoded in a weighted causal graph. The calculus has two main aims:

1. to formulate structural conditions under which fairness properties can be established from the causal graph;
2. to quantify the extent to which these conditions fail.

Although the calculus is defined over weighted causal graphs, the structural conditions themselves do not depend on the weights representing causal strength and confidence.^{††} This quantitative information is nevertheless preserved, since it is essential for defining both a graph-relative likelihood of individual fairness failure and a graph-relative measure of counterfactual fairness failure.

Inference Rules

The inference relation of the calculus TNDPQ is defined by the atomic rules, logical rules, rules for conditional independence and structural rules. Table 14 contains a single rule for atomic variables, which formalizes in TNDPQ the probabilistic decision resulting from the data analyzed in Phase I. To be more precise, each trained classifier corresponds to a set of instances of the rule *AtQuery*. The probabilistic rules that deal with logical operators are adapted from (12). The only change is the addition of the weighted causal graph on the left-hand side of the sequent and the formulation for both protected and unprotected variables. As an exemplification of rules for connectives, Table 15 contains the probabilistic rules for the introduction and elimination of conjunction on the right of the sequent.

^{††}The problem of formulating conditions for verifying fairness properties that also depend on these weights is left for future work.

Table 14. Rules for Atomic Query

$$\frac{}{\text{wcgph}, \text{at}\sigma \vdash v_i : \delta_p^{ij}} \text{AtQuery}$$

The system TNDPQ is substructural, since the usual structural rules are not valid in general. Extending the information relating to a datapoint object of classification *does* (in general) change the decision of the classifier. This should not come as a surprise, since substructurality is a common feature of probabilistic logics. In Table 16, we evaluate in which cases the structural rules are admissible. The rule of Contraction is vacuously admissible, since σ is defined as a set of judgments. The rule of *Weakening* (denoted as *W* in the proof trees for brevity) requires the extra premise $v_u \parallel_{\text{wcgph}}^{\sigma} v_t$, formalizing the conditional independence of v_t on v_u in the weighted graph when condition σ is used on the left-hand side of the sequent. Table 17 defines the conditions for $v_u \parallel_{\text{wcgph}}^{\sigma} v_t$, relying on d-separation – a *criterion* that verifies whether two variables are independent conditional on a set of other variables (5). The rule of *Cut* holds only in its additive formulation, that is, with the same context in both premises and conclusion. This is due to the fact that, while with shared context *Cut* removes redundancies from the current datapoint (removing pieces of information that are derivable), with independent contexts it also merges the information contained in the premises. Multiplicative *Cut* is unjustified, because the derivability of $v_u : \delta^{u_i}$ from σ and the coherence of a context $\sigma', v_u : \delta^{u_i}$ (used for the decision $v_t : \delta^{t_j}$) do not entail the coherence of the context σ, σ' (or *a fortiori* that it would lead to the same decision regarding v_t). For an obvious example, consider *Age* : $x < 18$ as σ , *marital_status* : *unmarried* as $v_u : \delta^{u_i}$ and *Age* : $18 < x < 30$ as σ' , and observe that σ, σ' is not a proper context, since it violates Definition 0.11. Notice that explicit structural rules are *not* justified by probabilistic reasoning, but by graph inspection, formalized in *wcgph*. Probabilistic rules can obviously make some structural rules derivable, but there is no need to add explicitly these instances of rules to our palette.

Once the graph has been translated into TNDPQ labels, fairness verification can be executed. Three kinds of analysis are performed:

1. Individual Fairness;
2. Intersectional Fairness;
3. Counterfactual Fairness.

Verifying Individual and Intersectional Fairness

For Individual Fairness, the task to be submitted for formal verification is the following:

Both structural and conditional independence rules are restricted to atomic variables and values. We leave the non-atomic extension to further investigation.

Table 15. Rules for $\times$

$$\begin{array}{c}
\frac{\text{wcgph}, \sigma, V_t : \Delta^i \vdash V_u : \Delta_p^{u_j} \quad \text{wcgph}, \sigma \vdash V_t : \Delta_q^{t_i}}{\text{wcgph}, \sigma \vdash \langle V_t, V_u \rangle : (\Delta^i \times \Delta^{u_j})_{pq}} \text{I}\times 1 \\
\\
\frac{\text{wcgph}, \sigma, V_t : \Delta^i \vdash V_u : \Delta_p^{u_j} \quad \text{wcgph}, \sigma \vdash V_t : \Delta_q^{t_i}}{\text{wcgph}, \sigma \vdash \langle V_u, V_t \rangle : (\Delta^{u_j} \times \Delta^i)_{pq}} \text{I}\times 2 \\
\\
\frac{\text{wcgph}, \sigma \vdash V_t : (\Delta^{u_i} \times \Delta^{v_j})_p \quad \text{wcgph}, \sigma, f st(V_t) : \Delta^{u_i} \vdash snd(V_t) : \Delta_{q \neq 0}^{v_j}}{\text{wcgph}, \sigma \vdash f st(V_t) : \Delta_{p/q}^{u_i}} \text{E}\times 1a \\
\\
\frac{\text{wcgph}, \sigma \vdash V_t : (\Delta^{u_i} \times \Delta^{v_j})_p \quad \text{wcgph}, \sigma, snd(V_t) : \Delta^{v_j} \vdash f st(V_t) : \Delta_{q \neq 0}^{u_i}}{\text{wcgph}, \sigma \vdash snd(V_t) : \Delta_{p/q}^{v_j}} \text{E}\times 1b \\
\\
\frac{\text{wcgph}, \sigma \vdash V_t : (\Delta^{u_i} \times \Delta^{v_j})_p \quad \text{wcgph}, \sigma \vdash f st(V_t) : \Delta_{q \neq 0}^{u_i}}{\text{wcgph}, \sigma, f st(V_t) : \Delta^{u_i} \vdash snd(V_t) : \Delta_{p/q}^{v_j}} \text{E}\times 2a \\
\\
\frac{\text{wcgph}, \sigma \vdash V_t : (\Delta^{u_i} \times \Delta^{v_j})_p \quad \text{wcgph}, \sigma \vdash snd(V_t) : \Delta_{q \neq 0}^{v_j}}{\text{wcgph}, \sigma, snd(V_t) : \Delta^{v_j} \vdash f st(V_t) : \Delta_{p/q}^{u_i}} \text{E}\times 2b
\end{array}$$

Table 16. Structural Rules

$$\begin{array}{c}
\frac{\text{wcgph}, at\sigma \vdash v_t : \delta_p^{t_i} \quad v_u \parallel_{\text{wcgph}}^{at\sigma} v_t}{\text{wcgph}, at\sigma, v_u : \delta^{u_j} \vdash v_t : \delta_p^{t_i}} w \\
\\
\frac{\text{wcgph}, at\sigma \vdash v_u : \delta_1^{u_i} \quad \text{wcgph}, at\sigma, v_u : \delta^{u_i} \vdash v_t : \delta_p^{t_j}}{\text{wcgph}, at\sigma \vdash v_t : \delta_p^{t_j}} cut
\end{array}$$

Can a protected variable be added to the context without changing the probability assigned to the target?

Table 17. Conditions for conditional independence

Given wcgph , $\text{at}\sigma$ and v_u, v_t atomic, let $\text{wcgph}^\diamond$ be the closure of wcgph under the rules of Table 15, and $\text{Var}_{\text{at}\sigma}$ the variables occurring in $\text{at}\sigma$. Hence, $v_u \parallel_{\text{wcgph}}^{\text{at}\sigma} v_t$ iff:

Condition1 $v_u \not\triangleright v_t$ and $v_t \not\triangleright v_u$; and

Condition2 For every $v_u \diamond_N^M v_t$ in $\text{wcgph}^\diamond$, either $M \cap \text{Var}_{\text{at}\sigma} \neq \emptyset$ or $\exists S \in N(\exists x \in S \wedge S \cap \text{Var}_{\text{at}\sigma} = \emptyset)$.

On the contrary, $v_u \not\parallel_{\text{wcgph}}^{\text{at}\sigma} v_t$ iff:

Condition1 $v_u \triangleright v_t$ or $v_t \triangleright v_u$; or

Condition2 For some $v_u \diamond_N^M v_t$ in $\text{wcgph}^\diamond$, $M \cap \text{Var}_{\text{at}\sigma} = \emptyset$ and $\forall S \in N(\exists x \in S \rightarrow S \cap \text{Var}_{\text{at}\sigma} \neq \emptyset)$.

Proof-theoretically, this question requires a *criterion* of applicability for the rule of *Weakening*, i.e. one needs to check that the following rule is admissible:

$$\frac{\text{wcgph}, \text{at}\sigma \vdash v_t : \delta_p^{t_i} \quad a_u \parallel_{\text{wcgph}}^{\text{at}\sigma} v_t}{\text{wcgph}, \text{at}\sigma, a_u : \alpha^{u_j} \vdash v_t : \delta_p^{t_i}} \text{w} \quad (1)$$

Notice that Rule 1 is a special case of the one occurring in Table 16, since it imposes that the variable introduced in the conclusion be protected. Hence, we define a rule 2 for Individual Fairness which relies on Conditional Independence and ensures that the desired application of *Weakening* is admissible. The dual rule 3, entailing that one of the premises of the correlated application of *Weakening* does not hold, checks for failures of Individual Fairness.

$$\frac{a_1 \parallel_{\text{wcgph}}^{\text{at}\sigma} v_t \quad \dots \quad a_n \parallel_{\text{wcgph}}^{\text{at}\sigma} v_t \quad a_1, \dots, a_n \in \text{AtA}}{IF[(a_1, \dots, a_n), \sigma, v_t]} \text{IF} \quad (2)$$

$$\frac{a_i \not\parallel_{\text{wcgph}}^{\text{at}\sigma} v_t \quad a_1, \dots, a_i, \dots, a_n \in \text{AtA}}{\overline{IF}[(a_1, \dots, a_i, \dots, a_n), \sigma, v_t]} \text{IF} \quad (3)$$

More formally, we have:

Lemma 0.1. *Given a set of AtQuery rules, representing the decisions of a classifier about a dataset, a weighted causal graph $\text{wcgph} \in \text{WCGph}$ describing the weighted causal relations holding between the features of the dataset, and a set of protected atomic variable $\{a_1, \dots, a_n\} \subseteq \text{AtA}$, a specific classification represented by the derivable sequent $\text{wcgph}, \sigma \vdash v_t : \delta_p^t$ is individually fair regarding the features $\{a_1, \dots, a_n\}$ iff for all $1 \leq i \leq n$, it holds $a_i \parallel_{\text{wcgph}}^{\text{at}\sigma} v_t$.*

Proof. Individual fairness requires that the addition of extra information on a protected attribute is indifferent to the decision of the classifier. The rule of *Weakening* formalizes this property, and its admissibility requires that having conditions on the variables in σ makes the target variable independent of the protected one (Table 16).

Example 3. Let $\sigma = \text{residence_duration} : \beta^1, \text{employment_duration} : \beta^2, \text{credit_history} : \beta^3, \text{credit_risk} : \beta^4, \text{job} : \beta^5, \text{saving/bonds} : \beta^6, \text{status} : \beta^7$ and focus on *gender* as protected feature. Given the immediate causal relations in *wcgp*, we have:

$$\frac{g. \Vdash_{\text{wcgp}} t_label \quad g. \in \text{AtA}}{IF[(\text{gender}), \sigma, t_label]} \text{ IF}$$

and so also

$$\frac{\text{wcgp}, \sigma \vdash t_label : \beta_p \quad g. \Vdash_{\text{wcgp}} t_label}{\text{wcgp}, \sigma, g. : \alpha \vdash t_label : \beta_p} \text{ W}$$

On the contrary, with $\sigma' = \text{employment_duration} : \beta^2, \text{credit_history} : \beta^3, \text{credit_risk} : \beta^4, \text{job} : \beta^5, \text{saving/bonds} : \beta^6$, *Weakening* does not necessarily hold, since *residence_duration* and *status* are needed for the second premise of the rule. Moreover, *Weakening* does not hold for $\sigma'' = \text{residence_duration} : \beta^1, \text{employment_duration} : \beta^2, \text{credit_history} : \beta^3, \text{credit_risk} : \beta^4, \text{job} : \beta^5, \text{saving/bonds} : \beta^6, \text{status} : \beta^7, \text{installment_plans} : \beta^8$ either, since *installment_plans* opens a path between *personal_status_and_sex* and *other_debtors* which leads to *predicted_label*. Hence, we have:

$$\frac{g. \Vdash_{\text{wcgp}}^{\sigma'} t_l \quad g. \in \text{AtA}}{\overline{IF}[(g.), \sigma', t_l]} \text{ IF} \quad \frac{g. \Vdash_{\text{wcgp}}^{\sigma''} t_l \quad g. \in \text{AtA}}{\overline{IF}[(g.), \sigma'', t_l]} \text{ IF}$$

Notice that in TNDPQ individual fairness is checked for specific contexts and protected features, enabling specific applications of *Weakening*. The classifier is fair regarding protected variable a , to decide the value of target variable v_t , using (all and only the) variables in $\text{at}\sigma$. Even when multiple protected variables are given, they are checked one at a time. Hence, this symbolic step makes it possible to reason about fairness of context-specific applications: fairness may require the use of different sets of (unprotected) variables for different decisions. Moreover, the symbolic aspect of TNDPQ makes it possible to make inferences regarding fairness for logically complex decisions (or logically complex information).

Example 4. Extending Example 3, for σ s.t. $\text{residence_duration} : \beta^3, \text{employment_duration} : \beta^4, \text{credit_history} : \beta^5, \text{credit_risk} : \beta^6, \text{job} : \beta^7, \text{saving/bonds} : \beta^8 \in \sigma$, but $\text{installment_plans} \notin \sigma$, we know that the following *weakening* holds:

$$\begin{array}{c}
\frac{\text{wcgph}, \sigma \vdash \text{housing} : \beta_p^1 \quad g. \parallel_{\text{wcgph}}^{\sigma} \text{housing}}{\text{wcgph}, \sigma, g. : \alpha \vdash \text{housing} : \beta_p^1} W \\
\\
\frac{\text{wcgph}, \sigma, \text{housing} : \beta^1 \vdash t_label : \beta_q^2 \quad g. \parallel_{\text{wcgph}}^{\sigma, \text{housing}} t_label}{\text{wcgph}, \sigma, \text{housing} : \beta^1, g. : \alpha \vdash t_label : \beta_q^2} W
\end{array}$$

Using the probability rules of TNDPQ, we can prove that also the following holds:

$$\frac{\text{wcgph}, \sigma \vdash \langle \text{housing}, \text{predicted_label} \rangle : (\beta^1 \times \beta^2)_r}{\text{wcgph}, \sigma, g. : \alpha \vdash \langle \text{housing}, \text{predicted_label} \rangle : (\beta^1 \times \beta^2)_r} W$$

Indeed, the third inference is derivable from the first two as follows:

$$\frac{\frac{\text{wcgph}, \sigma, g. : \alpha \vdash \langle h, p_l \rangle : (\beta^1 \times \beta^2)_r \quad \text{wcgph}, \sigma, h : \beta^1 \vdash p_l : \beta_q^2}{\text{wcgph}, \sigma \vdash \text{housing} : \beta_{rlq}^1 \quad g. \parallel_{\text{wcgph}}^{\sigma} \text{housing}} E \times \quad \frac{\frac{\text{wcgph}, \sigma, g. : \alpha \vdash \langle h, p_l \rangle : (\beta^1 \times \beta^2)_r \quad \text{wcgph}, \sigma \vdash h : \beta_{rlq}^1}{\text{wcgph}, \sigma, \text{housing} : \beta^1 \vdash p_l : \beta_q^2 \quad g. \parallel_{\text{wcgph}}^{\sigma, \text{housing}} p_l} E \times}{\text{wcgph}, \sigma, g. : \alpha \vdash \text{housing} : \beta_{rlq}^1 \quad \text{wcgph}, \sigma, \text{housing} : \beta^1, g. : \alpha \vdash p_l : \beta_q^2} W}{\text{wcgph}, \sigma, g. : \alpha \vdash \langle \text{housing}, p_l \rangle : (\beta^1 \times \beta^2)_r} I \times$$

Notice that the only restriction on the value q is that it be different from 0. For $q = 0$ the weakening of the conjunction is trivial.

Intersectionality requires that a system be fair not only regarding one single protected feature taken in isolation, but also regarding multiple protected attributes taken together. More formally, the classifier's output should not vary when we change at once the value of more protected attributes of the datapoint. Proof-theoretically, what is required is the admissibility of the version of the *Weakening* rule denoted as 4 introducing multiple new judgments at the same time, all regarding protected attributes:

$$\frac{\text{wcgph}, \sigma \vdash v_t : \delta_p^{t_i}}{\text{wcgph}, \sigma, a_1 : \alpha^1, \dots, a_n : \alpha^n \vdash v_t : \delta_p^{t_i}} W \quad (4)$$

Notice that, although Equation 2 for IF takes into account multiple protected variables, the *Weakening* rule in Equation 1, which interprets Individual Fairness proof-theoretically, allows adding only one judgment to its conclusion. This happens because the assumptions about Conditional Independence used in Equation 2 effectively license applications of *Weakening* with only *one* among the protected variables for which IF is proved. The situation with Intersectional Fairness is completely different, since multiple new judgments can be added. Rules for the intersectional version of Individual Fairness are grounded on Conditional Independence and justify the required applications of *Weakening*, see Rule 5. Analogously, the dual rule 6, capturing lack of Conditional Independence, checks for failures in intersectional Individual Fairness.

$$\frac{a_1 \Vdash_{\text{wcgph}}^{at\sigma} v_t \quad \dots \quad a_n \Vdash_{\text{wcgph}}^{at\sigma, a_1, \dots, a_{n-1}} v_t \quad a_1, \dots, a_n \in \text{AtA}}{IF_i[(a_1, \dots, a_n), \sigma, v_t]} \text{IF}_i \quad (5)$$

$$\frac{a_i \nVdash_{\text{wcgph}}^{at\sigma, a_1, \dots, a_{i-1}} v_t \quad a_1, \dots, a_i, \dots, a_n \in \text{AtA}}{\overline{IF}_i[(a_1, \dots, a_i, \dots, a_n), \sigma, v_t]} \overline{IF}_i \quad (6)$$

Lemma 0.2. *Given a set of AtQuery rules, representing the decisions of a classifier about a dataset, a weighted causal graph $\text{wcgph} \in \text{WCGph}$ describing the causal relations holding between the features of the dataset, and a set of protected atomic variable $\{a_1, \dots, a_n\} \subseteq \text{AtA}$, a specific classification represented by the derivable sequent $\text{wcgph}, \sigma \vdash v_t : \delta_p^{t_i}$ is intersectionally fair regarding the features $\{a_1, \dots, a_n\}$ iff for every $1 \leq i \leq n$, it holds $a_i \Vdash_{\text{wcgph}}^{at\sigma, a_1, \dots, a_{i-1}} v_t$.*

Proof. Intersectional fairness requires that the addition of extra information on protected attributes is indifferent to the output of the classifier, even when information is provided about more protected attributes at once. To check this, we have to inspect multiple applications of the rule of *Weakening*, taking into account the effect of the addition of the protected variables into the context, which if intersectionality holds make derivable Rule 4 as follows:

$$\frac{\text{wcgph}, \sigma \vdash v_t : \delta_p^{t_i} \quad a_1 \Vdash_{\text{wcgph}}^{\sigma} v_t}{\text{wcgph}, \sigma, a_1 : \alpha^1 \vdash v_t : \delta_p^{t_i}} \text{w} \quad \frac{a_2 \Vdash_{\text{wcgph}}^{\sigma, a_1} v_t}{\text{wcgph}, \sigma, a_1 : \alpha^1, a_2 : \alpha^2 \vdash v_t : \delta_p^{t_i}} \text{w}$$

$$\vdots$$

$$\frac{\text{wcgph}, \sigma, a_1 : \alpha^1, \dots, a_{n-1} : \alpha^{n-1} \vdash v_t : \delta_p^{t_i} \quad a_n \Vdash_{\text{wcgph}}^{\sigma, a_1, \dots, a_{n-1}} v_t}{\text{wcgph}, \sigma, a_1 : \alpha^1, \dots, a_n : \alpha^n \vdash v_t : \delta_p^{t_i}} \text{w}$$

Rule 5 certifies that such a derivation can be constructed, while Rule 6 certifies when it fails.

Example 5. For σ defined as in Example 3, it holds both $\text{gender} \Vdash_{\text{wcgph}}^{\sigma} t_l$ and $\text{age} \Vdash_{\text{wcgph}}^{\sigma, \text{gender}} t_l$. Moreover, both gender and age are protected attributes. Hence, we have:

$$\frac{\text{gender} \Vdash_{\text{wcgph}}^{\sigma} t_l \quad \text{age} \Vdash_{\text{wcgph}}^{\sigma, \text{gender}} t_l \quad \text{gender}, \text{age} \in \text{AtA}}{IF_i[(\text{gender}, \text{age}), \sigma, t_l]} \text{IF}_i$$

From the proof-theoretical point of view, this licenses the following application of *Weakening* (which is indeed derivable from that adopted in Table 16):

$$\frac{\text{wcgph}, \sigma \vdash v_t : \delta_p^{t_i}}{\text{wcgph}, \sigma, \text{gender} : \alpha^1, \text{age} : \alpha^2 \vdash v_t : \delta_p^{t_i}} \text{w}$$

While the qualitative judgment that individual fairness holds or fails is determined by IF-rule based on the conditional-independence relation of the calculus without reference to causal strength and confidence from the weighted causal graph, when individual fairness fails such quantitative information can be used to associate the detected failure with a graph-relative likelihood measure. The offered definition of such a measure, does not distinguish between active and inactive paths-relative to the conditioning context σ . Nor does it identify individual fairness with the absence of paths between the protected and target variables. A context-sensitive refinement restricting our measure to active paths would constitute an additional extension of the framework and is not introduced here. The empirical frequency with which individual fairness fails across different graphs learned from bootstrap samples would require evaluating individual fairness independently in every bootstrap graph and is not estimated here.

Lemma 0.3. Graph-relative likelihood of an individual-fairness failure. *Let a set of ATQUERY rules represent the decisions of a classifier about a dataset. Let $\text{cgph} \in \text{CGph}$ be the fixed reference causal graph, let $\text{wcgph} \in \text{WCGph}$ be its weighted extension, let wcgph^- be its direction-ignoring closure, and let $\{a_1, \dots, a_n\} \subseteq \text{AtA}$ be a set of protected atomic variables. Assume the classification represented by the derivable sequent $\text{cgph}, \sigma \vdash v_i : \delta_{p_i}^t$ has already been established to be not individually fair with respect to a_i . The graph-relative likelihood measure associated with the detected failure satisfies*

$$\lambda_{\text{IF}}(a_i, \sigma, v_i) = c(a_i, v_i)^-$$

Proof. For each undirected path connecting a_i and v_i , the rules of the weighted calculus assign the minimum of the edge weights occurring along that path. This value represents the weight of the path through its least stable connection. When several paths connect the same variables, the ALTERNATIVE PATHS rule selects the maximum of their respective minimum weights. The resulting value is therefore the greatest bottleneck weight among the connections between the protected and target variables in the fixed weighted reference graph. Accordingly, $\lambda_{\text{IF}}(a_i, \sigma, v_i)$ is a graph-theoretic likelihood measure of an IF failure that has already been detected in the reference graph. Although the edge weights from which it is computed are informed by bootstrap stability, the resulting min-max value is neither a bootstrap probability nor the empirical frequency with which IF fails in other learned graphs.

Example 6. *By taking into account the weighted graph wcgph , we can extend Example 3 with a calculation of the confidence that IF fails, when the condition on the left-hand side of the sequent does not allow for an application of Weakening. According to Lemma 0.3, the confidence in the failure of IF for gender is the greatest c s.t. $\text{gender } \bar{N}_c \text{ } t_1 \in \text{wcgph}^-$. Let us first check single paths. Using the*

rules in Table 13, we derive the connections in $wcgp\dot{h}^{\frac{\cdot}{\cdot}}$. We first derive:

$$\begin{aligned} & \text{personal_status_and_sex} \dot{\div}_{0.55} \text{gender} \quad \text{personal_status_and_sex} \dot{\div}_{0.35} \text{installment_plans} \\ & \text{credit_risk} \dot{\div}_{0.15} \text{installment_plans} \quad \text{credit_risk} \dot{\div}_{0.15} \text{telephone} \quad \text{job} \dot{\div}_{0.65} \text{telephone} \\ & \text{employment_duration} \dot{\div}_{0.65} \text{job} \quad \text{property} \dot{\div}_{0.90} \text{job} \quad \text{property} \dot{\div}_{1.00} \text{predicted_label} \end{aligned}$$

Hence, for $L = \{\text{personal_status_and_sex}, \text{installment_plans}, \text{credit_risk}, \text{telephone}, \text{job}, \text{employment_duration}\}$:

$$\text{gender } L \dot{\div}_{0.15} \text{predicted_label}$$

Moreover, in $wcgp\dot{h}^{\frac{\cdot}{\cdot}}$, we have also:

$$\begin{aligned} & \text{personal_status_and_sex} \dot{\div}_{0.55} \text{gender} \quad \text{personal_status_and_sex} \dot{\div}_{0.35} \text{installment_plans} \\ & \text{credit_risk} \dot{\div}_{0.15} \text{installment_plans} \quad \text{credit_risk} \dot{\div}_{0.15} \text{telephone} \quad \text{savings/bonds} \dot{\div}_{0.75} \text{telephone} \\ & \text{savings/bonds} \dot{\div}_{0.95} \text{predicted_label} \end{aligned}$$

Hence, for $M = \{\text{personal_status_and_sex}, \text{installment_plans}, \text{credit_risk}, \text{telephone}, \text{savings/bonds}\}$:

$$\text{gender } M \dot{\div}_{0.15} \text{predicted_label}$$

As last path, in $wcgp\dot{h}^{\frac{\cdot}{\cdot}}$, we have:

$$\begin{aligned} & \text{age_group} \dot{\div}_{0.30} \text{gender} \quad \text{employment_duration} \dot{\div}_{0.60} \text{age_group} \\ & \text{employment_duration} \dot{\div}_{1.00} \text{marital_status} \quad \text{savings/bonds} \dot{\div}_{0.40} \text{marital_status} \\ & \text{savings/bonds} \dot{\div}_{0.95} \text{predicted_label} \end{aligned}$$

Hence, for $N = \{\text{age_group}, \text{employment_duration}, \text{marital_status}, \text{savings/bonds}\}$:

$$\text{gender } N \dot{\div}_{0.30} \text{predicted_label}$$

Now, we can find the confidence of all the paths:

$$\text{gender } \overline{L, M, N}_{0.30} \text{predicted_label}$$

Hence, the detected IF failure for gender has graph-relative likelihood measure $\lambda_{IF} = 0.30$ in the fixed weighted reference graph.

Verifying Counterfactual Fairness

For Counterfactual Fairness, intervention rules are applied to evaluate whether changes in protected attributes modify the target outcome while relevant contextual variables are held fixed. Intuitively, the question is:

Does a counterfactual candidate obtained by intervening on the value of a protected attribute get assigned the same probability to the target as the original individual?

Counterfactual fairness compares predictions under interventions on a protected attribute while accounting for their effects on other features. For example, changing *ethnicity* while fixing residence and salary may overlook the consequences of segregation and intergenerational poverty. At minimum, information dependent on the original protected value cannot simply be carried over to the counterfactual case.

A fuller structural approach (5; 3) distinguishes exogenous background variables from endogenous variables determined by causal mechanisms. It intervenes on the protected attribute, holds the other exogenous variables fixed, and recomputes the remaining endogenous variables before checking whether the prediction remains invariant.

We adopt a minimal setup, since access to the exogenous information needed for this reconstruction cannot be assumed. Following the intervention, we retain the available inputs that are neither direct nor indirect effects of the protected variable and discard factual information about its descendants. The resulting classification therefore uses information that remains valid under the intervention according to the adopted causal model.

First, we extend the definition of Sequent, to account for counterfactuals:

Definition 0.20. Counterfactual Causal Candidate.

$$\text{WCGph}, [\Sigma]^{I(J)}, \Sigma \vdash J^P$$

Definition 0.21. Counterfactual Causal Sequent.

$$\text{WCGph}, [\Sigma]^{I(J)} \vdash J^P$$

Formally, the difference between a ‘counterfactual candidate’ and a proper counterfactual sequent is the presence or absence of Σ outside the square brackets. Given $\sigma \in \Sigma$ and $v_t : \delta^t \in J$, the result of the intervention assigning δ^t to v_t for a datapoint described by σ is denoted as $[\sigma]^{I(v_t:\delta^t)}$ and is interpreted as the corresponding counterfactual. Reference to a specific $\text{wcgph} \in \text{WCGph}$ is needed to obtain the consequences of the intervention. Hence, if there is no $\sigma' \in \Sigma$ outside the square brackets, the sequent expresses a proper counterfactual evaluation. On the contrary, if there is a $\sigma' \in \Sigma$ outside the square brackets, the sequent just says that we consider this datapoint as a candidate to be a counterfactual of the one in the square brackets. As a consequence, the sequents

$$\text{wcgph}, \sigma' \vdash v_t : \delta_p^t \quad \text{wcgph}, [\sigma]^{I(v_s:\delta^s)}, \sigma' \vdash v_t : \delta_p^t$$

denote the same output of the classifier, just considered under different interpretations: the first considers the datapoint described by σ' , the second considers it in relation to σ , as its counterfactual candidate.

Table 18. Rules for the counterfactual, with condition (*) for ν -Cut: $v_i : \delta^i \in \sigma$ and for no set of points M , $v_s \xrightarrow{M} v_i \in \text{cgph}^\blacktriangleright$.

$$\begin{array}{c}
 \frac{\text{wcgph}, at\sigma' \vdash v_i : \delta_p^i}{\text{wcgph}, [at\sigma]^{I(v_s : \delta^s)}, at\sigma' \vdash v_i : \delta_p^i} \text{C-W} \quad \frac{\text{wcgph}, [at\sigma]^{I(v_s : \delta^s)}, at\sigma', v_s : \delta^s \vdash v_i : \delta_p^i}{\text{wcgph}, [at\sigma]^{I(v_s : \delta^s)}, at\sigma' \vdash v_i : \delta_p^i} \text{I-Cut} \\
 \\
 \frac{\text{wcgph}, [at\sigma]^{I(v_s : \delta^s)}, at\sigma', v_i : \delta^i \vdash v_i : \delta_p^i}{\text{wcgph}, [at\sigma]^{I(v_s : \delta^s)}, at\sigma' \vdash v_i : \delta_p^i} \text{v-Cut}^*
 \end{array}$$

Proof-theoretically, we rely on the rules in Table 18 to detect the counterfactual of a datapoint. The rules concern only the atomic cases of datapoints, decisions and interventions. An extension to the logically complex cases is left for further research. *C-W* says that we can regard any datapoint $at\sigma'$ as a counterfactual *candidate* for another datapoint $at\sigma$ under a specific intervention $I(v_s : \delta^s)$ assigning a value (δ^s) to a variable (v_s). The other rules delete judgments from the antecedent of the sequent, and their role is to derive a proper counterfactual sequent from a counterfactual candidate. The rules *I-Cut* and *v-Cut* allow to do this under specific circumstances: we can drop a judgment if it corresponds to the content of our intervention, or if it is already contained in σ and it assigns a value to a variable that is not a (possibly mediated) effect of the variable on which we intervene.

The rules *I-Cut* and *v-Cut* are labeled this way as they are derivable from the rule of *Cut* in Table 16. The (additive) rule of *Cut* cuts away redundant information from the context, hence, it can be used to check how much counterfactual information is available already at the factual level. For example, *v-Cut* is derivable as follows:

$$\frac{\text{wcgph}, [at\sigma]^{I(v_s : \delta^s)} \vdash v_i : \delta_1^i \quad \text{wcgph}, [at\sigma]^{I(v_s : \delta^s)}, at\sigma', v_i : \delta^i \vdash v_i : \delta_p^i}{\text{wcgph}, [at\sigma]^{I(v_s : \delta^s)}, at\sigma' \vdash v_i : \delta_p^i} \text{Cut}$$

Where the first premise is obtainable because of the two conditions on *v-Cut*: $v_i : \delta^i \in at\sigma$ and v_i is not a (possibly indirect) consequence of v_s , the variable of intervention.

With these rules, we can give a method for deriving counterfactual sequents:

Lemma 0.4. *If $\sigma \vdash v_i : \delta_p^i$ is the counterfactual classification of σ' under intervention $I(v_s : \delta^s)$, operated under the assumptions of the weighted causal graph wcgph , then*

$$\text{wcgph}, [\sigma']^{I(v_s : \delta^s)} \vdash v_i : \delta_p^i$$

is derivable from it, by a single application of rule C-W and multiple applications of rules I-Cut and v-Cut.

Proof. If the candidate is a proper counterfactual, then all information in σ' is already contained in $[\sigma]^{I(v_s:\delta^s)}$, so it can be removed by applications of *Cut*. On the other hand, if these applications do not succeed in erasing all the formulas in σ' , it is because it contains information not available in $[\sigma]^{I(v_s:\delta^s)}$.

If the method fails, it gives a witness of its failure, that is, a list of judgments containing the information that is not counterfactually valid:

Lemma 0.5. *If $\sigma' \vdash v_t : \delta_p^t$ is not the counterfactual classification of σ under intervention $I(v_s : \delta^s)$, operated under the assumptions of the weighted causal graph wcgph , then, after a single application of *C-W* that introduces $[\sigma]^{I(v_s:\delta^s)}$ on the antecedent, it is not possible to cut away all the judgments of σ' by multiple applications of rules *I-Cut* and *v-Cut*. Moreover, if the minimal sequent obtainable by applying *I-Cut* and *v-Cut* multiple times is:*

$$\text{wcgph}, [\sigma]^{I(v_s:\delta^s)}, \sigma'' \vdash v_t : \delta_p^t$$

then, σ'' contains all the information in σ' that is not valid in the counterfactual of σ under intervention $I(v_s : \delta^s)$. We call σ'' the witness of the failed counterfactual.

Proof. The rule of *Cut* deletes all the information that is analytically contained in the antecedent. In particular, *I-Cut* and *v-Cut* delete the information that remains valid in the counterfactual of a specific datapoint. Hence, all the information that cannot be deleted is not valid in the counterfactual situation.

Relying on Lemmas 0.4 and 0.5 we can give a formal *criterion* for verifying counterfactual fairness:

Lemma 0.6. *Given a set of AtQuery rules, representing the decisions of a classifier about a datapoint, a weighted causal graph $\text{wcgph} \in \text{WCGph}$ describing the causal relations holding between the features of that datapoint and a protected atomic variable $a \in \text{AtA}$, the classification represented by the derivable sequent $\text{wcgph}, \sigma \vdash v_t : \delta_p^t$ is counterfactually fair regarding feature a iff for every $\alpha \in \mathcal{V}_a$ the sequent $\text{wcgph}, [\sigma]^{I(a:\alpha)} \vdash v_t : \delta_p^t$ is derivable as well. If $\text{wcgph}, [\sigma]^{I(a:\alpha)} \vdash v_t : \delta_p^t$ is not derivable, then the decision is not counterfactually fair and there is $q \neq p$ s.t. $\text{wcgph}, [\sigma]^{I(a:\alpha)} \vdash v_t : \delta_q^t$ is derivable.*

Proof. From the previous lemma, $\text{wcgph}, [\sigma]^{I(a:\alpha)} \vdash v_t : \delta_p^t$ represents the counterfactual sequent of $\text{wcgph}, \sigma \vdash v_t : \delta_p^t$ under an intervention assigning a specific value to a protected attribute. Hence, counterfactual fairness is established by checking that the probabilistic value is not altered under this intervention.

Lemma 0.6 justifies the rules for counterfactual fairness and for its failure in Table 19.

Table 19. Rules for counterfactual fairness. In $\overline{CF}$, we assume that $p \neq q$.

$wcgp, \sigma \vdash v_i : \delta_p^t$	$wcgp, [\sigma]^{I(a:\alpha^1)} \vdash v_i : \delta_p^t$	$\dots$	$wcgp, [\sigma]^{I(a:\alpha^n)} \vdash v_i : \delta_p^t$	CF
$CF[a, \sigma, v_i : \delta^t]$				
$wcgp, \sigma \vdash v_i : \delta_p^t$	$wcgp, [\sigma]^{I(a:\alpha^j)} \vdash v_i : \delta_p^t$	$wcgp, [\sigma]^{I(a:\alpha^k)} \vdash v_i : \delta_q^t$	$\overline{CF}$	
$\overline{CF}[a, \sigma, v_i : \delta^t]$				

Example 7. Given $\sigma = \text{duration} : \beta^1, \text{purpose} : \beta^2, \text{savings_bonds} : \beta^3, \text{employment_duration} : \beta^4, \text{residence_duration} : \beta^5, \text{p_s_and_sex} : \alpha^1$, we can check that the decision represented by the sequent

$$wcgp, \sigma \vdash t_l : \beta_p$$

is CF with respect to $p_s_and_sex$, by deriving

$$wcgp, \sigma' \vdash t_l : \beta_q$$

with $\sigma' = \text{duration} : \beta^1, \text{purpose} : \beta^2, \text{savings_bonds} : \beta^3, \text{employment_duration} : \beta^4, \text{p_s_and_sex} : \alpha^2$ and:

- deriving $wcgp, [\sigma]^{I(p_s_and_sex:\alpha^2)} \vdash t_l : \beta_q$ from $wcgp, \sigma' \vdash t_l : \beta_q$; and
- checking whether $p = q$.

This last step should be repeated for every $\alpha^i \in A_{p_s_and_sex}$.

$wcgp, [\sigma]^{I(p_s_and_sex:\alpha^2)} \vdash t_l : \beta_q$ can be derived from $wcgp, \sigma' \vdash t_l : \beta_q$ by first applying $C-W$, making it a counterfactual candidate for σ under intervention $I(p_s_and_sex : \alpha^2)$, and then applying $I-Cut$ and $v-Cut$ multiple times. The proof concludes $wcgp, [\sigma]^{I(p_s_and_sex:\alpha^2)} \vdash t_l : \beta_q$, because σ' does not contain judgments for the variables *residence_duration*, *property*, *housing*, *job*, *telephone*, *people_liable*, *number_of_credits*, *marital_status*, *age*, *age_group*, *gender*, which are all and only the consequences of $p_s_and_sex$.

Lemma 0.7. Graph-relative measure of counterfactual failure. Let σ'' be the witness of the failed counterfactual $\sigma' \vdash v_t : \delta_{p_t}^t$ of σ under the intervention $I(v_s : \delta^s)$, operated under the assumptions of the fixed reference weighted causal graph $wcgp \in WCGph$. The graph-relative measure of the counterfactual failure is

$$\lambda_{CF}(v_s, \sigma'') = \sum_{v_x \in \text{Var}(\sigma'')} w(v_s, v_x).$$

Proof. Counterfactual failure is established proof-theoretically, independently of the numerical weights, by the impossibility of reducing the counterfactual

candidate to the required counterfactual sequent. The judgments remaining in σ'' constitute the witness of this failure. First, the structural coefficients are composed multiplicatively along each directed causal path from the intervention variable v_s to a variable v_x occurring in the witness. When several directed paths connect v_s to v_x , their strengths are combined according to the alternative-path rules of the calculus. Summing these contributions over the variables in $\text{Var}(\sigma'')$ yields $w(v_s, v_x)$. The sum over the witness variables provides the graph-relative support for the detected counterfactual failure, yielding $\lambda_{\text{CF}}(v_s, \sigma'')$.

The qualitative proof-theoretic judgment that the counterfactual fails is established through *CF*-rules; the graph-relative likelihood $\lambda_{\text{CF}}(v_s, \sigma'')$ assigned to that already established failure in the fixed reference graph; although its edge-level inputs are informed by bootstrap stability, it is neither a bootstrap probability nor the empirical frequency with which the counterfactual failure would occur in graphs learned from other bootstrap samples. Such empirical frequency would require reconstructing and evaluating the counterfactual in every bootstrap graph G_b and is not computed in the present framework. Unlike the undirected connectivity measure used for individual fairness, the present construction uses directed causal paths, because it measures how the consequences of an intervention propagate from v_s to the variables occurring in the witness σ'' . No distinction between active and inactive paths is introduced here.

Example 8. Let us return to Example 7 and evaluate the strength and confidence of a failed counterfactual, applying Lemma 0.7. Consider $\sigma = \text{duration} : \beta^1, \text{purpose} : \beta^2, \text{savings_bonds} : \beta^3, \text{employment_duration} : \beta^4, \text{residence_duration} : \beta^5, \text{property} : \beta^6, \text{p_s_and_sex} : \alpha^1$. Moreover, consider $\sigma' = \text{duration} : \beta^1, \text{purpose} : \beta^2, \text{savings_bonds} : \beta^3, \text{employment_duration} : \beta^4, \text{residence_duration} : \beta^5, \text{property} : \beta^6, \text{p_s_and_sex} : \alpha^2$, as a counterfactual candidate for σ under intervention that assigns α^2 at *p_s_and_sex*.

The failure of counterfactual is proved by observing that $\text{cgph}, [\sigma]^{I(\text{p_s_and_sex} : \alpha^2)} \vdash t_l : \beta_q$ cannot be derived from $\text{cgph}, \sigma' \vdash t_l : \beta_q$. Indeed, after application of *C-W* to $\text{cgph}, \sigma' \vdash t_l : \beta_q$, the minimal sequent obtainable by applying *I-Cut* and *v-Cut* multiple times is:

$$\text{cgph}, [\sigma]^{I(\text{p_s_and_sex} : \alpha^2)}, \sigma'' \vdash t_l : \beta_q$$

With $\sigma'' = \text{residence_duration} : \beta^5, \text{property} : \beta^6$, which is the witness of failed counterfactual.

As a consequence, we can use σ'' to calculate the strength and confidence of the failure. There are just two directed paths from the variable of intervention to those in the witness:

$$\text{p_s_and_sex} \rightarrow \begin{smallmatrix} -0.172 \\ 0.50 \end{smallmatrix} \text{residence_duration} \quad \text{p_s_and_sex} \xrightarrow{\text{residence_duration}} \begin{smallmatrix} -0.027 \\ 0.50 \end{smallmatrix} \text{property}$$

Notice that these are directly derivable by the rules in Table 13, from the immediate causal relations:

$$\text{p_s_and_sex} \triangleright \begin{smallmatrix} -0.172 \\ 0.50 \end{smallmatrix} \text{residence_duration} \quad \text{residence_duration} \triangleright \begin{smallmatrix} -0.161 \\ 0.55 \end{smallmatrix} \text{property}$$

Hence, the graph-relative strength of the counterfactual failure is $S_{CF} = -0.199$, while its graph-relative likelihood measure in the fixed weighted reference graph is $\lambda_{CF} = 0.50$. The latter value is not the probability that the counterfactual will fail in another bootstrap-generated graph.

Metatheoretical Properties

The following properties characterize the proof-theoretic behavior of TNDPQ over the symbolic representation introduced in the previous section. Throughout this section, reasoning is performed over proof environments obtained from the graph injection together with the current observational context. Unless otherwise stated, the following results assume that the structural rule *Weakening* is admissible, meaning that Individual Fairness is satisfied by the corresponding proof objects.

Definition 0.22. Proof Environment. A *proof environment* Γ is a set of sequents, consisting of the conclusions of instances of the rule *AtQuery* of Table 14, representing the outputs of a trained classifier on a dataset:

$$\Gamma = \{WCGph, \Sigma \vdash J^P\}$$

where

- $WCGph$ is the weighted causal environment obtained by graph injection as by Definition 0.18;
- Σ is the current observational context containing facts and assumptions used during the proof;
- J^P is the decision of the classifier in the specific context represented by Σ .

A proof environment represents all the information that TNDPQ takes as input regarding a specific classifier and its intended application. A sequent $WCGph, \Sigma \vdash J^P$ is derivable whenever there exists a finite proof tree whose leaves are sequents belonging to Γ and whose internal nodes are obtained by repeated applications of the TNDPQ inference rules.

Definition 0.23. Proof State. Let the set of sequents $\Gamma = \{WCGph, \Sigma \vdash J^P\}$ be a proof environment. We will call it the *initial proof state* W_0 . A *proof state* is any finite set of TNDPQ sequents W such that $W_0 \subseteq W$, and every sequent

$$WCGph, \Sigma \vdash J^P \in W \setminus W_0$$

has a finite derivation in the TNDPQ calculus whose premises belong to W . A proof state is said to be *closed* if no additional sequent can be derived from W by further applications of the inference rules.

Definition 0.24. Deductive Closure. Let

$$W_0 = \{WCGph, \Sigma \vdash J^P\}$$

be a proof environment constituting an initial proof state. The *deductive closure* of W_0 , denoted by $W^\star$, is the smallest closed proof state containing W_0 . Equivalently,

$$W^\star = \{ \text{WCGph}, \Sigma \vdash J^P \mid \Gamma \vdash \text{WCGph}, \Sigma \vdash J^P \},$$

that is, the set of all TNDPQ sequents admitting a finite derivation from the proof environment according to the inference rules of the calculus. Equivalently, $W^\star$ is the least fixed point of the inference relation induced by the TNDPQ proof rules.

The inference rules of TNDPQ are defined only over well-typed sequents and preserve the typing discipline of the language. The following lemma establishes that no derivation can produce an ill-formed sequent.

Lemma 0.8. Well-Formedness Preservation. *Let $\Gamma = \{ \text{WCGph}, \Sigma \vdash J^P \}$ be a proof environment. If $\Gamma \vdash \text{WCGph}, \Sigma \vdash J^P$, then the sequent $\text{WCGph}, \Sigma \vdash J^P$ is well formed according to the syntax and typing rules of the TNDPQ calculus.*

Proof. Sketch. The proof proceeds by induction on the height of the derivation tree. For the base case, every sequent derivable by an instance of *AtQuery* consists of an atomic context *at σ* and an injected causal environment *wcgp* on the left and an atomic probabilistic judgment on the right. By construction of the graph injection (Definition 0.18) and by the definition of proof environments, all such judgments are syntactically valid and satisfy the typing constraints of the calculus. For the inductive step, assume that every premise of an inference rule is well formed. Each inference rule of TNDPQ is defined over well-typed premises and constructs only judgments whose syntactic structure and typing are prescribed by the grammar of the language. Consequently, the conclusion produced by the rule is itself well formed. Therefore, every sequent derivable in TNDPQ is syntactically and type-correct.

Theorem 0.1. Existence and Uniqueness of the Deductive Closure. *Let $\Gamma = \{ \text{WCGph}, \Sigma \vdash J^P \}$ be a finite proof environment. Given that the TNDPQ inference system consists of finitely many finitary inference rules, there exists a unique deductive closure $W^\star$ associated with Γ . Moreover,*

$$W^\star = \{ \text{WCGph}, \Sigma \vdash J^P \mid \Gamma \vdash \text{WCGph}, \Sigma \vdash J^P \},$$

is the least closed proof state containing the initial proof state $W_0 = \Gamma$.

Proof. Starting from the initial proof state $W_0 = \Gamma$, construct the sequence $W_0 \subseteq W_1 \subseteq \dots$ by repeatedly applying every applicable TNDPQ inference rule. Since every inference rule is finitary and every application introduces only derivable TNDPQ sequents, the sequence is monotone. Because the proof environment is finite and the language of TNDPQ admits only finitely many well-formed sequents over the symbols occurring in Γ , no new sequents can be generated indefinitely modulo logical equivalence. Consequently, there exists an index n such

that $W_n = W_{n+1}$. The proof state W_n is therefore closed under the inference rules and coincides with the deductive closure $W^\star$. Uniqueness follows from minimality. If W' is another closed proof state containing W_0 , then every sequent generated during the construction of $W^\star$ also belongs to W' . Hence $W^\star \subseteq W'$. Therefore $W^\star$ is the unique least closed proof state containing the initial proof state.

Definition 0.25. P-Consistency. Let $\Gamma = \{\text{WCGph}, \Sigma \vdash \text{J}^P\}$ be a finite proof environment. Γ is p-consistent iff:

1. For every well-formed $\text{wcgph}, \sigma \vdash V_i : \Delta^i$, there is at most one value $p \in [0, 1]$ such that $\text{wcgph}, \sigma \vdash V_i : \Delta_p^i$ occurs in Γ .
2. Let $P(\text{wcgph}, \sigma \vdash v_i : \delta^{ij})$ refer to the $p \in [0, 1]$ s.t. $\text{wcgph}, \sigma \vdash v_i : \delta_p^{ij}$ occurs in Γ . For every choice of $v_1, \dots, v_n \in \text{AtV}$, let F be the set of functions $f : v_i \rightarrow \delta^{ij}$ that associate each v_i with one of its values in Val_{v_i} . For every $\sigma \in \Sigma$:

$$\sum_{f \in F} P(\text{wcgph}, \sigma \vdash v_1 : f(v_1)) \times \dots \times P(\text{wcgph}, \sigma, v_1 : f(v_1), \dots, v_{n-1} : f(v_{n-1}) \vdash v_n : f(v_n)) = 1$$

3. For every $\sigma \in \Sigma$, $V_i, V_j \in \text{V}$, and $\Delta^m, \Delta^n \in \text{Val}$, it holds that:

$$P(\text{wcgph}, \sigma, V_i : \Delta^m \vdash V_j : \Delta^n) \times P(\text{wcgph}, \sigma \vdash V_i : \Delta^m) = P(\text{wcgph}, \sigma, V_j : \Delta^n \vdash V_i : \Delta^m) \times P(\text{wcgph}, \sigma \vdash V_j : \Delta^n)$$

Theorem 0.2. P-Consistency Preservation. Let Γ be a p-consistent proof environment and let $W^\star$ be its deductive closure. Then $W^\star$ is p-consistent.

Proof. The proof proceeds as follows:

1. If Γ is p-consistent, then the values p occurring in J^P are probabilities, i.e. they behave according to the probability axioms;
2. The rules of TNDPQ preserve p-consistency, because they are probabilistically valid.

Let us start with the first point. P-consistency requires the conjunction of three properties. The first property just requires that there is at most one probability associated with each sequent. The second property requires that at least one of the elementary events of the sample space occurs with certainty. In TNDPQ we express conditional probabilities, so elementary events need to be “reconstructed”: given a set of atomic variables and a $\sigma \in \Sigma$, an elementary event is the assignment of an atomic value to each of them, and corresponds to a sequent like:

$$\text{wcgph}, \sigma \vdash \langle v_1, \dots, v_n \rangle : (\delta^{1j_1} \times \dots \times \delta^{n j_n})_p$$

Since in proof environments there are only atomic sequents, we cannot directly use this sequent, but we have to rely on conditional probabilities instead, multiplying the probabilities $p_1, \dots, p_n$ of the sequents:

$$\text{wcgph}, \sigma \vdash v_1 : \delta_{p_1}^{1_{j_1}} \quad \dots \quad \text{wcgph}, \sigma, v_1 : \delta^{1_{j_1}} \dots, v_{n-1} : \delta^{n-1_{j_{n-1}}} \vdash v_n : \delta_{p_n}^{n_{j_n}}$$

From the structure of elementary events, we can use the set F of functions $f : v_i \rightarrow \delta^{i_j}$ that associate each v_i with one of its values in Val_{v_i} to detect all the elementary events. The probability axiom requiring additivity of the probabilities of mutually exclusive events is dealt with by the rules of TNDPQ for disjunction. The third property of p-consistency is needed because TNDPQ starts with conditional probabilities rather than the probability of the intersection of events. The usual definition of Conditional probability

$$P(A \mid B) =_{\text{def}} \frac{P(A \cap B)}{P(B)}$$

entails that $P(A \mid B) \times P(B) = P(B \mid A) \times P(A)$, since both correspond to $P(A \cap B)$. In TNDPQ, we start with a conditional probability built in the structure of sequents, and we use it to define the probability of the conjunction. Hence, we need to impose that conditional probabilities are mutually coherent. This excludes situations such as:

$$\begin{array}{ll} \text{wcgph}, \sigma, V_i : \Delta^m \vdash V_j : \Delta_{1/2}^n & \text{wcgph}, \sigma \vdash V_i : \Delta_{1/3}^m \\ \text{wcgph}, \sigma, V_j : \Delta^n \vdash V_i : \Delta_{1/2}^m & \text{wcgph}, \sigma \vdash V_j : \Delta_{1/4}^n \end{array}$$

which are problematic, as they lead to incoherence when conjunctions are derived:

$$\frac{\text{wcgph}, V_i : \Delta^m \vdash V_j : \Delta_{1/2}^n \quad \text{wcgph} \vdash V_i : \Delta_{1/3}^m}{\text{wcgph} \vdash \langle V_i, V_j \rangle : (\Delta^m \times \Delta^n)_{1/6}} \text{I}\times 1 \quad \frac{\text{wcgph}, V_j : \Delta^n \vdash V_i : \Delta_{1/2}^m \quad \text{wcgph} \vdash V_j : \Delta_{1/4}^n}{\text{wcgph} \vdash \langle V_i, V_j \rangle : (\Delta^m \times \Delta^n)_{1/8}} \text{I}\times 2$$

The second point, stressing that the rules of TNDPQ are probabilistically valid, can be checked by carefully inspecting the rules. Since the single steps are straightforward and uninteresting, they are omitted. From these two points, and the trivial observation that p-consistency holds when the p of the judgments behaves according to the axioms of probabilities, it follows that the deductive closure of a p-consistent proof environment is p-consistent.

These two theorems guarantee that the next synthesis phase operates on well-defined, consistent finite proof-objects.

Phase IV: Constraint Synthesis via Proof Analysis

The objective of the synthesis phase is to compile fairness proofs into logical constraints. The synthesis process operates over admissible proof certificates, thereby preserving the logical justification of every generated constraint.

Constraint Language

The output of the synthesis phase is a logical theory expressed in a first-order constraint language $\mathcal{L}_C$. The language is generated inductively from probability terms and logical formulas. Throughout this section, let $\mathcal{A}$ denote the finite set of protected attributes identified during the fairness analysis, and let Y denote the target prediction variable. For every protected attribute $A \in \mathcal{A}$, let $\text{Dom}(A)$ denote the finite domain of admissible values of A .

Definition 0.26. Terms. The set of probability terms is inductively defined by

$$P ::= P(Y \mid \sigma) \mid P(Y \mid \sigma_{A \leftarrow a}),$$

where Y denotes the prediction, σ denotes a factual datapoint and $\sigma_{A \leftarrow a}$ the corresponding counterfactual datapoint obtained by intervening on the protected attribute A .

Definition 0.27. Constraint Language. The formulas of the constraint language $\mathcal{L}_C$ are generated by the grammar

$$\varphi ::= \text{Equal}(P, P) \mid \text{ValidCF}(A, a_i, a_j, \sigma) \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid (\varphi \rightarrow \varphi) \mid \forall x \varphi \mid \exists x \varphi.$$

The intended meaning of the atomic predicates is as follows.

- $\text{Equal}(P_1, P_2)$ asserts that terms P_1 and P_2 have equal predictive probability;
- $\text{ValidCF}(A, a_i, a_j, \sigma)$ asserts that the intervention changing the protected attribute A from the factual value a_i to the counterfactual value a_j defines a valid counterfactual situation for the datapoint σ .

Definition 0.28. Constraint Theory. A *logical constraint theory* is a finite set $\mathcal{T} = \{\varphi_1, \dots, \varphi_n\} \subseteq \mathcal{L}_C$ such that each φ_i is synthesized from an admissible TNDPQ proof and represents a fairness obligation to be enforced during parameter learning.

Definition 0.29. Proof Diagnostic Report. A *proof diagnostic report* is a finite set $\mathcal{D} = \{d_1, \dots, d_m\}$, whose elements record proof attempts that fail to satisfy the admissibility conditions required for fairness certification. Each diagnostic entry consists of components $d_i = (c_i, \sigma_i, r_i, e_i)$, where

- c_i is the probabilistic constraint involved in the failed derivation;
- σ_i is the context in which the certification failed;
- r_i is the structural rule whose application failed;
- e_i is an explanatory annotation describing the reason for the failure.

The diagnostic report provides an explanation of unsuccessful certification attempts without introducing additional formulas into the constraint language.

Definition 0.30. Constraint Synthesis Operator. During the construction of the deductive closure W^* , the proof procedure retains a finite proof forest containing the derivations of the fairness judgments occurring in such closure. Let Π^* denote the finite proof forest actually produced by the proof-search procedure. Constraint synthesis is defined by $\mathcal{S} : W^* \longrightarrow (\mathcal{T}, \mathcal{D})$, where

- $\mathcal{T}$ is the synthesized logical constraint theory;
- $\mathcal{D}$ is the diagnostic report.

The synthesis operator is realized by $\mathcal{S} = (\mathbf{R} \circ \mathcal{N}, \mathcal{R}) \circ \mathcal{E}$, where

- $\mathcal{E} : W^* \rightarrow \Pi^*$ accesses the proof forest retained during the construction of the deductive closure;
- $\mathbf{R}$ compiles admissible proofs into logical constraints;
- $\mathcal{N}$ normalizes the synthesized constraint theory;
- $\mathcal{R}$ records failed proof attempts together with the structural reasons preventing their certification.

Successful proof objects contribute logical constraints, whereas failed proof attempts contribute diagnostic information that can be exploited during subsequent model revision. A pseudocode implementation of the operator is provided in Algorithm 1.

Algorithm 1: Proof-Guided Constraint Synthesis

Input: Deductive closure W^*
Output: Constraint theory $\mathcal{T}$, diagnostic report $\mathcal{D}$
 $\mathcal{T} \leftarrow \emptyset$;
 $\mathcal{D} \leftarrow \emptyset$;
 $\Pi^* \leftarrow \mathcal{E}(W^*)$;
foreach $\pi \in \Pi^*$ **do**
 if $\pi \in \Pi_A$ **then**
 identify the compilation rule $\rho \in \mathbf{R}$;
 $\mathcal{T} \leftarrow \mathcal{T} \cup \rho(\pi)$;
 else
 if $\pi \in \Pi_{NA}$ **then**
 $\mathcal{D} \leftarrow \mathcal{D} \cup \mathcal{R}(\pi)$;
 $\mathcal{T} \leftarrow \mathcal{N}(\mathcal{T})$;
return $(\mathcal{T}, \mathcal{D})$

Admissible Proofs

Constraint synthesis compiles only admissible proof objects from the deductive closure into the logical constraint theory. Admissibility depends on the structural

conditions of the rule certifying fairness; derivations of the same TNDPQ judgment may therefore yield different constraints when certified by different rules. Failed proof attempts are recorded in the diagnostic report together with the structural reasons preventing certification.

Definition 0.31. Admissible Proof. Let $\pi : \Gamma \vdash J^P$ be a TNDPQ proof. The proof π is said to be *admissible for constraint synthesis* if the fairness property is established through one of

- rule *IF* from Equation 2, which establishes individual fairness;
- rule *IF_i* from Equation 5, which establishes intersectional fairness;
- rule *CF* from Table 19 which establishes counterfactual fairness.

The set of admissible proofs is denoted by Π_A . Each proof $\pi \in \Pi_A$ is compiled into one or more logical constraints of the synthesized constraint theory $\mathcal{T}$.

Compilation Rules

The synthesis operator translates admissible TNDPQ proofs into logical constraints through two compilation rules. The first generates invariance constraints from proofs containing *IF* for individual fairness and extends directly to *IF_i* for intersectional fairness. The second generates counterfactual fairness constraints from proofs containing *CF*.

Definition 0.32. Weakening Compilation Rule. Let $\pi : \Gamma \vdash J^P$ be an admissible TNDPQ proof certified by the occurrence of the rule *IF*, hence the admissibility of the rule *W*. The compilation rule

$$\rho_W : \Pi_A \cap \Pi_W \longrightarrow \mathcal{L}_C$$

maps the proof π to a first-order logical constraint expressing invariance of the predictive distribution over all contexts in which the structural rule *W* is admissible, and hence *IF* valid. The generated constraint preserves the fairness invariant established by the proof and is subsequently enforced during parameter learning.

Proposition 1. Compilation of Individual Fairness. Let $\pi \in \Pi_A \cap \Pi_W$ be an admissible proof establishing fairness with respect to the value assignment $V_u : \Delta^{u_j}$ and set of datapoints Σ . Then the compilation rule ρ_W produces the constraint

$$\forall \Delta^{u_j} \in \text{Dom}(V_u), \text{Equal}(P(Y \mid \sigma_\pi), P(Y \mid \sigma_\pi, V_u : \Delta^{u_j})).$$

The generated constraint encodes the invariant of individual fairness for the certified context σ_π , and any demonstrably equivalent context.

The compilation of Proposition 1 generalizes easily to the case $V_u = (A_1, \dots, A_k)$, which corresponds to the constraints produced by any π in which there is a valid instance of *IF_i* and hence certifies intersectional fairness.

Definition 0.33. Cut Compilation Rule. Let $\pi : \Gamma \vdash J^P$ be an admissible TNDPQ proof certified by the occurrence of the rule CF , hence the admissibility of the rule Cut . The compilation rule

$$\rho_{CF} : \Pi_A \cap \Pi_{CF} \longrightarrow \mathcal{L}_C$$

maps the proof π to a first-order implication whose antecedent certifies the validity of the counterfactual derivation and whose consequent enforces equality of the corresponding predictive distributions.

Proposition 2. Compilation of Counterfactual Fairness. *Let $\pi \in \Pi_A \cap \Pi_{CF}$ be an admissible proof establishing counterfactual fairness with respect to the protected attribute $A \in \mathcal{A}$. Then the compilation rule $\rho_{CF}(\pi)$ produces the first-order constraint*

$$\forall a_i, a_j \in \text{Dom}(A), \left(\text{ValidCF}(A, a_i, a_j, \sigma_\pi) \rightarrow \text{Equal}\left(P(Y \mid \sigma_{\pi(A=a_i)}), P(Y \mid \sigma_{\pi(A=a_j)})\right) \right).$$

The generated constraint encodes the certified invariant of counterfactual fairness for the certified context σ_π with respect to its counterpart with $A \leftarrow a_j$, and any demonstrably equivalent context and counterpart.

Non-admissible Proofs

Diagnostics from failed proofs guide repair: semantic diagnostics are compiled into differentiable objectives for parameter retraining, while structural diagnostics suggest changes to the learned causal graph, followed by renewed parameter estimation.

Definition 0.34. Non-Admissible Fairness Proof. Let $\pi : \Gamma \vdash J$ be a TNDPQ proof object associated with a fairness-certification task. The proof π is said to be *non-admissible for constraint synthesis* when the failure of the relevant fairness property is formally established through one of the negative fairness rules, namely:

- rule $\overline{\text{IF}}$ from Equation 3, which establishes the failure of individual fairness;
- rule $\overline{\text{IF}}_i$ from Equation 6, which establishes the failure of intersectional fairness;
- rule $\overline{\text{CF}}$ from Table 19, which establishes the failure of counterfactual fairness.

The set of non-admissible fairness proofs is denoted by Π_{NA} . Each proof $\pi \in \Pi_{\text{NA}}$ constitutes a negative fairness certificate.

The class Π_{NA} is not defined as the complement $\Pi^\star \setminus \Pi_A$: proof objects unrelated to fairness certification belong to neither class, while tasks for which neither a positive fairness rule nor its negative counterpart is derivable remain

unresolved. The diagnostic operator $\mathcal{R} : \Pi_{\text{NA}} \longrightarrow \mathcal{D}, \mathcal{D} = \{d_1, \dots, d_m\}$ maps each non-admissible proof object to an entry recording the failed obligation, its context, the rule establishing failure, and the proof-theoretic explanation. As specified in Definition 0.29, entries contain semantic and structural information supporting two complementary repair strategies:

1. *Semantic repair* translates probabilistic equalities or independence requirements violated by the current model into differentiable objectives for updating its parameters.
2. *Structural repair* modifies the causal graph according to the diagnostic explanation, then re-estimates probabilistic parameters. Diagnosed obstacles include omitted conditioning variables, inadmissible conditioning on colliders, and other violations of the encoded causal assumptions.

Example of Constraint Synthesis

We illustrate the synthesis procedure by compiling the admissible proof of Example 3. Besides synthesizing logical fairness constraints, the procedure also produces a diagnostic report collecting proof attempts that cannot be certified. Recall the observational context

$$\sigma = \left\{ \begin{array}{l} \text{residence_duration} : \beta^1, \\ \text{employment_duration} : \beta^2, \\ \text{credit_history} : \beta^3, \\ \text{credit_risk} : \beta^4, \\ \text{job} : \beta^5, \\ \text{saving/bonds} : \beta^6, \\ \text{status} : \beta^7 \end{array} \right\}.$$

and let g denote the protected attribute **gender**. Since the conditional independence judgment $g \parallel_{\text{wcgph}}^\sigma t_label$ holds, the following admissible application of W is derivable:

$$\frac{\text{wcgph}, \sigma \vdash t_label : \beta_p \quad g \parallel_{\text{wcgph}}^\sigma t_label}{\text{wcgph}, \sigma, g : \alpha \vdash t_label : \beta_p} W$$

Applying the compilation rule ρ_W produces the fairness constraint

$$\forall \alpha \in \text{Dom}(g), \text{Equal}(P(t_label \mid \sigma), P(t_label \mid \sigma, g : \alpha)),$$

which is inserted into the synthesized constraint theory $\mathcal{T}$. The resulting formula requires that extending the admissible observational context with any value of the protected attribute leaves the predictive distribution of the target variable unchanged. Consider next the reduced observational context

$$\sigma' = \left\{ \begin{array}{l} \text{employment_duration} : \beta^2, \\ \text{credit_history} : \beta^3, \\ \text{credit_risk} : \beta^4, \\ \text{job} : \beta^5, \\ \text{saving/bonds} : \beta^6 \end{array} \right\}.$$

In this case, the conditional independence premise of W cannot be established because the variables `residence_duration` and `status`, required to block every causal path between `gender` and `predicted_label`, are absent. Consequently, no admissible proof exists and no fairness constraint is generated. Instead, the failed proof attempt is recorded in the diagnostic report together with the structural rule involved and the reason for the failure. Finally, consider the extended context

$$\sigma'' = \sigma \cup \{\text{installment_plans} : \beta^8\}.$$

Although σ'' contains every assignment of σ , the additional variable `installment_plans` opens a collider path connecting `personal_status_and_sex` to `predicted_label`. Hence the conditional independence judgment $g \parallel_{\text{wcgph}}^{\sigma''} t_label$ is no longer derivable. As before, the synthesis procedure produces no logical constraint, while the failed certification is recorded in the diagnostic report together with its structural explanation. The synthesis procedure therefore produces the pair $(\mathcal{T}, \mathcal{D})$, where

$$\mathcal{T} = \{\forall \alpha \in \text{Dom}(g), \text{Equal}(P(t_label \mid \sigma), P(t_label \mid \sigma, g : \alpha))\},$$

and the diagnostic report contains two failed certification attempts,

$$\mathcal{D} = \left\{ (c_{\sigma'}, \sigma', W, \text{missing conditioning variables}), \right. \\ \left. (c_{\sigma''}, \sigma'', W, \text{collider introduced by installment_plans}) \right\}.$$

where e.g.

$$c_{\sigma'} = \{\forall \alpha \in \text{Dom}(g), \text{Equal}(P(t_label \mid \sigma'), P(t_label \mid \sigma', g : \alpha))\}$$

This example illustrates the complementary roles of the two synthesis outputs. The logical constraint theory contains only fairness obligations justified by admissible TNDPQ proofs, while the diagnostic report preserves the proof-theoretic explanation of unsuccessful certification attempts. During subsequent model revision, these diagnostics provide precise information about the structural conditions preventing fairness certification and can therefore guide feature selection, graph refinement, or additional data collection.

Phase V: Diagnostic-Guided Repair and Retraining

Phase IV produces the constraint theory $\mathcal{T}$ from admissible proofs (Definition 0.28) and the diagnostic report $\mathcal{D}$ from diagnosed failures (Definition 0.29). Constraints in $\mathcal{T}$ express certified properties to preserve, while $C(\mathcal{D}) = \{c_1, \dots, c_n\}$ contains violated probabilistic constraints extracted by C to guide retraining. Structural diagnostics that cannot yield quantitative obligations on model parameters are handled by the structural repair procedure below.

Let θ_0 denote the parameters before repair and $\theta \in \Theta$ a candidate update, such as neural-network weights, Bayesian-network parameters, or statistical coefficients. Since parameter retraining affects only some constraints in $\mathcal{T}$, we distinguish model-dependent constraints from structural constraints.

Definition 0.35. Model-dependent and structural constraints. Let Γ denote a proof environment, and write $(\theta, \Gamma) \models \varphi$ when the model parameterized by θ satisfies $\varphi \in \mathcal{T}$ relative to Γ . The set of model-dependent constraints is

$$\mathcal{T}_\theta = \{ \varphi \in \mathcal{T} : \exists \theta \in \Theta \text{ such that } (\theta_0, \Gamma) \models \varphi \text{ and } (\theta, \Gamma) \not\models \varphi \}.$$

Thus, $\mathcal{T}_\theta$ contains exactly those certified constraints whose satisfaction may be lost through parameter retraining while the proof environment remains fixed. The complementary set of structural constraints is $\mathcal{T}_{\text{str}} = \mathcal{T} \setminus \mathcal{T}_\theta$ whose validity depends instead on the causal graph, the conditioning contexts, or other components of the proof environment.

This distinction is relative to a fixed causal and proof-theoretic environment: a constraint may belong to $\mathcal{T}_\theta$ during parameter retraining but require complete recertification if the underlying graph, context, or proof assumptions are subsequently modified.

The Retraining Problem

The retraining problem is to repair the violations identified in $C(\mathcal{D})$ while preserving every model-dependent invariant in $\mathcal{T}_\theta$. At the highest level of abstraction, this is expressed as

$$\theta^* \in \arg \min_{\theta \in \Theta} L_{\text{pred}}(\theta) + \lambda_{\text{rep}} L_{\text{rep}}(\theta)$$

subject to $\theta \models \varphi_i$, for every $\varphi_i \in \mathcal{T}_\theta$ and where

- $L_{\text{pred}}(\theta)$ is the predictive loss of the underlying learning model;
- $L_{\text{rep}}(\theta)$ aggregates the violations of the obligations in $C(\mathcal{D})$;
- $\lambda_{\text{rep}} > 0$ determines the trade-off between predictive performance and repair whenever $C(\mathcal{D}) \neq \emptyset$. When $C(\mathcal{D}) = \emptyset$, $L_{\text{rep}}(\theta) = 0$, and the value of λ_{rep} is immaterial.

The two outputs of Phase IV therefore play different roles. The constraints in $\mathcal{T}_\theta$ determine the admissible region of optimization: they specify which model updates preserve the properties already certified. The obligations in $C(\mathcal{D})$ determine the direction of repair within that region: they identify the discrepancies that retraining should reduce.

Invariant-Preservation Violations and the Admissible Region

Let $\mathcal{T}_\theta = \{\varphi_1, \dots, \varphi_m\}$, and let $\Sigma_{\varphi_i} \neq \emptyset$ denote the finite set of admissible contexts over which the invariant φ_i is evaluated. The set Σ_{φ_i} contains the contexts to which the universally quantified constraint applies, as determined by logical equivalence with respect to the current context of analysis. Suppose that $\varphi_i \in \mathcal{T}_\theta$ has the form

$$\varphi_i \equiv \forall \sigma \in \Sigma_{\varphi_i}, \text{Equal} \left(P(t_label \mid \sigma), P'(t_label \mid \sigma') \right).$$

The formula expresses the probabilistic relation certified by the admissible proof and is evaluated using P_θ and P'_θ . The prime denotes the distribution in the modified context σ' . Both distributions refer to the same parameterized model. For every $\varphi_i \in \mathcal{T}_\theta$, let τ_{φ_i} denote the context transformation prescribed by φ_i , and let $\sigma' = \tau_{\varphi_i}(\sigma)$. Depending on the fairness notion, τ_{φ_i} may modify the protected-attribute assignments for an individual or intersectional comparison, or encode the intervention required for a counterfactual comparison.

Definition 0.36. Total-variation discrepancy for invariant preservation. For every $\varphi_i \in \mathcal{T}_\theta$ and $\sigma \in \Sigma_{\varphi_i}$, the total-variation discrepancy associated with φ_i is

$$\Delta_{\varphi_i}(\sigma, \theta) = d_{\text{TV}} \left(P_\theta(t_label \mid \sigma), P'_\theta(t_label \mid \sigma') \right).$$

For a discrete outcome t_label , this is

$$\Delta_{\varphi_i}(\sigma, \theta) = \frac{1}{2} \sum_{y \in \text{Dom}(t_label)} \left| P_\theta(t_label = y \mid \sigma) - P'_\theta(t_label = y \mid \sigma') \right|.$$

For a binary outcome t_label , total variation reduces to

$$\Delta_{\varphi_i}(\sigma, \theta) = \left| P_\theta(t_label = 1 \mid \sigma) - P'_\theta(t_label = 1 \mid \sigma') \right|.$$

Total variation applies uniformly to binary and finite categorical outcomes, is symmetric, and takes values in the bounded interval $[0, 1]$. For continuous outcomes, a fixed probability metric, e.g. Wasserstein distance, may be used instead. Let $\varepsilon_{\varphi_i} \geq 0$ be the preservation tolerance for φ_i . Since φ_i derives from an admissible proof, under the adopted evaluation semantics, $\Delta_{\varphi_i}(\sigma, \theta_0) \leq \varepsilon_{\varphi_i}, \forall \sigma \in \Sigma_{\varphi_i}$. The normalized preservation violation is

$$V_{\text{pres}}(\varphi_i, \theta) = \frac{1}{|\Sigma_{\varphi_i}|} \sum_{\sigma \in \Sigma_{\varphi_i}} \left(\max \left\{ 0, \Delta_{\varphi_i}(\sigma, \theta) - \varepsilon_{\varphi_i} \right\} \right)^2.$$

Thus,

$$V_{\text{pres}}(\varphi_i, \theta) = 0 \iff \forall \sigma \in \Sigma_{\varphi_i}, \Delta_{\varphi_i}(\sigma, \theta) \leq \varepsilon_{\varphi_i}.$$

A parameter vector preserves all certified model-dependent invariants, over their respective evaluation domains, when $\forall \varphi_i \in \mathcal{T}_\theta, V_{\text{pres}}(\varphi_i, \theta) = 0$. Accordingly, the invariant-preserving admissible region is

$$\Theta_{\mathcal{T}} = \{\theta \in \Theta : V_{\text{pres}}(\varphi, \theta) = 0 \text{ for every } \varphi \in \mathcal{T}_\theta\}.$$

Since the original model satisfies the constraints compiled from its admissible proofs, $\theta_0 \in \Theta_{\mathcal{T}}$ relative to the same evaluation domains and tolerances. The set $\Theta_{\mathcal{T}}$ therefore contains exactly the candidate parameter vectors that preserve the certified model-dependent invariants and determines the admissible region of optimization.

Note that preservation does not require the repaired model to reproduce every prediction of the original model. In particular, it does not generally require $d_{\text{TV}}(P_{\theta_0}, P_\theta) = 0$. Instead, both sides of each invariant are evaluated under the candidate model θ . What must be preserved is the probabilistic relation certified by φ_i , rather than the complete predictive behaviour of the original model.

Repair Violations and the Direction of Optimization

The violated obligations in $C(\mathcal{D})$ guide model updates within the invariant-preserving region $\Theta_{\mathcal{T}}$. For each $c_i \in C(\mathcal{D})$, let $\Sigma_{c_i} \neq \emptyset$ be the finite set of admissible evaluation contexts, derived from the diagnostic context through logical equivalence relative to the current analysis. Suppose that c_i has the form

$$c_i \equiv \forall \sigma \in \Sigma_{c_i}, \text{Equal}(P(t_label \mid \sigma), P'(t_label \mid \sigma')).$$

We denote by τ_{c_i} the context transformation prescribed by i , and let $\sigma' = \tau_{c_i}(\sigma)$. As above, depending on the fairness notion, τ_{c_i} may modify the protected-attribute assignments for an individual or intersectional comparison, or encode the intervention required for a counterfactual comparison.

Definition 0.37. Total-variation discrepancy for a repair obligation. For every $c_i \in C(\mathcal{D})$ and $\sigma \in \Sigma_{c_i}$, the total-variation discrepancy associated with c_i is

$$\Delta_{c_i}(\sigma, \theta) = d_{\text{TV}}(P_\theta(t_label \mid \sigma), P'_\theta(t_label \mid \sigma')).$$

For a discrete outcome t_label , this is

$$\Delta_{c_i}(\sigma, \theta) = \frac{1}{2} \sum_{y \in \text{Dom}(t_label)} |P_\theta(t_label = y \mid \sigma) - P'_\theta(t_label = y \mid \sigma')|.$$

For a binary outcome t_label , total variation reduces to

$$\Delta_{c_i}(\sigma, \theta) = \left| P_{\theta}(t_label = 1 \mid \sigma) - P'_{\theta}(t_label = 1 \mid \sigma') \right|.$$

Let $\varepsilon_{c_i} \geq 0$ denote the admissible tolerance associated with c_i . Since c_i represents a failed obligation, the original model satisfies $\Delta_{c_i}(\sigma, \theta_0) > \varepsilon_{c_i}$ for at least one context $\sigma \in \Sigma_{c_i}$. The logical violating domain of c_i is

$$\mathcal{V}_{c_i}(\theta) = \left\{ \sigma \in \Sigma_{c_i} : \Delta_{c_i}(\sigma, \theta) > \varepsilon_{c_i} \right\}.$$

The most direct proof-theoretic measure of violation is therefore

$$V_{\log}(c_i, \theta) = |\mathcal{V}_{c_i}(\theta)| = \sum_{\sigma \in \Sigma_{c_i}} \mathbf{1}_{\Delta_{c_i}(\sigma, \theta) > \varepsilon_{c_i}},$$

which counts the counterexamples to the universally quantified repair obligation. Although this measure directly reflects the logical semantics, its indicator function is discontinuous and is therefore unsuitable for gradient-based optimization. To obtain a compatible optimization objective, the indicator is replaced by a continuous relaxation. Let $f : [0, \infty) \rightarrow [0, \infty)$ be monotone increasing, satisfy $f(0) = 0$, and be chosen to be differentiable or subdifferentiable. A general normalized relaxation is

$$\tilde{V}_{\text{rep}}(c_i, \theta) = \frac{1}{|\Sigma_{c_i}|} \sum_{\sigma \in \Sigma_{c_i}} f\left(\max\left\{0, \Delta_{c_i}(\sigma, \theta) - \varepsilon_{c_i}\right\}\right).$$

Different choices of f yield different notions of differentiable or subdifferentiable violation while preserving the same underlying proof-theoretic interpretation. Typical choices include power functions ($f(x) = x^p, p \geq 1$), which recover absolute and squared penalties (for $p = 1$ and $p = 2$, respectively), as well as smooth approximations of the logical indicator. Using the squared relaxation, the normalized repair violation is

$$V_{\text{rep}}(c_i, \theta) = \frac{1}{|\Sigma_{c_i}|} \sum_{\sigma \in \Sigma_{c_i}} \left(\max\left\{0, \Delta_{c_i}(\sigma, \theta) - \varepsilon_{c_i}\right\}\right)^2.$$

Thus,

$$V_{\text{rep}}(c_i, \theta) = 0 \iff \Delta_{c_i}(\sigma, \theta) \leq \varepsilon_{c_i}, \forall \sigma \in \Sigma_{c_i}.$$

When $C(\mathcal{D}) \neq \emptyset$, the complete repair loss is defined as the weighted mean

$$L_{\text{rep}}(\theta) = \frac{\sum_{c_i \in C(\mathcal{D})} w_{c_i} V_{\text{rep}}(c_i, \theta)}{\sum_{c_i \in C(\mathcal{D})} w_{c_i}},$$

where $w_{c_i} \geq 0$ and $\sum_{c_i \in C(\mathcal{D})} w_{c_i} > 0$. When $C(\mathcal{D}) \neq \emptyset$, we require $\sum_{c_i \in C(\mathcal{D})} w_{c_i} > 0$, with positive weight for every mandatory obligation. Uniform weights treat obligations equally; application-specific weights may prioritize legally protected attributes, severe violations, or safety-critical decisions. Normalization over $\sum_{c_i}$ prevents larger context sets from receiving greater weight, while normalization by the total weight makes the aggregate invariant under uniform rescaling of the weights. When $C(\mathcal{D}) = \emptyset$, we set $L_{\text{rep}}(\theta) = L_{\text{fair}}(\theta) = 0$.

Having defined both the invariant-preserving admissible region and the repair loss, the initial retraining problem can now be instantiated as

$$\theta^* \in \arg \min_{\theta \in \Theta} L_{\text{pred}}(\theta) + \lambda_{\text{rep}} L_{\text{rep}}(\theta)$$

subject to $V_{\text{pres}}(\varphi, \theta) = 0$, for every $\varphi \in \mathcal{T}_\theta$. Invariant preservation is imposed as a condition on admissible repairs rather than as an objective that may be traded against predictive performance or the repair obligations. This construction follows the general distinction between optimization objectives and explicitly enforced constraints used in constrained learning (27) and in provable model-repair methods (28; 29).

When direct optimization over $\Theta_{\mathcal{T}}$ is unavailable, preservation violations may be added as weighted penalties. These allow trade-offs with predictive performance and repair, so preservation is not guaranteed. Accepting the retrained model therefore requires verifying that every constraint in $\mathcal{T}_\theta$ remains satisfied. Any incompatibility between preservation and repair must be reported as a conflict, without sacrificing certified invariants.

Structural Repair

Constraints in $\mathcal{T}_{\text{str}}$ need not enter $\Theta_{\mathcal{T}}$, since parameter retraining alone cannot affect their satisfaction. They remain valid while the causal graph, conditioning contexts, and proof assumptions remain fixed. Structural changes require repeating the affected causal-discovery and proof-theoretic phases and recertifying the relevant constraints. Structural diagnostics identify graph or context conditions that block certification and propose candidate repairs rather than differentiable objectives. Changes to topology require new evidence, background knowledge, or explicit causal assumptions and must proceed through causal rediscovery.

For a missing conditioning variable Z , the immediate repair revises the query context to $\sigma^+ = \sigma \cup \{Z = z\}$, and reevaluates the fairness judgment. Conditioning on Z is distinct from adding it to the classifier’s inputs. If feature inclusion is separately adopted as a learning-level repair, its effects require empirical evaluation and renewed certification.

Collider and path-dependent independence diagnostics may similarly motivate revised conditioning contexts, feature restrictions, or justified background knowledge for renewed discovery. Each intervention is applied at the appropriate proof, learning, or discovery level, followed by recomputation of the affected

graph structure, d -separation relations, and fairness judgments. Success is not guaranteed: a structural repair succeeds only when the reconstructed obligation becomes derivable and every still-applicable invariant is recertified in the revised proof environment.

Empirical Validation

As an illustrative validation procedure, we consider the structural repair strategy prescribed by the failed application of the *Weakening* rule discussed in Example 3. The diagnostic report identifies the omission of `residence_duration` from the conditioning context as the structural reason preventing the derivation of the desired fairness certificate. According to the causal graph, removing this variable opens an active dependency between the protected attribute `gender` and the prediction target, thereby invalidating the conditional independence required by the proof. The prescribed repair therefore consists of restoring the missing conditioning variable before learning again the predictive model. To evaluate the effectiveness of this intervention, two classifiers are trained on the German Credit dataset. The first classifier, denoted by M_{-res} , is trained using the feature set

`{employment_duration, credit_history, job, savings/bonds, status},`

where `residence_duration` has been removed. The second classifier, M_{+res} , is trained using

`{employment_duration, credit_history, job,
savings/bonds, status, residence_duration},`

thus restoring the conditioning variable required by the proof. Both classifiers are implemented as Random Forest models trained under identical conditions. Each forest contains 500 decision trees, uses balanced class weights to compensate for the unequal distribution of the target classes, imposes no fixed maximum tree depth, and requires at least two observations in each terminal leaf. Since the German Credit variables are represented by integer-valued category codes, all predictors are treated as categorical and transformed through one-hot encoding before training. Model performance is estimated using five-fold stratified group cross-validation, with the same fold assignment used for M_{-res} and M_{+res} . Instances sharing the same values for the features common to both models are assigned to the same fold, thereby preventing repeated synthetic patterns from appearing simultaneously in the training and validation partitions. For each observation, the prediction used in the subsequent fairness analysis is therefore an out-of-fold prediction produced by a model that was not trained on that observation or on an identical common-feature pattern. A fixed random seed of 42 and a probability threshold of 0.5 are used throughout. Consequently,

Table 20. Structural effect of restoring `residence_duration`.

Gender	Instances	Changed predictions	Repair rate
Male	622	83	13.34%
Female	378	56	14.81%
Total	1000	139	13.90%

the only difference between the two training procedures is the inclusion of `residence_duration` in M_{+res} .

Since the failed *Weakening* rule concerns the conditioning structure rather than individual predictions, the first validation measures the extent to which restoring `residence_duration` modifies the learned decision function. Given the predictions $\hat{Y}_{-res}$ and $\hat{Y}_{+res}$ produced by the two classifiers on the same test set, we define the empirical repair rate

$$R_{struct} = \frac{|\{x : \hat{Y}_{-res}(x) \neq \hat{Y}_{+res}(x)\}|}{N},$$

where N denotes the number of evaluated observations. Across the 1000 observations of D_{aug} , restoring `residence_duration` changes the prediction for 139 observations, corresponding to a repair rate of 13.9%. The changes are distributed across the protected groups as reported in Table 20, indicating that the repaired conditioning structure modifies the learned classifier for both genders.

The second validation evaluates whether the structural repair also improves established fairness measures. Since the diagnostic predicts that restoring the missing conditioning variable should reduce the spurious dependence between gender and the prediction, we compare the two classifiers using standard group fairness metrics. Table 21 reports the resulting comparison. Restoring `residence_duration` consistently reduces all measured group disparities. In particular, the statistical parity difference decreases from 3.55% to 3.13%, while the gaps in true positive rate, false positive rate, equalized odds and average odds are substantially reduced. These improvements are accompanied by only a modest reduction in predictive performance, illustrating the expected fairness–accuracy trade-off.

Taken together, the two analyses support the structural diagnosis produced by the proof system. Restoring the conditioning variable prescribed by the failed *Weakening* rule produces a measurable modification of the learned classifier and consistently reduces standard fairness disparities. The experiment therefore provides empirical evidence that structural repairs derived from failed proof obligations can improve fairness (even in the absence of direct optimization of fairness-specific loss functions).

Table 21. External fairness validation after restoring `residence_duration`. Lower values indicate smaller disparities between the protected groups.

Measure	M _{+res}	M _{-res}	Difference
Statistical parity difference	3.13%	3.55%	−0.42 pp
TPR gap	1.01%	11.33%	−10.32 pp
FPR gap	1.70%	6.52%	−4.82 pp
Equalized odds gap	1.70%	11.33%	−9.63 pp
Average odds gap	1.36%	8.93%	−7.57 pp
Accuracy gap	5.39%	11.00%	−5.61 pp
Accuracy	72.4%	74.3%	−1.9 pp
Balanced accuracy	63.90%	67.98%	−4.08 pp
F1-score	48.31%	55.61%	−7.30 pp

Conclusions and Future Work

This work introduced a neuro-symbolic framework integrating causal discovery, probabilistic typed natural deduction, machine-checked verification, diagnosis, and retraining to certify and repair fairness in classifiers. Learned causal models provide symbolic premises for formally verified proof obligations. Successful derivations yield fairness certificates, while diagnosed failures identify structural or probabilistic obstacles and guide changes to the causal model, conditioning context, or learning objective. Fairness verification thus informs model adaptation beyond retrospective bias assessment. Future work will extend this methodology to agentic AI and errors beyond bias. Agents acquire information, reason, plan, act, and revise their representations. Their certification must therefore address the causal organization of data, reasoning, and interventions over time. We identify three connected research directions.

First, agents should reason causally about both their data and their own operations. Causal models must represent information sources, reasoning steps, intermediate conclusions, and actions. Extending the proof system with annotations for the reliability, fairness, and causal effects of these operations would help distinguish failures originating in data, causal assumptions, or reasoning procedures.

Second, abstraction should identify reusable causal and inferential patterns across datasets, environments, and reasoning episodes. The central challenge is to formalize when different structures instantiate the same schema and which transformations permit certificates to transfer between instances. This would support generalization across tasks while making the assumptions governing certificate transfer explicit.

Third, planning should enable agents to revise learned schemas in response to new evidence or normative constraints and select interventions that satisfy certification requirements. Integrating causal intervention with belief revision would support changes to factual beliefs, causal dependencies, inferential rules,

and goal priorities. Plans would comprise justified sequences of epistemic, structural, and operational interventions.

The long-term objective is a formally grounded architecture for adaptive agents that can represent and verify their decision processes, generalize certified patterns across contexts, and revise those patterns through controlled interventions.

Acknowledgements

The authors used OpenAI’s GPT-5.6 Sol and Astra models, accessed through ChatGPT Work, as an assistive tool during the preparation and revision of this article. Its use included: language and stylistic revision; refinement (but not formulation) of the presentation of formal definitions, notation, algorithms, and pseudocode; $\text{\LaTeX}$ formatting and troubleshooting; and consistency checks of the formalism across the protocol. The model was also used to test human-produced code templates for causal-graph extraction, graph pruning, classifier training, and empirical analysis, as well as for verification of bibliographic metadata.

All material submitted to model analysis was treated as provisional and was critically reviewed, corrected, and, where necessary, rejected or independently verified by the authors. The authors retained full control over the conceptual framework, methodological and experimental choices, formal definitions and proofs, implementation decisions, interpretation of the results, and selection and verification of the cited literature. No model output was treated as an autonomous source of evidence. The authors take full responsibility for the accuracy, originality, and final content of the article.

The authors acknowledge financial support from the Italian Ministry of University and Research through the project “sMARTEST: Simulation of Probabilistic Systems for the Age of the Digital Twin” (MUR–PRIN 2022, project no. 20223E8Y4X) and through the project “Departments of Excellence 2023–2027”, awarded to the Department of Philosophy “Piero Martinetti” of the University of Milan.

References

- [1] Hardt M, Price E and Srebro N. Equality of opportunity in supervised learning. *NeurIPS* 2016; .
- [2] *Fairness through awareness*, 2012. URL <https://arxiv.org/abs/1104.3913>.
- [3] Kusner M, Loftus J, Russell C et al. Counterfactual fairness. In *NeurIPS*.
- [4] Coraglia G, D’Asaro FA, Genco FA et al. Brioxalkemy: a bias detecting tool. In Boella G, D’Asaro FA, Dyoub A et al. (eds.) *Proceedings of the 2nd Workshop on Bias, Ethical AI, Explainability and the role of Logic and Logic Programming co-located with the 22nd International Conference of the Italian Association for Artificial Intelligence (AI*IA 2023), Rome, Italy, November 6, 2023, CEUR Workshop Proceedings*, volume 3615. CEUR-WS.org, pp. 44–60. URL <https://ceur-ws.org/Vol-3615/paper4.pdf>.
- [5] Pearl J. *Causality: Models, Reasoning, and Inference*. 2 ed. Cambridge University Press, 2009.

- [6] d’Avila Garcez A, Besold TR, de Raedt L et al. Neural-symbolic learning and reasoning: Contributions and challenges. *AAAI Spring Symposium* 2015; .
- [7] Spirtes P, Glymour C and Scheines R. *Causation, Prediction, and Search*. 2 ed. MIT Press, 2001.
- [8] Termine A and Primiero G. Causality problems in machine learning systems. In Illari P and Russo F (eds.) *The Routledge Handbook of Causality and Causal Methods*. Routledge, 2025. pp. 325–341. DOI:10.4324/9781003528937-37.
- [9] Kilbertus N, Rojas-Carulla M, Parascandolo G et al. Avoiding discrimination through causal reasoning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS’17, Red Hook, NY, USA: Curran Associates Inc. ISBN 9781510860964, p. 656–666.
- [10] Nabi R and Shpitser I. Fair inference on outcomes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.
- [11] Makhoul K, Zhioua S and Palamidessi C. When causality meets fairness: A survey. *Journal of Logical and Algebraic Methods in Programming* 2024; 141: 101000. DOI:<https://doi.org/10.1016/j.jlamp.2024.101000>. URL <https://www.sciencedirect.com/science/article/pii/S2352220824000543>.
- [12] Ceragioli L and Primiero G. Trustworthiness preservation by copies of machine learning systems. *International Journal of Approximate Reasoning* 2026; 192: 109638. DOI:<https://doi.org/10.1016/j.ijar.2026.109638>. URL <https://www.sciencedirect.com/science/article/pii/S0888613X26000149>.
- [13] D’Asaro FA, Genco FA and Primiero G. Checking trustworthiness of probabilistic computations in a typed natural deduction system. *J Log Comput* 2025; 35(6). DOI: 10.1093/LOGCOM/EXAF003. URL <https://doi.org/10.1093/logcom/exaf003>.
- [14] Kubyshkina E and Primiero G. A possible worlds semantics for trustworthy non-deterministic computations. *Int J Approx Reason* 2024; 172: 109212. DOI:10.1016/J.IJAR.2024.109212. URL <https://doi.org/10.1016/j.ijar.2024.109212>.
- [15] Ceragioli L and Primiero G. A proof system with causal labels (part i): checking individual fairness and intersectionality. In *Proceedings of the 7th International Workshop on Artificial Intelligence and Formal Verification, Logic, Automata, and Synthesis (OVERLAY 2025)*. pp. 173–179.
- [16] Ceragioli L and Primiero G. A proof system with causal labels (part ii): checking counterfactual fairness. In *Proceedings of the 7th International Workshop on Artificial Intelligence and Formal Verification, Logic, Automata, and Synthesis (OVERLAY 2025)*. pp. 181–187.
- [17] Manganini C and Primiero G. *Reasoning With and About Bias*. Cham: Springer Nature Switzerland. ISBN 978-3-031-77892-6, 2024. pp. 127–154. DOI:10.1007/978-3-031-77892-6_7. URL https://doi.org/10.1007/978-3-031-77892-6_7.
- [18] Manganini C, Corsi EA and Primiero G. Data speak but sometimes lie: A game-theoretic approach to data bias and algorithmic fairness. *Int J Approx Reason* 2026; 190: 109608. DOI:10.1016/J.IJAR.2025.109608. URL <https://doi.org/10.1016/j.ijar.2025.109608>.

- [19] Hitzler P, Eberhart A, Ebrahimi M et al. Neuro-symbolic approaches in artificial intelligence. *National Science Review* 2022; 9(6): nwac035. DOI:10.1093/nsr/nwac035.
- [20] d'Avila Garcez A and Lamb LC. Neuro-symbolic ai: The 3rd wave. *Artificial Intelligence Review* 2023; 56(12): 14013–14035. DOI:10.1007/s10462-023-10448-w.
- [21] Marra G, Dumančić S, Manhaeve R et al. From statistical relational to neurosymbolic artificial intelligence: A survey. *Artificial Intelligence* 2024; 328: 104062. DOI:10.1016/j.artint.2023.104062.
- [22] Badreddine S, d'Avila Garcez A, Serafini L et al. Logic tensor networks. *Artificial Intelligence* 2022; 303: 103649. DOI:<https://doi.org/10.1016/j.artint.2021.103649>. URL <https://www.sciencedirect.com/science/article/pii/S0004370221002009>.
- [23] Greco G, Alberici F, Palmonari M et al. Declarative encoding of fairness in logic tensor networks. In Gal K, Nowé A, Nalepa GJ et al. (eds.) *Proceedings of the 26th European Conference on Artificial Intelligence (ECAI 2023)*, *Frontiers in Artificial Intelligence and Applications*, volume 372. IOS Press, pp. 908–915. DOI:10.3233/FAIA230360.
- [24] Heilmann X, Manganini C, Cerrato M et al. A neurosymbolic approach to counterfactual fairness. *Neurosymbolic Artificial Intelligence* 2026; 2: 1–19. DOI:10.1177/29498732261443184.
- [25] Xu J, Zhang Z, Friedman T et al. A semantic loss function for deep learning with symbolic knowledge. In *Proceedings of the 35th International Conference on Machine Learning, Proceedings of Machine Learning Research*, volume 80. PMLR, pp. 5502–5511. URL <https://proceedings.mlr.press/v80/xu18h.html>.
- [26] Fischer M, Balunovic M, Drachslers-Cohen D et al. DL2: Training and querying neural networks with logic. In *Proceedings of the 36th International Conference on Machine Learning, Proceedings of Machine Learning Research*, volume 97. PMLR, pp. 1931–1941. URL <https://proceedings.mlr.press/v97/fischer19a.html>.
- [27] Cotter A, Jiang H, Gupta M et al. Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. *Journal of Machine Learning Research* 2019; 20(172): 1–59. URL <https://jmlr.org/papers/v20/18-616.html>.
- [28] Sotoudeh M and Thakur AV. Provable repair of deep neural networks. In *Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation*. Association for Computing Machinery, pp. 588–603. DOI:10.1145/3453483.3454064. URL <https://doi.org/10.1145/3453483.3454064>.
- [29] Fayyazi A, Xiao Y, Nuzzo P et al. Efficient counterexample-guided fairness verification and repair of neural networks using satisfiability modulo convex programming. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, pp. 367–375. DOI:10.24963/ijcai.2025/42. URL <https://doi.org/10.24963/ijcai.2025/42>.

- [30] Hofmann H. Statlog (German Credit Data). UCI Machine Learning Repository, 1994. DOI: <https://doi.org/10.24432/C5NC77>.
- [31] Renero J. Causalexplain: A library to infer causal-effect relationships from tabular data, 2026. URL <https://github.com/renero/causalexplain>.
- [32] Renero J, Maestre R and Ochoa I. ReX: Causal discovery based on machine learning and explainability techniques. *Pattern Recognition* 2026; 172: 112491. DOI:10.1016/j.patcog.2025.112491.
- [33] Zheng Y, Huang B, Chen W et al. Causal-learn: Causal discovery in python. *Journal of Machine Learning Research* 2024; 25(60): 1–8. DOI:10.48550/arXiv.2307.16405.