Reasoning With Bias

Chiara Manganini[1,*,†], Giuseppe Primiero[1,†]

[1]Logic, Uncertainty, Computation and Information Group, Department of Philosophy, University of Milan

Abstract

In recent years, the problem of evaluating the trustworthiness of machine learning systems has become more urgent than ever. A directly related issue is that of assessing the fairness of their decisions. In this work, we adopt a primarily logical perspective on the topic, by trying to highlight the basic logical characteristics of the inferential setting in which a biased prediction occurs. To do so, we first identify and formalise four key desiderata for a logic capable of modelling the behaviour of a biased system, namely: skewness, dependency on data and model, non-monotonicity, and the existence of a minimal distinction between types of bias. On this basis, we define two metrics, one for group and one for individual fairness.

Keywords

Bias, Logic, Fairness,

1. Introduction

The widespread emergence of phenomena of biased predictions counts certainly among the most adverse impacts of new data-intensive science and technologies. This questions the trustworthiness of results, especially when the opaque nature of the models very often prevents us from precisely knowing or examining their inner structure [1].

A possible strategy to assess trustworthiness in these opaque settings is to check the actual model behaviour against a desirable one. Logics have been recently designed to formalise trustworthiness for probabilistic programs and to reason about them [2, 3, 4], with a specific focus on determining statistical distance measures for such systems with respect to their desirable or expected behaviour. A specific formulation has also been offered for modelling classifiers whose wrong output might be due to forms of bias, using a non-symmetric distance which reflects the systematic skeweness of the result [5]. In this sense, such logics offer verification and reasoning tools on the trustworthiness of black-box model with respect to explainable surrogate models, as those obtained by symbolic knowledge-extraction (SKE) [6, 7].

However, a grounded logical formalisation of how to define measures of bias in such contexts and how to reason about them is still at early stages in the literature [8, 9, 10, 11]. From the point of view of symbolic reasoning, and especially when it comes to the task of designing models for scientific inference in the era of data science, this means to extend the vast families

Aequitas 2023: Workshop on Fairness and Bias in AI | co-located with ECAI 2023, Kraków, Poland

*Corresponding author.

†These authors contributed equally.

chiara.manganini@unimi.it (C. Manganini); giuseppe.primiero@unimi.it (G. Primiero)

of defeasible logics and logics for uncertain reasoning with new ones, capable of accounting for the phenomenon of machine learning bias.

In this paper, we aim to sketch a first understanding of bias usable in the context of logical reasoning. In order to do so, in Section 2 we first start formulating the desiderata for a logic designed to reason in the presence of bias. In Section 3 we propose a logical formalisation of the notion of biased machine learning system and we show how it accommodates the mentioned desiderata. After introducing a correction distance as a measure of the monotonicity of a prediction (Section 4), in Section 5 we exploit it to quantitatively assess the presence of bias. Finally, we explain how this logical framework may be expanded, refined, applied in the future (Section 6).

2. Logical desiderata for reasoning with bias

As a preliminary work, we start from the observation that a logic for reasoning *with* bias – i.e., for drawing valid conclusions within a possibly biased inferential system – should be able to capture the following set of central properties characterising the phenomenon of algorithmic discrimination.

2.1. Skewness of Incorrect Predictions

A first and fundamental distinction our logic should account for is that between biased predictions and merely erroneous ones. In statistics, error is defined as the difference between the true value of a measurement and its recorded value, and can be either random or systematic. Since random error, or variability, has no preferred direction, its impact can be minimised with large sample sizes. Systematic error, instead, refers to a consistent over- or underestimation of the observations. It has a net direction so that averaging over a large number of observations does not eliminate its effect.

In the machine learning context, the notion of bias we want to focus upon is that of a systematic error against a certain protected category such as race, gender, age, geographical provenance, or economic status [12]. Here "against" is to be intended as "opposed to the direction given by the most favourable prediction". Such a direction of course varies in relation to the type of scenario considered. In this work, we focus on a very coarse-grained distinction between what we can call "allocative" and "punitive" contexts: in the former case, some beneficial resource (a job, a loan, a subsidy, etc.) is to be allocated to individuals based on a prediction; in the latter, some punitive action is imposed to individuals, based on some risk measure (fraud, recidivism, etc.). The reason for distinguishing between two symmetrical predictive scenarios is related to the different interpretation we give of the negation rule for our logical framework, which will become clearer in Section 3.

Summarising, two types of skewness should be considered in relation to bias:

1. Skewness as systematicity of error. The evaluation is skewed, in the sense that it shows a net direction, which means it results from a systematic rather than a random error. The direction itself is also relevant, as it emerges either in terms of false positives in punitive contexts, or in terms of false negatives in allocative contexts.

2. Skewness as partiality of the domain affected by the error. The systematic error negatively impacts only a subgroup of the population which is identified by its belonging to a certain protected category.

   Example. A classifier D is used to predict the risk of insurance fraud. To each new entry in a set of insurance claims, it either assigns the target label "Fraud" or "¬Fraud". D uses the protected binary feature "MalePolicyholder" for its prediction. The desideratum of skewness requires that, for the system to be considered gender-biased a significant imbalance in the prediction errors across male and female individuals must be observed in its output. For example, this can happen when false positives are significantly more in the female group with respect to the male one, so that the former class is negatively impacted by the decisions (predictions) of D.

All of these fundamental aspects need to be captured in our formal representation of a reasoning system for bias.

2.2. Dependency on Data and Model

A second property of the bias phenomenon we wish to capture is its dependency on both data and model. For instance, it is well acknowledged in the literature that a skew in frequency of the classes in the training set leads to disparate error rates on the underrepresented attributes [12]. This is not the only issue affecting training data, though. [13] highlights that other types of data bias (due to disproportionate measurement, representation, aggregation, and evaluation among the different classes) can lead to discriminatory results as well. In general, the presence of bias largely depends on the training data, and this is definitely something our logic should account for.

Also model design choices concerning loss function, optimiser and hyper-parameters, made to maximise test-set accuracy, can result in the occurrence of machine learning bias [13]. It has been highlighted that even more subtle choices – like learning rate and length of training – can impact on fairness due to the fact that underrepresented features are learnt later in the training process [14].

The actual manifestation of bias, instead, depends on the test set. Actually, there can be a situation in which a biased model does not in fact result in a discrimination due to contingent conditions.

> **Example.** Reconsider the gender-biased learning algorithm D for fraud risk detection. Imagine now that, due to a poor selection of inputs for the testing process, the totality of test inputs are male, i.e., each of them satisfies the predicate "Male". What can we say about a possible gender bias of the algorithm? On the one hand, an imbalance in the frequency of the gender classes in the historical recordings on which D has been trained causes the model to have the disposition to produce biased outcomes against women. On the other hand, no gender-biased prediction is actually produced, since this disposition never emerges.

Hence, our formal modelling will have to be parametric with respect to these three variables.

2.3. Non-Monotonicity

> **Example.** Gender-biased model D for fraud risk detection is fed step-by-step with features from a predetermined list that contains age, gender, address, etc. about datapoint a. At each step, D has to classify a. Imagine now that, while at step m of this process D classifies a as "Fraud", the information provided at step $m + i$ makes D revise the confidence in this prediction under a certain threshold, and the system outputs value "¬ Fraud".

An increase in both accuracy and available information can lead to a change of classification (from "Fraud" to "¬ Fraud"). At the beginning of this process, a "minimal-knowledge" condition, possibly corresponding to one piece of information to make the prediction, matches with the minimum accuracy possible; on the other hand, an "omniscience" condition, in which all the available pieces of information have been provided matches with the maximum accuracy possible. Theoretically, this should always be associated with a correct classification (the ground truth).

This simplified story just loosely models the behaviour of a learning system that corrects its own prediction on a new input a as the amount of available information increases. The above example captures the basic intuition at the heart of non-monotonicity, which we think useful to account for clarifying of the relationship between a biased outcome and the information used to generate it, both qualitatively and quantitatively. We start from the following claims:

Proposition 1. *Possessing all available correct information about datapoint x is sufficient for its correct classification.*

Proposition 2. *There is a minimal amount N of information about datapoint x, sufficient for its correct classification.*

Proposition 3. *Any amount of information $M < N$ about datapoint x is not sufficient for its correct classification.*

Note that, in practice, a classifier assumes $N = 1$ to be the weighted value of the whole set of features it uses, although it could be taken to be a smaller value (i.e., the prediction can be correct even in absence of some information). And $M < 1$ is always the weighted value of the set of features used at any point in time for an incorrect prediction. When incomplete information is already sufficient for a correct prediction, then $M < N < 1$.

With this in mind, let us now take into consideration a machine learning system, which we know to be biased according to the analysis in Subsections 2.1 and 2.2. In other words, this means that:

1. the system generates systematic (i.e., with a net direction) erroneous predictions against a certain protected category of individuals,
2. the errors depend on the data on which it has been trained and on the design choices of the model itself.

In this scenario, we are interested in exploring how the presence of bias in a system quantitatively depends on the amount of information used to return an incorrect prediction. A first intuition is that, on the one hand, the system may be considered maximally biased against a certain protected attribute when, given a new datapoint with that attribute and for which a wrong classification is predicted, no additional information may allow to correct it. Conversely, the system may be considered minimally biased against the same protected attribute when a single additional piece of information allows to correct it.

To recapitulate, in the presence of a biased system, we expect our logic to assess how much information is required to fail monotonicity as a proxy of the amount of bias affecting the system. Bias will manifest as an imbalance for such measure with respect to distinct groups, or distinct protected features.

2.4. A Minimal Distinction Between Types of Bias

When it comes to define the types of bias, a number of nomenclatures can be found in the literature [13, 12, 15, 16], most of which tend to characterise qualitative differences among biases based on the their different origins. For instance, in [13], among the statistical factors possibly resulting in a biased prediction we find: the non-random sampling of subgroups (sampling bias), the lack of diversity of the sample (representation bias), and the distortions that can emerge from the aggregation of datapoints (aggregation bias, Simpson's paradox). In this work, we rather want to focus our attention on two quantitative aspects connected to biased outcomes, along which a minimal distinction between types of bias can be traced:

1. biases engendered by an insufficient amount of information available to the system to generate the prediction.
2. biases engendered by an incorrect assignment of the feature relevance.

> **Example.** In the gender-biased learning algorithm D for fraud risk detection, the two different kinds of bias illustrated above would be illustrated as follows. First, consider a version of D which is never given any age information on the claimant, e.g. because such feature was not required at design stage; as a result, a potential bias towards some class (e.g. "female") might emerge which does not recognise that a given subclass (e.g. "above 60 years old") is actually strongly correlated with claims. Now consider a deployment of D which is fed only with occurrences of cases by female drivers and learns accordingly. Its trained version D' would eventually associates all cases of frauds to female driver and assign low scores to male drivers.

Hence, another component in bias emergence needs to be identified in the amount of information available and its relevance for the classification at hand.

3. Formalising Bias

For sake of simplicity, in this work we focus on the minimal case of a binary classification in the presence of a single binary protected attribute.[1] We abstract here from many technical details, and in particular we present only a sketch of a formalization based on a derivability relation $\vdash$, leaving the details of a corresponding consequence relation $\models$ to further work. Moreover, the current presentation is intended to offer only some useful intuitions about how to model the properties presented above in Section 2 within a logical setting; actual logical systems can then be designed with these characteristics to deal with specific bias cases.

Consider a model S trained on a dataset $\mathtt{d}$ and characterised by a set of design features (in terms of loss function, optimiser, and hyper-parameters used for its training) $\mathtt{m}$. The trained model returns a classification P for the datapoint a of the test set on a target feature $\mathbb{F}_i$. We use the derivability relation to express the classification by formulas of the form:

$$S \vdash^{d,m} P(a)$$

This judgement can be further enhanced: first, by a measure of the amount of features used by the classifier at any given moment, as a rational number on the left-hand side of the derivability relation; second, by a degree of accuracy on the classification expressed as a rational number in $[0, 1]$ on the right-hand side of the relation, then using a cut-off point to reduce the classification to a binary value. We leave these technicalities aside for the present moment and deal only with Boolean judgements. Further, let us assume this classification to be *incorrect*. This means that, in a given model Z – which we consider isomorphic to the real world, i.e., representing it correctly or, more agnostically, at least expressing what the designer considers to be a correct model for the test set – datapoint a should be correctly classified as $\neg P$:

$$Z \vdash \neg P(a)$$

To start off, here is how the desiderata mentioned in the previous Section can fit into a logical framework:

- What mentioned in Subsection 2.1 indicates that inferential and/or semantic validity may be indeed appropriate to model the skewness of bias in classification: fixed the properties of the system S, the error in the prediction arises in terms of evaluations for some predicates and some individuals; hence, a subclass of predicates and individuals of the domain in the test set will make biased evaluations emerge, where such classes can be at least partially identified by comparison with the training set.
- For the discussion in Subsection 2.2, we know that the incorrect inference returned by system S can depend on both the data d with which it was trained or on the characteristics m of the model itself. This aspect requires therefore that such parameters are made transparently available in the formalisation of the system we are after.

[1] Our proposal can be quite easily translated into a multi-label setting by introducing a probabilistic evaluation on the target predicate. Adding further protected attributes could be more challenging, due to the problem of modelling intersectionality of discrimination. These tasks, however, go beyond the scope of the present work.

- The aspects of non-monotonicity considered in Subsection 2.3 are to be expressed in terms of the dynamic evaluation made by S for a given datapoint, as the information under which the prediction is derived increases. We aim therefore at defining a measure of such a change, by invalidating standard classical inference rules (respectively, making a consequence relation non-monotonic).
- As illustrated in Subsection 2.4, S can be differently biased depending either (1) on how much information is provided by the available features used as predictors, or (2) on how relevant each of the selected feature has been taken to be by the system. These aspects will be rendered by formulating explicitly features on the classifier and their weights.

Consider now a classification system $(S, \vdash^{d,m})$ within a language $\mathscr{L} = \{\mathbb{D}, \mathbb{P}, <^w\}$ where:

- $\mathbb{D} = \{a, b, c, \dots, n\}$ is a finite set of elements of a domain denoting the datapoints of the test set;
- $\mathbb{P} = \{P, Q, R, ..., Z\}$ is the finite set of the predicates in the language such that:

  1. a partition of $\mathbb{P}$ is the set $\mathscr{P} = \{\mathbb{P}\mathbb{r}, \mathbb{N}\mathbb{P}\mathbb{r}, \mathbb{T}\}$ denoting, respectively, the set of protected predicates, the set of non-protected ones, and the target predicate to be predicted;
  2. another partition of $\mathbb{P}$ is the set of features $\mathscr{F} = \{\mathbb{F}_1, \mathbb{F}_2, ..., \mathbb{F}_m\}$ which, for simplicity, we assume to be all binary.

  In other words, $\mathbb{P}$ enumerates all the features used by the system. Every class, in turn, denotes a possible value of a certain feature, be it protected, non-protected, or the target feature.
- $<^w \subseteq \mathbb{P} \times \mathbb{P}$ is an ordering on the predicates $\mathbb{P}$ based on a weight function $w : \mathbb{P} \mapsto [0, 1]$ that assigns a measure of relevance to each predicate for a given feature $\mathbb{F}_i$, so that $\langle P_i, P_j \rangle \in <^w$ iff $w(P_i) \leq WwP_j)$ for every $P_i, P_j \in \mathbb{P}$. Moreover, we take that weights are normalised, i.e., that $\sum_n w(P_n) \leq 1$.

  Example. A classifier is used to predict applicants' eligibility for a loan, therefore $\mathbb{T} = \{Loan\}$, where $Loan(a)$ can be true (if a is eligible for a loan) or false (if a is not eligible). The system classifies on the basis of the following sets of predicates: $\mathbb{PR} = \{Male\}$, and $\mathbb{NPR} = \{FullTimeEmployed, HouseholdOwner, UniversityDegree, Married\}$. It is known that two features are more relevant than the others for the classification: $w(FullTimeEmployed) = 0.3$ and $w(HouseholdOwner) = 0.25$. All the other features are equally relevant for the classification, and they all weigh $\frac{1-0.3-0.25}{3} = 0.15$.

An atomic formula of $\mathscr{L}$ is therefore a predication of a class P for a given datapoint a.[2] In this context, proof-theoretically the information available in S expresses the current knowledge about a such that given the available inference rules in $\vdash$ it allows to derive that a has (resp. has

[2]The choice of connectives in $\mathscr{L}$ will be determined by requirements of the system and will define the inference relation $\vdash$ (in the following, we will only make general comments on the use of negation).

not) target feature P. Semantically, it says that the model in which all information contained in S is true, will also make $P(a)$ true (resp. false).

In the following we focus on two structural elements required to model bias formally as intended above: the role of the rules for negation, and the quantification of information for non-monotonicity of the inference relation.

3.1. Negation

The initial situation of skewness might be represented as follows. In punitive contexts, bias emerges as systematic error in terms of false positives, according to which

$$S \vdash^{\mathrm{d},\mathrm{m}} P(a) \quad \textit{while} \quad Z \vdash \neg P(a)$$

i.e., the real or chosen model of the world sees individual a not to satisfy the property P, while the system S predicts it to have it. In allocative contexts, bias emerges as systematic error in terms of false negatives, according to which:

$$S \vdash^{\mathrm{d},\mathrm{m}} \neg P(a) \quad \textit{while} \quad Z \vdash P(a)$$

i.e., the real or chosen model of the world sees individual a satisfying the property P, while the system S predicts the opposite. In both contexts, the systematic error concerns a subset of the domain $\mathbb{D}$, identified by a partition with respect to a predicate $P \in \mathbb{Pr}$.

The logical setting we are defining allows to model the difference between the two scenarios by a different interpretation of the rule for negation. In the allocative context, a correctly working classifier should see implemented the interpretation of Negation as Contradiction:

$$\frac{S \vdash \neg P(a)}{S \vdash P(a) \rightarrow \bot} \quad \textit{while} \quad Z \vdash P(a)$$

i.e., if the system predicts the negation of the intended property (say: not being eligible for welfare credits), the assignment of a certain benefit should be blocked. On the other hand, in a punitive context it seems that a correctly working classification system should express the principle of Negation as Failure:

$$\frac{S \nvdash P(a)}{S \vdash \neg P(a)} \quad \textit{while} \quad Z \vdash \neg P(a)$$

in other words, if the system does not predict the intended property (say: being guilty), it should infer its negation (not being guilty). This allows us to meet the desideratum of "skewness as systematicity of error" specified at point (1) of Section 2.1. In simple words, we required that a logic for reasoning with bias should be able to capture the idea that the systematic prediction error we call bias can either occur in terms of false positives (in punitive setting) or in terms of false negatives (in allocative ones). This difference can be in fact accommodated by the semantics of our logic, through the choice of an appropriate set of logical rules for negation.

3.2. Quantifying Non-Monotonicity

Under the assumption of an incorrect prediction, we want to quantify the amount of information needed for the (supposedly wrong) inference to be corrected. We know that the inability of an inferential system to accommodate such a change is expressed by the structural rule of Weakening:

$$\frac{S \vdash P(a)}{S, Q_1(a), \dots, Q_n(a) \vdash P(a)} Weak \quad while \quad Z \vdash \neg P(a)$$

for

$$\{Q_1, \dots, Q_n\} \subseteq (\mathbb{P} - \{P\})$$

i.e., while adding new predicates to the datapoint, the ability of the system to continue inferring a valid (target) property, should not diminish. Note that the quantification over the possible predicates for which the Weakening rule holds is fundamental here. For instance, if $\{Q_1(a), \dots, Q_n(a)\} = \mathbb{P} - \{P\}$, this means that S incorrectly predicts the class P for the datapoint a (for some P and some a), and no additional amount of information allows to change this inference:

$$\nexists Q_m(a) \quad s.t. \quad \frac{S \vdash P(a)}{S, Q_1(a), \dots, Q_n(a), Q_m(a) \vdash \neg P(a)} \tag{1}$$

This is, for simplicity, expressed here by boolean predicates, while a worked out analysis would make use of the anticipated probabilistic assignment of predicates to constants and the corresponding measure on the amount of features evaluated.

What one would want to model, therefore, is a system whose incorrect prediction is very "close" to be amended, provided a new piece of information becomes available:

$$\exists Q_m(a) \quad s.t. \quad \frac{S \vdash P(a)}{S, Q_1(a), \dots, Q_n(a), Q_m(a) \vdash \neg P(a)} \tag{2}$$

Example. The fraud detection classifier should implement a negation as failure principle

$$\frac{D \nvdash Fraud(a)}{D \vdash \neg Fraud(a)}$$

as long as enough information is provided to prove the contrary, i.e.

$$Z \vdash Fraud(a)$$

In particular, the aim is to show how much information allows for the following

$$\frac{D \vdash \neg Fraud(a)}{D, Q_1(a), \dots, Q_m(a) \vdash Fraud(a)}$$

Hence, our next tasks are

1. to evaluate the amount of information of $Q_1(a), \dots, Q_n(a)$ so that certain predicates are more relevant than others;
2. use it to quantify non-monotonicity.

In other words, assuming S's initial incorrect prediction for a datapoint a, the degree of monotonicity expresses "how far the system is from a correction of the prediction". Both these goals will be addressed in the following section.

4. Correction Distance

Consider the task of inferring the correct prediction $S \vdash P(a)$. Assume:

$$\{Q_1(a), \dots, Q_n(a)\}^{f(w_1,\dots,w_n)=N} \vdash P(a)$$

where N expresses the amount of information computed by a chosen function f over the weights of the predicates $Q_1, \dots, Q_n$ and sufficient for the correct classification. We abbreviate with $\mid S \mid= N$ the overall weight of the set of formulas $\{Q_1(a), \dots, Q_n(a)\}^{f(w_1,\dots,w_n)=N}$ when used in the system S.

Consider moreover:

$$\{Q_1(a), \dots, Q_m(a)\}^{f(w_1,\dots,w_m)=M} \vdash \neg P(a)$$

where M is the amount of information currently available and used for a presumably or possibly wrong prediction $S \vdash \neg P(a)$, assuming the same function f. We abbreviate with $\mid S \mid= M$ the overall weight of the set of formulas $\{Q_1(a), \dots, Q_m(a)\}^{f(w_1,\dots,w_m)=M}$ when used in the system S.

The difference $(N - M)$ expresses the amount of additional information that would be sufficient to correct our wrong prediction, normalized over the accuracy of the system. Namely, it expresses the amount of additional information that would be sufficient for the rule $Weak$ to fail. Hence, these elements collectively express an amount of non-monotonicity, weighted on a measure of certainty of the prediction:

Definition 1 (Correction Distance). *The correction distance for a system $(S, \vdash^{d,m})$, given a target predicate P and an individual a, is the inverse of the amount of additional information sufficient for S to correct its evaluation on a, weighted on certainty (as expressed by accuracy known for the system from previous analyses)*

$$C(S, \vdash, P(a)) = 1 - (N - M) * accuracy$$

Given our definitions of N and M – as the amounts of information sufficient to give, respectively, a correct and an incorrect prediction – the measure C expresses an evaluation of how far the system is from correcting a wrong classification. We have to further impose that $(N - M) > 0$ if we want to model the non-monotonic setting exemplified in Section 2.3, in which an increase of the available information is assumed to always result in an improvement (or, in a binary setting like ours, a correction) of the prediction made before (see Proposition

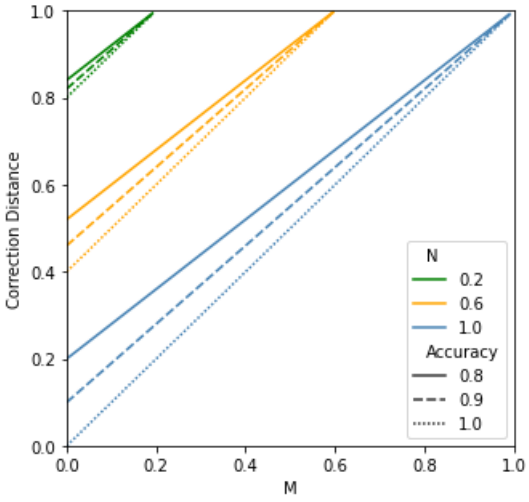

Figure 1: Examples of correction distance for incorrect predictions, at different levels of N and Accuracy

1). Moreover, in defining the order relation $<^w$, we said that all the weighted features of the system's feature space together sum to 1. Therefore, $N \leq \sum_{P_i \in \mathbb{P}}^{n} w(P_n) = 1$. Note, finally, that $C(S, \vdash, P(a)) \in [0, 1]$ and that it could be interpreted as a probabilistic measure. For instance, if $C(S, \vdash, P(a)) = 10\%$, we could say that there is a probability of 90% that the system will eventually reach the correct prediction for a.

> **Example.** Recall the anti-fraud system mentioned in previous examples. Figure 1 shows how $C(S, \vdash, P(a))$ varies in relation to M, N, and accuracy, assuming that an incorrect classification is always returned by $(S, \vdash^{d,m})$ for datapoint a. On the x-axis we measure the available information M and on the y-axis the corresponding correction distance.
>
> If we focus on the system denoted by the dashed yellow line, where the amount of information required for a correct classification is 60% of the total and accuracy is known to be 90%, in the virtual case in which no information is used for the prediction, the correction distance attributed to the classification is already set at 46%. The more the system proceeds with the acquisition of new information, say 40%, the correction distance attributed to a wrong classification rises to 82%.
>
> For comparison, if we take a system with maximal accuracy (100%) and little information required to infer a correct classification (20%), a wrong classification without any information is already assumed to be 80% distant from its correction

(dotted green line).

Finally, a system with low accuracy (80%) and which requires total information to infer correct classifications (100%) – like the one represented by the solid blue line – will be evaluated to be 36% distant from correction for the case of $M = 20\%$. While with $M = 90\%$, if a wrong is still being returned, its correction distance will be evaluated at 92%.

In the light of the above, we can now get back to the two limit cases discussed informally at the end of Section 3. In fact, we can now give a much more refined definition of a system which is maximally (resp. minimally) distant from the correct prediction, by simply translating Formula 1 (resp. 2) in terms of correction distance. We obtain that a system is maximally distant from the correction of an incorrect prediction $P(a)$, based on an amount of information M, then there is no additional information M' such that the sum of $M + M'$ results in an amount of information sufficient of predicting correctly $\neg P(a)$.

Definition 2 (Maximally Distant-From-Correction System). *Given* $(S, \vdash^{d,m}, P(a))$ *with* $| S |= M$ *and* $\nexists M'$ *s.t.* $M + M' = N$, *then* $C(S, \vdash^{d,m}, P(a)) = 1$

All things being equal, a system is correctable if there exists some additional information M' such that the sum of $M+M'$ results in an amount of information sufficient of predicting correctly $\neg P(a)$.

Definition 3 (Correctable System). *Given* $(S, \vdash^{d,m})$ *with* $| S |= M$ *and* $\exists M'$ *s.t.* $M + M' = N$, *then* $C(S, \vdash^{d,m}, P(a)) < 1$

Note that, intuitively, it makes sense to present the Correction Distance as a measure from a possibly incorrect to a correct prediction. Nonetheless, in the next section, we make use of the Correction Distance as a criterion to establish whether a classifier is biased towards or against a certain class by measuring whether associated correction distances differ. In this sense, one can even abstract from the assumption of initial incorrectness of the system, and ask what would take for the system to change the result of a classification for a given datapoint, and whether that amount of information is different for distinct classes of individuals.

5. Measuring Bias as Correction Distance Imbalance

To reach a proper definition of bias, we just need to accommodate the last desideratum left unaddressed so far, i.e., the idea of skewness as "partiality of the domain affected by the error" introduced in Section 2.1 at point (2). Put differently, while in the previous Section our correction distance has been defined over a single datapoint a, now it must be generalized over a subdomain of $\mathbb{D}$ in order to fully account for the notion of bias relevant in machine learning contexts. Adopting the distinction between group (or statistical) vs. individual (or similarity-based) unfairness criteria, well known in the bias literature [15], we first define a measure of group unfairness in terms of imbalance of correction distance of a classifier used to predict a target class P – denoted by $C(S, \vdash^{d,m}, P)$ – across the subdomains of $\mathbb{D}$ determined by a binary protected category Q, with respect to a tolerance threshold ϵ.

Definition 4 (Measure of Group Fairness).

$$Bias_{group}(S, \vdash^{d,m}, P) =\mid C(S, \vdash^{d,m}, P(a)) - C(S, \vdash^{d,m}, P(b)) \mid> \epsilon$$

where

$$Q \in \mathbb{P}\mathrm{r}, \forall a, b \in \mathbb{D} \mid Q(a) \wedge \neg Q(b)$$

and where ϵ is a threshold value for the difference of the correction measures above which fairness is considered to fail.

Coming to the measure of individual fairness, we explore a strategy inspired by the principle of Fairness Through Unawareness or Blindness [17, 18, 19]. To implement it, we adopt the following similarity criterion.

In general, given two sets A and B, the Jaccard Index of A and B, denoted by $J(A, B)$, is defined as:

Definition 5 (Jaccard Index).

$$J(A, B) = \frac{\mid A \cap B \mid}{\mid A \cup B \mid}$$

Taken two individuals a and b in the test set $\mathbb{D}$, given the sets of the predicates they respectively satisfy (denoted by $\mathbb{L}_a$ and $\mathbb{L}_b$ with slight abuse of notation), they are called blindly similar – expressed by $BSim(a, b)$ – if and only if they only differ with respect to the protected predicate Q, and they show more similarities than differences:

Definition 6 (Blind Similarity). *Two datapoints a and b are blindly similar, denoted by $BSim(a, b)$, iff:*

1. $(\mathbb{L}_a \cup \mathbb{L}_b) \setminus (\mathbb{L}_a \cap \mathbb{L}_b) = \{Q\} \in \mathbb{PR}$
2. $J(\mathbb{L}_a, \mathbb{L}_b) > \frac{1}{2}$

Definition 7 (Measure of Individual Unfairness).

$$Bias_{Ind}(S, \vdash^{d,m}, P) =\mid C(S, \vdash^{d,m}, P(a)) - C(S, \vdash^{d,m}, P(b)) \mid> \epsilon$$

where

$$\forall a, b \in \mathbb{D}.BSim(a, b)$$

and where ϵ is a threshold value for the difference of the correction measures above which fairness is considered to fail.

Namely, we define a measure for individual unfairness in terms of an imbalance in the correction distance – denoted by $C(S, \vdash^{d,m}, P)$ – across the subdomain of similar individuals determined by Definition 6, with respect to a tolerance threshold ϵ.

6. Further developments

We presented an attempt to bring a logical contribution to the research on fairness in machine learning. To make this proposal complete and applicable, much work remains to be done.

First of all, it will be crucial to contextualise our approach within current technical, philosophical, and logical research on bias. This means, in particular, connecting our proposal to other recent formalisations of bias [8, 9, 10, 11] on the one hand, and to the wide literature on fairness measures on the other. Understanding how our proposal aligns with and contributes to the current debate on fairness will provide a better insight on its potential implications and applications.

Secondly, formal and structural features of our logical framework need further clarification, starting from inferential rules and semantic clauses. In this spirit, some remarks made in Section 3 highlighted that the negation rule seems to require different interpretations, based on the setting considered (either allocative or punitive). More generally, the definition of inference rules and semantic clauses must be functional to measure inferences (respectively, define valid consequences) in terms of correction distance, to be used for the computation of the two fairness measures proposed. Moreover, it will be important to further refine the notions of ground truth, training data, and test sample.

Also a probabilistic refinement of the proposed logical framework is essential to better model machine learning inference. A probabilistic evaluation on the target predicate, to be interpreted as a measure of confidence or accuracy of the prediction returned by the system, will in turn require to suitably formalise its relationship with the other used notions of quantity of information and feature relevance.

From a semantical point of view, a modal setting may be helpful to reason about biased predictions. Kripke models can be interpreted as sets of evaluations of attributes on datapoints, and modalities to access new worlds can be understood as evaluations of new attributes (including the target one). Under this interpretation, verifying that a machine learning model S is individually fair (as defined in Definition 7) relatively to a certain protected attribute means to check as a safety property that similar possible worlds (based on the similarity definition given in Definition 6) provide access to the same possible worlds. Developing model checking and correctness algorithms remains the main goal for the future development of the present work.

References

[1] P. Linardatos, V. Papastefanopoulos, S. Kotsiantis, Explainable ai: A review of machine learning interpretability methods, Entropy 23 (2020) 18. URL: http://dx.doi.org/10.3390/e23010018. doi:10.3390/e23010018.

[2] F. A. D'Asaro, G. Primiero, Probabilistic typed natural deduction for trustworthy computations, in: Proceedings of the 22nd International Workshop on Trust in Agent Societies (TRUST2021 @ AAMAS), 2021.

[3] F. A. D'Asaro, F. Genco, G. Primiero, Checking trustworthiness of probabilistic computations in a typed natural deduction system, 2023. arXiv:2206.12934.

[4] F. A. Genco, G. Primiero, A typed lambda-calculus for establishing trust in probabilistic

programs, CoRR abs/2302.00958 (2023). URL: https://doi.org/10.48550/arXiv.2302.00958. doi:10.48550/arXiv.2302.00958. arXiv:2302.00958.
[5] G. Primiero, F. A. D'Asaro, Proof-checking bias in labeling methods, in: G. Boella, F. A. D'Asaro, A. Dyoub, G. Primiero (Eds.), Proceedings of 1st Workshop on Bias, Ethical AI, Explainability and the Role of Logic and Logic Programming (BEWARE 2022) co-located with the 21st International Conference of the Italian Association for Artificial Intelligence (AI*IA 2022), Udine, Italy, December 2, 2022, volume 3319 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 9–19. URL: http://ceur-ws.org/Vol-3319/paper1.pdf.
[6] R. Calegari, G. Ciatto, A. Omicini, On the integration of symbolic and sub-symbolic techniques for XAI: A survey, Intelligenza Artificiale 14 (2020) 7–32. URL: https://doi.org/10.3233/IA-190036. doi:10.3233/IA-190036.
[7] R. Calegari, F. Sabbatini, The psyke technology for trustworthy artificial intelligence, in: A. Dovier, A. Montanari, A. Orlandini (Eds.), AIxIA 2022 - Advances in Artificial Intelligence - XXIst International Conference of the Italian Association for Artificial Intelligence, AIxIA 2022, Udine, Italy, November 28 - December 2, 2022, Proceedings, volume 13796 of *Lecture Notes in Computer Science*, Springer, 2022, pp. 3–16. URL: https://doi.org/10.1007/978-3-031-27181-6_1. doi:10.1007/978-3-031-27181-6_1.
[8] A. Ignatiev, M. C. Cooper, M. Siala, E. Hebrard, J. Marques-Silva, Towards formal fairness in machine learning, in: Principles and Practice of Constraint Programming: 26th International Conference, CP 2020, Louvain-La-Neuve, Belgium, September 7–11, 2020, Proceedings, Springer-Verlag, Berlin, Heidelberg, 2020, p. 846–867. URL: https://doi.org/10.1007/978-3-030-58475-7_49. doi:10.1007/978-3-030-58475-7_49.
[9] Y. Kawamoto, An epistemic approach to the formal specification of statistical machine learning, Software and Systems Modeling 20 (2020) 293–310. URL: http://dx.doi.org/10.1007/s10270-020-00825-2. doi:10.1007/s10270-020-00825-2.
[10] V. Belle, Toward a logical theory of fairness and bias, Theory and Practice of Logic Programming (2023) 1–19. doi:10.1017/S1471068423000157.
[11] X. Liu, E. Lorini, A unified logical framework for explanations in classifier systems, Journal of Logic and Computation 33 (2023) 485–515. doi:10.1093/logcom/exac102.
[12] H. Suresh, J. V. Guttag, A framework for understanding sources of harm throughout the machine learning life cycle, Equity and Access in Algorithms, Mechanisms, and Optimization (2019).
[13] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, A. Galstyan, A survey on bias and fairness in machine learning, 2022. arXiv:1908.09635.
[14] S. Hooker, Moving beyond "algorithmic bias is a data problem", Patterns 2 (2021) 100241. doi:10.1016/j.patter.2021.100241.
[15] A. Castelnovo, R. Crupi, G. Greco, D. Regoli, The zoo of fairness metrics in machine learning, CoRR abs/2106.00467 (2021). URL: https://arxiv.org/abs/2106.00467. arXiv:2106.00467.
[16] S. R. Das, M. Donini, J. Gelman, K. Haas, M. Hardt, J. Katzman, K. Kenthapadi, P. Larroy, P. Yilmaz, B. Zafar, Fairness measures for machine learning in finance, in: The Journal of Financial Data Science, 2021.
[17] N. Grgic-Hlaca, M. B. Zafar, K. P. Gummadi, A. Weller, The case for process fairness in learning: Feature selection for fair decision making, 2016. URL: https://api.semanticscholar.org/CorpusID:13633339.

[18] S. Verma, J. Rubin, Fairness definitions explained, in: Proceedings of the International Workshop on Software Fairness, FairWare '18, Association for Computing Machinery, New York, NY, USA, 2018, p. 1–7. URL: https://doi.org/10.1145/3194770.3194776. doi:10.1145/3194770.3194776.

[19] M. J. Kusner, J. R. Loftus, C. Russell, R. Silva, Counterfactual fairness, 2018. arXiv:1703.06856.