Interpretable Fair Distance Learning for Categorical Data

Interpretable Fair Distance Learning for Categorical Data

Alessio Famiani[1], Federico Peiretti[1][0000−0001−7648−162X], and Ruggero G. Pensa[1][0000−0001−5145−3438] (✉)

Department of Computer Science, University of Turin, Italy
{alessio.famiani,federico.peiretti,ruggero.pensa}@unito.it

Abstract. Categorical features are widespread in many decision support systems relying on personal and sensitive data, such as credit scoring or personalized medicine and are not exempt of bias and fairness concerns. Unfortunately, bias mitigation techniques based on representation learning for categorical data are poorly studied and most solutions are limited to using the same approaches designed for numeric data on one-hot encoded features. To fill this gap, we propose FairDILCA, a fair extension of a known framework for learning distances on categorical data, which exploits co-distributions of attributes values for computing distances. FairDILCA considers the correlation of the features w.r.t. the protected one to create an unbiased representation of the data, making any subsequent analysis and learning task fairer. Furthermore, it also represents a more interpretable option than typical representation learning approaches, since it relies on deterministic and clear computational steps. Thanks to extensive experiments, we show the effectiveness of our framework also when applied to a classification task and in comparison with a state-of-the-art method pursuing a similar objective.

Keywords: Categorical features · Distance learning · Fairness.

1 Introduction

Throughout the years, machine learning gained an increased notoriety for several infamous incidents that harmed people by reinforcing prejudices and discriminatory patterns. Such events have raised awareness of the topic and gave birth to a more trustworthy branch of AI, called Responsible AI, other than to laws, regulation and guidelines [10] aimed at dealing with ethical issues like privacy, explainability, transparency, fairness, or environmental sustainability [28]. Some of these concerns have origin from prejudices encoded in the data used to train ML models, either because the captured phenomenon is itself biased or for other factors. In fact, a dataset recording information about individuals may contain the so called *protected features* (i.e. gender, sexual orientation, etc...), details that can be used to infer membership to some social categories of people, hence can be exploited in a decision making setting and may result in unfair outcomes. Direct discrimination, known under the name of "disparate treatment" [3], occurs when a protected attribute is used explicitly in the decision process and it is prohibited by various laws and regulations. The simple omission or oversight of such features may not be enough to solve all unfairness issues within a dataset. In fact, other features could encode parts of the sensitive information and act as proxies for the protected characteristics. This is especially true in high-dimensional cases. This more indirect and maybe involuntary manner of discriminating is closer to the notion of "disparate impact" [3]. Consequently, a sub-branch of Responsible AI, called Fair ML, studies debiasing strategies with the goal of dealing with discrimination and prejudices in data and models [24]. Most existing fairness-aware ML techniques are task-dependent and try to mitigate discriminating descriptive or predictive patterns in the output model [31, 1]. Another branch of research focuses on learning new fair, and often latent, representations of data, hence considered task-independent [30, 25]. Most state-of-the-art approaches assume that the data are numeric or that categorical data have either been one-hot encoded or roughly handled. However, it has been shown that commonly used data engineering steps within the ML pipeline, such as data encoding [20] or other data transformations techniques [6], can have a big impact on fairness. For example, evidences found in studies like these suggest that one-hot encoding tends to discriminate more in terms of equality of opportunity than target encoding and that transformation techniques, especially the ones altering data distribution, along with factors like dataset size or the classifier choice, can have a harmful influence on fairness as well [20]. Nonetheless, categorical attributes are ubiquitous in tabular data and in many sensitive applications using personal data, such as credit approval, hiring and healthcare decision support systems. In addition, very few works in literature focus on learning distances and stating similarities between objects in a fair manner [21, 34, 16, 32], especially on categorical data.

To fill this gap, we propose a method to learn fair distances in categorical data starting from a known framework for distance computation called DILCA [11], which assumes that the co-distributions of feature values within the dataset can help define more accurate distances between attribute values. Nevertheless, distances computed by DILCA are based on data and they are not immune to fairness concerns. Consequently, its application can result in biased outcomes, both in the computed distances and in the subsequent learning tasks based upon them. Hence, in this paper we introduce an extension of DILCA, called FairDILCA, that includes fairness considerations in all computational steps in order to produce a debiased version of the pairwise distance matrix. We show experimentally that FairDILCA can lead to fairer distances and, consequently to less discriminating usages of them in typical ML tasks, also compared to a recent state-of-the-art pre-processing technique [29]. To the best of our knowledge, FairDILCA is the first framework specifically tailored to categorical data. Interestingly, since all algorithms implementing the framework are deterministic and based on fully interpretable steps, the outcomes of the overall framework can be easily explained, thus meeting transparency requirements.

2 Background and Related Work

In this section, we present the background notions required to understand how our framework works, and a literature overview of related concepts in the field.

2.1 Fair ML approaches for categorical data

Fair ML refers to algorithmic solutions used to assess and mitigate biases in various steps of the ML pipeline. In literature, fair ML algorithms can be roughly grouped into three types of approaches [2] depending on where the bias reduction process is employed within the pipeline. Some approaches are based on *in-processing* and *post-processing* techniques. The former apply the debiasing procedure at training time [14, 33], while the latter focus on transforming a model output considering protected attribute values and group membership [13]. Our method, instead, can be classified as a pre-processing technique.

Pre-processing methods apply bias mitigation ahead of the learning process, and focus on creating a new representation or sample of the data containing less cues about protected features. This can be achieved, for instance, by randomly mapping each sample while considering fairness, utility and distortion constraints [7] or reweighing instances depending on label and group membership or sampling data according to the inferred weights [12]. Studies regarding fair representation through Adversarial Learning are pretty popular as well, and they usually rely on *Generative Adversarial Networks* (GANs) to be able to generate unbiased synthetic data [30] or on learning representations for individual fairness via logical constraints [25]. The main advantage of these approaches is that they usually are agnostic to tasks and models and, because of this, in literature they are considered among the most flexible solutions. However, representations constructed in the attempt of obfuscating sensitive information might belong to latent spaces, thus raising explainability and transparency concerns. Instead, in [29], the authors define a method that applies a linear transformation to the non-sensitive feature columns in order to remove their correlation with the sensitive ones while retaining as much information as possible. Even in this case, the transformation can not be applied on categorical features directly.

The majority of state-of-the-art studies on Fair ML for categorical data are task-dependent and involve in-process mechanisms using basic distance measures (such as the overlap) to effectively deal with categorical data as in [26]. Instead, in [1] a regularisation term is integrated into the objective function and a clustering is considered fair if it is both coherent, when measured on non-sensitive attributes, and approximate data distribution when measured on protected features. For what concerns fair distance learning, few works exists in literature [21], [34], [16], [32]. For example, [34] describe a way for computing distances between two instances by leveraging on causal notions with the aim of focusing on a subset of relevant non protected attributes. In [16] instead, a weighted euclidean distance is learnt following the idea that features with the highest impact on the target should contribute the most. However, again, categorical data is usually encoded in a raw numeric fashion or handled using overlap measures.

This unrefined treatment of such type of data, other than diminishing utility, can introduce biases and have a major impact on model fairness. To the best of our knowledge, there is no literature on fair distance learning on categorical attributes.

2.2 The DILCA framework

DILCA [11] is an unsupervised framework created with the purpose of learning context-based distances between pairs of values of a categorical attribute. This is achieved by seeing how these values and those of other attributes are co-distributed within a given dataset. The underlying key idea is that the context is crucial for defining similarities between values, thus between objects. For example, consider an attribute like *Animal* whose values are inside the following set $\{Cat, Dog, Cougar\}$, depending on the context, *Cat* could be more similar to *Dog*, if one is talking about pets, or to *Cougar*, if one is talking about felines.

DILCA has two major steps: *context selection*, where, for a given target attribute, a subset of attributes considered relevant is computed; *distance computation*, where the distances between attribute pair values are measured and can be used to calculate the distance between objects. More formally, given a dataset of n data objects $D = \{o_1, \dots o_n\}$ each described by a set of m categorical features denoted by $F = \{X_1, X_2, ..., X_m\}$ and a feature $Y \in F$, referred to as the target[1], i.e., the one on which the context computation is required, the relevant features collected for the target form the so called context of Y, denoted by $C_Y \subseteq F \backslash \{Y\}$. Additionally, the j-th value of the i-th feature X_i of the dataset is referred as $x_j \in X_i$ and, similarly, y_i denotes the i-th value of Y.

In DILCA, the context selection is performed in two different ways: a parametric one called $DILCA_M$, and an automatic feature selection one called $DILCA_{RR}$. They both use an information gain based metric called *Symmetric Uncertainty* (SU), to measure the correlation between two attributes. It is defined as:

$$SU(Y, X) = 2 \cdot \frac{H(Y) - H(Y|X)}{H(Y) + H(X)} \tag{1}$$

where H is the *entropy*, determined as:

$$H(Y) = -\sum_i P(y_i) log_2(P(y_i))$$

$P(y_i)$ being the probability of the value y_i of Y. Analogously,

$$H(Y|X) = -\sum_j P(x_j) \sum_i P(y_i|x_i) log_2(P(y_i))$$

where $P(y_i|x_i)$ is the probability that $Y = y_i$ given the observation $X = x_i$. In the remainder of the paper, the notation $SU_Y(X_i)$ indicates the Symmetric Uncertainty of X_i for the target Y.

[1] Not to be confused with the task-related target (class) variable.

The parametric method $DILCA_M$ add X_i into the context of a target feature Y if $SU_Y(X_i)$ is above the mean value, $\overline{SU}_Y$, whose influence is controlled by a parameter $\sigma \in [0, 1]$. Formally, the mean of SU_Y is determined as follows:

$$\overline{SU}_Y = \frac{\sum_{x_i \in F \setminus Y} SU_Y(X_i)}{|F \setminus \{Y\}|}$$

and the context for a given target variable Y is built by collecting attributes X_i that satisfy the following condition:

$$C_Y = \{X_i \in F | X_i \neq Y \wedge SU_Y(X_i) \geq \sigma \cdot \overline{SU}_Y\}.$$

$DILCA_{RR}$, instead, is based on a feature selection algorithm that first identifies a set of significant features and then removes the ones which are considered redundant [23]. More precisely, in the *relevancy step*, features $X_i \neq Y$ are ranked by decreasing order of $SU_Y(X_i)$. In the *redundancy step* features that carry similar informative details are removed from the ranking obtained in the previous step. Given two attributes, X_1 and X_2, both considered relevant to Y, X_2 is considered redundant w.r.t. X_1 if $SU_{X_1}(X_2) \geq SU_Y(X_2)$.

Once an appropriate context is returned for a target feature Y, the distance $d(y_i, y_j)$ between every pair of its values (y_i, y_j) can be determined as follows:

$$d(y_i, y_j) = \sqrt{\frac{\sum_{X_i \in C_Y} \sum_{x_k \in X_i} \sum (P(y_i|x_k) - P(y_j|x_k))^2}{\sum_{X_i \in C_Y} |X_i|}} \tag{2}$$

Finally, after all the distances between categorical attributes values X_i are determined and stored in a matrix $\mathcal{M}_{X_i}$, the usual distance measures can be applied, e.g. the one below:

$$d(o_k, o_j) = \sqrt{\sum_{X_i \in F} \mathcal{M}_{X_i}(o_k.X_i, o_j.X_i)^2} \tag{3}$$

where o_k is the k-th data object and $o_k.X_i$ the value of attribute X_i in o_k.

DILCA has been applied in different distance-based application scenarios, such as ensemble learning [27] and clustering of large data [18]. More recently, in the context of Responsible AI, a differentially-private extension has been proposed [4]. Our contribution, called FairDILCA, extends DILCA to meet fairness needs, resulting in a task-agnostic approach that natively supports categorical data and is both more explainable and transparent.

3 The FairDILCA framework

The application of DILCA can reinforce existing biases in data, either because they are directly encoded by sensitive features or indirectly by features that act as proxies. Consider for instance a simple dataset like the one in Table 1 that contains details about hiring outcomes in the tech industry. Each person is described by four characteristics: their gender, the job position they applied to, their level of experience for the job in question, if they apply for a part-time job or not, plus whether they were hired and got the job or rejected. Intuitively, relevant attributes for this setting are the job position and, more importantly, the level of experience. *Gender*, which is the sensitive attribute S, should not be taken into account in the decision of hiring (or not) someone from a fair decision process perspective. Also note that *Part-time* is a known proxy for gender.

Table 1. Toy dataset

ID	**Gender** (G)	**Job Position** (J)	**Experience** (E)	**Part-time** (P)	**Hired**
1	Male	Software Engineer	High	False	True
2	Female	Software Engineer	High	True	False
3	Male	Data Scientist	Medium	False	True
4	Female	Data Scientist	High	True	False
5	Male	Web Developer	Low	False	False
6	Female	DB Admin	Low	False	True
7	Female	UIX Designer	Medium	True	True
8	Male	Software Engineer	High	False	True
9	Male	Software Engineer	Medium	False	False
10	Female	Help Desk	Medium	True	True
11	Female	Cybersecurity Analyst	High	True	False
12	Female	Network Admin	High	True	False
13	Female	Cybersecurity Analyst	Medium	False	False
14	Male	Cybersecurity Analyst	Medium	False	True
15	Male	Network Admin	Medium	True	True
16	Male	Network Admin	Low	True	True

Discrimination within DILCA can happen while selecting a context for a target attribute Y. Relevant features are collected with the aim of later computing distances between pairs of its attribute values. Hence, it is crucial to act at this stage. Assume th use of $DILCA_M$, after the computation of the matrices needed for storing the SU and related metrics (see equations in Section 2.2), the contexts for the four categorical attributes (the target variable *Hired* is not taken into account by the framework) would be respectively, $\{J, P\}$ for *Gender*, $\{E, P\}$ for *Job Position*, $\{J\}$ for *Experience* and $\{G, J\}$ for *Part-time*. Take the context of *Gender* as instance. Since $DILCA_M$ is used, attributes are selected according to the values of SU (in this case, the column of the matrix concerning G are $SU_{Y=G}(J) \simeq 0.087$, $SU_{Y=G}(E) \simeq 0.021$, $SU_{Y=G}(P) \simeq 0.094$) that exceed the mean value $\overline{SU}_{Y=G} \simeq 0.067$. As it can be noted, the protected attribute is more correlated to *Part-time* (with an $SU_{Y=G}(P) \simeq 0.094 > 0.067$), which, as we said, is a known proxy of *Gender*, and *Job Position* (with an $SU_{Y=G}(J) \simeq 0.087$). DILCA isn't aware of ethical concepts, and if it takes directly into consideration such sensitive information, so does the subsequent learning algorithm, leading to possible discrimination towards people in the learning task (i.e. classification). Take as example object 14: according to DILCA object 3 is more similar to it because *Gender* is directly taken into account and has the same importance of the other attributes. However, object 13 would be a better option since, aside *Gender*, the job position and the other attributes values are the same. In an automated decision system scenario, data generated by DILCA could eventually contribute to a gender bias in a subsequent applied classification task.

To avoid issues like these, we introduce FairDILCA (Fair DIstance Learning for Categorical Attributes), consisting of a regularisation approach inside the context selection step, promoting the selection of attributes that are as accurate and unbiased as possible w.r.t. a sensitive attribute S, and a debiased distance computation method, where a term weighs the contribution of the different attributes to the distance according to their correlation to S. In addition, a parameter can regulate the degree of fairness involved and so the accuracy trade-off, making FairDILCA a more suitable option in scenarios requiring more control by the users. It is worth noting that, in this paper, we consider a single specified sensitive attribute S at a time. However, the framework can be easily extended for considering data with multiple sensitive attributes.

3.1 Fair Context Selection

In the standard DILCA framework, the context selection step is performed to gain relevant attributes w.r.t. a feature taken as the target Y. Simply removing a sensitive feature can improve fairness, but is not as effective as removing the informative content of it from the whole dataset. To decide whether an attribute is part of the context C_Y, not only must we consider how a feature is linked to the target Y, but also how it is correlated to the sensitive attribute S.

To this purpose, we define a new correlation coefficient based on the Symmetric Uncertainty which we call *Fair Correlation Coefficient* (FCC), that states how much an attribute X_i is a proxy for the protected attribute S w.r.t. another attribute X_j. In addition, FCC relies on a parameter, named α, that regulates the amount of fairness one want to achieve and ranges between 0, which corresponds to the non-fair case, and 1, corresponding to a case with the highest level of fairness. The new correlation coefficient is defined as follows.

Definition 1 (Fair Correlation Coefficient). *Given two attributes $X_i, X_j \in F$ and a parameter $\alpha \in [0,1]$, the Fair Correlation Coefficient is computed as*

$$FCC_{X_j}(X_i) = (1-\alpha)SU_{X_j}(X_i) + \alpha(1 - SU_S(X_i)) \tag{4}$$

where $SU_{X_j}(X_i)$ is the Symmetrical Uncertainty between X_i and X_j as defined in Equation 1.

If the value of α is 0, v_i becomes equal to $SU_{X_j}(X_i)$, which leads to the same outcome of the original non-fair $DILCA_M$. Instead, if $\alpha = 1$, $FCC_{X_j}(X_i)$ becomes equal to $(1 - SU_S(X_i))$ and states how much X_i is a proxy for the sensitive attribute S. Note that, although the FCC is based on the Symmetric Uncertainty, it is not symmetrical. Furthermore, in this setting, the context of the target variable Y should not include Y itself, nor the sensitive attribute S.

As the original framework, FairDILCA has two main ways of computing a relevant context, $FairDILCA_M$ and $FairDILCA_{RR}$, which are the fair counterparts of $DILCA_M$ and $DILCA_{RR}$ (see Section 2.2). Additionally, we also define a parameterless algorithm, called $FairDILCA_{PL}$, which does not rely on the FCC and, consequently, does not require the setting of parameter α.

FairDILCA$_{\mathbf{M}}$ Like the original $DILCA_M$ selection method, $FairDILCA_M$ focuses on selecting the relevant subset of features by observing a mean value whose influence is controlled by the original σ parameter. However, for the computation of the mean value, instead of using $SU_Y(X_i)$, it takes into account $FCC_Y(X_i)$ in the following way:

$$\overline{FCC}_Y = \frac{\sum_{X_i \in F \setminus \{Y \cup S\}} FCC_Y(X_i)}{|F \setminus \{Y \cup S\}|} \tag{5}$$

Then, an attribute X_i is included in the context for a given target variable Y when it satisfies the condition:

$$C_Y = \{X_i \in F | X_i \neq Y \wedge X_i \neq S \wedge FCC_Y(X_i) \geq \sigma \cdot \overline{FCC}_Y\} \tag{6}$$

Consider again the example in Table 1. After calculating the FCC values upon the SU matrix, $FairDILCA_M$ would collect the attributes whose FCCs are above the mean. This leads to computing the following contexts for the four categorical attributes: $\{\varnothing\}$ for *Gender*, $\{E\}$ for *Job Position*, $\{J\}$ for *Experience*, and $\{E\}$ for *Part-time*. Focusing on *Job Position*, the only collectable attribute above the mean value is *Experience* (with a value of $FCC_J(E) \simeq 0.97$ against the $FCC_J(P) \simeq 0.89$ of *Part-time* and a mean value of $\overline{FCC_J} \simeq 0.93$[2]). As expected, neither the sensitive attribute nor its proxy are counted as relevant attributes giving space to *Job Position* and *Experience*, making object 13 more similar to object 14. This way we ensure that distances won't depend on the informative parts of *Gender* in the data, making any later learning fairer.

Algorithm 1 FairDILCA$_{\mathrm{M}}$(D, F, Y, S, σ, α)

```
$\mathcal{M}_{FCC} = (FCC_\alpha(X_i, X_j))_{i=1\ldots m, j=1\ldots m}$
$\overline{FCC}_Y = \frac{1}{m-2} \sum_{X_i \neq Y} \mathcal{M}_{FCC}(X_i, X_j)$
$C_Y = \{\varnothing\}$
for all $X_i \in F \setminus \{Y \cup S\}$ do                ▷ the context cannot contain $Y$ nor $S$
    if $\mathcal{M}_{FCC}(Y, X_i) \geq \sigma \cdot \overline{FCC}_Y$ then
        $C_Y = C_Y \cup \{X_i\}$
    end if
end for
return $FairDistanceComputation(D, Y, C_Y)$
```

In Algorithm 1, we report the pseudocode of $FairDILCA_M$. After the matrix $\mathcal{M}_{FCC}$ recording all FCC values between attributes is computed[3], the mean of the FCC values between the target attribute Y and the other attributes is calculated excluding Y itself and the sensitive attribute S. Then, the context selection is performed according to the rule of Equation 6. Once the context is fully determined, the fair version of the distance computation is performed to compute the matrix of distances between any pair of values of Y.

[2] Parameter α was set to 0.99

[3] We remind that $\mathcal{M}_{FCC}$ is not symmetric, as, in general $FCC_{X_i}(X_j) \neq FCC_{X_j}(X_i)$.

FairDILCA$_{\mathbf{RR}}$ This method (described in Algorithm 2) is substantially similar to $DILCA_{RR}$ (see Section 2.2), but instead of using the Symmetric Uncertainty, it computes the FCC (see Equation 4). In the *relevancy step* the candidate attributes X_i are ranked according to the values of $FCC_Y(X_i)$, in decreasing order. In this way, the ranking takes into account both the correlation of X_i w.r.t the target Y and that with the sensitive feature S, depending on the fairness parameter α. In the *redundancy step*, candidate attributes for the C_Y are collected if an attribute X_i has $FCC_Y(X_i) > FCC_{X_j}(X_i)\ \forall X_j \in F \setminus \{Y, X_i\}$. Better explained, X_i is not redundant if the FCC that links X_i to Y, $FCC_Y(X_i)$, is greater than the FCC linking X_i to every other attribute X_j with $j \neq i$.

Algorithm 2 FairDILCA$_{\mathrm{RR}}$(D, F, Y, S, α)

```
$\mathcal{M}_{FCC} = (FCC_\alpha(X_i, X_j))_{i=1\ldots m, j=1\ldots m}$
$C_Y = \{X \in F \backslash \{Y \cup S\}\}$
for all $X_i \in C_Y$ in descending order w.r.t. $\mathcal{M}_{FCC}(Y, X_i)$ do        ▷ Relevancy step
    for all $X_j$ s.t. $\mathcal{M}_{FCC}(Y, X_j) \leq \mathcal{M}_{FCC}(Y, X_i)$ do
        if $\mathcal{M}_{FCC}(X_i, X_j) \geq \mathcal{M}_{FCC}(Y, X_j)$ then          ▷ Redundancy step
            $C_Y = C_Y \setminus \{X_j\}$
        end if
    end for
end for
return $FairDistanceComputation(D, Y, C_Y)$
```

FairDILCA$_{\mathbf{PL}}$ In this variant, an additional step, the sensitivity check step, is added to the original $DILCA_{RR}$ strategy to remove those features, excluding Y, that act as proxies for the sensitive feature S. Basically, an attribute X_i is removed from C_Y if the following condition is satisfied:

$$SU_S(X_i) > SU_Y(X_i) \tag{7}$$

More intuitively, an attribute is removed from C_Y if it is more informative for the sensitive attribute than the target attribute. The last variant, with the context computed regardless of α is showcased in Algorithm 3.

Note that, for a given Y, when all other attributes X_i are too strongly correlated with S, the resulting context could be empty. When it happens, it means that Y should be filtered out during computation of the pairwise object distance.

3.2 Fair Distance Computation

To compute the distance between any pair of values of the same target attribute Y, the fair method differs from its non fair counterpart by the inclusion of a term γ_i that weighs the contribution of the probability differences between each pair of attribute values. More formally:

$$d_{fair}(y_i, y_j) = \sqrt{\frac{\sum_{X_i \in C_Y} \sum_{x_k \in X_i} \gamma_i \left(P(y_i|x_k) - P(y_j|x_k)\right)^2}{\sum_{X_i \in C_Y} \gamma_i |X_i|}} \tag{8}$$

Algorithm 3 FairDILCA$_{\text{PL}}$(D, F, Y, S)

```
$\mathcal{M}_{SU} = (SU(X_i, X_j))_{i=1\dots m, j=1\dots m}$
$C_Y = \{X \in F \backslash \{Y \cup S\}\}$
for all $X_i \in C_Y$ in descending order w.r.t. $\mathcal{M}_{SU}(Y, X_i)$ do
    if $\mathcal{M}_{SU}(S, X_i) \geq \mathcal{M}_{SU}(Y, X_i)$ then        ▷ Sensitivity step
        $C_Y = C_Y \setminus \{X_i\}$
    end if
end for
for all $X_i \in C_Y$ taken in descending order w.r.t. $\mathcal{M}_{SU}(Y, X_i)$ do        ▷ Relevancy step
    for all $X_j$ s.t. $\mathcal{M}_{SU}(Y, X_j) \leq \mathcal{M}_{SU}(Y, X_i)$ do
        if $\mathcal{M}_{SU}(X_i, X_j) \geq \mathcal{M}_{SU}(Y, X_j)$ then        ▷ Redundancy step
            $C_Y = C_Y \setminus \{X_j\}$
        end if
    end for
end for
return $FairDistanceComputation(D, Y, C_Y)$
```

where $\gamma_i = \frac{1-SU_S(X_i)}{n-\sum_i SU_S(X_i)}$ and $n = |C_Y|$. Note that γ_i is directly proportional to $\frac{1}{SU_S(X_i)}$ and $\sum \alpha_i = 1$. The procedure is described in Algorithm 4.

Algorithm 4 FairDistanceComputation(D, Y, C_Y)

```
for all $X_i \in C_Y$ do        ▷ Compute all $\gamma_i$ coefficients
    $\gamma_i = \frac{1-SU_S(X_i)}{|C_Y|-\sum_{X_i \in C_Y} SU_S(X_i)}$
end for
for all $y_i, y_j \in Y$ s.t. $y_i \neq y_j$ do
    $\mathcal{M}_Y(y_i, y_j) = \sqrt{\frac{\sum_{X_i \in C_Y} \sum_{x_k \in X_i} \gamma_i \left(\sum (P(y_i|x_k) - P(y_j|x_k))^2\right)}{\sum_{X_i \in C_Y} \gamma_i |X_i|}}$
end for
return $\mathcal{M}_Y$
```

As final remark, it is worth noting that, when computing the distance between object pairs, the sensitive attribute and all other attributes with empty contexts should not be taken into consideration. If C_Y is empty for all Y, than computing a fair distance in unfeasible and the algorithm fails. However, this condition never happens in our experiments.

3.3 Characteristics of the FairDILCA framework

FairDILCA offers various key features, depending on the specific version adopted. First of all, it belongs to the family of pre-processing techniques and can produce a debiased version of the input data. This can then be used by subsequent distance-based learning algorithms that do not necessarily need to consider fairness constraints in order to produce fairer outcomes. In other terms, it is task-agnostic.

Another non-trivial benefit is the availability of both parametric and non-parametric variants. On the one hand, the parameter α allows the user to control the level of desired fairness involved and, consequently, the trade-off between accuracy and bias reduction, making FairDILCA a flexible option that can adapt to different contexts. On the other hand, $FairDILCA_{PL}$ can be used in all application scenarios when deciding a trade-doff is difficult or even impossible.

Furthermore, due to its nature, FairDILCA works in a white-box fashion, as it returns distances between object pairs that can be traced back to attributes contexts, created by following a features selection process based on notions like relevancy and correlation to the protected attributes. This makes FairDILCA more explainable and transparent, and thus more trustworthy.

4 Experiments and discussion

This section reports the experiments conducted to assess the performance of FairDILCA in terms of accuracy and fairness. Two types of experiments were conducted. The first aimed at assessing the output of the framework alone, specifically the distance matrices. The second was designed to assess the accuracy and fairness of a classification task applied to the distance matrices learnt by FairDILCA. As direct competitor, we use CorrelationRemover [29] (hereinafter referred to as CR), which distorts the original data matrix by smoothing the correlation of any variable w.r.t. the sensitive one. When using CR, the data are not discretized, but categorical attributes are converted into numeric ones using one-hot encoding (as CR works on numeric data only). Consequently, distances of data processed by CR are computed according to the standard Euclidean (ℓ^2) norm. CR adopts a parameter (also called α) that controls the amount of fairness one wants to guarantee when debiasing the data.

All DILCA variants are implemented in Python. We use the implementation of CR provided in the Fairlearn package[4], while we use the machine learning algorithms available in Scikit-learn. The source code for reproducing all our experiments is available online[5].

The datasets used in our experiments are all downloaded from the UCI Machine Learning Repository [15], and are typically adopted in the evaluation of fairness frameworks for tabular data. Their size in the number of data objects and features, alongside the details about the protected attributes, the number of categorical features available, the task-specific target attributes are given in Table 2. Note that, since FairDILCA and its non-fair counterpart only support categorical data, before applying them, we discretize the numerical features using K-Means with five bins.

4.1 Assessment of the distances learnt by FairDILCA

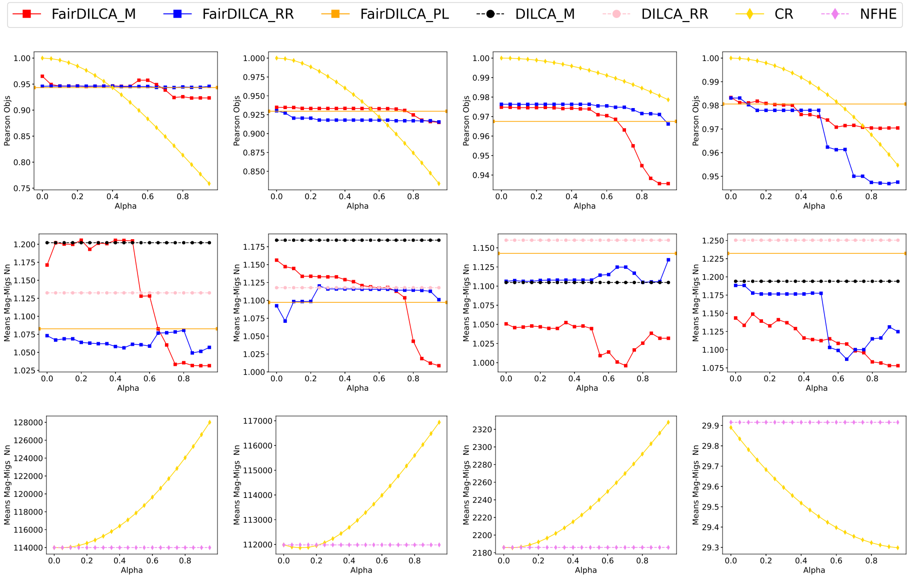
As first experiment, we assess the distance matrices resulting from the application of FairDILCA to categorical datasets with the goal of measuring their distortion compared to the original ones (as computed by DILCA) as well as their actual fairness. To measure the distortion, we compute the Pearson's correlation

[4] https://github.com/fairlearn/fairlearn

[5] https://github.com/alessiofamiani/FairDILCA

Table 2. Datasets used for performance evaluation

Dataset	#Objects	#Features	#Cat. Feat.	Sens. Attr.	Task Target
Adult (S) [5]	32 561	15	9	Sex	Income
Adult (R) [5]	32 561	15	9	Race	Income
Obesity [22]	2 111	17	9	Gender	Obesity Lv.
Bank Marketing [19]	45 211	17	10	Marital status	Deposit sub.

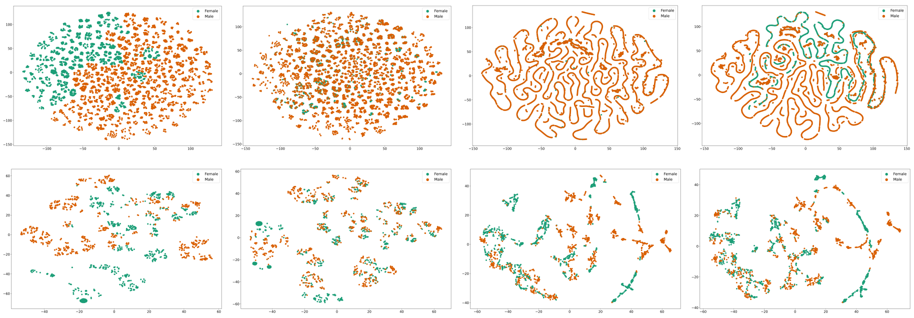


(a) Adult (S) (b) Adult (R) (c) Bank Mark. (d) Obesity

Fig. 1. Pearson correlation coefficient on pairwise object distances (top) and Average distances between the largest protected groups and all the other groups for FairDILCA (middle) and CR (bottom). NFHE stands for Non Fair one-Hot Encoding.

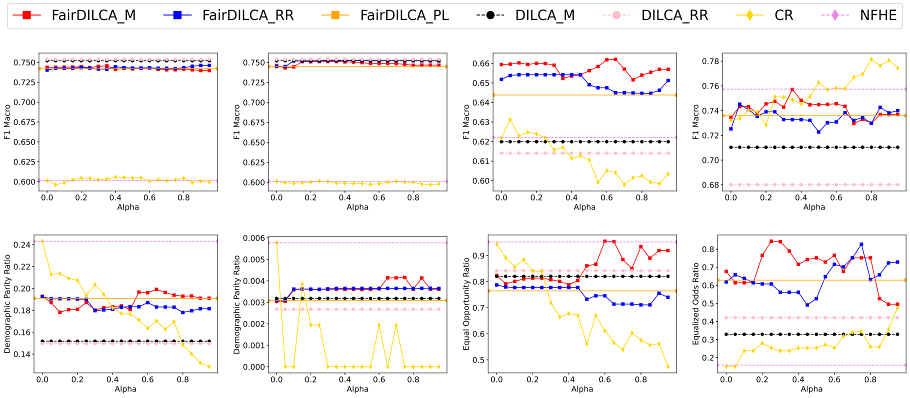
coefficient between the pairwise object distance matrix after the application of FairDILCA and CR, and the ones computed, respectively, by DILCA on categorical data and by CR on numeric ones. The fairness is measured by computing the average distance between objects belonging to the minority groups (MiGs), i.e., the least represented groups within the dataset, and those belonging to the majority one (MaG), i.e., the most represented group within the dataset. The rationale is that the more fairness is demanded (i.e., the greater the value of α), the more similar the objects belonging to different protected groups. Consequently, when α increases, we expect a decrease in the value of the average distances between objects in the MiGs and objects in the MaG.

The results are given in Figure 1. In the top row, we report the values of the Pearson correlation coefficient for increasing values of α. Interestingly, while the decrease of the Pearson's correlation coefficient is smooth, FairDILCA has a stable behavior for low and medium values of α. Then it decreases faster, although it remains on higher values compared to CR. Instead, the decrease on the average distances between MaG and MiGs is unexpectedly counterintuitive in CR (bottom row), with the exception of Obesity, while it is more coherent in FairDILCA (middle row). This confirms that our method actually reduces the distances between data objects belonging to different protected categories, while containing the distortion within reasonable limits.



(a) DILCA$_M$ (b) FairDILCA$_M$ (c) NFHE (d) CR

Fig. 2. t-SNE computed on Adult (S) in the top row and Obesity (bottom).



(a) Adult (S) (b) Adult (R) (c) Bank Mark. (d) Obesity

Fig. 3. Macro-averaged F1-score (top) and fairness measures (bottom) of kNN.

To better investigate the behavior of FairDILCA and CR, we apply t-SNE [17] to their representation in comparison with their non fair counterparts. More in details, we use the original numeric data and the representations learnt by

$DILCA_M$, $FairDILCA_M$ with $\alpha = 0.5$ and CR with the same α. In Figure 2 we report the 2D visualization where each point is colored according to the protected group membership for Adult (top) and Obesity (bottom). The plots confirm that both algorithms work reasonably well on Obesity, while on Adult (S) the transformation applied by CR makes the data even more biased compared to its non fair counterpart. It is worth noting that, without the fair adjustment, DILCA stresses the differences between Males and Females in Adult.

4.2 Results on classification

Here, we discuss the results of the application of a traditional machine learning task on the representation learnt by FairDILCA, compared with the same task applied to the the representation computed by CR and DILCA. In more detail, we consider a classification task using kNN with $k = 23$ in all experiments[6]. After splitting the dataset into training set (70% of the data instances) and test set (the remaining 30%), we first learn all transformations on the training data and apply them on both sets. Hence, for the fair and non fair versions of DILCA, we discretize the whole data according to the bins learnt on the training set only. Then, all (Fair)DILCA distances are learnt on the training set and applied on the whole dataset. As regards CR, after applying one-hot encoding on categorical attributes, the fair transformation is learnt on the training set only and then applied on the whole dataset. Finally, for measuring the performance of the classifier, we compute the macro-averaged F1-score on the test set. To assess the fairness of the results, we use three different metrics depending on the specific application of each dataset, as suggested in [8]. So, we compute the demographic parity ratio (DPR) for Adult, the equalized odds ratio (EOD) for Obesity and the equal opportunity ratio (EOP) for Bank Marketing, three recognized measures for assessing the presence of biases in decision support systems [9]. For all these three measures, higher values imply fairer predictions.

The results are reported in Figure 3. The top row displays the macro-averaged F1-score of the classifier. It indicates that DILCA generally performs better on categorical data than one-hot encoding approaches on it. Additionally, the F1-score of FairDILCA is more stable and, in some cases, even better than the non-fair variants. FairDILCA also produces fairer predictions than DILCA. Interestingly, CR exhibits a counterintuitive behavior in all datasets except Obesity, as its fairness decrease for higher values of α.

5 Conclusion

We have introduced FairDILCA, an extension of a popular distance learning framework for categorical data, which takes into account fairness observations in both the context selection and the distance computation stages. To this purpose, we have defined a parametric fair correlation measure together with two different

[6] We choose a rather high value of k to stabilize the behavior of the classifier.

strategies to use it in the context computation stage. Moreover, we have defined a fully parameterless strategy relying on the standard symmetrical uncertainty, already used by the non fair version of DILCA. As clues have showed, FairDILCA have promising performances in terms of fairness with a reasonable perturbation degree in comparison with the distance matrices returned by the original DILCA.

One technical limitation to the effectiveness of FairDILCA lies in the need for discretization of numerical data in order to compute the conditional probabilities used in the computation. This may have an impact on the fairness itself, as discretization can introduce bias in the data, and we plan to investigate deeper this issue as future work. Another opportunity of improvement for FairDILCA lies in the support of multiple sensitive attributes at a time, an aspect that is generally overlooked in the literature, but of crucial importance for the correct management of fairness issues in data science and machine learning applications.

Acknowledgments. The work presented in this paper is supported by Fondazione CRT (Grant No. 2022-0720). R.G. Pensa is member of Gruppo Nazionale Calcolo Scientifico – Istituto Nazionale di Alta Matematica (GNCS-INdAM).

References

1. Abraham, S.S., P, D., Sundaram, S.S.: Fairness in clustering with multiple sensitive attributes. In: Proc. of EDBT 2020. pp. 287–298 (2020)
2. Barocas, S., Hardt, M., Narayanan, A.: Fairness and Machine Learning: Limitations and Opportunities. MIT Press (2023)
3. Barocas, S., Selbst, A.D.: Big data's disparate impact. Calif. L. Rev. **104**, 671 (2016)
4. Battaglia, E., Celano, S., Pensa, R.G.: Differentially private distance learning in categorical data. Data Min. Knowl. Discov. **35**(5), 2050–2088 (2021)
5. Becker, B., Kohavi, R.: Adult. UCI Machine Learning Repository (1996), DOI: https://doi.org/10.24432/C5XW20
6. Biswas, S., Rajan, H.: Fair preprocessing: towards understanding compositional fairness of data transformers in machine learning pipeline. In: Proc. of ACM ESEC/FSE 2021. pp. 981–993 (2021)
7. Calmon, F., Wei, D., Vinzamuri, B., Natesan Ramamurthy, K., Varshney, K.R.: Optimized pre-processing for discrimination prevention. Advances in neural information processing systems **30** (2017)
8. Castelnovo, A., Crupi, R., Greco, G., Regoli, D., Penco, I.G., Cosentini, A.: A clarification of the nuances in the fairness metrics landscape. Sci. Rep. **12** (2021)
9. Chen, Z., Zhang, J.M., Hort, M., Harman, M., Sarro, F.: Fairness testing: A comprehensive survey and analysis of trends. ACM Trans. Softw. Eng. Methodol. (2024)
10. Council of European Union: Regulation (eu) 2024/1689, https://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX%3A32024R1689
11. Ienco, D., Pensa, R.G., Meo, R.: From context to distance: Learning dissimilarity for categorical data clustering. ACM Trans. Knowl. Discov. Data **6**(1), 1:1–1:25 (2012)
12. Kamiran, F., Calders, T.: Data preprocessing techniques for classification without discrimination. Knowl. Inf. Syst. **33**(1), 1–33 (2011)

13. Kamiran, F., Karim, A., Zhang, X.: Decision theory for discrimination-aware classification. In: Proc. of IEEE ICDM 2012. pp. 924–929 (2012)
14. Kamishima, T., Akaho, S., Asoh, H., Sakuma, J.: Fairness-aware classifier with prejudice remover regularizer. In: Proc. of ECML PKDD 2012. pp. 35–50 (2012)
15. Kelly, M., Longjohn, R., Nottingham, K.: The UCI Machine Learning Repository, https://archive.ics.uci.edu
16. Lenders, D., Calders, T.: Learning a fair distance function for situation testing. In: Proc. of BIAS @ ECML PKDD 2021. pp. 631–646 (2021)
17. der Maaten, V., Laurens, Hinton, Geoffrey: Visualizing data using t-sne. J. Mach. Learn. Res. **9**, 2579–2605 (2008)
18. Mau, T.N., Huynh, V.: An lsh-based k-representatives clustering method for large categorical data. Neurocomputing **463**, 29–44 (2021)
19. Moro, S., Rita, P., Cortez, P.: Bank Marketing. UCI Machine Learning Repository (2012), DOI: https://doi.org/10.24432/C5K306
20. Mougan, C., Álvarez, J.M., Ruggieri, S., Staab, S.: Fairness implications of encoding protected categorical attributes. In: Proc. of AAAI/ACM AIES 2023. pp. 454–465 (2023)
21. Mukherjee, D., Yurochkin, M., Banerjee, M., Sun, Y.: Two simple ways to learn individual fairness metrics from data. In: Proc. of ICML. pp. 7097–7107 (2020)
22. Palechor, F.M., de la Hoz Manotas, A.: Dataset for estimation of obesity levels based on eating habits and physical condition in individuals from colombia, peru and mexico. Data in Brief **25**, 104344 (2019)
23. Peng, H., Long, F., Ding, C.H.Q.: Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell. **27**(8), 1226–1238 (2005)
24. Pessach, D., Shmueli, E.: A review on fairness in machine learning. ACM Comput. Surv. **55**(3), 51:1–51:44 (2023)
25. Ruoss, A., Balunovic, M., Fischer, M., Vechev, M.: Learning certified individually fair representations. Advances in neural information processing systems **33**, 7584–7596 (2020)
26. Santos-Mangudo, C., Heras, A.J.: A fair-multicluster approach to clustering of categorical data. Central Eur. J. Oper. Res. **31**(2), 583–604 (2023)
27. Su, B., Ming, C., Sun, Y., Liu, K.: Clustering ensemble method based DILCA distance. In: Proc. of IEEE ICMLC 2013. pp. 29–34 (2013)
28. Voeneky, S., Kellmeyer, P., Müller, O., Burgard, W. (eds.): The Cambridge Handbook of Responsible Artificial Intelligence - Interdisciplinary Perspectives. Cambridge University Press (2022)
29. Weerts, H.J.P., Dudík, M., Edgar, R., Jalali, A., Lutz, R., Madaio, M.: Fairlearn: Assessing and improving fairness of AI systems. J. Mach. Learn. Res. **24**, 257:1–257:8 (2023)
30. Xu, D., Yuan, S., Zhang, L., Wu, X.: Fairgan: Fairness-aware generative adversarial networks. In: Proc. of IEEE Big Data 2018. pp. 570–575. IEEE (2018)
31. Yang, S., Cerrato, M., Ienco, D., Pensa, R.G., Esposito, R.: Fairswirl: fair semi-supervised classification with representation learning. Mach. Learn. **112**(9), 3051–3076 (2023)
32. Zemel, R.S., Wu, Y., Swersky, K., Pitassi, T., Dwork, C.: Learning fair representations. In: Proc. of ICML 2013. pp. 325–333 (2013)
33. Zhang, B.H., Lemoine, B., Mitchell, M.: Mitigating unwanted biases with adversarial learning. In: Proc. of AAAI/ACM AIES 2018. pp. 335–340 (2018)
34. Zhang, L., Wu, Y., Wu, X.: Situation testing-based discrimination discovery: A causal inference approach. In: Proc. of IJCAI 2016. pp. 2718–2724 (2016)