

A Proof System with Causal Labels (Part II): checking Counterfactual Fairness

Leonardo Ceragioli^{1,*†}, Giuseppe Primiero^{1,*†}

¹University of Milan, Via Festa del Perdono 7, Milano, 20122, Italy

Abstract

In this article we propose an extension to the typed natural deduction calculus **TNDPQ** to model verification of counterfactual fairness in probabilistic classifiers. This is obtained formulating specific structural conditions for causal labels and checking that evaluation is robust under their variation.

Keywords

Fairness, Probabilistic Logic, Causal Graphs, Structural Rules

1. Introduction

The calculus **TPTND** (*Trustworthy Probabilistic Typed Natural Deduction* [1, 2]) is designed to evaluate *post-hoc* the trustworthiness of the behavior of opaque systems. The system is implemented for verification of dataframes in the tool BRIO [3, 4]. In [5], we introduced **TNDPQ** (*Typed Natural Deduction for Probabilistic Queries*), a variation of the previous system in which a probabilistic output is associated to a target variable when a Data Point – consisting of a list of values attributions for a set of variables – is provided. In this paper, we extend this system with tools to verify counterfactual fairness of classifiers.

We start with a formal definition of classifiers. Let \mathcal{A} be a set of protected variables a_1, \dots, a_n , \mathcal{X} be a (disjoint) set of non-protected variables x_1, \dots, x_n and t be a target variable. Moreover, let \mathcal{V}_{a_i} be a set of values $\alpha^{i_1}, \alpha^{i_2}, \dots, \alpha^{i_j}$ that a_i can receive, \mathcal{V}_A the set of all \mathcal{V}_{a_i} , \mathcal{V}_{x_i} be a set of values $\beta^{i_1}, \beta^{i_2}, \dots, \beta^{i_j}$ that x_i can receive, \mathcal{V}_X the set of all \mathcal{V}_{x_i} , and \mathcal{V}_t the set $\delta^1, \delta^2, \dots, \delta^j$ of values that t can receive. Let us use v_1, \dots, v_n to denote elements of $\mathcal{A} \cup \mathcal{X}$ (that is, variables regardless of their protected or unprotected status), and $\gamma^{i_1}, \dots, \gamma^{i_j}$ to denote the values that v_i can receive. We use $a_i : \alpha^{i_j}$ (respectively $x_i : \beta^{i_j}$) to express the *judgment* that variable a_i receives value α^{i_j} (respectively, variable x_i receives value β^{i_j}), and $t : \delta_{p_1}^1, \dots, \delta_{p_j}^j$ to express the *probabilistic judgment* that $\delta^1, \dots, \delta^j$ are all the possible values that variable t can receive and that, for $1 \leq k \leq j$, it receives value δ^k with probability p_k .¹ We use $\mathcal{J}^{\mathcal{A}}$ for the set of all the judgments about protected variables, $\mathcal{J}^{\mathcal{X}}$ for the set of all the judgments about non-protected variables, and $\mathcal{J}^{\mathcal{P}}$ for the set of all probabilistic judgments. Moreover, we use $\sigma^{\mathcal{A}}$ to express a set of judgments about protected variables such that each element of \mathcal{A} receives at most one value, $\sigma^{\mathcal{X}}$ to express a set of judgments about non-protected variables such that each element of \mathcal{X} receives at most one value, $\Sigma^{\mathcal{A}}$ to refer to the set of all $\sigma^{\mathcal{A}}$, and $\Sigma^{\mathcal{X}}$ to refer to the set of all $\sigma^{\mathcal{X}}$. σ is used to express the union of a $\sigma^{\mathcal{A}}$ and a $\sigma^{\mathcal{X}}$, and Σ is used to refer to the set of all σ . More formally:

$$\Sigma^{\mathcal{A}} =_{def} \{ \sigma^{\mathcal{A}} \subseteq \mathcal{J}^{\mathcal{A}} \mid \forall i (a_i : \alpha^{i_l} \in \sigma^{\mathcal{A}} \wedge a_i : \alpha^{i_m} \in \sigma^{\mathcal{A}} \rightarrow l = m) \}$$

7th International Workshop on Artificial Intelligence and fOrmal VERification, Logic, Automata, and sYnthesis, OVERLAY 2025, Oct 25 – 26, Bologna, Italy

*Corresponding author.

[†]These authors contributed equally.

✉ leonardo.ceragioli@unimi.it (L. Ceragioli); giuseppe.primiero@unimi.it (G. Primiero)

🆔 0000-0001-5250-9720 (L. Ceragioli); 0000-0003-3264-7100 (G. Primiero)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

¹ We assume that the values for t (also for the elements of \mathcal{A} and \mathcal{X} but this is irrelevant here) are all mutually exclusive, and so $\sum_{k=1}^j p_k = 1$. Note that we make no assumption regarding whether $t \in \mathcal{A} \cup \mathcal{X}$ and so on whether $\mathcal{V}_t = \mathcal{V}_{a_i}$ or $\mathcal{V}_t = \mathcal{V}_{x_i}$ for some i .

$$\begin{aligned}\Sigma^{\mathcal{X}} &=_{def} \{\sigma^{\mathcal{X}} \subseteq \mathcal{J}^{\mathcal{X}} \mid \forall i(x_i : \beta^{i_l} \in \sigma^{\mathcal{X}} \wedge x_i : \beta^{i_m} \in \sigma^{\mathcal{X}} \rightarrow l = m)\} \\ \Sigma &=_{def} \{\sigma^{\mathcal{A}} \cup \sigma^{\mathcal{X}} \mid \sigma^{\mathcal{A}} \in \Sigma^{\mathcal{A}} \wedge \sigma^{\mathcal{X}} \in \Sigma^{\mathcal{X}}\}\end{aligned}$$

A classifier $\hat{\mathcal{F}} \in \hat{\mathcal{F}}$ is a function from Σ to $\mathcal{J}^{\mathcal{P}}$, where each $\sigma \in \Sigma$ describes a Data Point, that is what we know about a subject, and the probabilistic judgment $t : \delta_{p_1}^1, \dots, \delta_{p_j}^j$ in $\mathcal{J}^{\mathcal{P}}$ represents the output of the classifier regarding the probability distribution of the possible values for the target variable t .

TNDPQ is a proof system working with sequents describing the result of queries for classifiers. More precisely, each classifier $\hat{\mathcal{F}}$ is characterized by a set of ground sequents of the form:²

$$\sigma \mid \sim t : \delta_p \quad (1)$$

For readability reasons, the sequents focus on only one possible value for the target variable at a time. In [5] we show how to extend **TNDPQ** with sequents working with logically complex sequents – possibly non-atomic variables receiving possibly non-atomic values. As an example, the following sequent expresses the probability that a non-white 27 years old man who is married or divorced and has a gross annual income of 65000 receives a loan:

$$Age : 27, Gen. : m, MS : married + divorced, Etn. : white^\perp, GAI : 65K \mid \sim Loan : yes_{0.60}$$

TNDPQ was initially designed to investigate the preservation of trustworthiness under the composition of logically simpler queries and then extended with causal labels to verify individual and intersectional fairness for a probabilistic classifier via structural properties, see [6]. In this paper, we focus only on the atomic fragment and provide a *criterion* to verify counterfactual fairness for a probabilistic classifier.

2. Counterfactual Fairness

Counterfactual fairness requires that a subject would not have been treated differently had their protected attributes been different [7]. Formally, it can be defined as follows:

Definition 2.1 (Counterfactual Fairness (CF)). An algorithm is counterfactually fair regarding a protected variable a if, given a Data Point σ describing an actual individual, the algorithm gives the same output to both σ and to the Data Point σ' describing how the individual would have been, had the protected variable a received a different value.

As an example, we could wonder whether the probability of receiving a loan in the previous example would have still been 60%, had the subject been a woman? If we do not consider the connections between the features, this question just corresponds to whether or not the following sequent is derivable in the system:³

$$Age : 27, Gen. : f, MS : married + divorced, Etn. : white^\perp, GAI : 65K \mid \sim Loan : yes_{0.60}$$

However, when causal relations are taken into account, it is a trivial observation that gender influences other features (both directly and indirectly). For example, both through objective physical differences and prejudices, gender influences job opportunities, and therefore the counterpart of a subject having a different gender would probably not have the same *GAI*. This makes the individuation of the Data Point σ' a lot more complicated.

Hence, assessing counterfactual situations **CF** requires some more precise formal tools. The usual approaches are: possible worlds semantics [8] and causal models [9]. In the following we choose the second method.

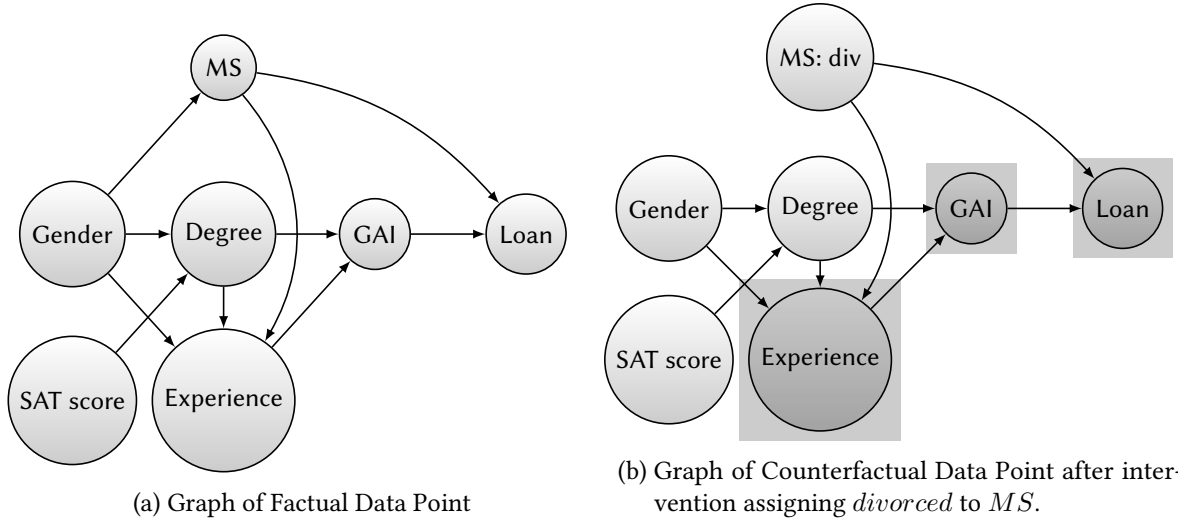


Figure 1: Graphs corresponding to factual and counterfactual Data Points. The squares in figure (b) surround variables that depend on the variable we intervene on, and which for this reason cannot be used to decide the target.

3. Causal Relations

We first consider how counterfactual situations are dealt with in causal models, and then internalize both causal relations and the characterization of counterfactuals in our calculus **TNDPQ**. For the purposes of this work, we define causal graphs as follows:

Definition 3.1 (Causal Graph). A causal graph is an acyclic directed graph with nodes representing events (variables receiving values) and edges representing immediate causal relations.

By closing edges under transitivity, we obtain the notion of mediate cause. For purely formal reasons, we close the notion of cause under reflexivity as well. The usual extension of deterministic causal graphs with functions to compute the value of a node on the basis of those of all the immediate parent nodes is here expressed by sequents like in equation 1.⁴

As already shown, to capture counterfactuals one cannot just change the protected attribute and leave all other variables fixed: the properties of the subject that do not depend on the protected variable need to be identified and kept fixed. For this, the usual approach (followed, for example, in [9, 7]) is to rely on the distinction between exogenous and endogenous variables: exogenous variables represent attributes that have no direct cause in the graph, while endogenous ones represent their consequences. By keeping fixed the values of exogenous variables (possibly with the exclusion of the protected variable), we make sure that the causal graph represents the counterfactual situation. In fact, exogenous variables cannot be consequences of the protected variable. Moreover, under the assumptions of causal models, it is possible to calculate the values of the endogenous variables from those of the exogenous ones.

According to this usual approach, the protected variable can be either exogenous (such as gender) or endogenous (such as marital status, which, for example, depends on gender: there are more widows than widowers). When the protected variable is endogenous, we erase all the edges that enter it, since we want to assign its value *ad arbitrium*.

In summary, to capture counterfactual situations, the usual approach prescribes to intervene on the graph as follows [9, 7]:

1. *impose* some value to the protected variable;

²Technically, we should add a subscript in the equation specifying the classifier we are focusing on. However, this will not be needed here, since we will not compare outputs of different classifiers.

³Note that this would make counterfactual and individual fairness indistinguishable from one another.

⁴However, note that in causal models the function does not assign value in a probabilistic way and probabilities come in play only at a later stage. On the contrary, our equations are probabilistic in the strictest sense.

2. *keep* all the (other) exogenous variables fixed;
3. *erase* all the edges that enter in the protected variable;
4. *calculate* the values of the endogenous variables (particularly of the target), using those of the exogenous variables.

The third point is relevant only when the protected variable is not exogenous; otherwise, it is vacuously satisfied. Now, to check **CF** we just need to control whether the resulting value of the target variable is the same.

We slightly modify this approach to apply it to ML classifiers. The assignment of a value to the target variable depends on the selection of a set of exogenous variables, so we have to be sure to work with an adequate set of such variables. Moreover, while in causal models we can require to have a sufficiently vast set of exogenous variables, we cannot ask the same regarding the set of entries of a classifier, which we cannot change. For this reason, instead of relying only on the exogenous variables, we will use all and only the variables that are not (direct or indirect) effects of the protected one, and ignore all the others. This will allow to use all the factual information that remains valid in the counterfactual situation in order to derive the counterfactual classification. An example of the graphs corresponding to a factual and a counterfactual Data Point is shown in figure 1.

4. Adding Counterfactuals to TNDPQ

To express the fact that a Data Point is the counterfactual of another, we need to internalize both causal relations and interventions in our calculus **TNDPQ**. For this purpose, we use the methodology of labeled calculi [10, 11]. First, we extend the language with the following relational predicates for variables and expressions for interventions on Data Points, as already introduced in [6]:

Immediate Causal Relations $v_i \triangleright v_j =_{def} v_i$ is an immediate cause of v_j .

Mediate Causal Relations $v_i \blacktriangleright^M v_j =_{def} v_i$ is a mediate cause of v_j , with intermediate causes M .

Intervention on Data Point $[\triangleright_{\hat{f}}, \sigma]I(a : \alpha) =_{def}$ an intervention assigning the value α to variable a is operated on the Data Point σ and its associated graph $\triangleright_{\hat{f}}$.⁵

Then, we reformulate **TNDPQ** sequents by extending their left-hand side:

$$\triangleright_{\hat{f}}, \sigma \mid \sim t : \delta_p \quad [\triangleright_{\hat{f}}, \sigma]I(a : \alpha) \mid \sim t : \delta_q \quad (2)$$

Let us use Var_{σ} to indicate the set of variables that occur in σ . We use $\triangleright_{\hat{f}}$ to indicate all the immediate causal relations among features in the classifier. $\blacktriangleright_{\hat{f}}$ denotes all the mediate causal relations in the resulting graph and is derivable as the closure of $\triangleright_{\hat{f}}$ under reflexivity and transitivity. We will use $\triangleright'_{\hat{f}}$ and $\blacktriangleright'_{\hat{f}}$ respectively for different sets of direct and indirect causal relations. Hence, the first sequent of equation 2 is a sequent that gives an output for the target t in the actual situation (that is, σ), also specifying the immediate causal relations that hold between the features of the classifier ($\triangleright_{\hat{f}}$), while the second is a hypersequent that gives an output for the target t in the counterfactual situation resulting from $\triangleright_{\hat{f}}, \sigma$ by the intervention that assigns α to a . We call a the variable of intervention. Although $\triangleright_{\hat{f}}$ is not actually used by the classifier to evaluate t , it is relevant in the calculus to check whether a Data Point is the counterfactual of another.

Example 1 (Factual and Counterfactual sequents). *The following sequents express, respectively, that the probability of receiving a loan for a 27 years old person with a gross annual income of 40.000 is 60%, and that it would have been 50% had them been 35 years old:*

$$Age \triangleright MS, Age \triangleright GAI, Age \triangleright Loan, GAI \triangleright Loan, Age : 27, GAI : 40K \mid \sim Loan : yes_{0.60}$$

⁵Notice that we focus on interventions on protected variables, although interventions on unprotected variables could also be studied.

Table 1

Rules for the counterfactual, with the following conditions: (*) $v_k \neq a$; (**) v_i s.t. $v_i : \gamma^i \in \sigma'$ and for no set of points M , $a \blacktriangleright^M v_i \in \blacktriangleright'_{\hat{f}}$.

$$\begin{array}{c}
\frac{\triangleright_{\hat{f}}, \sigma \mid \sim t : \delta_p}{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha), \triangleright_{\hat{f}}, \sigma \mid \sim t : \delta_p} \text{C-Weakening} \quad \frac{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha), \triangleright_{\hat{f}}, \sigma, a : \alpha \mid \sim t : \delta_p}{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha), \triangleright_{\hat{f}}, \sigma \mid \sim t : \delta_p} \text{I-Cut} \\
\frac{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha), \triangleright_{\hat{f}}, v_i \triangleright v_k, \sigma \mid \sim t : \delta_p}{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha), \triangleright_{\hat{f}}, \sigma \mid \sim t : \delta_p} \triangleright\text{-Cut}^* \quad \frac{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha), \triangleright_{\hat{f}}, \sigma, v_i : \gamma^i \mid \sim t : \delta_p}{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha), \triangleright_{\hat{f}}, \sigma \mid \sim t : \delta_p} v\text{-Cut}^{**}
\end{array}$$

$$[Age \triangleright MS, Age \triangleright GAI, Age \triangleright Loan, GAI \triangleright Loan, Age : 27, GAI : 40K] I(Age : 35) \mid \sim Loan : yes_{0.50}$$

While sequents describing actual decisions of a classifier, such as the one on the left of equation 2, are assumptions of **TNDPQ**, those describing counterfactual decisions, such as the one on the right of the equation, are derivable. To derive them, we start with a plausible sequent for the counterfactual, which can be obtained using only the features that do not depend on the variable of intervention to run the classifier:

$$\triangleright'_{\hat{f}}, \sigma' \mid \sim t : \delta_q$$

Let us call this the counterfactual candidate. Then, we apply the rule C-Weakening in table 1, adding $[\triangleright_{\hat{f}}, \sigma] I(a : \alpha)$. The resulting hypersequent $[\triangleright_{\hat{f}}, \sigma] I(a : \alpha) \triangleright'_{\hat{f}}, \sigma' \mid \sim t : \delta_q$ can be interpreted as saying that the classifier assigns probability q to the value assignment $t : \delta$ for the Data Point σ' , which we regard as a counterfactual candidate for the Data Point σ after intervention assigning to a the value α . The formulas $\triangleright_{\hat{f}}$ and $\triangleright'_{\hat{f}}$ represent, respectively, the graph describing the causal relations between the variables of the classifier and the same graph after the intervention.

If $\triangleright'_{\hat{f}}, \sigma'$ is really the counterfactual of $\triangleright_{\hat{f}}, \sigma$ after intervention $I(a : \alpha)$, by applying the rules of Cut in table 1 we end with a sequent of the form:

$$[\triangleright_{\hat{f}}, \sigma] I(a : \alpha) \mid \sim t : \delta_q$$

More precisely, the rule I-Cut erases $a : \alpha$ from the premise, that is, the value assignment to the protected variable imposed by the intervention. Hence, if the counterfactual candidate contains $a : \alpha$, it can be erased. The rule \triangleright -Cut erases $v_i \triangleright v_k$, under the condition that $k \neq j$, enabling the erasure of all direct causal relations that do not enter the protected variable. The rule v -Cut erases $v_i : \gamma^i$, that is, the assignment of value given to v_i by the original Data Point, under the condition that v_i is not a consequence of v_j . This is established by checking $\blacktriangleright_{\hat{f}}$, that is, the set of mediate causal relations resulting from the graph before intervention. Hence, all the assignments of values to the variables that are not consequences of the protected variable in the original graph can be erased.

The label *Cut* of these rules refers to the fact that they can be seen as contractions of Cut applications of the following kind:⁶

$$\frac{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha) \mid \sim a : \alpha_1 \quad \triangleright_{\hat{f}}, \sigma, a : \alpha \mid \sim t : \delta_p}{[\triangleright'_{\hat{f}}, \sigma'] I(a : \alpha), \triangleright_{\hat{f}}, \sigma \mid \sim t : \delta_p} \text{Cut}$$

Where the first premise says that in the counterfactual Data Point obtained from $\triangleright'_{\hat{f}}, \sigma'$ by the intervention $I(a : \alpha)$, variable a receives the value α with probability 1 (that is, with certainty), and the second premise gives the probability of $t : \delta$ in the counterfactual candidate. \triangleright -Cut and v -Cut are contractions of similar Cut applications.

⁶See note 1 for the occurrence of a as target variable.

If by applying all these rules we can erase all the formulas in the counterfactual candidate, then this is really the Data Point corresponding to the counterfactual of the factual Data Point. Now, all we have to do to check **CF** is to compare the probabilities p and q . This can be done either by requiring their identity or by requiring a threshold on their difference.

Example 2 (Evaluation of **CF** for a classifier). *Let us use $\triangleright_{\hat{f}}$ for the causal relations resulting from graph (a) of figure 1 and $\triangleright'_{\hat{f}}$ for the causal relations resulting from graph (b) of the same figure. Moreover, let us assume that the following sequent describe decisions of a classifier:*

$$\triangleright_{\hat{f}}, G. : m, MS : mar, SAT : 1100, GAI : 65K, Deg. : PhD, Exp : 5y \mid \sim Loan : yes_{0.60} \quad (3)$$

$$\triangleright'_{\hat{f}}, G. : m, MS : div, SAT : 1100, Deg. : PhD \mid \sim Loan : yes_{0.60} \quad (4)$$

We can show that sequent 4 is the counterfactual of 3 after intervention $I(MS : div)$, since it entails the sequent:

$$[\triangleright_{\hat{f}}, G. : m, MS : mar, SAT : 1100, GAI : 65K, Deg. : PhD, Exp : 5y]I(MS : div) \mid \sim Loan : yes_{0.60} \quad (5)$$

The derivation can be constructed as follows, using $[...]$ for $[\triangleright_{\hat{f}}, G. : m, MS : mar, SAT : 1100, GAI : 65K, Deg. : PhD, Exp : 5y]$ and relying only on the rules in table 1:

$$\frac{\frac{\frac{\frac{\triangleright'_{\hat{f}}, G. : m, MS : div, SAT : 1100, Deg. : PhD \mid \sim Ln : y_{0.60}}{[...]I(MS : div), \triangleright'_{\hat{f}}, G. : m, MS : div, SAT : 1100, Deg. : PhD \mid \sim Ln : y_{0.60}} \text{ c-w}}{[...]I(MS : div), \triangleright'_{\hat{f}}, G. : m, SAT : 1100, Deg. : PhD \mid \sim Ln : y_{0.60}} \text{ I-Cut}}{[...]I(MS : div), \triangleright_{\hat{f}}, G. : m, SAT : 1100, Deg. : PhD \mid \sim Ln : y_{0.60}} \triangleright\text{-Cut}}{[\triangleright_{\hat{f}}, G. : m, MS : mar, SAT : 1100, GAI : 65K, Deg. : PhD, Exp : 5y]I(MS : div) \mid \sim Ln : y_{0.60}} \text{ v-Cut}$$

Moreover, since the probability that the variable Ln receives value y is the same in both sequents, this application of the classifier satisfies **CF**.

5. Conclusion

This work focuses on formal tools to check counterfactual fairness of probabilistic classifiers. We have seen how causal models allow to formalize counterfactuals, and argued that a modified version of their approach is suitable to test fairness for classifiers. We have proposed a typed natural deduction calculus **TNDPQ** extended with labels representing causal relations and expressions for interventions to internalize this approach.

Acknowledgments

This research was supported by the Ministero dell'Università e della Ricerca (MUR) through PRIN 2022 Project SMARTEST – Simulation of Probabilistic Systems for the Age of the Digital Twin (20223E8Y4X), and through the Project “Departments of Excellence 2023-2027” awarded to the Department of Philosophy “Piero Martinetti” of the University of Milan.

Declaration on Generative AI

During the preparation of this work, the authors used Grammarly in order to: Grammar and spelling check. After using these tool, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

- [1] F. A. D’Asaro, F. A. Genco, G. Primiero, Checking trustworthiness of probabilistic computations in a typed natural deduction system, *Journal of Logic and Computation* (2025) exaf003. URL: <https://doi.org/10.1093/logcom/exaf003>. doi:10.1093/logcom/exaf003.
- [2] E. Kubyshkina, G. Primiero, A possible worlds semantics for trustworthy non-deterministic computations, *International Journal of Approximate Reasoning* 172 (2024) 109212. URL: <https://www.sciencedirect.com/science/article/pii/S0888613X24000999>. doi:<https://doi.org/10.1016/j.ijar.2024.109212>.
- [3] G. Coraglia, F. A. D’Asaro, F. A. Genco, D. Giannuzzi, D. Posillipo, G. Primiero, C. Quaggio, Brioxalkemy: a bias detecting tool, in: G. Boella, F. A. D’Asaro, A. Dyoub, L. Gorrieri, F. A. Lisi, C. Manganini, G. Primiero (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, volume 3615 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 44–60. URL: <https://ceur-ws.org/Vol-3615/paper4.pdf>.
- [4] G. Coraglia, F. A. Genco, P. Piantadosi, E. Bagli, P. Giuffrida, D. Posillipo, G. Primiero, Evaluating ai fairness in credit scoring with the brio tool, 2024. URL: <https://arxiv.org/abs/2406.03292>. arXiv:2406.03292.
- [5] L. Ceragioli, G. Primiero, Trustworthiness preservation by copies of machine learning systems (submitted). URL: <https://doi.org/10.48550/arXiv.2506.05203>.
- [6] L. Ceragioli, G. Primiero, 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*, Bologna, Italy, October 26th, 2025, OVERLAY Workshop Proceedings, CEUR-WS.org, 2025.
- [7] M. Kusner, J. Loftus, C. Russell, R. Silva, *Counterfactual fairness*, volume 30, Massachusetts Institute of Technology Press, 2017, pp. 4067–4077.
- [8] D. K. Lewis, *Counterfactuals*, Blackwell, Malden, Mass., 1973.
- [9] J. Pearl, M. Glymour, N. Jewell, *Causal Inference in Statistics: A Primer*, Wiley, 2017. URL: <https://books.google.it/books?id=L3G-CgAAQBAJ>.
- [10] Viganò, *Labelled Non-Classical Logics*, Springer, New York, 2000.
- [11] S. Negri, J. von Plato, *Proof Analysis: A Contribution to Hilbert’s Last Problem*, Cambridge University Press, Cambridge, 2011.