Learning Fair Naive Bayes Classifiers by Discovering and Eliminating Discrimination Patterns

YooJung Choi[*]
University of California, Los Angeles
yjchoi@cs.ucla.edu

Golnoosh Farnadi[*]
Polytechnique Montréal, Canada
golnoosh.farnadi@polymtl.ca

Behrouz Babaki[*]
Polytechnique Montréal, Canada
behrouz.babaki@polymtl.ca

Guy Van den Broeck
University of California, Los Angeles
guyvdb@cs.ucla.edu

Abstract

As machine learning is increasingly used to make real-world decisions, recent research efforts aim to define and ensure fairness in algorithmic decision making. Existing methods often assume a fixed set of observable features to define individuals, but lack a discussion of certain features not being observed at test time. In this paper, we study fairness of naive Bayes classifiers, which allow partial observations. In particular, we introduce the notion of a discrimination pattern, which refers to an individual receiving different classifications depending on whether some sensitive attributes were observed. Then a model is considered fair if it has no such pattern. We propose an algorithm to discover and mine for discrimination patterns in a naive Bayes classifier, and show how to learn maximum-likelihood parameters subject to these fairness constraints. Our approach iteratively discovers and eliminates discrimination patterns until a fair model is learned. An empirical evaluation on three real-world datasets demonstrates that we can remove exponentially many discrimination patterns by only adding a small fraction of them as constraints.

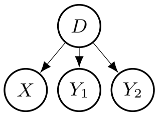
1 Introduction

With the increasing societal impact of machine learning come increasing concerns about the fairness properties of machine learning models and how they affect decision making. For example, concerns about fairness come up in policing [18], recidivism prediction [2], insurance pricing [17], hiring [4], and credit rating [10]. The algorithmic fairness literature has proposed various solutions, from limiting the disparate treatment of similar individuals to giving statistical guarantees on how classifiers behave towards different populations. Key approaches include fairness through awareness [6], individual fairness [23], statistical parity, disparate impact, and group fairness [2, 8, 12], counterfactual fairness [17], preference-based fairness [22], and equality of opportunity [9]. The goal in these works is usually to assure the fair treatment of individuals or groups that are identified by sensitive attributes.

In this paper, we study fairness properties of probabilistic classifiers that represent joint distributions over the features and a decision variable. In particular, Bayesian network classifiers treat the classification or decision-making task as a probabilistic inference problem: given observed features, compute the probability of the decision variable. Such models have a unique ability that they can naturally handle missing features, by simply marginalizing them out of the distribution when they are not observed at prediction time. Hence, a Bayesian network classifier effectively embeds exponentially many traditional classifiers, one for each subset of observable features. We ask whether

[*] Equal contribution

Preprint. Under review.



$P(d)$
0.2

D	$P(x\|D)$
$+$	0.8
$-$	0.5

D	$P(y_1\|D)$
$+$	0.7
$-$	0.1

D	$P(y_2\|D)$
$+$	0.8
$-$	0.3

Figure 1: Naive Bayes classifier with a sensitive attribute X and non-sensitive attributes Y_1, Y_2

such classifiers exhibit patterns of discrimination where similar individuals receive markedly different outcomes purely because they disclosed a sensitive attribute.

The first key contribution of this paper is an algorithm to verify whether a Bayesian classifier is fair, or else to mine the classifier for discrimination patterns. We propose two alternative criteria for identifying the most important discrimination patterns that are present in the classifier. We specialize our pattern miner to efficiently discover discrimination patterns in naive Bayes models using branch-and-bound search. These classifiers are often used in practice because of their simplicity and tractability, and they allow for the development of effective bounds. Our empirical evaluation shows that naive Bayes models indeed exhibit vast numbers of discrimination patterns, and that our pattern mining algorithm is able to find them by traversing only a small fraction of the search space.

The second key contribution of this paper is a parameter learning algorithm for naive Bayes classifiers that eliminates discrimination patterns from the learned distribution. We propose a signomial programming approach to eliminate individual patterns of discrimination during maximum-likelihood learning. Moreover, to efficiently eliminate the exponential number of patterns that could exist in a naive Bayes classifier, we propose a cutting-plane approach that uses our discrimination pattern miner to find and iteratively eliminate discrimination patterns until the entire learned model is fair. Our empirical evaluation shows that this process converges in a small number of iteration, effectively removing millions of discrimination patterns. Moreover, the learned fair models are of high quality, achieving likelihoods that are close to the best likelihoods achieved by models that are not fair.

2 Problem formalization

We use uppercase letters for random variables and lowercase letters for their assignments. Sets of variables and their joint assignments are written in bold. Negation of a binary assignment x is denoted $\bar{x}$, and $\mathbf{x} \models \mathbf{y}$ means that $\mathbf{x}$ logically implies $\mathbf{y}$. Concatenation of sets $\mathbf{XY}$ denotes their union.

Each individual is characterized by an assignment to a set of discrete variables $\mathbf{Z}$, called attributes or features. Assignment d to a binary decision variable D represents a decision made in favor of the individual (e.g., a loan approval). A set of *sensitive attributes* $\mathbf{S} \subset \mathbf{Z}$ specifies a group of entities protected often by law, such as gender and race. We now define the notion of a discrimination pattern.

Definition 1. *Let P be a distribution over $D \cup \mathbf{Z}$. Let $\mathbf{x}$ and $\mathbf{y}$ be joint assignments to $\mathbf{X} \subseteq \mathbf{S}$ and $\mathbf{Y} \subseteq \mathbf{Z} \backslash \mathbf{X}$, respectively. The* degree of discrimination *of $\mathbf{xy}$ is: $\Delta_{P,d}(\mathbf{x}, \mathbf{y}) \triangleq P(d \,|\, \mathbf{xy}) - P(d \,|\, \mathbf{y})$.*

The assignment $\mathbf{y}$ identifies a group of similar individuals, and the degree of discrimination quantifies how disclosing sensitive information $\mathbf{x}$ affects the decision for this group. Note that sensitive attributes missing from $\mathbf{x}$ can still be used to define $\mathbf{y}$. We drop the subscripts P, d when clear from context.

Definition 2. *Let P be a distribution over $D \cup \mathbf{Z}$, and $\delta \in [0, 1]$ a threshold. Joint assignments $\mathbf{x}$ and $\mathbf{y}$ form a* discrimination pattern *w.r.t. P and δ if: (1) $\mathbf{X} \subseteq \mathbf{S}$ and $\mathbf{Y} \subseteq \mathbf{Z} \backslash \mathbf{X}$; and (2) $|\Delta_{P,d}(\mathbf{x}, \mathbf{y})| > \delta$.*

Intuitively, we do not want information about the sensitive attributes to significantly affect the probability of getting a favorable decision. Let us consider two special cases of discrimination patterns. First, if $\mathbf{Y} = \emptyset$, then a small discrimination score $|\Delta(\mathbf{x}, \emptyset)|$ can be interpreted as an approximation of statistical parity, which is achieved when $P(d \,|\, \mathbf{x}) = P(d)$. For example, the naive Bayes network in Figure 1 satisfies approximate parity for $\delta = 0.2$ as $|\Delta(x, \emptyset)| = 0.086 \leq \delta$ and $|\Delta(\bar{x}, \emptyset)| = 0.109 \leq \delta$. Second, suppose $\mathbf{X} = \mathbf{S}$ and $\mathbf{Y} = \mathbf{Z} \backslash \mathbf{S}$. Then bounding $|\Delta(\mathbf{x}, \mathbf{y})|$ for all joint states $\mathbf{x}$ and $\mathbf{y}$ is equivalent to enforcing individual fairness where two individuals are considered similar if their non-sensitive attributes $\mathbf{y}$ are equal. The network in Figure 1 is also individually fair for $\delta = 0.2$ because $\max_{xy_1y_2} |\Delta(x, y_1 y_2)| = 0.167 \leq \delta$.[2]

[2] The highest discrimination score is at $\Delta(\bar{x}, y_1 \bar{y}_2) = -0.167$.

Algorithm 1 Disc-Patterns$(\mathbf{x}, \mathbf{y}, \mathbf{E})$

Input: P : Distribution over $D \cup \mathbf{Z}$, δ : discrimination threshold **Output:** Discrimination patterns L
Data: $\mathbf{x} \leftarrow \emptyset, \mathbf{y} \leftarrow \emptyset, \mathbf{E} \leftarrow \emptyset, L \leftarrow []$

```
for all assignments $z$ to some selected variable $Z \in \mathbf{Z} \setminus \mathbf{XYE}$ do
    if $Z \in \mathbf{S}$ then
        if $|\Delta(\mathbf{x}z, \mathbf{y})| > \delta$ then add $(\mathbf{x}z, \mathbf{y})$ to $L$
        if UB$(\mathbf{x}z, \mathbf{y}, \mathbf{E}) > \delta$ then Disc-Patterns$(\mathbf{x}z, \mathbf{y}, \mathbf{E})$
    if $|\Delta(\mathbf{x}, \mathbf{y}z)| > \delta$ then add $(\mathbf{x}, \mathbf{y}z)$ to $L$
    if UB$(\mathbf{x}, \mathbf{y}z, \mathbf{E}) > \delta$ then Disc-Patterns$(\mathbf{x}, \mathbf{y}z, \mathbf{E})$
if UB$(\mathbf{x}, \mathbf{y}, \mathbf{E} \cup \{Z\}) > \delta$ then Disc-Patterns$(\mathbf{x}, \mathbf{y}, \mathbf{E} \cup \{Z\})$
```

Even though the example network has no discrimination pattern at the group level nor at the individual level (with fully observed features), it may still produce a discrimination pattern. In particular, $|\Delta(\bar{x}, y_1)| = 0.225 > \delta$. That is, a person with $\bar{x}$ and y_1 observed and the value of Y_2 undisclosed would receive a much more favorable decision had they not disclosed X as well. Hence, naturally we wish to ensure that there exists no discrimination pattern across all subsets of observable features.

Definition 3. *A distribution P is δ-*fair *if there exists no discrimination pattern w.r.t P and δ.*

Although our notion of fairness applies to any distribution, finding discrimination patterns can be computationally challenging: computing the degree of discrimination involves probabilistic inference, which is hard in general, and a given distribution may have exponentially many patterns. In this paper, we demonstrate how to discover and eliminate discrimination patterns of a naive Bayes classifier effectively by exploiting its independence assumptions. Concretely, we answer the following questions: (1) Can we certify that a classifier is δ-fair?; (2) If not, can we find the most important discrimination patterns?; (3) Can we learn a naive Bayes classifier that is entirely δ-fair?

3 Discovering discrimination patterns and verifying δ-fairness

This section describes our approach to finding discrimination patterns or checking that there are none.

3.1 Searching for discrimination patterns

One may naively enumerate all possible patterns and compute their degrees of discrimination. However, this would be very inefficient as there are exponentially many subsets and assignments to consider. We instead use branch-and-bound search to more efficiently decide if a model is fair.

Algorithm 1 finds discrimination patterns. It recursively adds variable instantiations and checks the discrimination score at each step. If the input distribution is δ-fair, the algorithm returns no pattern; otherwise, it returns the set of all discriminating patterns. Note that computing Δ requires probabilistic inference on distribution P. This can be done efficiently for large classes of graphical models [3, 5, 15, 19], and particularly for naive Bayes networks, which will be our main focus.

Furthermore, the algorithm relies on a good upper bound to prune the search tree and avoid enumerating all possible patterns. Here, UB$(\mathbf{x}, \mathbf{y}, \mathbf{E})$ bounds the degree of discrimination achievable by observing more features after $\mathbf{xy}$ while excluding features $\mathbf{E}$.

Proposition 1. *Let P be a naive Bayes distribution over $D \cup \mathbf{Z}$, and let $\mathbf{x}$ and $\mathbf{y}$ be joint assignments to $\mathbf{X} \subseteq \mathbf{S}$ and $\mathbf{Y} \subseteq \mathbf{Z} \setminus \mathbf{X}$. Let $\mathbf{x}'_u$ (resp. $\mathbf{x}'_l$) be an assignment to $\mathbf{X}' = \mathbf{S} \setminus \mathbf{X}$ that maximizes (resp. minimizes) $P(d \mid \mathbf{xx}')$. Suppose $l \leq P(d \mid \mathbf{yy}') \leq u$ for all possible assignments $\mathbf{y}'$ to $\mathbf{Y}' = \mathbf{Z} \setminus (\mathbf{XY})$. Then the degrees of discrimination for all patterns $\mathbf{xx}'\mathbf{yy}'$ that extend $\mathbf{xy}$ are bounded as follows:*

$$\min_{l \leq \gamma \leq u} \widetilde{\Delta}\left(P(\mathbf{xx}'_l \mid d), P(\mathbf{xx}'_l \mid \overline{d}), \gamma\right) \leq \Delta_{P,d}(\mathbf{xx}', \mathbf{yy}') \leq \max_{l \leq \gamma \leq u} \widetilde{\Delta}\left(P(\mathbf{xx}'_u \mid d), P(\mathbf{xx}'_u \mid \overline{d}), \gamma\right),$$

where $\widetilde{\Delta}(\alpha, \beta, \gamma) \triangleq \frac{\alpha\gamma}{\alpha\gamma + \beta(1-\gamma)} - \gamma$.

Here, $\widetilde{\Delta} : [0,1]^3 \rightarrow [0,1]$ is introduced to relax the discrete problem of minimizing or maximizing the degree of discrimination into a continuous one. In particular, $\widetilde{\Delta}\left(P(\mathbf{x}|d), P(\mathbf{x}|\overline{d}), P(d|\mathbf{y})\right)$ equals the degree of discrimination $\Delta(\mathbf{x}, \mathbf{y})$. This relaxation allows us to compute bounds efficiently, as closed-form solutions. We refer to the Appendix for full proofs and details.

To apply above proposition, we need to find $\mathbf{x}'_u, \mathbf{x}'_l, l, u$ by maximizing/minimizing $P(d|\mathbf{x}\mathbf{x}')$ and $P(d|\mathbf{y}\mathbf{y}')$ for a given pattern $\mathbf{x}\mathbf{y}$. Fortunately, this can be done efficiently for naive Bayes classifiers.

Lemma 1. *Given a naive Bayes distribution P over $D \cup \mathbf{Z}$, a subset $\mathbf{V} = \{V_i\}_{i=1}^n \subset \mathbf{Z}$, and an assignment $\mathbf{w}$ to $\mathbf{W} \subseteq \mathbf{Z} \setminus \mathbf{V}$, we have:* $\arg\max_{\mathbf{v}} P(d|\mathbf{v}\mathbf{w}) = \left\{\arg\max_{v_i} P(v_i|d)/P(v_i|\overline{d})\right\}_{i=1}^n$.

That is, the joint observation $\mathbf{v}$ that will maximize the probability of the decision can be found by optimizing each variable V_i independently; the same holds when minimizing. Hence, we can use Proposition 1 to compute upper bounds on discrimination scores of extended patterns in linear time.

3.2 Searching for top-k ranked patterns

If a distribution is significantly unfair, Algorithm 1 may return exponentially many discrimination patterns. This is not only very expensive but makes it difficult to interpret the discrimination patterns. Instead, we would like to return a smaller set of "interesting" discrimination patterns.

An obvious choice is to return a small number of discrimination patterns with the highest absolute degree of discrimination. Searching for the k most discriminating patterns can be done with a small modification to Algorithm 1. First, the size of list L is limited to k. The conditions in Lines 3–7 are modified to check the current discrimination score and upper bounds against the smallest discrimination score of patterns in L, instead of the threshold δ.

Nevertheless, ranking patterns by their discrimination score may return patterns of very low probability. For example, the most discriminating pattern of a naive Bayes classifier learned on the COMPAS dataset[3] has a high discrimination score of 0.42, but only has a 0.02% probability of occurring. The probability of a discrimination pattern denotes the proportion of the population (according to the distribution) that could be affected unfairly, and thus a pattern with extremely low probability could be of lesser interest. To address this concern, we propose a more sophisticated ranking of the discrimination patterns that also takes into account the probabilities of patterns.

Definition 4. *Let P be a distribution over $D \cup \mathbf{Z}$. Let $\mathbf{x}$ and $\mathbf{y}$ be joint instantiations to subsets $\mathbf{X} \subseteq \mathbf{S}$ and $\mathbf{Y} \subseteq \mathbf{Z} \setminus \mathbf{X}$, respectively. The* divergence score *of* $\mathbf{x}\mathbf{y}$ *is:*

$$\mathrm{Div}_{P,d,\delta}(\mathbf{x}, \mathbf{y}) \triangleq \min_{Q} \mathrm{KL}\left(P \parallel Q\right) \textit{ s.t. } |\Delta_{Q,d}(\mathbf{x}, \mathbf{y})| \leq \delta \textit{ and } P(d\mathbf{z}) = Q(d\mathbf{z}),\ \forall\, d\mathbf{z} \not\models \mathbf{x}\mathbf{y} \quad (1)$$

The divergence score assigns to a pattern $\mathbf{x}\mathbf{y}$ the minimum Kullback-Leibler divergence between current distribution P and a hypothetical distribution Q that is fair on the pattern $\mathbf{x}\mathbf{y}$ and differs from P only on the assignments that satisfy the pattern (namely $d\mathbf{x}\mathbf{y}$ and $\overline{d}\mathbf{x}\mathbf{y}$). Informally, the divergence score approximates how much the current distribution P needs to be changed in order for $\mathbf{x}\mathbf{y}$ to no longer be a discrimination pattern. Hence, patterns with higher divergence score will tend to have not only higher discrimination score but also higher probabilities.

For instance, the pattern with the highest divergence score on the COMPAS dataset has a discrimination score of 0.19 which is not insignificant, but also has a relatively high probability of 3.33% – more than two orders of magnitude larger than that of the most discriminating pattern. Therefore, such a general pattern could be more interesting for the user studying this classifier.

To find the top-k patterns with the divergence score, we need to be able to compute the score and its upper bound efficiently. The key insights we exploit are that KL divergence is convex and that Q, in Equation 1, can freely differ from P only on one probability value (either that of $d\mathbf{x}\mathbf{y}$ or $\overline{d}\mathbf{x}\mathbf{y}$). Then:

$$\mathrm{Div}_{P,d,\delta}(\mathbf{x}, \mathbf{y}) = P(d\mathbf{x}\mathbf{y}) \log\left(\frac{P(d\mathbf{x}\mathbf{y})}{P(d\mathbf{x}\mathbf{y}) + r}\right) + P(\overline{d}\mathbf{x}\mathbf{y}) \log\left(\frac{P(\overline{d}\mathbf{x}\mathbf{y})}{P(\overline{d}\mathbf{x}\mathbf{y}) - r}\right), \quad (2)$$

where $r = 0$ if $|\Delta_{P,d}(\mathbf{x}, \mathbf{y})| \leq \delta$; $r = \frac{\delta - \Delta_{P,d}(\mathbf{x},\mathbf{y})}{1/P(\mathbf{x}\mathbf{y}) - 1/P(\mathbf{y})}$ if $\Delta_{P,d}(\mathbf{x}, \mathbf{y}) > \delta$; and $r = \frac{-\delta - \Delta_{P,d}(\mathbf{x},\mathbf{y})}{1/P(\mathbf{x}\mathbf{y}) - 1/P(\mathbf{y})}$ if $\Delta_{P,d}(\mathbf{x}, \mathbf{y}) < -\delta$. Intuitively, r represents the minimum necessary change to $P(d\mathbf{x}\mathbf{y})$ for $\mathbf{x}\mathbf{y}$ to be non-discriminating in the new distribution. Note that the smallest divergence score $\mathrm{Div}_{P,d,\delta}(\mathbf{x}, \mathbf{y}) = 0$ is attained when the pattern is already fair.

[3] https://github.com/propublica/compas-analysis

Table 1: Data statistics (number of training instances, sensitive features S, non-sensitive features N, and potential patterns) and the proportion of patterns explored during the search

Dataset Statistics						Divergence score			Discrimination score		
Dataset	Size	S	N	# Pat.	k	$\delta = 0.01$	$\delta = 0.05$	$\delta = 0.10$	$\delta = 0.01$	$\delta = 0.05$	$\delta = 0.10$
COMPAS	48,834	4	3	15K	1	6.387e-01	5.634e-01	3.874e-01	8.188e-03	8.188e-03	8.188e-03
					10	7.139e-01	5.996e-01	4.200e-01	3.464e-02	3.464e-02	3.464e-02
					100	8.222e-01	6.605e-01	4.335e-01	9.914e-02	9.914e-02	9.914e-02
Adult	32,561	4	9	11M	1	3.052e-06	7.260e-06	1.248e-05	2.451e-04	2.451e-04	2.451e-04
					10	7.030e-06	1.154e-05	1.809e-05	2.467e-04	2.467e-04	2.467e-04
					100	1.458e-05	1.969e-05	2.509e-05	2.600e-04	2.600e-04	2.597e-04
German	1,000	4	16	23B	1	5.075e-07	2.731e-06	2.374e-06	7.450e-08	7.450e-08	7.450e-08
					10	9.312e-07	3.398e-06	2.753e-06	1.592e-06	1.592e-06	1.592e-06
					100	1.454e-06	4.495e-06	3.407e-06	5.897e-06	5.897e-06	5.897e-06

Lastly, we refer to the Appendix for two upper bounds of the divergence score, which utilize the bound on discrimination score of Proposition 1 and can be computed efficiently using Lemma 1.

3.3 Empirical evaluation of discrimination pattern miner

In this section, we report the experimental results on the performance of our pattern mining algorithms. All experiments were run on an AMD Opteron 275 processor (2.2GHz) and 4GB of RAM running Linux Centos 7. Execution time is limited to 1800 seconds.

Data and pre-processing. We use three datasets: The *Adult* dataset and *German* dataset are used for predicting income level and credit risk, repectively, and are obtained from the UCI machine learning repository[4]. The *COMPAS* dataset is used for predicting recidivism. As pre-processing, we removed unique features (e.g. names of individuals) and duplicate features.[5] See Table 1 for a summary.

Q1. Does our pattern miner find discrimination patterns more efficiently than by enumerating all possible patterns? We answer this question by inspecting the fraction of all possible patterns that our pattern miner visits during the search. Table 1 shows the results on three datasets, using two rank heuristics (discrimination and divergence) and three threshold values (0.01, 0.05, and 0.1). The results are reported for mining the top-k patterns when k is 1, 10, and 100. A naive method has to enumerate all possible patterns to discover the discriminating ones, while our algorithm visits only a small fraction of patterns (e.g., one in every several millions on the German dataset).

Q2. Does the KLD heuristic find discrimination patterns with both a high discrimination score and high probability? Figure 2 shows the *probability* and *discrimination score* of all patterns in the COMPAS dataset. The top-10 patterns according to three measures (degree of discrimination, divergence score, and probability) are highlighted in the figure. The observed trade-off between probability and difference score indicates that picking the top patterns according to each measure will yield low quality patterns according to the other measure. The KLD score, however, balances the two measures and returns patterns that have high probability and difference scores. Also observe that the patterns selected by the divergence score lie in the Pareto front. This in fact always holds by the definition of this heuristic; fixing the probability and increasing the degree of discrimination will also increase the KLD score, and vice versa.

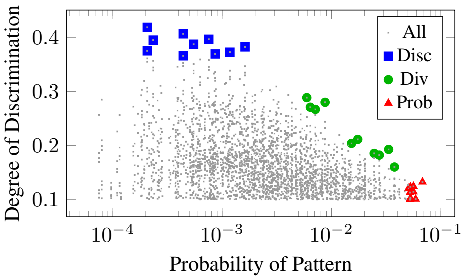


Figure 2: Discrimination patterns with $\delta = 0.1$ for the max-likelihood NB classifier on COMPAS.

[4] `https://archive.ics.uci.edu/ml/datasets.html`

[5] The processed data and our implementation of the algorithm are available at `https://github.com/UCLA-StarAI/LearnFairNB`.

4 Learning fair naive Bayes classifiers

We now describe our approach to learning the maximum-likelihood parameters of a naive Bayes model from data while eliminating discrimination patterns. It is based on formulating the learning subject to fairness constraints as a signomial program, an optimization problem of the form:

$$\text{minimize } f_0(x), \quad \text{s.t.} \quad f_i(x) \leq 1, \quad g_j(x) = 1 \quad \forall\, i,j$$

where each f_i is signomial while g_j is monomial. A *signomial* is a function of the form $\sum_k c_k x_1^{a_{1k}} \cdots x_n^{a_{1n}}$ defined over real positive variables $x_1 \ldots x_n$ where $c_k, a_{ij} \in \mathbb{R}$; a *monomial* is of the form $cx_1^{a_1} \cdots x_n^{a_n}$ where $c > 0$ and $a_i \in \mathbb{R}$. Signomial programs are not globally convex, and only a locally optimal solution can be computed efficiently, unlike the closely related class of geometric programs, for which the globally optimum can be found efficiently [7].

4.1 Parameter learning with fairness constraints

The likelihood of a Bayesian network given data $\mathcal{D}$ is $P_\theta(\mathcal{D}) = \prod_i \theta_i^{n_i}$ where n_i is the number of examples in $\mathcal{D}$ that satisfy the assignment corresponding to parameter θ_i. To learn the maximum-likelihood parameters, we minimize the inverse of likelihood which is a monomial: $\theta_{\text{ml}} = \arg\min_\theta \prod_i \theta_i^{-n_i}$. The parameters of a naive Bayes network with binary class consist of $\theta_d, \theta_{\bar{d}}$, and $\theta_{z\,|\,d}, \theta_{z\,|\,\bar{d}}$ for all z.

Next, we show the constraints for our optimization problem. To learn a valid distribution, we need to ensure that probabilities are non-negative and sum to one. The former assumption is inherent to signomial programs. To enforce the latter, for each instantiation d and feature Z, we need that $\sum_z \theta_{z\,|\,d} = 1$, or as signomial inequality constraints: $\sum_z \theta_{z\,|\,d} \leq 1$ and $2 - \sum_z \theta_{z\,|\,d} \leq 1$.

Finally, we derive the constraints to ensure that a given pattern $\mathbf{xy}$ is non-discriminating.

Proposition 2. *Let P_θ be a naive Bayes distribution over $D \cup \mathbf{Z}$, and let $\mathbf{x}$ and $\mathbf{y}$ be joint assignments to $\mathbf{X} \subseteq \mathbf{S}$ and $\mathbf{Y} \subseteq \mathbf{Z} \setminus \mathbf{X}$. Then $|\Delta_{P_\theta,d}(\mathbf{x},\mathbf{y})| \leq \delta$ for a threshold $\delta \in [0,1]$ iff the following holds:*

$$r_{\mathbf{x}} = \frac{\prod_x \theta_{x\,|\,\bar{d}}}{\prod_x \theta_{x\,|\,d}}, \qquad r_{\mathbf{y}} = \frac{\theta_{\bar{d}} \prod_y \theta_{y\,|\,\bar{d}}}{\theta_d \prod_y \theta_{y\,|\,d}},$$

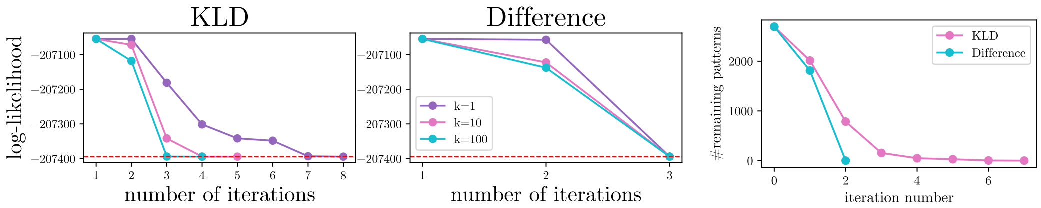
$$\left(\frac{1-\delta}{\delta}\right) r_{\mathbf{x}} r_{\mathbf{y}} - \left(\frac{1+\delta}{\delta}\right) r_{\mathbf{y}} - r_{\mathbf{x}} r_{\mathbf{y}}^2 \leq 1, \qquad -\left(\frac{1+\delta}{\delta}\right) r_{\mathbf{x}} r_{\mathbf{y}} + \left(\frac{1-\delta}{\delta}\right) r_{\mathbf{y}} - r_{\mathbf{x}} r_{\mathbf{y}}^2 \leq 1.$$

Note that above equalities and inequalities are valid signomial program constraints. Thus, we can learn the maximum-likelihood parameters of a naive Bayes network while ensuring a certain pattern is fair by solving a signomial program. Furthermore, we can eliminate multiple patterns by adding the constraints in Proposition 2 for each of them. However, learning a model that is entirely fair with this approach will introduce an exponential number of constraints. Not only does this make the optimization more challenging, but listing all patterns may simply be infeasible.

4.2 Learning δ-fair parameters

To address the aforementioned challenge of removing an exponential number of discrimination patterns, we propose an approach based on the *cutting plane* method. That is, we iterate between *parameter learning* and *constraint extraction*, gradually adding fairness constraints to the optimization. The parameter learning component is as described in the previous section, where we add the constraints of Proposition 2 for each discrimination pattern that has been extracted so far. For constraint extraction we use the top-k pattern miner presented in Section 3.2. At each iteration, we learn the maximum-likelihood parameters subject to fairness constraints, and find k more patterns using the updated parameters to add to the set of constraints in the next iteration. This process is repeated until the search algorithm finds no more discrimination pattern.

In the worst case, our algorithm may add exponentially many fairness constraints whilst solving multiple optimization problems. However, as we will later show empirically, we can learn a δ-fair model by explicitly enforcing only a small fraction of fairness constraints. The efficacy of our approach depends on strategically extracting patterns that are significant in the overall distribution. Here, we again use a ranking by discrimination or divergence score, which we also evaluate empirically.



(a) Log-likelihood (b) Number of remaining patterns

Figure 3: Log-likelihood and the number of remaining discrimination patterns after each iteration of learning on COMPAS dataset with $\delta = 0.1$.

Table 2: Log-likelihood of models learned without fairness constraints, with the δ-fair learner ($\delta = 0.1$), and by making sensitive variables independent from the decision variable.

Dataset	Unconstrained	δ-Fair	Independent
COMPAS	-207,055	-207,395	-208,639
Adult	-226,375	-228,763	-232,180
German	-12,630	-12,635	-12,649

Table 3: Number of remaining patterns with $\delta = 0.1$ in naive Bayes models trained on discrimination-free data, where λ determines the tradeoff between fairness and accuracy in the data repair step [8].

Dataset	λ =0.5	λ =0.9	λ =0.95	λ =0.99	λ =1.0
COMPAS	11,512	7,862	8,872	8,926	0
Adult	>1e6	1,078	1,123	1,087	0
German	>1e6	1	9	0	0

4.3 Empirical evaluation of δ-fair learner

We will now evaluate our iterative algorithm for learning δ-fair naive Bayes models. We use the same datasets and hardware as in Section 3.3. To solve the signomial programs, we use *GPkit*, which finds local solutions to these problems using a convex optimization solver as its backend.[6]

Q1. Can we learn a δ-fair model in a small number of iterations while only asserting a small number of fairness constraints? We train a naive Bayes model on the COMPAS dataset subject to δ-fairness constraints. Fig. 3a shows how the iterative method converges to a δ-fair model, whose likelihood is indicated by the dotted line. Our approach converges to a fair model in a few iterations, including only a small fraction of the fairness constraints. In particular, adding only the most discriminating pattern as a constraint at each iteration learns an entirely δ-fair model with only three fairness constraints.[7] Moreover, Fig. 3b shows the number of remaining discrimination patterns after each iteration of learning with $k = 1$. Note that enforcing a single fairness constraint can eliminate a large number of remaining ones. Eventually, a few constraints subsume all discrimination patterns.

We also evaluated our δ-fair learner on the other two datasets; see Appendix D for plots. We observed that more than a million discrimination patterns that exist in the unconstrained maximum-likelihood models were eliminated using a few dozen to, even in the worst case, a few thousand fairness constraints. Furthermore, stricter fairness requirements (smaller δ) tend to require more iterations, as would be expected. An interesting observation is that neither of the two rankings consistently dominate the other in terms of the number of iterations to converge.

Q2. How does the quality of naive Bayes models from our fair learner compare to ones that make the sensitive attributes independent of the decision? and to the best model without fairness constraints? A simple method to guarantee that a naive Bayes model is δ-fair is to make all sensitive variables independent from the target value. The obvious downside is the negative effect on the predictive power of the model. We compare the models learned by our approach with: (1) a maximum-likelihood model with no fairness constraints (unconstrained) and (2) a model in which the sensitive variables are independent of the decision variable, and the remaining parameters are learned using the maximum-likelihood criterion (independent).

These models lie at two opposite ends of the spectrum of the trade-off between fairness and accuracy. The δ-fair model falls between these extremes, balancing approximate fairness and prediction power. Table 2 shows the log-likelihood of these models for three datasets. The δ-fair models achieve

[6] We use Mosek (`www.mosek.com`) as backend.

[7] There are 2695 discrimination patterns w.r.t. unconstrained naive Bayes on COMPAS and $\delta = 0.1$.

likelihoods that are much closer to those of the unconstrained models than the independent ones. This shows that it is possible to enforce the fairness constraints without a major reduction in model quality.

Q3. Do discrimination patterns still occur when learning Naive Bayes models from fair data? We first use the data repair algorithm in [8] to remove discrimination from data, and learn a naive Bayes model from it. Table 3 shows the number of remaining discrimination patterns in such model. The results indicate that as long as preserving some degree of accuracy is in the objective, this method leaves lots of discrimination patterns, whereas our method removes all patterns.

5 Related work

Most prominent definitions of fairness in machine learning can be largely categorized into *individual fairness* and *group fairness*. Individual fairness is based on the intuition that similar individuals should be treated similarly. For instance, the Lipschitz condition [6] requires that the statistical distance between classifier outputs of two individuals are bounded by a task-specific distance between them. As hinted to earlier, our proposed notion of δ-fairness satisfies the Lipschitz condition if two individuals who differ only in the sensitive attributes are considered similar. However, our definition cannot represent more nuanced similarity metrics that consider relationships between feature values.

An example of group fairness definition is statistical (demographic) parity, which states that a model is fair if the probability of getting a positive decision is equal between two groups defined by the sensitive attribute, i.e. $P(d|s)=P(d|\bar{s})$ where d and S are positive decision and sensitive variable, respectively. Approximate measures of statistical parity include CV-discrimination score [1]: $P(d|s)-P(d|\bar{s})$; and disparate impact (or $p\%$-rule) [8, 21]: $P(d|\bar{s})/P(d|s)$. Our definition of δ-fairness is strictly stronger than requiring a small CV-discrimination score, as a violation of (approximate) statistical parity corresponds to a discrimination pattern with only the sensitive attribute (i.e. empty $\mathbf{y}$). Even though the $p\%$-rule was not explicitly discussed in this paper, our notion of discrimination pattern can be extended to require a small relative (instead of absolute) difference for partial feature observations.

Moreover, the inadequacy of statistical parity in detecting bias for subgroups or individuals has been pointed out numerous times. We resolve such issue by eliminating discrimination patterns for all subgroups that can be expressed as assignments to subsets of features. In fact, we satisfy approximate statistical parity for any subgroup defined over the set of sensitive attributes, as any subgroup can be expressed as a union of joint assignments to the sensitive features, each of which has a bounded discrimination score. Kearns et al. [13] showed that auditing fairness at this arbitrary subgroup level (i.e. detecting *fairness gerrymandering*) is computationally hard.

Other notions of group fairness include equalized true positive rates (equality of opportunity), false positive rates, or both (equalized odds [9]) among groups defined by the sensitive attributes. These definitions are "oblivious" to features other than the sensitive attribute. Moreover, our method still applies in decision making scenarios where a true label is not well defined or hard to observe.

Our approach differs from causal approaches to fairness [14, 16, 20] which are more concerned with the causal mechanism of the real world that generated a potentially unfair decision, whereas we study the effect of sensitive information on a known classifier.

Lastly, current works on learning a fair naive Bayes model include modifying the data [11], changing the model structure [1], and adding a regularizer [12, 23]. We formulate the problem as constrained optimization, an approach often used to ensure fairness in other models [6, 13].

6 Discussion and conclusion

In this paper we introduced a novel definition of fair probability distribution in terms of discrimination patterns which considers exponentially many (partial) observations of features. We have also presented algorithms to search for discrimination patterns in naive Bayes networks and to learn a high-quality fair naive Bayes classifier from data. We empirically demonstrated the efficiency of our search algorithm and the ability to eliminate exponentially many discrimination patterns by iteratively removing a small fraction at a time.

We have shown that our approach of fair distribution implies group fairness such as statistical parity. However, ensuring group fairness in general is always with respect to a distribution and is only valid under the assumption that this distribution is truthful. While our approach guarantees some level of group fairness of naive Bayes classifiers, this is only true if the naive Bayes assumption holds. That is, the group fairness guarantees do not extend to using the classifier on an arbitrary population.

There is always a tension between three criteria of a probabilistic model: its fidelity, fairness, and tractability. Our approach aims to strike a balance between them by giving up some likelihood to be tractable (naive Bayes assumption) and more fair. There are certainly other valid approaches: learning a more general graphical model to increase fairness and truthfulness, which would in general make it intractable, or making the model less fair in order to make it more truthful and tractable.

Lastly, real-world algorithmic fairness problems are only solved by domain experts understanding the process that generated the data, its inherent biases, and which modeling assumptions are appropriate. Our algorithm is only a tool to assist such experts in learning fair distributions: it can provide the domain expert with discrimination patterns, who can then decide which patterns need to be eliminated.

Acknowledgments

This work is partially supported by NSF grants #IIS-1657613, #IIS-1633857, #CCF-1837129, DARPA XAI grant #N66001-17-2-4032, NEC Research, and gifts from Intel and Facebook Research. Golnoosh Farnadi and Behrouz Babaki are supported by postdoctoral scholarships from IVADO through the Canada First Research Excellence Fund (CFREF) grant.

References

[1] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2):277–292, 2010.

[2] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *CoRR*, abs/1703.00056, 2017.

[3] Adnan Darwiche. *Modeling and reasoning with Bayesian networks*. Cambridge University Press, 2009.

[4] Amit Datta, Michael Carl Tschantz, and Anupam Datta. Automated experiments on ad privacy settings. *Proceedings on Privacy Enhancing Technologies*, 2015(1):92–112, 2015.

[5] Rina Dechter. Reasoning with probabilistic and deterministic graphical models: Exact algorithms. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 7(3):1–191, 2013.

[6] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226. ACM, 2012.

[7] Joseph G Ecker. Geometric programming: methods, computations and applications. *SIAM review*, 22(3):338–362, 1980.

[8] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 259–268. ACM, 2015.

[9] Moritz Hardt, Eric Price, Nati Srebro, et al. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

[10] Loren Henderson, Cedric Herring, Hayward Derrick Horton, and Melvin Thomas. Credit where credit is due?: Race, gender, and discrimination in the credit scores of business startups. *The Review of Black Political Economy*, 42(4):459–479, 2015.

[11] Faisal Kamiran and Toon Calders. Classifying without discriminating. In *2009 2nd International Conference on Computer, Control and Communication*, pages 1–6. IEEE, 2009.

[12] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer, 2012.

[13] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *Proceedings of the 35th International Conference on Machine Learning*, pages 2564–2572, 2018.

[14] Niki Kilbertus, Mateo Rojas Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. Avoiding discrimination through causal reasoning. In *Advances in Neural Information Processing Systems*, pages 656–666, 2017.

[15] Doga Kisa, Guy Van den Broeck, Arthur Choi, and Adnan Darwiche. Probabilistic sentential decision diagrams. In *KR*, 2014.

[16] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, pages 4066–4076, 2017.

[17] Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *NIPS*, pages 4069–4079, 2017.

[18] George Mohler, Rajeev Raje, Jeremy Carter, Matthew Valasik, and Jeffrey Brantingham. A penalized likelihood method for balancing accuracy and fairness in predictive policing. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2454–2459. IEEE, 2018.

[19] Hoifung Poon and Pedro Domingos. Sum-product networks: a new deep architecture. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pages 337–346. AUAI Press, 2011.

[20] Chris Russell, Matt J Kusner, Joshua Loftus, and Ricardo Silva. When worlds collide: integrating different counterfactual assumptions in fairness. In *Advances in Neural Information Processing Systems*, pages 6414–6423, 2017.

[21] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. *arXiv preprint arXiv:1507.05259*, 2015.

[22] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, Krishna P. Gummadi, and Adrian Weller. From parity to preference-based notions of fairness in classification. In *NIPS*, pages 228–238, 2017.

[23] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333, 2013.

A Degree of Discrimination Bound

A.1 Proof of Proposition 1

We first derive how $\widetilde{\Delta}$ represents the degree of discrimination Δ for some pattern $\mathbf{xy}$.

$$\begin{aligned}
\Delta_{P,d}(\mathbf{x},\mathbf{y}) &= P(d\,|\,\mathbf{xy}) - P(d\,|\,\mathbf{y})\\
&= \frac{P(\mathbf{x}\,|\,d)P(d\mathbf{y})}{P(\mathbf{x}\,|\,d)P(d\mathbf{y}) + P(\mathbf{x}\,|\,\bar{d})P(\bar{d}\mathbf{y})} - P(d\,|\,\mathbf{y})\\
&= \frac{P(\mathbf{x}\,|\,d)P(d\,|\,\mathbf{y})}{P(\mathbf{x}\,|\,d)P(d\,|\,\mathbf{y}) + P(\mathbf{x}\,|\,\bar{d})P(\bar{d}\,|\,\mathbf{y})} - P(d\,|\,\mathbf{y})\\
&= \widetilde{\Delta}(P(\mathbf{x}\,|\,d), P(\mathbf{x}\,|\,\bar{d}), P(d\,|\,\mathbf{y}))
\end{aligned}$$

Clearly, if $l \leq \gamma \leq u$ then $\min_{l\leq\gamma\leq u} \widetilde{\Delta}(\alpha,\beta,\gamma) \leq \widetilde{\Delta}(\alpha,\beta,\gamma) \leq \max_{l\leq\gamma\leq u} \widetilde{\Delta}(\alpha,\beta,\gamma)$. Therefore, if $l \leq P(d\,|\,\mathbf{yy}') \leq u$, then the following holds for any $\mathbf{x}$:

$$\begin{aligned}
\min_{l\leq\gamma\leq u} \widetilde{\Delta}(P(\mathbf{x}\,|\,d), P(\mathbf{x}\,|\,\bar{d}), \gamma) &\leq \widetilde{\Delta}(P(\mathbf{x}\,|\,d), P(\mathbf{x}\,|\,\bar{d}), P(d\,|\,\mathbf{yy}'))\\
&= \Delta_{P,d}(\mathbf{x},\mathbf{yy}') \leq \max_{l\leq\gamma\leq u} \widetilde{\Delta}(P(\mathbf{x}\,|\,d), P(\mathbf{x}\,|\,\bar{d}), \gamma).
\end{aligned}$$

Next, suppose $\mathbf{x}'_u = \arg\max_{\mathbf{x}'} P(d\,|\,\mathbf{xx}')$ and $\mathbf{x}'_l = \arg\min_{\mathbf{x}'} P(d\,|\,\mathbf{xx}')$. Then from Lemma 1, we also have that $\mathbf{x}'_u = \arg\max_{\mathbf{x}'} P(d\,|\,\mathbf{xx}'\mathbf{yy}')$ and $\mathbf{x}'_l = \arg\min_{\mathbf{x}'} P(d\,|\,\mathbf{xx}'\mathbf{yy}')$ for any $\mathbf{yy}'$. Therefore,

$$\begin{aligned}
&\min_{l\leq\gamma\leq u} \widetilde{\Delta}\left(P(\mathbf{xx}'_l\,|\,d), P(\mathbf{xx}'_l\,|\,\bar{d}), \gamma\right)\\
&\leq \widetilde{\Delta}\left(P(\mathbf{xx}'_l\,|\,d), P(\mathbf{xx}'_l\,|\,\bar{d}), P(d\,|\,\mathbf{yy}')\right) = \Delta_{P,d}(\mathbf{xx}'_l, \mathbf{yy}') = P(d\,|\,\mathbf{xx}'_l\mathbf{yy}') - P(d\,|\,\mathbf{yy}')\\
&\leq P(d\,|\,\mathbf{xx}'\mathbf{yy}') - P(d\,|\,\mathbf{yy}') = \Delta_{P,d}(\mathbf{xx}', \mathbf{yy}')\\
&\leq \Delta_{P,d}(\mathbf{xx}'_u, \mathbf{yy}') = \widetilde{\Delta}\left(P(\mathbf{xx}'_u\,|\,d), P(\mathbf{xx}'_u\,|\,\bar{d}), P(d\,|\,\mathbf{yy}')\right)\\
&\leq \max_{l\leq\gamma\leq u} \widetilde{\Delta}\left(P(\mathbf{xx}'_u\,|\,d), P(\mathbf{xx}'_u\,|\,\bar{d}), \gamma\right). \qquad \square
\end{aligned}$$

A.2 Computing the Discrimination Bound

If $\alpha = P(\mathbf{x}\,|\,d) = 0$ and $\beta = P(\mathbf{x}\,|\,\bar{d}) = 0$, then the probability of $\mathbf{x}$ is zero and thus $P(d\,|\,\mathbf{xy})$ is ill-defined. Therefore, we will assume that either α or β is nonzero.

Let us write $\widetilde{\Delta}_{\alpha,\beta}(\gamma) = \widetilde{\Delta}(\alpha,\beta,\gamma)$ to denote the function restricted to fixed α and β. If $\alpha = \beta$, then $\widetilde{\Delta}_{\alpha,\beta} = 0$. Also, $\widetilde{\Delta}_{0,\beta}(\gamma) = -\gamma$ and $\widetilde{\Delta}_{\alpha,0}(\gamma) = 1 - \gamma$. Thus, in the following analysis we assume α and β are non-zero and distinct.

If $0 < \alpha \leq \beta \leq 1$, $\widetilde{\Delta}_{\alpha,\beta}$ is negative and convex in γ within $0 \leq \gamma \leq 1$. On the other hand, if $0 < \beta \leq \alpha \leq 1$, then $\widetilde{\Delta}_{\alpha,\beta,\gamma}$ is positive and concave. This can quickly be checked using the following derivatives.

$$\frac{d}{d\gamma}\widetilde{\Delta}_{\alpha,\beta}(\gamma) = \frac{\alpha\beta}{(\alpha\gamma + \beta(1-\gamma))^2} - 1, \quad \frac{d^2}{d\gamma^2}\widetilde{\Delta}_{\alpha,\beta}(\gamma) = \frac{-2\alpha\beta(\alpha-\beta)}{(\alpha\gamma + \beta(1-\gamma))^3}$$

Furthermore, the sign of the derivative at $\gamma = 0$ is different from that at $\gamma = 1$, and thus there must exist a unique optimum in $0 \leq \gamma \leq 1$.

Solving for $\frac{d}{d\gamma}\widetilde{\Delta}_{\alpha,\beta}(\gamma) = 0$, we get $\gamma = \frac{\beta\pm\sqrt{\alpha\beta}}{\beta-\alpha}$. The solution corresponding to the feasible space $0 \leq \gamma \leq 1$ is: $\gamma_{\text{opt}} = \frac{\beta-\sqrt{\alpha\beta}}{\beta-\alpha}$. The optimal value is derived as the following.

$$\widetilde{\Delta}_{\alpha,\beta}(\gamma_{\text{opt}}) = \frac{\alpha\left(\frac{\beta-\sqrt{\alpha\beta}}{\beta-\alpha}\right)}{(\alpha-\beta)\left(\frac{\beta-\sqrt{\alpha\beta}}{\beta-\alpha}\right)+\beta} - \frac{\beta-\sqrt{\alpha\beta}}{\beta-\alpha} = \frac{\alpha(\beta-\sqrt{\alpha\beta})}{\sqrt{\alpha\beta}(\beta-\alpha)} - \frac{\beta-\sqrt{\alpha\beta}}{\beta-\alpha} = \frac{2\sqrt{\alpha\beta}-\alpha-\beta}{\beta-\alpha}$$

Next, suppose that the feasible space is restricted to $l \leq \gamma \leq u$. Then the optimal solution is: γ_{opt} if $l \leq \gamma_{\text{opt}} \leq u$; l if $\gamma_{\text{opt}} < l$; and u if $\gamma_{\text{opt}} > u$.

A.3 Proof of Lemma 1

Now we prove that we can maximize the posterior decision probability by maximizing each variable independently. It suffices to prove that for a single variable V and all evidence $\mathbf{w}$, $\arg\max_v P(d \,|\, v\mathbf{w}) = \arg\max_v \frac{P(v \,|\, d)}{P(v \,|\, \bar{d})}$. We first express $P(d \,|\, v\mathbf{w})$ as the following:

$$P(d \,|\, v\mathbf{w}) = \frac{P(v \,|\, d)P(d \,|\, \mathbf{w})}{P(v \,|\, d)P(d \,|\, \mathbf{w}) + P(v \,|\, \bar{d})P(\bar{d} \,|\, \mathbf{w})} = \frac{1}{1 + \frac{P(v \,|\, \bar{d})P(\bar{d} \,|\, \mathbf{w})}{P(v \,|\, d)P(d \,|\, \mathbf{w})}}$$

Then clearly,

$$\arg\max_v P(d \,|\, v\mathbf{w}) = \arg\min_v \frac{P(v \,|\, \bar{d})P(\bar{d} \,|\, \mathbf{w})}{P(v \,|\, d)P(d \,|\, \mathbf{w})} = \arg\max_v \frac{P(v \,|\, d)}{P(v \,|\, \bar{d})}. \qquad \square$$

B Divergence Score

B.1 Derivation of Equation 2

We want to find the closed form solution of the optimization problem in Equation 1. Because P and Q differs only in two assignments, we can write the KL divergence as follows:

$$\mathrm{KL}\left(P \parallel Q\right) = \sum_{d\mathbf{z}} P(d\mathbf{z}) \log\left(\frac{P(d\mathbf{z})}{Q(d\mathbf{z})}\right) = P(d\mathbf{xy}) \log\left(\frac{P(d\mathbf{xy})}{Q(d\mathbf{xy})}\right) + P(\bar{d}\mathbf{xy}) \log\left(\frac{P(\bar{d}\mathbf{xy})}{Q(\bar{d}\mathbf{xy})}\right)$$

Let r be the change in probability of $d\mathbf{xy}$. That is, $r = Q(d\mathbf{xy}) - P(d\mathbf{xy})$. For Q to be a valid probability distribution, we must have $Q(d\mathbf{xy}) + Q(\bar{d}\mathbf{xy}) = P(\mathbf{xy})$. Then we have $Q(d\mathbf{xy}) = P(d\mathbf{xy}) + r$, and $Q(\bar{d}\mathbf{xy}) = P(\mathbf{xy}) - Q(d\mathbf{xy}) = P(\bar{d}\mathbf{xy}) - r$. We can then express the KL divergence between P and Q as a function of P and r:

$$g_{P,d,\mathbf{x},\mathbf{y}}(r) \triangleq P(d\mathbf{xy}) \log\left(\frac{P(d\mathbf{xy})}{P(d\mathbf{xy}) + r}\right) + P(\bar{d}\mathbf{xy}) \log\left(\frac{P(\bar{d}\mathbf{xy})}{P(\bar{d}\mathbf{xy}) - r}\right)$$

Moreover, the discrimination score of pattern $\mathbf{xy}$ w.r.t Q can be expressed using P and r as the following:

$$\begin{aligned} Q(d \,|\, \mathbf{xy}) - Q(d \,|\, \mathbf{y}) &= \frac{P(d\mathbf{xy}) + r}{P(\mathbf{xy})} - \frac{P(d\mathbf{y}) + r}{P(\mathbf{y})} = P(d \,|\, \mathbf{xy}) - P(d \,|\, \mathbf{y}) + r\left(\frac{1}{P(\mathbf{xy})} - \frac{1}{P(\mathbf{y})}\right) \\ &= \Delta_{P,d}(\mathbf{x},\mathbf{y}) + r\left(\frac{1}{P(\mathbf{xy})} - \frac{1}{P(\mathbf{y})}\right). \end{aligned}$$

The heuristic $\mathrm{Div}_{P,d,\delta}(\mathbf{x},\mathbf{y})$ is then written using r as follows:

$$\min_r \; g_{P,d,\mathbf{x},\mathbf{y}}(r) \;\; \text{s.t.} \;\; \left|\Delta_{P,d}(\mathbf{x},\mathbf{y}) + r\left(\frac{1}{P(\mathbf{xy})} - \frac{1}{P(\mathbf{y})}\right)\right| \leq \delta \qquad (3)$$
$$-P(d\mathbf{xy}) \leq r \leq P(\bar{d}\mathbf{xy})$$

The objective function $g_{P,d,\mathbf{x},\mathbf{y}}$ is convex in r with its unconstrained global minimum at $r = 0$. Note that this is a feasible point if and only if $|\Delta_{P,d}(\mathbf{x},\mathbf{y})| \leq \delta$; in other words, when the pattern $\mathbf{xy}$ is already fair. Otherwise, the optimum must be either of the extreme points of the feasible space, whichever is closer to 0. The extreme points for the first set of inequalities are:

$$r_1 = \frac{\delta - P(d \,|\, \mathbf{xy}) + P(d \,|\, \mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})}, \quad r_2 = \frac{-\delta - P(d \,|\, \mathbf{xy}) + P(d \,|\, \mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})}.$$

If $\Delta_{P,d}(\mathbf{x},\mathbf{y}) > \delta$, then $r_2 \leq r_1 < 0$. In such case, $g(r_2) \geq g(r_1)$ and $-P(d\mathbf{xy}) \leq r_1 \leq P(\bar{d}\mathbf{xy})$ as shown below:

$r_1 < 0 \leq P(\bar{d}\mathbf{xy})$,

$$-r_1 = \frac{-\delta + P(d\,|\,\mathbf{xy}) - P(d\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} \leq \frac{P(d\,|\,\mathbf{xy}) - P(d\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} \leq \frac{P(d\,|\,\mathbf{xy}) - P(d\mathbf{x}\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} = P(d\mathbf{xy})$$

Similarly, if $\Delta_{P,d}(\mathbf{x},\mathbf{y}) < -\delta$, then $r_1 \geq r_2 > 0$. Also, $g(r_1) \geq g(r_2)$ and $-P(d\mathbf{xy}) \leq r_2 \leq P(\bar{d}\mathbf{xy})$ as shown below:

$$\begin{aligned} &r_2 > 0 \geq -P(d\mathbf{xy}), \\ &r_2 \leq \frac{-P(d\,|\,\mathbf{xy}) + P(d\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} \leq \frac{P(\bar{d}\,|\,\mathbf{xy}) - P(\bar{d}\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} = P(\bar{d}\mathbf{xy}) \end{aligned}$$

Hence, the optimal solution $r^\star$ is

$$r^\star = \begin{cases} 0, & \text{if } |\Delta_{P,d}(\mathbf{x},\mathbf{y})| \leq \delta, \\ \frac{\delta - \Delta_{P,d}(\mathbf{x},\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})}, & \text{if } \Delta_{P,d}(\mathbf{x},\mathbf{y}) > \delta, \\ \frac{-\delta - \Delta_{P,d}(\mathbf{x},\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})}, & \text{if } \Delta_{P,d}(\mathbf{x},\mathbf{y}) < -\delta, \end{cases}$$

and the divergence score is $\mathrm{Div}_{P,d,\delta}(\mathbf{x},\mathbf{y}) = g_{P,d,\mathbf{x},\mathbf{y}}(r^\star)$.

B.2 Upper Bounds on Divergence Score

Here we present two upper bounds on the divergence score for pruning the search tree. The first bound uses the observation that the hypothetical distribution Q with $\Delta_{Q,d}(\mathbf{x},\mathbf{y}) = 0$ is always a feasible hypothetical fair distribution.

Proposition 3. *Let P be a Naive Bayes distribution over $D \cup \mathbf{Z}$, and let $\mathbf{x}$ and $\mathbf{y}$ be joint assignments to $\mathbf{X} \subseteq \mathbf{S}$ and $\mathbf{Y} \subseteq \mathbf{Z} \setminus \mathbf{X}$. For all possible valid extensions $\mathbf{x}'$ and $\mathbf{y}'$, the following holds:*

$$\mathrm{Div}_{P,d,\delta}(\mathbf{xx}',\mathbf{yy}') \leq \; P(d\mathbf{xy}) \log \frac{\max_{\mathbf{z} \models \mathbf{xy}} P(d\,|\,\mathbf{z})}{\min_{\mathbf{z} \models \mathbf{y}} P(d\,|\,\mathbf{z})} + P(\bar{d}\mathbf{xy}) \log \frac{\max_{\mathbf{z} \models \mathbf{xy}} P(\bar{d}\,|\,\mathbf{z})}{\min_{\mathbf{z} \models \mathbf{y}} P(\bar{d}\,|\,\mathbf{z})}$$

Proof. Consider the following point:

$$r_0 = \frac{-P(d\,|\,\mathbf{xy}) + P(d\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})}.$$

First, we show that above r_0 is always a feasible point in Problem 3:

$$\left| \Delta_{P,d}(\mathbf{x},\mathbf{y}) + r_0 \left(\frac{1}{P(\mathbf{xy})} - \frac{1}{P(\mathbf{y})} \right) \right| = |\Delta_{P,d}(\mathbf{x},\mathbf{y}) - \Delta_{P,d}(\mathbf{x},\mathbf{y})| = 0 \leq \delta,$$

$$\begin{aligned} r_0 &= \frac{P(\bar{d}\,|\,\mathbf{xy}) - P(\bar{d}\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} \leq \frac{P(\bar{d}\,|\,\mathbf{xy}) - P(\bar{d}\mathbf{x}\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} = P(\bar{d}\mathbf{xy}), \\ -r_0 &= \frac{P(d\,|\,\mathbf{xy}) - P(d\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} \leq \frac{P(d\,|\,\mathbf{xy}) - P(d\mathbf{x}\,|\,\mathbf{y})}{1/P(\mathbf{xy}) - 1/P(\mathbf{y})} = P(d\mathbf{xy}). \end{aligned}$$

Then the divergence score for any pattern must be smaller than $g_{P,d,\mathbf{x},\mathbf{y}}(r_0)$:

$$\begin{aligned} \mathrm{Div}_{P,d,\delta}(\mathbf{x},\mathbf{y}) \leq g_{P,d,\mathbf{x},\mathbf{y}}(r_0) &= P(d\mathbf{xy}) \log \frac{P(d\,|\,\mathbf{xy})}{P(d\,|\,\bar{\mathbf{x}}\mathbf{y})} + P(\bar{d}\mathbf{xy}) \log \frac{P(\bar{d}\,|\,\mathbf{xy})}{P(\bar{d}\,|\,\bar{\mathbf{x}}\mathbf{y})} \\ &\leq P(d\mathbf{xy}) \log \frac{P(d\,|\,\mathbf{xy})}{\min_{\mathbf{x}} P(d\,|\,\mathbf{xy})} + P(\bar{d}\mathbf{xy}) \log \frac{P(\bar{d}\,|\,\mathbf{xy})}{\min_{\mathbf{x}} P(\bar{d}\,|\,\mathbf{xy})}. \end{aligned}$$

Here, we use $\bar{\mathbf{x}}$ to mean that $\mathbf{x}$ does not hold. In other words,

$$P(d\,|\,\bar{\mathbf{x}}\mathbf{y}) = \frac{P(d\mathbf{y}) - P(d\mathbf{xy})}{P(\mathbf{y}) - P(\mathbf{xy})} = \sum_{\mathbf{x}} P(d\,|\,\mathbf{xy}) P(\mathbf{x}\,|\,\bar{\mathbf{x}}\mathbf{y}).$$

We can then use this to bound the divergence score any pattern extended from $\mathbf{xy}$:

$$\mathrm{Div}_{P,d,\delta}(\mathbf{xx}',\mathbf{yy}')$$

$$\leq P(d\mathbf{xx'yy'}) \log \frac{P(d \,|\, \mathbf{xx'yy'})}{\min_{\mathbf{xx'}} P(d \,|\, \mathbf{xx'yy'})} + P(\bar{d}\mathbf{xx'yy'}) \log \frac{P(\bar{d} \,|\, \mathbf{xx'yy'})}{\min_{\mathbf{xx'}} P(\bar{d} \,|\, \mathbf{xx'yy'})}$$

$$\leq P(d\mathbf{xy}) \log \frac{\max_{\mathbf{z} \models \mathbf{xy}} P(d \,|\, \mathbf{z})}{\min_{\mathbf{z} \models \mathbf{y}} P(d \,|\, \mathbf{z})} + P(\bar{d}\mathbf{xy}) \log \frac{\max_{\mathbf{z} \models \mathbf{xy}} P(\bar{d} \,|\, \mathbf{z})}{\min_{\mathbf{z} \models \mathbf{y}} P(\bar{d} \,|\, \mathbf{z})}.$$

□

We can also bound the divergence score using the maximum and minimum possible discrimination scores shown in Proposition 1, in place of the current pattern's discrimination. Let us denote the bounds for discrimination score as follows:

$$\overline{\Delta}(\mathbf{x}, \mathbf{y}) = \max_{l \leq \gamma \leq u} \widetilde{\Delta}\left(P(\mathbf{xx}'_u \,|\, d), P(\mathbf{xx}'_u \,|\, \bar{d}), \gamma\right), \; \underline{\Delta}(\mathbf{x}, \mathbf{y}) = \min_{l \leq \gamma \leq u} \widetilde{\Delta}\left(P(\mathbf{xx}'_l \,|\, d), P(\mathbf{xx}'_l \,|\, \bar{d}), \gamma\right).$$

Proposition 4. *Let P be a Naive Bayes distribution over $D \cup \mathbf{Z}$, and let $\mathbf{x}$ and $\mathbf{y}$ be joint assignments to $\mathbf{X} \subseteq \mathbf{S}$ and $\mathbf{Y} \subseteq \mathbf{Z} \setminus \mathbf{X}$. For all possible valid extensions $\mathbf{x}'$ and $\mathbf{y}'$, $\mathrm{Div}_{P,d,\delta}(\mathbf{xx}', \mathbf{yy}') \leq \max\left(g_{P,d,\mathbf{xx}',\mathbf{yy}'}(r_u), g_{P,d,\mathbf{xx}',\mathbf{yy}'}(r_l)\right)$ where*

$$r_u = \frac{\delta - \overline{\Delta}(\mathbf{x}, \mathbf{y})}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})}, \quad r_l = \frac{-\delta - \underline{\Delta}(\mathbf{x}, \mathbf{y})}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})}.$$

Proof. The proof proceeds by case analysis on the discrimination score of extended patterns $\mathbf{xx'yy'}$.

First, if $|\Delta(\mathbf{xx}', \mathbf{yy}')| \leq \delta$, $\mathrm{Div}_{P,d,\delta}(\mathbf{xx}', \mathbf{yy}') = 0$ which is the global minimum, and thus is smaller than both $g(r_u)$ and $g(r_l)$.

Next, suppose $\Delta(\mathbf{xx}', \mathbf{yy}') > \delta$. Then from Proposition 1,

$$r_u = \frac{\delta - \overline{\Delta}(\mathbf{x}, \mathbf{y})}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})} \leq r^\star = \frac{\delta - \Delta_{P,d}(\mathbf{xx}', \mathbf{yy}')}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})} < 0.$$

As g is convex with its minimum at 0, we can conclude $\mathrm{Div}_{P,d,\delta}(\mathbf{xx}', \mathbf{yy}') = g(r^\star) \leq g(r_u)$.

Finally, if $\Delta(\mathbf{xx}', \mathbf{yy}') < -\delta$, we have

$$r_l = \frac{-\delta - \underline{\Delta}(\mathbf{x}, \mathbf{y})}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})} \geq r^\star = \frac{-\delta - \Delta_{P,d}(\mathbf{xx}', \mathbf{yy}')}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})} > 0.$$

Similarly, this implies $\mathrm{Div}_{P,d,\delta}(\mathbf{xx}', \mathbf{yy}') = g(r^\star) \leq g(r_l)$. Because the divergence score is always smaller than either $g(r_u)$ or $g(r_l)$, it must be smaller than $\max(g(r_u), g(r_l))$. □

Lastly, we show how to efficiently compute an upper bound on $g_{P,d,\mathbf{xx}',\mathbf{yy}'}(r_u)$ $g_{P,d,\mathbf{xx}',\mathbf{yy}'}(r_l)$ from Proposition 4 for all patterns extended from $\mathbf{xy}$. This is necessary for pruning during the search for discrimination patterns with high divergence scores. First, note that r_u and r_l can be expressed as

$$\frac{c}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})}, \tag{4}$$

where $c = \delta - \overline{\Delta}(\mathbf{x}, \mathbf{y})$ for r_u and $c = -\delta - \underline{\Delta}(\mathbf{x}, \mathbf{y})$ for r_l. Hence, it suffices to derive the following bound.

$$\begin{aligned}
& g_{P,d,\mathbf{xx}',\mathbf{yy}'}\left(\frac{c}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})}\right) \\
&= P(d\mathbf{xx'yy'}) \log \left(\frac{P(d\mathbf{xx'yy'})}{P(d\mathbf{xx'yy'}) + \frac{c}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})}}\right) \\
&\quad + P(\bar{d}\mathbf{xx'yy'}) \log \left(\frac{P(\bar{d}\mathbf{xx'yy'})}{P(\bar{d}\mathbf{xx'yy'}) - \frac{c}{1/P(\mathbf{xx'yy'}) - 1/P(\mathbf{yy'})}}\right) \\
&= P(d\mathbf{xx'yy'}) \log \left(\frac{P(d \,|\, \mathbf{xx'yy'})(1 - P(\mathbf{xx}' \,|\, \mathbf{yy}'))}{P(d \,|\, \mathbf{xx'yy'})(1 - P(\mathbf{xx}' \,|\, \mathbf{yy}')) + c}\right) \\
&\quad + P(\bar{d}\mathbf{xx'yy'}) \log \left(\frac{P(\bar{d} \,|\, \mathbf{xx'yy'})(1 - P(\mathbf{xx}' \,|\, \mathbf{yy}'))}{P(\bar{d} \,|\, \mathbf{xx'yy'})(1 - P(\mathbf{xx}' \,|\, \mathbf{yy}')) - c}\right)
\end{aligned}$$

$$\leq \begin{cases} 0 & \text{if } c = 0 \\ P(d\mathbf{x}\mathbf{y}) \log \frac{(\max_{\mathbf{z} \models \mathbf{x}\mathbf{y}} P(d \,|\, \mathbf{z}))(1 - \min_{\mathbf{x}'\mathbf{y}'} P(\mathbf{x}\mathbf{x}' \,|\, \mathbf{y}\mathbf{y}'))}{(\min_{\mathbf{z} \models \mathbf{x}\mathbf{y}} P(d \,|\, \mathbf{z}))(1 - \max_{\mathbf{x}'\mathbf{y}'} P(\mathbf{x}\mathbf{x}' \,|\, \mathbf{y}\mathbf{y}')) + c} & \text{if } c < 0 \\ P(\bar{d}\mathbf{x}\mathbf{y}) \log \frac{(\max_{\mathbf{z} \models \mathbf{x}\mathbf{y}} P(\bar{d} \,|\, \mathbf{z}))(1 - \min_{\mathbf{x}'\mathbf{y}'} P(\mathbf{x}\mathbf{x}' \,|\, \mathbf{y}\mathbf{y}'))}{(\min_{\mathbf{z} \models \mathbf{x}\mathbf{y}} P(\bar{d} \,|\, \mathbf{z}))(1 - \max_{\mathbf{x}'\mathbf{y}'} P(\mathbf{x}\mathbf{x}' \,|\, \mathbf{y}\mathbf{y}')) - c} & \text{if } c > 0 \end{cases}$$

C Proof of Proposition 2

The probability values of positive decision in terms of naive Bayes parameters θ are as follows:

$$P_\theta(d \,|\, \mathbf{x}\mathbf{y}) = \frac{P_\theta(d\mathbf{x}\mathbf{y})}{P_\theta(\mathbf{x}\mathbf{y})} = \frac{\theta_d \prod_x \theta_{x \,|\, d} \prod_y \theta_{y \,|\, d}}{\theta_d \prod_x \theta_{x \,|\, d} \prod_y \theta_{y \,|\, d} + \theta_{\bar{d}} \prod_x \theta_{x \,|\, \bar{d}} \prod_y \theta_{y \,|\, \bar{d}}} = \frac{1}{1 + \frac{\theta_{\bar{d}} \prod_x \theta_{x \,|\, \bar{d}} \prod_y \theta_{y \,|\, \bar{d}}}{\theta_d \prod_x \theta_{x \,|\, d} \prod_y \theta_{y \,|\, d}}},$$

$$P_\theta(\bar{d} \,|\, \mathbf{y}) = \frac{P_\theta(d\mathbf{y})}{P_\theta(\mathbf{y})} = \frac{1}{1 + \frac{\theta_{\bar{d}} \prod_y \theta_{y \,|\, \bar{d}}}{\theta_d \prod_y \theta_{y \,|\, d}}}.$$

For simplicity of notation, let us write:

$$r_{\mathbf{x}} = \frac{\prod_x \theta_{x \,|\, \bar{d}}}{\prod_x \theta_{x \,|\, d}}, \quad r_{\mathbf{y}} = \frac{\theta_{\bar{d}} \prod_y \theta_{y \,|\, \bar{d}}}{\theta_d \prod_y \theta_{y \,|\, d}}. \tag{5}$$

Then the degree of discrimination is $\Delta_{P_\theta,d}(\mathbf{x},\mathbf{y}) = P_\theta(d \,|\, \mathbf{x}\mathbf{y}) - P_\theta(\bar{d} \,|\, \mathbf{y}) = \frac{1}{1+r_{\mathbf{x}}r_{\mathbf{y}}} - \frac{1}{1+r_{\mathbf{y}}}$. Now we express the fairness constraint $|\Delta_{P_\theta,d}(\mathbf{x},\mathbf{y})| \leq \delta$ as the following two inequalities:

$$-\delta \leq \frac{(1 + r_{\mathbf{y}}) - (1 + r_{\mathbf{x}}r_{\mathbf{y}})}{(1 + r_{\mathbf{x}}r_{\mathbf{y}}) \cdot (1 + r_{\mathbf{y}})} \leq \delta.$$

After simplifying,

$$r_{\mathbf{y}} - r_{\mathbf{x}}r_{\mathbf{y}} \geq -\delta(1 + r_{\mathbf{x}}r_{\mathbf{y}} + r_{\mathbf{y}} + r_{\mathbf{x}}r_{\mathbf{y}}^2), \quad r_{\mathbf{y}} - r_{\mathbf{x}}r_{\mathbf{y}} \qquad \leq \delta(1 + r_{\mathbf{x}}r_{\mathbf{y}} + r_{\mathbf{y}} + r_{\mathbf{x}}r_{\mathbf{y}}^2).$$

We further express this as the following two signomial inequality constraints:

$$\left(\frac{1-\delta}{\delta}\right) r_{\mathbf{x}}r_{\mathbf{y}} - \left(\frac{1+\delta}{\delta}\right) r_{\mathbf{y}} - r_{\mathbf{x}}r_{\mathbf{y}}^2 \leq 1, \quad -\left(\frac{1+\delta}{\delta}\right) r_{\mathbf{x}}r_{\mathbf{y}} + \left(\frac{1-\delta}{\delta}\right) r_{\mathbf{y}} - r_{\mathbf{x}}r_{\mathbf{y}}^2 \leq 1 \tag{6}$$

Note that $r_{\mathbf{x}}$ and $r_{\mathbf{y}}$ according to Equation 5 are monomials of θ, and thus above constraints are also signomial with respect to the optimization variables θ. □

D Additional Experiments

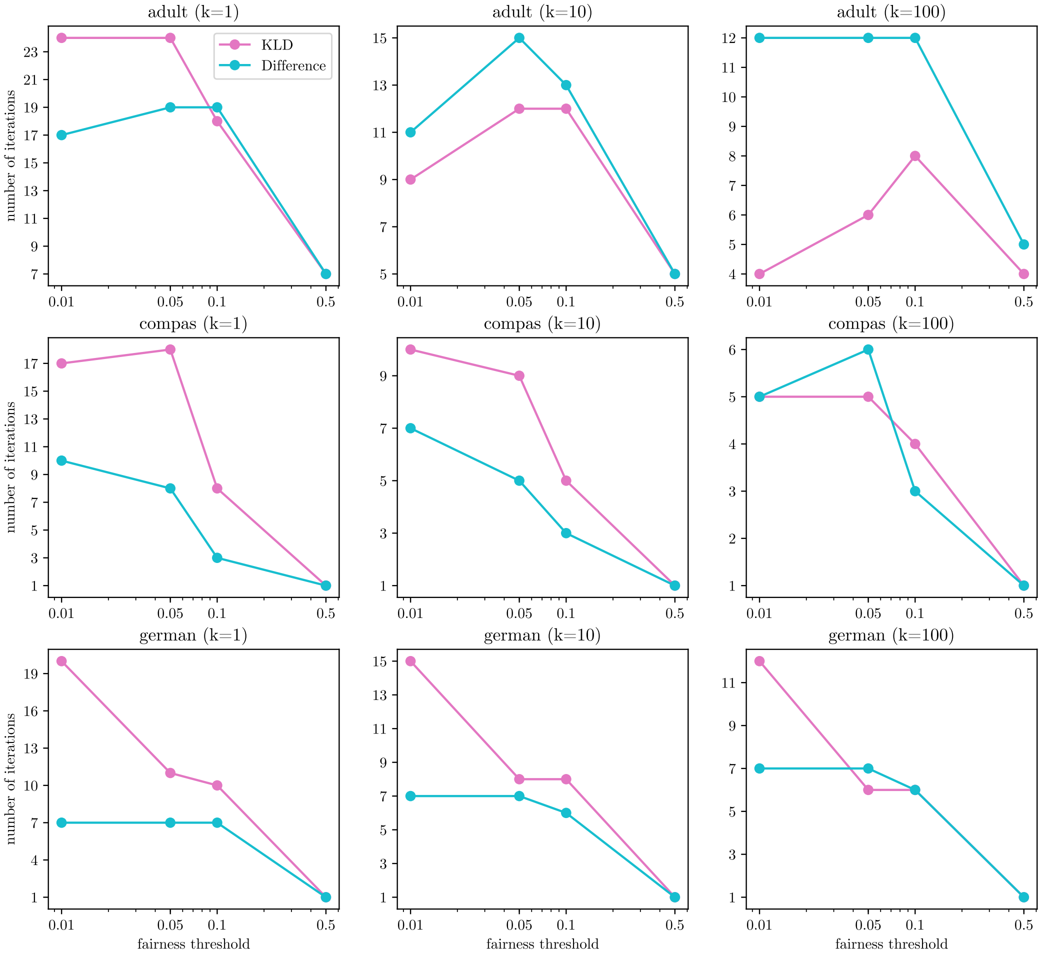


Figure 4: Number of iterations of δ-fair learner until convergence