A Frank System for Co-Evolutionary Hybrid Decision-Making

Federico Mazzoni[1]✉, Riccardo Guidotti[2], and Alessio Malizia[3]

[123]University of Pisa, Italy, [3]Molde University College, Norway
[1]{name.surname}@phd.unipi.it, [23]{name.surname}@unipi.it

Abstract. We introduce Frank, a human-in-the-loop system for co-evolutionary hybrid decision-making aiding the user to label records from an un-labeled dataset. Frank employs incremental learning to "evolve" in parallel with the user's decisions, by training an interpretable machine learning model on the records labeled by the user. Furthermore, Frank advances state-of-the-art approaches by offering inconsistency controls, explanations, fairness checks, and bad-faith safeguards simultaneously. We evaluate our proposal by simulating the users' behavior with various levels of expertise and reliance on Frank's suggestions. The experiments show that Frank's intervention leads to improvements in the accuracy and the fairness of the decisions.

Keywords: Human-Centered AI · Hybrid Decision Maker · Skeptical Learning · Incremental Learning · Explainable AI · Fairness Checking

1 Introduction

Automated decision-makers based on Machine Learning (ML) are still not widely adopted for high-stakes decisions such as medical diagnoses or court decisions [22]. In these fields, humans are aided but not replaced by Artificial Intelligence (AI), resulting in Hybrid Decision-Makers (HDM) [15]. While HDM literature is flourishing, certain key aspects have not yet been considered, preventing HDM systems from covering possible use cases. HDM systems promote the collaboration between human and AI decision-makers, resulting in a final set of "hybrid" decisions (some taken by the human, others by the machine). In *Learning-to-Defer* [10] systems, the machine plays the primary role, deferring decisions on records with a high degree of uncertainty to an external human supervisor. In [22], a rule-based AI model with inferred rules suggests replacing some user's decisions to maximize fairness, whereas in [9], the model mediates between a user and their supervisor if it is not confident in the user's decisions. On the other hand, in the Skeptical Learning (SL) paradigm, an ML model learns "in parallel" to the decisions taken by a human and queries them if it is "skeptical" of the human decision [4, 19, 23, 24]. SL aims to help the user remain consistent with their past decisions, still giving them veto power against the model's suggestions. SL has been extensively applied to personal context recognition [4, 24] and image classifications [19]. In [19], SL suggestions are also supported by *contrastive explanations*.

Our system employs and extends traditional SL, by taking into account simultaneously fairness aspects, explainable suggestions, and the involvement of the user's supervisor. In line with [4], our proposal is powered by a *Incremental Learning* (IL) model. IL, also known as Continual Learning, is an ML paradigm where the model is continuously trained on small data batches, potentially including only one data point, instead of the entirety of the training set [12, 21].

The *eXplainable AI (XAI)* research field aims to create humanly interpretable proxies of "black-box" ML models used for decision-making. An explanation is *global* if it unveils the whole model logic, or *local* if it justifies the decision of a specific record [7]. A global explanation can be achieved by approximating black-box models with interpretable-by-design ones, such as a decision tree, which also offers local explanations as decision rules [3]. Also, instance-based explanations make use of examples and counter-examples, i.e., similar records with the same/different decision by the AI system [6]. Our proposal offers both a model approximation, employing an interpretable decision tree, and instance-based (counter-)examples to explain the model's suggestions to the user.

Finally, we also account for the *fairness* of the decisions. Two major approaches have been proposed to quantify a dataset's fairness [2]. For *individual fairness*, similar individuals should receive similar treatment, while for *group fairness*, each group should receive a similar treatment [16]. The discriminatory feature to be monitored (e.g., *Race*, *Gender*) is often defined *sensitive* or *protected attribute* [20]. Given a sensitive attribute, our proposal checks both individual and group fairness, helping the user avoid discriminating behavior.

We propose Frank, a HDM system overcoming the current limitations of SL related to explainability, fairness, consistency, and bad-faith users. As in SL, if the user's label is inconsistent with Frank's prediction, the user is warned of possible contradictions with their past behavior and suggested to modify their decision. Besides, Frank provides explanations that become increasingly detailed as the model learns more from the user, who can, in turn, learn more about their behavior. Also, Frank can prevent bad-faith behavior and discriminating decisions. Ultimately, Frank and the human have a symbiotic co-evolutionary relationship, with Frank's model able to predict the user's behavior, thus aiding them, and the human feeding Frank's model with new data. Experimental results show that pairing Frank with less reliable users provides noticeable improvements in terms of accuracy and fairness, and that the usage of explanations increases the number of acceptance for suggestions in case of skepticism.

2 Setting the Stage

We keep the paper self-contained by reporting in the following a brief overview of concepts necessary to understand our proposal. We indicate with X, Y a dataset where $X = \{x_1, \dots, x_n\} \in \mathcal{X}^{(m)}$ is a set of n records described by m attributes (features), i.e., $x_i = \{(a_1, v_1), \dots, (a_m, v_m)\}$, where a_i is the attribute name and v_i is the corresponding value, and $\mathcal{X}^{(m)}$ is the feature space consisting of m input features, while $Y = \{y_1, \dots, y_n\} \in \mathcal{Y}$ is the set of the target variable in the target space $\mathcal{Y}$. With $A = \{a_1, \ldots, a_m\}$ we indicate the set of feature names, and for an instance $x \in X$, we write $x[a_k]$ to refer to the value v_k of attribute a_k. For classification problems, $y_i \in \{1, \ldots, l\} = L$ where L is the set of different class labels and l is the number of the classes, while when dealing with regression problems, $y_i \in \mathbb{R}$. Without losing in generality, we consider $l = 2$, i.e., binary classification problems. We indicate a trained decision-making model with a function $f : \mathcal{X}^{(m)} \rightarrow \mathcal{Y}$ that maps data instances x from the feature space $\mathcal{X}^{(m)}$ to the target space $\mathcal{Y}$. We write $f(x) = y$ to denote the decision y taken by f, and $f(X) = Y$ as a shorthand for $\{f(x_i) \mid x_i \in X\} = Y$.

Skeptical Learning. Given a ML model f and a dataset X, the user is tasked to assign a label y_i to each record $x_i \in X$. In SL, the user assigns to x_i the label $\hat{y}_i$, according to their own belief and background and, independently from them, f assigns the label $\tilde{y}_i$, i.e, $\tilde{y}_i = f(x_i)$. The ML model implementing f can be pre-trained on a small training set. If $\hat{y}_i \neq \tilde{y}_i$ and f is *skeptical* (see below), the user is asked if they want to accept $\tilde{y}_i$ as y_i. If they do, y_i takes the value $\tilde{y}_i$. If the user refuses, if $\hat{y}_i = \tilde{y}_i$ or if the model is not skeptical, y_i is assigned $\hat{y}_i$. The ML model is then incrementally trained on x_i and y_i.

The definition of the model's *skepticality* varies in the literature [19]. However, skepticism is always related to model's *epistemic uncertainty*, which is independent of the notion of *confidence* score towards a certain decision, i.e., the prediction probability[1]. Epistemic uncertainty is the model's *ignorance*, and given enough data, it should be minimized [8]. Only a limited number of ML model offers by-design access to epistemic uncertainty, e.g., Naive Bayes, Gaussian Process [4,8]. In the context of SL, it has been approximated by the *empirical accuracy* of past predictions both of the user and the model, i.e., the ratio between the number of times a label has been proposed by the user or predicted by the model, and the times it has been accepted as y [23]. Thus, given x_i and the prediction $\tilde{y}_i$, the skepticism towards the user's $\hat{y}_i$ is:

$$skpt(x_i, \tilde{y}_i, \hat{y}_i, Y, \tilde{Y}, \hat{Y}) = c(f, x_i, \tilde{y}_i) \cdot ea(\tilde{y}_i, Y, \tilde{Y}) - c(f, x_i, \hat{y}_i) \cdot ea(\hat{y}_i, Y, \hat{Y}) \quad (1)$$

where $c(f, x_i, \tilde{y}_i)$ and $c(f, x_i, \hat{y}_i)$ are the model confidence score towards $\tilde{y}_i$ and $\hat{y}_i$. The function ea computes the empirical accuracy of either the model or the user toward their respective label. The empirical accuracy is computed as the cardinality of the intersection between the subset of all their past decisions with label either $\hat{y}_i$ or $\tilde{y}_i$ and the corresponding subset in Y, i.e., the final decision, over the subset of all their past decisions with either $\hat{y}_i$ or $\tilde{y}_i$. Therefore, each possible label $l \in L$ has two accuracy values – following the user's and the model's track record. In [23], the user's accuracy values are initialized with 1, and the model's with 0 (therefore, the model is not skeptical of earlier decisions).

Incremental Decision Tree. We employ Extremely Fast Decision Tree (EFDT) [13], a variant of Hoeffding Tree, which offers performance on par with the non-incremental counterpart [1,5]. EFDT splits a node as soon as the split is deemed useful, with the possibility of later revisiting the decision [13]. Being a decision tree, EFDT can also be exploited to provide explanations to the user [7].

[1] Note that there's a general lack of normativity w.r.t. these terms; e.g., [23] uses the term *confidence* to refer to the epistemic uncertainty.

Preferential Sampling. We include an interactive variant of Preferential Sampling (PS), an algorithm increasing group fairness [11]. PS assumes that in the set of class labels L we can recognize a favorable $+$ and an unfavorable $-$ decision, i.e., $L = \{+, -\}$, while among A we can denote a binary sensitive attribute $sa \in A$, e.g., *Sex*. The possible values $\{v, \bar{v}\}$ of sa refers to a discriminated group v and privileged group $\bar{v}$, e.g., *Female* and *Male*. The algorithm identifies the size of the groups of Discriminated records with a Positive (DP) or Negative label (DN), and of Privileged records with a Positive (PP) or Negative label (PN). Given X, it computes the *dataset discrimination score* as:

$$disc(X, sa, v) = {}^{|PP|}/_{|PP \cup PN|} - {}^{|DP|}/_{|DP \cup DN|} \tag{2}$$

Then, it computes how many records from PP and DN should be removed, and how many from DP and PN should be duplicated to reach $disc \approx 0$. Records are selected w.r.t. the prediction probability of a classifier trained on X. A variant supporting non-binary sensitive attributes, and where the user does not need to know *a priori* the discriminated group(s), is presented in [14].

3 A Frank System

Frank is a system for HDM, learning from the decisions of the human decision-maker (typically identified as the "user"), continuously evolving with them, and aiding the human to remain consistent by offering suggestions and explanations. Frank is named after its frank behavior – it interacts with the user as soon as something "unexpected" happens. Other than Frank and the user, in line with [9], we also suppose a third agent, i.e., the user's *supervisor*. Depending on the context, the supervisor could be someone enforcing company policies to the user's decisions, e.g., making sure they are not biased by personal beliefs, or someone with higher expertise than the user, e.g., a senior doctor.

The pseudocode of Frank is reported in Algorithm 1. Frank requires a set of records to label X, which are received one by one, a set of rules R provided by the user's supervisor, a sensitive attribute sa, a skepticality threshold s, the number of iterations k after which a group fairness check is performed on the records and decisions analyzed so far, and a stopping condition *stp*. At this stage, we are very general about the stopping condition *stp* as it might be implemented as reaching a certain number of labeled records, or an accuracy higher than a threshold[2] for f. The initialization of $X', Y', \tilde{Y}, \hat{Y}, \ddot{Y}$ in line 1 can rely on empty sets for a cold start execution, or they might be initialized with records and decisions of previous runs. We use X' to collect the set of records analyzed so far, Y' for the set of final hybrid decisions taken on the records in X', $\tilde{Y}$ for the set of decisions of Frank's EFDT model f alone, $\hat{Y}$ for the set of decisions proposed by the user alone, and $\ddot{Y}$ to store the decisions taken by Frank and the user without re-labelling due to fairness corrections. Also, f might be completely untrained, pre-trained non-interactively on some records, or pre-trained in a past

[2] In our experiments, we consider as *stp* a certain number of instances to be analyzed, leaving for future work the study of measures automatically unveiling when to stop the training.

Algorithm 1: FRANK

Input : X - records to label, R - supervisor rule set, sa - sensitive attribute, s - skepticality thr, k - nbr of iter. for GFC, stp - stopping condition,

1 $X', Y', \tilde{Y}, \hat{Y}, \ddot{Y}, f \leftarrow initialize$; `// sets initialization`
2 **while** $stop \neq True$ **do** `// until a stop condition is met`
3 $\quad x_i \leftarrow receive\_record(X)$; `// receive a new un-label record`
4 $\quad \hat{y}_i \leftarrow user\_decision(x_i)$; `// get user decision`
5 $\quad \tilde{y}_i \leftarrow f(x_i)$; `// get model prediction`
6 $\quad$ **if** $ideal\_rule_R(x_i)$ **then** `// if` x_i `covered by expert rule`
IRC
7 $\quad\quad \bar{y}_i \leftarrow rule\_label_R(x_i)$; `// get` $\bar{y}_i$ `from rule`
8 $\quad\quad y_i \leftarrow \bar{y}_i$; `//` $\bar{y}_i$ `is compulsorily accepted`
9 $\quad$ **else if** $individual\_fairness_{sa}(x_i, X')$ **then** `// if` x_i `is similar to past records`
IFC
10 $\quad\quad y'_p \leftarrow get\_similar\_past\_label(x_i, X', Y')$; `// get` y'_p `from past records`
11 $\quad\quad$ **if** $\hat{y}_i \neq y'_p$ **then** `// conflict with a past decision`
12 $\quad\quad\quad \langle y_i, Y' \rangle \leftarrow solve\_conflict(x_i, y'_p, \hat{y}_i, Y')$; `// solve conflict & update` Y'
13 $\quad\quad$ **else** $y_i \leftarrow \hat{y}_i$; `// otherwise, user decision` $\hat{y}_i$ `is accepted`
14 $\quad$ **else if** $\hat{y}_i \neq \tilde{y}_i \wedge skept_s(f, x_i, \tilde{y}_i, \hat{y}_i, \ddot{Y}, \tilde{Y}, \hat{Y})$ **then** `// if clash & skepticism`
SLC
15 $\quad\quad$ **if** $is\_expl\_desired(x_i, \tilde{y}_i)$ **then** `// if an explanation for` $\tilde{y}_i$ `is desired`
16 $\quad\quad\quad e_i \leftarrow get\_and\_show\_expl(x_i, \tilde{y}_i, f, X')$; `// return explanation` e_i
17 $\quad\quad$ **if** $accept\_label\_change(x_i, \tilde{y}_i)$ **then** $y_i \leftarrow \tilde{y}_i$; `//` $\tilde{y}_i$ `is accepted`
18 $\quad\quad$ **else** $y_i \leftarrow \hat{y}_i$; `//` $\tilde{y}_i$ `is refused`
19 $\quad$ **else** $y_i \leftarrow \hat{y}_i$; `// otherwise` $\hat{y}_i$ `is accepted`
20 $\quad X' \leftarrow X' \cup \{x_i\}$; $Y' \leftarrow Y' \cup \{y_i\}$; $\ddot{Y} \leftarrow \ddot{Y} \cup \{y_i\}$; `// update sets`
21 $\quad \hat{Y} \leftarrow \hat{Y} \cup \{\hat{y}_i\}$; $\tilde{Y} \leftarrow \tilde{Y} \cup \{\tilde{y}_i\}$; $f \leftarrow update(f, x_i, y_i)$; `// update sets and model`
GFC
22 $\quad$ **if** $|Y'| \% k = 0$ **then** `// every` k `records`
23 $\quad\quad Y', f \leftarrow group\_fairness\_check_{sa}(X', Y', f)$; `// run GFC`

run of FRANK[3]. Until the stopping condition stp is met (line 2), FRANK receives a x_i from X (line 3). As in SL [19], the user assigns a label $\hat{y}_i$, and FRANK's model f a label $\tilde{y}_i$, i.e., the prediction (lines 4 and 5).

With *Ideal Rule Check* (IRC), FRANK checks if the record x_i is covered by a rule in the rule set R provided by the user's supervisor (line 7). If it is, then the decision $\bar{y}_i$ is derived from the rule and assigned to the final decision y_i (line 8). If none of the rules from R cover the record, with *Individual Fairness Check* (IFC), FRANK checks if the user's decision complies with the individual fairness condition by comparing $\hat{y}_i$ to the labels assigned to "similar" past records (lines 9-13). The definition of similarity is further defined below. *Skeptical Learning Check* (SLC) is triggered if no similar records exist and the user's decision $\hat{y}_i$ and FRANK's prediction $\tilde{y}_i$ are not the same. If FRANK is skeptical of $\hat{y}_i$, the user is asked if they want an explanation for $\tilde{y}_i$ (line 15). If the user accepts, they are shown the explanation e_i (line 16). Regardless, the user is then asked if they accept $\tilde{y}_i$ as the final decision y_i (lines 17). If the user refuses (line 18), if FRANK

[3] In our experiments, we consider the sets $X', Y', \tilde{Y}, \hat{Y}, \ddot{Y}$ initialized with empty sets and f pre-trained non-interactively on 500 records. Future works will investigate further these aspects.

is not skeptical, or if it agrees with the user (line 19), the user's decision $\hat{y}_i$ is accepted as the final decision y_i. Regardless of the triggered checks, x_i and y_i are added respectively to X' and Y' (line 20), and are used to update Frank's model f (line 21). Similarly, $\tilde{y}_i$ and $\hat{y}_i$ are added to $\tilde{Y}$ and $\hat{Y}$, respectively. Also, y_i is added to $\ddot{Y}$, which might differ from the set of labels Y' in the case of relabeling. Finally, every k records, Frank performs *Group Fairness Check* (GFC, lines 24-25), asking the user if they want to change the label of some past records to reduce the dataset's discrimination as computed by Prefential Sampling [11]. Frank prioritizes IRC to follow the guidelines of the supervisor, then IFC for fairness among similar records, and, finally, SLC. To avoid contradictions, once a final label y_i is set, checks with less priority are never triggered, and GFC cannot relabel records labeled by IRC or IFC. We stress that the user *has* to accommodate suggestions offered by IRC and IFC. On the other hand, the user is free to disregard suggestions by SLC and GFC. Depending on the use cases, certain checks might be turned off, e.g., IFC and GFC in health contexts. As some functions cycle the previously-seen records, Frank's algorithmic complexity is $O(n)$ with $n = |X'|$.

In the following, we provide a detailed explanation of Frank's four checks.

Ideal Rule Check. Each rule $r \in R$ includes a set of conditions and a label $\bar{y}$. The *ideal_rule* function checks if x_i follows the conditions of one of the rules in R (line 6), and if it is, it provides the label $\bar{y}_i$ (line 7), which is selected as the final decision y_i, regardless of the user's label $\hat{y}_i$. In case of divergence between the user's decision and the supervisor's rule, the user is notified that their decision is not compliant. Since IRC leaves no freedom of choice, the rules R should only cover very limited, specific, and ideal cases, describing records which should *absolutely* receive a certain label. The supervisor should also make sure the rules R are mutually exclusive. Besides, to avoid conflicts with fairness-related functions, the rules' conditions should not rely on sensitive attributes.

Individual Fairness Check. IFC is meant to reduce the pairs of records violating individual fairness condition, i.e., similar individuals should be treated similarly, by assessing if records similar to x_i received a different label than $\hat{y}_i$. Frank checks through the *individual_fairness* function (line 9) if there is at least one past record $x'_p \in X'$ identical or "similar" to the current record x_i. Given a binary sensitive attribute $sa \in A$, Frank defines two records x_i and x'_p *similar* if $v_j = v'_j \forall a_j \in A - \{sa\}$, i.e., x_i and x'_p are similar if they are identical, save for the value of sa. More than one similar or identical record $x'_p \in X$ can be found, and, by construction, they have all the same past label $y'_p \in Y'$ (line 10). If there is a disagreement between the current decision and past decisions, i.e., $y'_p \neq \hat{y}_i$ (line 11), then in line 12 *solve_conflict* prompts the user either to change their decision to make it compliant with past records, i.e., to select y'_p as y_i, or to keep the decision but relabel past records with $\hat{y}_i$, i.e., modifying the labels in Y'[4]. If the latter is chosen, f is also retrained, accounting for the modified labels. Otherwise, if $y'_p = \hat{y}_i$, x_i is assigned $\hat{y}_i$, i.e., the user's decision is accepted (line 13) as it is consistent with past records.

[4] Note that $\ddot{Y}$ is not modified, nor taken into account by IFC.

Skeptical Learning Check. If there is a disagreement between the decision of the user and f, i.e., $\hat{y}_i \neq \tilde{y}_i$, the *skept* function (line 14) computes FRANK's *skepticality* following Eq. 1. If it is higher than s, FRANK is skeptical. Empirical accuracy values are initialized as in Sec. 2. We emphasize that *skept* does not take as input Y', i.e., the set of decisions after possible re-labeling, but $\dddot{Y}$, i.e., the set of decisions made by the user after FRANK's checks for each record[5]. If skeptical, FRANK proposes $\tilde{y}_i$ for y_i, and asks the user if they want an explanation e_i (line 15). The user is then asked to accept $\tilde{y}_i$ (line 17). The user has the full veto power against FRANK, and if they reject $\tilde{y}_i$, the user label is collected as the final decision y_i (line 18). If the user accepts to see an explanation, FRANK runs the *get_and_show_expl* function and provides it to the user (line 16). FRANK can provide *Logic-based Explanations*, where a global representation of the EFDT is shown alongside the local decision rule followed for the record x_i and $\tilde{y}_i$ (such as *IF Years_of_Experience > 5 AND Attitude = True THEN Hire*), or *Instance-based Explanations*, i.e., records similar to x_i which can be either *real* or *synthetic*. These records are classified by f either with $\tilde{y}_i$, i.e., an *example* of FRANK's decision, or $\hat{y}_i$, i.e., a *counter-example*. FRANK's explanations are the result of a *co-evolutionary relationship* with the user, leading to more detailed justifications over time. Thus, the user should progressively trust FRANK more.

Group Fairness Check. GFC checks if one of the value of a binary sensitive attribute $sa \in A$ are discriminating against the other group w.r.t. Y'. GFC is independent from the other checks, and it is always triggered every k records (see lines 22-23). FRANK computes *disc* and the DN, DP, PN, and PP groups of the set of records X' w.r.t. the labels Y', following [14]. Then, it orders the records from DN and PP following the prediction probability of f, and calculates how many of them should be removed. Finally, the records with higher probability are shown to the user, who can choose to change their label. The new labels replace the older ones in Y', and f is retrained from scratch. Thus, GFC is an interactive implementation of PS, where the user is made aware of their discriminating behavior and is asked to relabel past records to mitigate the discrimination.

4 Experiments

We evaluated FRANK[6] on three real-world datasets and, in line with [4, 10], we employed simulated users to assess its impact in a variety of conditions.

Users. We employed five kinds of *simulated* users: the *Real Expert*, who always makes decisions following the ground truth (which is unknown in a real scenario), the *Absent-Minded*, an easily-distracted expert who follows the ground truth 75% of times, the *Coin-Tosser*, who makes decisions by flipping a coin, and the *Bayesian* and *Similarity* experts, simulated by Naive Bayes and KNN [18]. For IFC, we suppose that all the users have conservative behavior w.r.t. their past decisions, with 80.00% of chance of changing the label assigned to the current record x_i, instead of re-labeling past records. For SLC, we set a threshold s

[5] Y' and $\dddot{Y}$ coincide until the user relabel older records if prompted by IFC or GFC.
[6] The Python code is available here: `https://github.com/FedericoMz/Frank`.

Table 1. Ablation study of FRANK's checks.

		None					*oIRC*					*oIFC*					*oGFC*				
		CA	MA	CD	MD	UC	CA	MA	CD	MD	UC	CA	MA	CD	MD	UC	CA	MA	CD	MD	UC
`Adult`	*Real*	1.0	.83	.21	.18	7.0	.96	.82	.23	.15	7.0	1.0	.84	.22	.17	0.0	.84	.75	-.02	.01	6.0
	Abs.	.75	.77	.10	.09	5.3	.74	.76	.13	.09	5.3	.75	.77	.11	.12	0.0	.78	.76	.01	.04	4.2
	Coin	.50	.56	.00	.02	5.6	.52	.51	.03	-.01	5.6	.50	.52	.00	.00	0.0	.55	.55	.03	.04	5.3
	Bayes	.80	.77	.12	.07	0.0	.79	.76	.11	.09	0.0	.80	.77	.12	.07	0.0	.80	.77	.09	.09	0.0
	Sim.	.79	.76	.20	.24	1.0	.79	.76	.20	.24	1.0	.79	.76	.20	.24	0.0	.80	.77	.03	.17	0.0
`COMPAS`	*Real*	1.0	.69	-.14	-.21	42.	.65	.61	-.15	-.21	18.	.98	.68	-.14	-.24	0.0	.75	.64	-.06	-.15	17.
	Abs.	.75	.63	-.07	-.19	50.	.60	.61	-.12	-.21	24.	.74	.64	-.08	-.19	0.0	.64	.62	-.03	-.17	24.
	Coin	.50	.57	.00	-.17	56.	.54	.55	-.09	-.08	27.	.50	.52	-0.0	-.09	0.0	.49	.48	.01	-.01	32.
	Bayes	.63	.63	-.20	-.19	0.0	.59	.62	-.18	-.25	0.0	.63	.63	-.20	-.19	0.0	.61	.63	-.15	-.18	0.0
	Sim.	.63	.66	-.31	-.18	25.	.58	.62	-.21	-.25	15.	.62	.66	-.28	-.17	0.0	.63	.66	-.01	-.21	17.
`HR`	*Real*	1.0	.93	-.01	.00	39.	.99	.89	-.02	0.02	39.	.98	.93	-.01	.00	.99	.94	.93	.00	.00	31.
	Abs.	.75	.93	-.01	.00	24.	.74	.92	-.01	-0.01	24.	.74	.93	.00	.00	0.0	.85	.93	.00	.00	20.
	Coin	.50	.93	-.01	.00	21.	.50	.83	-.02	-.06	21.	.50	.93	.01	.00	0.0	.62	.65	.01	.06	19.
	Bayes	.89	.92	.00	-.02	0.0	.89	.92	.00	-.02	0.0	.89	.92	.00	-.02	0.0	.90	.93	.00	.00	0.0
	Sim.	.89	.93	.00	.00	0.0	.89	.91	.00	-.02	0.0	.89	.93	.00	.00	0.0	.89	.93	.00	.00	0.0

of 0.05, increasing the times FRANK is skeptical. We assumed that the users can always *accept* or *decline* FRANK's suggestions, or *randomly* choose. For Bayesian and Similarity experts, we also envisioned users who request *explanations*, i.e., five synthetic examples and counterexamples, monitoring their reaction[7]. If they agree with more than half, they accept FRANK's suggestions. For GFC, we suppose that the user selects to re-label the top 25% DN and PP records.

Datasets. The `Adult`, `COMPAS` and `HR` datasets[8] simulate classification tasks for granting credits, predicting recidivism, or giving a promotion, i.e., possible real use-cases for FRANK. In `HR`, only 8% of records belong to the positive class, compared to the 25% and the 50.00% in `Adult` and `COMPAS`, which are, however, highly discriminating [17]. In contrast, `HR` is fair w.r.t. *Sex*. After removing duplicated or incomplete records, we randomly selected 2,000 records to incrementally train FRANK, i.e., X. We set labeling all the records in X as our stopping condition *stp*. The Naive Bayes and KNN models were trained on an additional 500 records. Half of them were also used to pre-train FRANK's ML model f. Finally, a dataset X_T includes 500 records reserved to test f. For IRC, we set the following rules: for `Adult`, IF *capital_gain* > 9000 THEN $\bar{y} = +$; for `COMPAS`, IF *priors_count* > 0 THEN $\bar{y} = +$; for `HR`, IF *awards_won* $=$ *True* THEN $\bar{y} = +$.

Evaluation Measures. We measured the *Co-evolutionary Accuracy* (CA) by comparing Y' with the ground truth Y, and the Model Accuracy (MA) by comparing the prediction of f on X_T with its ground truth Y_T. Likewise, we measured the *Co-evolutionary Discrimination* (CD) and the Model Discrimination (MD). The *disc* score was computed towards *Female* for all datasets[9]. Finally, we counted the number of Unfair Couples (UC), i.e., similar records violating individual fairness by receiving a different label. Ideal values are 1 for CA and MA, 0 for the others. Each experiment was repeated 10 times. The tables report the average results, standard deviations are very low and not reported.

[7] As synthetic records lack a ground truth, this option cannot be implemented with the other users.
[8] `kaggle.com/datasets/`.
[9] Note that a negative *disc* implies that *Male* is discriminated.

Results. As an ablation study of Frank's structure, in Table 1, we report the results when *None* of Frank's functions are enabled, and when only IRC, IFC, or GFC are enabled (*oIRC*, *oIFC*, *oGFC*). The impact of IRC is minimal on `Adult` and `HR`, whereas it negatively affects all the experts except for the *Coin-Tosser* in `COMPAS`. This is probably due to the selected rules, either too narrow in scope or inaccurate. These results highlight the importance of selecting good rules for Frank. On the other hand, comparing the *oIFC* and *oGFC* columns to *None*, we can see a significant improvement in terms of fairness. IFC always successfully minimizes *UC* with no side effects, whereas GFC consistently reduces both *CD* and *MD*. For `Adult` and `COMPAS` and with the *Real Expert*, this is at the expense of *CA* and *MA*. However, we should stress that the "accuracy" of very biased datasets does not necessarily mirror "right" decisions. In fact, on the already balanced `HR`, the impact on *CA* and *MA* with the *Real Expert* is minimal. Additionally, with `Adult` and `HR`, GFC improves the accuracy of *Absent-Minded* and *Coin-Tosser* experts without negatively impacting the model-based ones.

Table 2 compares traditional SL [19] with Frank with everything enabled, except for IRC in `COMPAS`. As mentioned for IRC, Frank consistently minimizes UC. In `Adult`, Frank provides each expert better CA and MA if they always accept the suggestions, whereas CD and sometimes MD is slightly better with SL. By declining the suggestions or randomizing the choices with SL, the *Real Expert* gets better CA and MA, but worse CD and MD. With other experts, Frank is better than, or very close to, SL for CA and MA, while consistently improving CD and MD. In `COMPAS`, Frank always has a better CD, and often a better MD. When the *Real Expert* and the *Absent-Minded* randomize or decline, this is at the expense of CA and, to a lesser extent, MA, with a strong fairness-performance trade-off. In other cases, Frank performs a bit better or on par with SL. As for `HR`, the two methods are very close for the *Real*, *Bayesian*, and *Similarity* experts, with SL slightly better. With the *Absent-Minded* and the *Coin-Tosser*, declining or randomizing decision greatly enhances the CA. In fact, the randomizing *Coin-Tosser* reaches a CA comparable to the *Absent-Minded*'s. Also, with the same example we can notice a lower MA than SL's. This might be due to the fact that IRC, IFC, and GFC are not triggered when f makes decisions on X_T.

Figure 1 shows CD and CA over time for different experts, randomly accepting the suggestions from Frank and SL. Plots are in log scale along the x-axis. At first, for each user Frank and SL follow a similar pattern, both in terms of CA and CD. Their lines then diverge due to fairness interventions. In `Adult`, this results in a drop of CA for the *Real Expert*, and in `COMPAS` also for the *Absent-Minded*. In `HR`, the *Real Expert* is far less affected, as the dataset is less biased. On the other hand, the *Absent-Minded* and *Coin-Tosser* receive a noticeable boost in terms of CA. In `Adult` and `COMPAS`, the *Real* and the *Similarity* experts make biased decisions while paired with SL, whereas their CD with Frank is near 0. Frank's CD lines tend to converge to 0 for all the datasets.

Table 3 compares the impact of having users accepting Frank's suggestions randomly (RND) against users deciding on top of Frank's explanations (XAI).

Table 2. FRANK vs traditional SL. Best scorer in bold, parity in italics.

			Real Expert		*Absent-Minded*		*Coin-Tosser*		*Bayesian*		*Similarity*	
			SL	FRANK	SL	FRANK	SL	FRANK	SL	FRANK	SL	FRANK
Adult	*accept*	CA	0.74	**0.78**	0.73	**0.78**	0.73	**0.78**	0.73	**0.78**	0.74	**0.77**
		MA	0.66	**0.75**	0.66	**0.75**	0.65	**0.75**	0.64	**0.75**	0.74	**0.75**
		CD	**0.02**	0.05	**0.03**	0.05	**0.04**	0.05	**0.04**	0.05	**0.00**	0.01
		MD	-0.10	**0.05**	-0.06	**0.05**	**-0.01**	0.05	**0.00**	0.05	-0.02	**0.01**
		UC	1.00	**0.00**	1.00	**0.00**	1.00	**0.00**	1.00	**0.00**	*0.00*	*0.00*
	decline	CA	**1.00**	0.83	0.75	**0.77**	0.50	**0.57**	**0.80**	0.79	0.79	**0.80**
		MA	**0.83**	0.75	**0.77**	0.75	0.56	**0.58**	**0.77**	0.76	0.76	**0.77**
		CD	0.21	**0.03**	0.11	**0.01**	*-0.01*	*-0.01*	0.12	**0.11**	0.20	**0.03**
		MD	0.18	**0.05**	0.12	**0.05**	**-0.04**	0.05	**0.07**	0.09	0.24	**0.17**
		UC	7.00	**0.00**	6.10	**0.00**	5.60	**0.00**	0.00	**0.00**	1.00	**0.00**
	random	CA	**0.89**	0.80	0.74	**0.77**	0.55	**0.57**	*0.79*	*0.79*	*0.77*	*0.77*
		MA	**0.76**	0.75	0.71	**0.75**	*0.58*	*0.58*	*0.76*	*0.76*	0.73	**0.75**
		CD	0.14	**0.03**	0.09	**0.01**	0.04	**0.01**	0.10	**0.08**	0.14	**0.00**
		MD	0.09	**0.05**	**0.04**	0.05	0.09	**-0.01**	**0.07**	0.08	0.14	**0.02**
		UC	4.60	**0.00**	3.10	**0.00**	4.20	**0.00**	0.10	**0.00**	0.60	**0.00**
COMPAS	*accept*	CA	*0.52*	*0.52*	0.51	**0.53**	0.52	**0.54**	*0.52*	*0.52*	0.55	**0.58**
		MA	**0.52**	0.51	0.49	**0.54**	0.49	**0.55**	**0.52**	0.51	0.56	**0.62**
		CD	-0.02	**0.00**	-0.01	**0.00**	-0.01	**0.00**	-0.02	**0.00**	-0.09	**-0.05**
		MD	**-0.02**	-0.04	**0.01**	-0.07	**0.01**	-0.08	**-0.02**	-0.04	**-0.03**	-0.11
		UC	8.00	**0.00**	17.60	**0.00**	17.60	**0.00**	8.00	**0.00**	1.00	**0.00**
	decline	CA	**1.00**	0.77	**0.75**	0.64	*0.50*	*0.50*	**0.63**	0.61	**0.63**	0.62
		MA	**0.69**	0.66	**0.65**	0.62	**0.53**	0.52	*0.63*	*0.63*	**0.66**	0.65
		CD	-0.14	**0.00**	-0.06	**-0.01**	0.01	**0.00**	-0.20	**-0.15**	-0.31	**-0.04**
		MD	-0.21	**-0.19**	-0.21	**-0.13**	**-0.07**	-0.10	-0.19	**-0.18**	*-0.18*	*-0.18*
		UC	42.00	**0.00**	50.00	**0.00**	51.10	**0.00**	*0.00*	*0.00*	25.00	**0.00**
	random	CA	**0.80**	0.66	**0.65**	0.58	0.50	**0.54**	**0.62**	0.60	*0.62*	*0.62*
		MA	**0.64**	0.61	**0.57**	0.56	0.48	**0.57**	**0.63**	0.60	0.65	0.65
		CD	-0.15	**-0.01**	-0.10	**0.00**	-0.02	**-0.01**	-0.18	**-0.08**	-0.23	**-0.05**
		MD	-0.16	**-0.12**	-0.17	**-0.11**	**0.00**	-0.08	-0.18	**-0.15**	-0.17	**-0.16**
		UC	34.00	**0.00**	45.20	**0.00**	43.30	**0.00**	3.20	**0.00**	23.70	**0.00**
HR	*accept*	CA	**0.90**	0.89	**0.90**	0.86	**0.90**	0.88	**0.90**	0.89	**0.90**	0.89
		MA	**0.93**	0.92	**0.93**	0.89	**0.93**	0.91	**0.93**	0.92	**0.93**	0.92
		CD	*0.00*	0.01	*0.00*	0.01	*0.00*	*0.00*	*0.00*	*0.00*	*0.00*	*0.00*
		MD	**0.00**	-0.02	**0.00**	-0.02	**0.00**	-0.02	**0.00**	-0.02	**0.00**	-0.02
		UC	*0.00*	*0.00*	*0.00*	*0.00*	*0.00*	*0.00*	*0.00*	*0.00*	*0.00*	*0.00*
	decline	CA	**1.00**	0.91	0.75	**0.83**	0.50	**0.66**	*0.89*	*0.89*	*0.89*	*0.89*
		MA	**0.93**	0.92	**0.93**	0.92	**0.90**	0.71	*0.92*	*0.92*	**0.93**	0.92
		CD	*-0.01*	*-0.01*	-0.01	**0.00**	*-0.01*	*-0.01*	*0.00*	*0.00*	*0.00*	*0.00*
		MD	**0.00**	-0.02	-0.01	**0.00**	**-0.01**	0.05	*-0.02*	*-0.02*	**0.00**	-0.02
		UC	39.00	**0.00**	26.9	**0.00**	22.6	**0.00**	*0.00*	*0.00*	*0.00*	*0.00*
	random	CA	**0.95**	0.91	0.82	**0.86**	0.70	**0.82**	*0.89*	*0.89*	*0.89*	*0.89*
		MA	**0.93**	0.92	**0.93**	0.90	**0.93**	0.86	**0.93**	0.92	**0.93**	0.92
		CD	-0.01	**0.00**	*0.00*	*0.00*	**0.00**	0.01	*0.00*	*0.00*	*0.00*	*0.00*
		MD	**0.00**	-0.02	**0.00**	-0.02	**0.00**	0.03	**0.00**	-0.02	**0.00**	-0.02
		UC	18.10	**0.00**	17.40	**0.00**	16.20	**0.00**	*0.00*	*0.00*	*0.00*	*0.00*

The first three rows report the percentage of Agreements, Skepticism, and Disagreement between the user and FRANK. We notice that they tend to agree, and the disagreement almost always leads to skepticism. The fourth and fifth rows show the percentage of the Accepted and Declined FRANK's suggestions. When XAI is used, we observe a lower agreement rate (Agr) in `Adult` and `COMPAS`, but ultimately, looking at the acceptance rate (Acc), these users rely on FRANK more than their randomizing counterparts, also resulting in a better CD at the expense of CA. This confirms that FRANK is able to provide satisfying explanations to the *Bayesian* and *Similarity* users. We underline that the *Similarity* expert on `Adult` is the exception, as they tend to decline. Finally, in `HR`, SLC

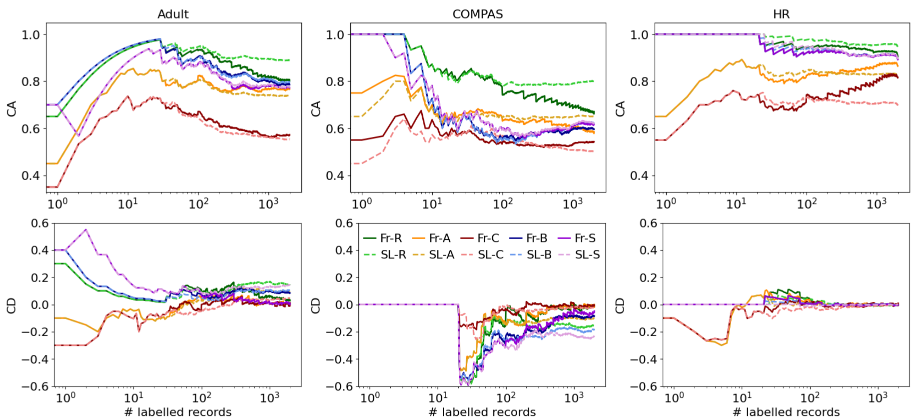

Fig. 1. CA and CD evolution over time with different experts.

Table 3. Users accepting suggestions randomly (RND) or w.r.t. explanations (XAI).

	Adult				COMPAS				HR			
	Bayesian		*Similarity*		*Bayesian*		*Similarity*		*Bayesian*		*Similarity*	
	RND	XAI	RND	XAI	RND	XAI	RND	XAI	RND	XAI	RND	XAI
Agr %	96.38	88.72	77.14	77.37	89.64	74.53	68.83	60.42	100.00	100.00	99.12	99.16
Ske %	3.49	11.20	22.47	22.47	10.32	25.41	31.04	39.45	0.00	0.00	0.79	0.73
Dis %	0.11	0.05	0.37	0.15	0.03	0.05	0.12	0.13	0.00	0.00	0.08	0.10
Acc %	51.99	94.03	49.61	37.58	50.49	93.94	50.37	74.02	N/A	N/A	54.30	0.00
Dec %	48.01	5.97	50.39	62.42	49.51	6.06	49.63	25.98	N/A	N/A	45.70	100.00
CA	0.79	0.77	0.77	0.76	0.60	0.53	0.62	0.59	0.89	0.89	0.89	0.89
CD	0.08	0.03	0.0	0.01	-0.08	-0.01	-0.05	-0.02	0.00	0.00	0.00	0.00

was never triggered by the *Bayesian*, and only 14 times by the *Similarity* expert (who then declined the 14 suggestions, hence the anomalous percentage).

5 Conclusion

We have presented Frank, a system based on Skeptical Learning that evolves with the user. Compared to traditional SL, Frank checks the fairness of the decisions, if they are compliant with external rules, and provides explanations for the suggestions. Through these additional functions, Frank successfully improves the fairness of the datasets and of the model, often outperforming SL in terms of accuracy, especially with less-skilled users. Moreover, we noticed that our simulated users accept Frank's explanations most of the time. However, at the moment, Frank is limited to tabular data and better suitable to those of low dimensionality. Future works might extend Frank to other data types and decision models, explore alternative stopping conditions, and focus on the Frank-user relationships. For example, Frank could build trust or distrust towards the user, and react accordingly. Finally, after being trained in the co-evolutionary process, Frank's model f could be used within a Learning-to-Defer system, with Frank making decisions and asking the user when uncertain.

Acknowledgment. This work is partially supported by the EU NextGenerationEU programme under the funding schemes PNRR-PE-AI FAIR, PNRR-SoBigData.it - Prot. IR0000013, H2020-INFRAIA-2019-1: Res. Infr. G.A. 871042 *SoBigData++*, TANGO G.A. 101120763, and ERC-2018-ADG G.A. 834756 *XAI*. We thank Andrea Bontempelli for his insights on Skeptical Learning.

References

1. A. Bifet and R. Gavaldà. Adaptive learning from evolving data streams. In *IDA*, volume 5772 of *LNCS*, pages 249–260. Springer, 2009.
2. R. Binns. On the apparent conflict between individual and group fairness. In *FAT**, pages 514–524. ACM, 2020.
3. A. Blanco-Justicia et al. Machine learning explainability through comprehensible decision trees. In *CD-MAKE*, volume 11713 of *LNCS*, pages 15–26. Springer, 2019.
4. A. Bontempelli, S. Teso, et al. Learning in the wild with incremental skeptical gaussian processes. In *IJCAI*, pages 2886–2892. ijcai.org, 2020.
5. P. Domingos et al. Mining high-speed data streams. In *KDD*, page 71. ACM, 2000.
6. R. Guidotti. Counterfactual explanations and how to find them: literature review and benchmarking. *Data Mining and Knowledge Discovery*, pages 1–55, 2022.
7. R. Guidotti, A. Monreale, S. Ruggieri, et al. A survey of methods for explaining black box models. *ACM CSUR*, 51(5):93:1–93:42, 2019.
8. E. Hüllermeier and W. Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *ML*, 110(3):457–506, 2021.
9. D. Jarrett et al. Online decision mediation. In *NeurIPS*, 2022.
10. S. Joshi, S. Parbhoo, and F. Doshi-Velez. Pre-emptive learning-to-defer for sequential medical decision-making under uncertainty. *CoRR*, abs/2109.06312, 2021.
11. F. Kamiran and T. Calders. Classification with no discrimination by preferential sampling. In *BNAIC*, volume 1. Citeseer, 2010.
12. M. D. Lange et al. A continual learning survey: Defying forgetting in classification tasks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(7):3366–3385, 2022.
13. C. Manapragada, G. I. Webb, and M. Salehi. Extremely fast decision tree. In *KDD*, pages 1953–1962. ACM, 2018.
14. F. Mazzoni et al. Genfair: A genetic fairness-enhancing data generation framework. In *DS*, volume 14276 of *LNCS*, pages 356–371. Springer, 2023.
15. K. L. Mosier et al. Human decision makers and automated decision aids: Made for each other? In *Automation and human performance*, pages 201–220. CRC, 2018.
16. D. Pessach and E. Shmueli. A review on fairness in machine learning. *ACM CSUR*, 55(3):51:1–51:44, 2023.
17. T. L. Quy et al. A survey on datasets for fairness-aware machine learning. *WIREs Data Mining Knowl. Discov.*, 12(3), 2022.
18. P.-N. Tan et al. *Introduction to data mining.* Pearson Education India, 2016.
19. S. Teso, A. Bontempelli, F. Giunchiglia, and A. Passerini. Interactive label cleaning with example-based explanations. In *NeurIPS*, pages 12966–12977, 2021.
20. S. Verma et al. Fairness definitions explained. In *ICSE*, pages 1–7. ACM, 2018.
21. L. Wang et al. A comprehensive survey of continual learning: Theory, method and application. *CoRR*, abs/2302.00487, 2023.
22. T. Wang and M. Saar-Tsechansky. Augmented fairness: An interpretable model augmenting decision-makers' fairness. *CoRR*, abs/2011.08398, 2020.
23. M. Zeni et al. Fixing mislabeling by human annotators leveraging conflict resolution and prior knowledge. *ACM IMWUT*, 3(1):32:1–32:23, 2019.
24. W. Zhang. Personal context recognition via skeptical learning. In *IJCAI*, pages 6482–6483. ijcai.org, 2019.