2013 IEEE 13th International Conference on Data Mining Workshops

# The Independence of Fairness-aware Classifiers

Toshihiro Kamishima*,Shotaro Akaho*, Hideki Asoh*, and Jun Sakuma†
*National Institute of Advanced Industrial Science and Technology (AIST),
AIST Tsukuba Central 2, Umezono 1-1-1, Tsukuba, Ibaraki, 305-8568 Japan,
Email: mail@kamishima.net (http://www.kamishima.net/), s.akaho@aist.go.jp, and h.asoh@aist.go.jp
†University of Tsukuba, 1-1-1 Tennodai, Tsukuba, 305-8577 Japan, Email: jun@cs.tsukuba.ac.jp

***Abstract*—Due to the spread of data mining technologies, such technologies are being used for determinations that seriously affect individuals' lives. For example, credit scoring is frequently determined based on the records of past credit data together with statistical prediction techniques. Needless to say, such determinations must be nondiscriminatory and fair in sensitive features, such as race, gender, religion, and so on. The goal of fairness-aware classifiers is to classify data while taking into account the potential issues of fairness, discrimination, neutrality, and/or independence. In this paper, after reviewing fairness-aware classification methods, we focus on one such method, Calders and Verwer's two-naive-Bayes method. This method has been shown superior to the other classifiers in terms of fairness, which is formalized as the statistical independence between a class and a sensitive feature. However, the cause of the superiority is unclear, because it utilizes a somewhat heuristic post-processing technique rather than an explicitly formalized model. We clarify the cause by comparing this method with an alternative naive Bayes classifier, which is modified by a modeling technique called *hypothetical fair-factorization*. This investigation reveals the theoretical background of the two-naive-Bayes method and its connections with other methods. Based on these findings, we develop another naive Bayes method with an *actual fair-factorization* technique and empirically show that this new method can achieve an equal level of fairness as that of the two-naive-Bayes classifier.**

***Keywords*—fairness, discrimination, generative model, naive Bayes classifier**

## I. Introduction

The goal of fairness-aware data mining is to analyze data while taking into account issues or potential issues of fairness, discrimination, neutrality, and independence. A typical application of these mining techniques is to avoid social discrimination. Due to the accumulation of vast stores of digitized personal data, data mining techniques are being increasingly used for serious decisions that affect individual's lives such as credit, insurance rates, employment applications, and so on. For example, credit scoring is frequently decided based on the records of past credit data together with statistical prediction techniques. Such decisions are considered unfair in both a social and legal sense if they have been made based on sensitive features such as gender, religion, race, ethnicity, handicaps, political convictions, and so on. Pedreschi et al. first proposed the concept of fairness-aware data mining [1] to detect such unfair determinations. After the publication of this pioneering work, several types of fairness-aware data mining tasks have been proposed.

Fairness-aware data-mining tasks can currently be classified into two groups [2]: unfairness discovery from data and unfairness prevention in data mining. The first unfairness discovery task aims to check whether specified decisions depended on sensitive features and to enumerate all unfair decisions in a given database [1]. Methods for correcting these detected patterns of unfairness have been discussed, too. The second task, unfairness prevention, aims to learn a statistical model for prediction or decision from potentially unfair data sets so that the sensitive feature does not influence the prediction or decision. There are two approaches to this task. The goal of fairness-aware learning is to design machine learning methods while taking the fairness of the analysis results into account [3]. In fairness-aware data publication, potentially unfair data sets are converted so that the sensitive feature does not influence the other variables of the data [4], and the data sets are then processed by standard mining methods.

Our first contribution is to review fairness-aware classifiers. An unfairness prevention task is generally carried out by a variant of standard analysis methods that is modified so as to enhance the fairness. Several types of analysis tasks are targeted: classification [3], recommendation [5], and clustering [6]. Among these tasks, we focus on the use of fairness aware-classifiers, which is a classification method designed to prevent unfairness.

Our second contribution is to examine a fairness-aware classifier, Calders and Verwer's two-naive-Bayes method [3]. This two-naive-Bayes classifier has achieved a higher level of fairness than other fairness-aware classifiers, but it is not clear why. This is because its fairness is enhanced by a somewhat heuristic post-processing technique; it remains obscure what statistical model is learned.

To clear up the cause of this phenomenon, we first introduce a simple alternative model, and then show the reason why the model is inferior to the two-naive-Bayes model. This simple alternative model is applied a technique of *hypothetical fair-factorization* to a naive Bayes model. The aim of the hypothetical fair-factorization is to create probabilistic generative models so that the models make fair decisions. It is easy to understand the characteristics of this alternative model because it is simple and its global optimum can be derived analyticially. We next show its connections with a two-naive-Bayes method and Kamiran et al.'s decision theory [7]. We empirically confirm that the degree of fairness of this alternative model is inferior to that of the two-naive-Bayes model, like the other fairness-aware classifiers.

We then hypothesize two reasons why this simple alternative performed poorly and show experimental results on synthetic data to validate our hypotheses. The first reason is

978-0-7695-5109-8/13 $31.00 © 2013 IEEE
DOI 10.1109/ICDMW.2013.133

a model bias, which makes an estimated distribution different from a true distribution. This difference damages the fairness of the learned classifier. The second reason is the deterministic Bayes decision rule. Though class labels are in fact chosen according to a deterministic Bayes decision rule, our simple version assumes that labels are probabilistically decided. After discussing the theoretical backgrounds of these two reasons, we show experimental results on synthetic data to validate our hypotheses.

Our final contribution is to develop a modeling technique that maintains an equal level of fairness to that attained by the two-naive-Bayes method. This technique, which we call *actual fair-factorization*, eliminates the above two defects by a method to adjust for the deviations caused by model bias and the deterministic decision rule. The performance of this new technique showed drastic improvement.

This paper is organized as follows. In section II, we briefly review concepts and tasks of fairness-aware data mining and then introduce the sub-task of fairness-aware classification. In section III, we propose a hypothetical fair-factorization technique and show its experimental results on a benchmark data set. In section IV, we analyze why this hypothetical fair-factorization model failed and empirically validate our hypotheses. In section V, we develop an actual fair-factorization technique to eliminate the defects of the hypothetical version, and successfully test this new model for effectiveness. In section VI, we summarize our conclusions.

## II. Fairness-aware Data Mining

This section summarizes fairness-aware data mining. Following the definitions of notations, we sequentially review the formal notion of fairness, a few example of applications, and tasks of fairness-aware data mining. We then focus on methods for fairness-aware classification, especially Calders and Verwer's two-naive-Bayes and Kamiran's decision theory.

### A. Notations

The goal of *fairness-aware data mining* (FADM) is to analyze data and simultaneously to take into account issues or potential issues of fairness, discrimination, neutrality, and independence. Three types of variables are used in FADM. The random variables $S$ and $\mathbf{X}$ respectively denote *sensitive* and *non-sensitive features*. Techniques of FADM maintain fairness regarding the information expressed by any sensitive features. For example, in the case of avoiding discrimination as described in the introduction, a sensitive feature may correspond to gender, religion, race, or some other feature specified from a social or legal viewpoint. Non-sensitive features consist of all features other than a sensitive feature. The random variable $Y$ denotes a *target variable* that expresses the information in which the data analysts are interested. In a case of the credit application, $Y$ expresses a binary determination of whether to approve or deny the application. Target variables may be continuous or discrete according to a goal of the data mining task. $A \perp\!\!\!\perp B$ denotes the (unconditional) independence between variables $A$ and $B$, and $A \perp\!\!\!\perp B \,|\, C$ denotes the conditional independence between $A$ and $B$ given $C$.

Each object is represented by a pair of instances, $(\mathbf{x}, s)$, which are generated from a true distribution, $\Pr[\mathbf{X}, S]$. Given

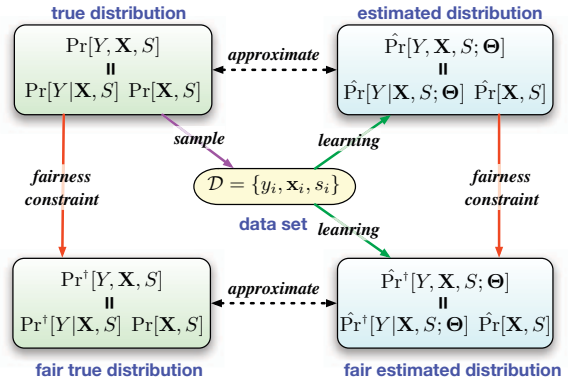


Fig. 1. Notations of distributions

the object, the target instance value, $y$, is generated from a conditional true distribution, $\Pr[Y|\mathbf{x}, s]$. It should be noted that this true distribution, $\Pr[Y|\mathbf{X}, S]$, may make a potentially unfair decision that depends on the sensitive feature, $S$. We cannot know these true distributions themselves, but we can observe data sampled from a joint true distribution, $\Pr[Y, \mathbf{X}, S] = \Pr[Y|\mathbf{X}, S]\Pr[\mathbf{X}, S]$. A *data set*, $\mathcal{D} = \{(y_i, \mathbf{x}_i, s_i)\}$, $i = 1, \ldots, N$, is generated by repeating this process $N$ times. $\mathcal{D}[\mathrm{cond}]$ denotes a set of data that consists of all the data in $\mathcal{D}$ that satisfies the condition cond. Given a parametric family of models, $\hat{\Pr}[Y|\mathbf{X}, S; \mathbf{\Theta}]$, and a training data set, $\mathcal{D}$, the goal of the standard fitting problem is to optimize the parameter, $\mathbf{\Theta}$, so that the resultant distribution would best approximate a true distribution.

We assume the existence of a fair true distribution, $\Pr^\dagger[Y|\mathbf{X}, S]$, that is made fair by imposing a pre-specified fairness constraint on a true distribution, $\Pr[Y|\mathbf{X}, S]$. Unlike a true distribution, we cannot observe even samples generated from this fair true distribution, because actual decisions in the real world may not satisfy the fairness constraint. Therefore, we use a set of training data that are sampled not from a fair true distribution, but from a true distribution. Given a parametric family of fair models, $\hat{\Pr}^\dagger[Y|\mathbf{X}, S; \mathbf{\Theta}]$, that satisfy the same fairness constraint as that of the fair true distribution, the goal of fairness-aware fitting is to optimize parameters so that a fair estimated distribution can best approximate a fair true distribution. The notations of distributions are summarized in Figure 1.

### B. Fairness in Data Mining

Here we review formal definitions of fairness in data mining. A fairness constraint is formally the inequalities that fairness indexes should satisfy. Fairness indexes measure the degree of fairness based on observed or estimated distributions over $(Y, \mathbf{X}, S)$. Many types of fairness indexes have been proposed: extended lift [1], discrimination score [3], mutual information [6], [8], $\chi^2$-statistics [7], [9], $\eta$-neutrality [10], and a combination of statistical parity and the Lipschitz condition [11], [4]. If these fairness indexes are worse than a specified level, the corresponding decisions are considered unfair.

Almost all the fairness indexes are fundamentally related to the statistical independence between a target variable, $Y$,

and a sensitive feature, $S$ [12]. The simple elimination of a sensitive feature from calculations is insufficient for avoiding inappropriate determination processes, due to the indirect influence of sensitive information. Consider the case in which some variable $X$ in a non-sensitive vector is strongly correlated with a sensitive feature. For example, the sensitive feature of race could be correlated with a non-sensitive feature, such as address, if people of a specific race live in a specific area. In this case, the target variable can be indirectly influenced by the sensitive feature. Such a phenomenon is called a *red-lining effect* [3]. Formally, this red-lining effect is produced because $Y$ and $S$ are conditionally independent, $Y \perp\!\!\!\perp S \,|\, \mathbf{X}$, but not unconditionally independent, $Y \not\perp\!\!\!\perp S$.

An example of a red-lining effect in online ad delivery has been reported [9]. When querying by the term of full names to the Web search engines, online ads with negative words are more frequently displayed for first names that are frequent in African descents than those in European descents, though no information about users' race, and that about their first names was not used. Online ads have been unfairly delivered as the result of automatic optimization of the clicking rate based on users' feedbacks.

### C. Applications of Fairness-aware Data Mining

We here show three applications of FADM. The first application is discrimination-aware data mining whose purpose is to eliminate socially unfair treatment [1], as shown in the introduction. Data mining techniques are increasingly being used for serious determinations such as credit, insurance rates, employment applications, and so on, which are represented by $Y$. Needless to say, such serious determinations must guarantee fairness from both the social and legal viewpoints; that is, they must be fair and nondiscriminatory in relation to sensitive features such as gender, religion, race, ethnicity, handicaps, political convictions, and so on, which are represented by $S$. Discovery of discriminative treatments [13], [14], [15], [16], [1], [17], [18], [9], learning non-discriminative estimators [3], [13], [7], and transforming data for non-discriminative analysis [11], [4] have been discussed.

The second type of application is an information-neutral recommender system [5], [19]. Recommendations are made while maintaining neutrality regarding particular viewpoints specified by users. For such a purpose, FADM techniques can be used by regarding recommendation results and viewpoints as $Y$ and $S$, respectively.

The third application is excluding useless information from analysis results. For this purpose, non-redundant clustering was proposed [6], [20]. This method was used for clustering facial images. Simple clustering methods found two clusters: one contained only faces, and the other contained faces with shoulders. Here, a data analyst is assumed to consider that this clustering is useless for his/her purpose of the analysis. If this clustering result is treated as a sensitive feature, the non-redundant clustering technique can find more useful clusters that are composed of male and female images by ignoring the information of the useless clustering.

### D. Formal Tasks of Fairness-aware Data Mining

Formal tasks of fairness-aware data mining can be currently classified into two groups: unfairness discovery and unfairness prevention [2].

*a) Unfairness Discovery:* Given a data set $\mathcal{D} = \{y_i, \mathbf{x}_i, s_i\}^N$, the goal of an unfairness discovery task is to find or enumerate patterns that violate a pre-specified fairness constraint. Numerous methods for unfairness detection tasks have been developed in the literature [13], [14], [15], [16], [1], [17], [18], [9].

*b) Unfairness Prevention:* The second unfairness prevention task aims to learn a statistical model for prediction or decision-making from potentially unfair data sets so that the sensitive feature does not influence the prediction or decision. Two different approaches, fairness-aware data publication and fairness-aware learning, are described below

In the first approaches, fairness-aware data publication, a given data set without target values $\mathcal{D} = \{\mathbf{x}_i, s_i\}^N$ is transformed into a data set, $\mathcal{D}'$. This transformation is designed so that the potential unfairness in the data set is removed. The transformed data set is then published, and recipients of the set process them by standard analysis methods. A few examples of fairness-aware data publication tasks have been reported [11], [4]. This approach has an advantage in that many standard analysis methods can be used, but more information can be lost than in the case of fairness-aware learning by transformation.

In the second approach, fairness-aware learning, potentially unfair data are directly processed by methods designed so as to enhance the independence between the target variable and the sensitive feature. In the case of supervised learning, the goal of fairness-aware classification or regression is to find a fair estimated distribution that approximates a fair true distribution, given a training data set $\mathcal{D} = \{y_i, \mathbf{x}_i, s_i\}^N$. There are several examples of fairness-aware classification [3], [13], [8], [7], fairness-aware regression [10], and an information-neutral recommendation [5], [19]. In the case of unsupervised learning, given a data set without target values $\mathcal{D} = \{\mathbf{x}_i, s_i\}^N$, the goal of fairness-aware clustering is to predict latent labels as well as to satisfy a pre-specified fairness constraint. Fairness-aware clustering has been described in the literature [6], [20].

### E. Fairness-aware Classification

This paper focuses on a fairness-aware classification task. The target variable is discrete and represents a class in fairness-ware classification. In this paper, we further restricts the types of variables, $Y$, $S$, and $\mathbf{X}$. A target variable $Y$ represents a binary class whose domain is $\{0, 1\}$. The classes, 0 and 1, are represent unfavorable and favorable outcomes, such as denial and approval of a loan request, respectively. $S$ is also restricted to a binary variable whose domain is $\{0, 1\}$. Objects whose sensitive values are 1 and 0 are said to be *non-protected* and *protected* states, respectively. Protected objects represent individuals or entities that should be protected from socially unfair treatments, typically minorities. The groups of all individuals who are in a protected state constitute a protected group, and the rest of the individuals comprise an unprotected group. $\mathbf{X}$ is composed of $K$ random variables, $X^{(1)}, \ldots, X^{(K)}$, each of which can be discrete or continuous.

Figure 2 geometrically represents a task of fairness-aware classification. The entire of the figure corresponds to a family of all distributions over $(Y, \mathbf{X}, S)$. A vertical plane depicts

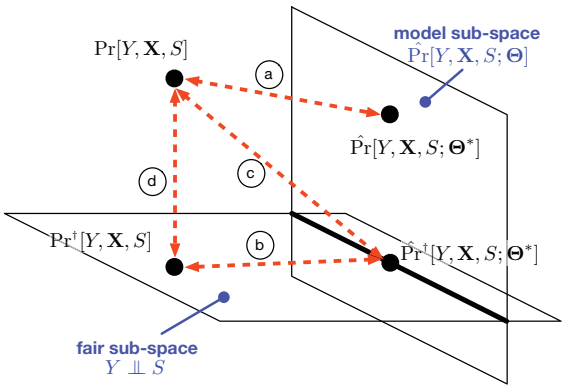


Fig. 2. A geometrical interpretation of fairness-aware classification

a model sub-space of distributions that are represented by a parametric model, $\hat{\Pr}[Y, \mathbf{X}, S; \boldsymbol{\Theta}]$. In the case of a standard classification task, the goal of the task is to find the best parameter, $\boldsymbol{\Theta}^*$, such that the resulting distribution, $\hat{\Pr}[Y, \mathbf{X}, S; \boldsymbol{\Theta}^*]$, best approximates a true distribution, $\Pr[Y, \mathbf{X}, S]$. As in the figure, a true distribution may not be in a model sub-space, while a parametric model of distributions must be in the sub-space. Therefore, the best estimated distribution is chosen so as to minimize the divergence between $\Pr[Y, \mathbf{X}, S]$ and $\hat{\Pr}[Y, \mathbf{X}, S; \boldsymbol{\Theta}^*]$ (ⓐ in the figure.) As is well known, when adopting a maximum likelihood estimator, the divergence is measured by a Kullback-Leibler divergence.

We turn to a case of fairness-aware classification. The goal of a fairness-aware classification task is to find a fair estimated model, $\hat{\Pr}^\dagger[Y, \mathbf{X}, S; \boldsymbol{\Theta}^*]$, that best approximates a fair true distribution, $\Pr^\dagger[Y, \mathbf{X}, S]$. A horizontal plane depicts a fair sub-space of distributions that satisfies a pre-specified fairness constraint. A fair true distribution, $\Pr^\dagger[Y, \mathbf{X}, S]$, must be in this fair sub-space. A parametric model of fair estimated distributions, $\hat{\Pr}^\dagger[Y, \mathbf{X}, S; \boldsymbol{\Theta}]$, must be in the product sub-space of fair and model sub-spaces, depicted by a thick line in the figure. Our goal is to find the best parameter so as to minimize the divergence between a fair true distribution and a fair estimated distribution (ⓑ in the figure.) Unfortunately, we cannot sample from a fair true distribution due to the potential unfairness of actual decisions in real world. We therefore tried to minimize the divergence between a true distribution and a fair estimated distribution (ⓒ in the figure.) We here assume that minimizing the divergence ⓒ is equivalent to minimizing the divergence ⓑ when the divergence between a true distribution and a fair true distribution (ⓓ in the figure) is constant. Whether this assumption is the case depends on the divergence; what kind of divergence should be theoretically adopted is currently an open problem.

*1) Calders & Verwer's two-naive Bayes:* We introduce Calders and Verwer's two-naive-Bayes method (CV2NB for short) [3], whose theoretical backgrounds we will discuss. The generative model of this method is

$$\hat{\Pr}[Y, \mathbf{X}, S] = \hat{\Pr}[Y|S]\,\hat{\Pr}[S]\prod_k \hat{\Pr}[X^{(k)}|Y, S]. \quad (1)$$

In a standard naive Bayes mode, each $X^{(k)}$ only depends on $Y$; in this CV2NB model, it also depends on $S$. Note that

1 Calculate a CV score, $disc$, of the predicted classes by the current model.
2 `while` $disc > 0$
3 $numpos$ is the number of positive samples classified by the current model.
4 `if` $numpos <$ the number of positive samples in $\mathcal{D}$ `then`
5 $N(Y{=}1, S{=}0) \leftarrow N(Y{=}1, S{=}0) + \Delta N(Y{=}0, S{=}1)$
6 $N(Y{=}0, S{=}0) \leftarrow N(Y{=}0, S{=}0) - \Delta N(Y{=}0, S{=}1)$
7 `else`
8 $N(Y{=}0, S{=}1) \leftarrow N(Y{=}0, S{=}1) + \Delta N(Y{=}1, S{=}0)$
9 $N(Y{=}1, S{=}1) \leftarrow N(Y{=}1, S{=}1) - \Delta N(Y{=}1, S{=}0)$
10 `if` any of $N(Y, S)$ is negative `then`
cancel the previous update of $N(Y, S)$ and abort
11 Recalculate $\Pr[Y|S]$ and a CV score, $disc$ based on updated $N(Y, S)$

Fig. 3. A modification algorithm for a two-naive-Bayes model

this method was named "two-naive-Bayes" because it is as if two naive Bayes classifiers are learned depending on each sensitive value. Because $Y$ and $S$ are not mutually independent in this model, the final determination may be unfair. A joint distribution $\hat{\Pr}[Y, S] = \hat{\Pr}[Y|S]\,\hat{\Pr}[S]$ is therefore modified so as to satisfy the fairness constraint.

This CV2NB method adopts Calders-Verwer's discrimination score (*CV score* for short) as a fairness index. This CV score is defined by subtracting the probability that protected individuals get favorable treatment from the probability that unprotected individuals do:

$$\hat{\Pr}[Y{=}1|S{=}1] - \hat{\Pr}[Y{=}1|S{=}0]. \quad (2)$$

As this score increases, the members of unprotected group get favorable treatment more frequently while those in the protected group get favorable treatment less frequently. It is easy to show that when both $Y$ and $S$ are binary, the zero CV score implies that $Y$ and $S$ are statistically independent, $Y \perp\!\!\!\perp S$ [12].

To satisfy the fairness constraint, this CV score should be close to zero. For this purpose, $\hat{\Pr}[Y, S]$ is modified by the algorithms in Figure 3. After stopping this algorithm, a model parameter $\hat{\Pr}[y, s]$ can be induced from $N(y, s)$, which is the number of instances that $Y = y$ and $S = s$. This algorithm updates $N[Y, S]$ so that the resultant CV score approaches zero. Note that we slightly modified the original algorithm by adding line 10 in Figure 3, which guarantees $N(Y, S)$ to be non-negative, because the original algorithm may fail to stop.

*2) Reject Option Based Classification:* Kamiran et al. discussed a theory to determine class labels based on a class posterior distribution so that a fairness constraint is satisfied [7], [21]. In standard classification, objects are classified to the calass 1 if the class posteriors satisfy the inequality $\hat{\Pr}[Y{=}1|\mathbf{X}] \geq \hat{\Pr}[Y{=}0|\mathbf{X}]$, which is equivalent to $\hat{\Pr}[Y{=}1|\mathbf{X}] \geq 0.5$. The threshold $0.5$ is referred as a decision boundary.

The authors proposed a method, which they call *Reject Option based Classification* (ROC), to change the decision boundary to make fair classification. For members of a protected group, the boundary is decreased so that they will get favorable treatments more frequently. Conversely, the boundary is increased for members of a non-protected group. Their idea is to change class labels so as to satisfy a fairness constraint as well as not to damage prediction accuracy so much. This is achieved by changing labels of objects that lie in the neighbor of a decision boundary because the confidences of classification are considered relatively low in this region.

Formally, we introduce a threshold parameter, $0.5 \leq t < 1$. Objects such that $S{=}0$ are classified to the class 1 if $\hat{\Pr}[Y{=}1|\mathbf{X}, S{=}0] \geq 1-t$. Inversely, objects such that $S{=}1$ are classified to the class 1 if $\hat{\Pr}[Y{=}1|\mathbf{X}, S{=}1] \geq t$.

The authors pointed out the connection of this decision rule with a theory of *cost-sensitive learning* [22]. The goal of cost-sensitive learning is classifying objects so that their misclassification costs are minimized. When classifying an object, a misclassification cost is a penalty that is added when an estimated class of the object is different from its true class. The following relations can be derived according to the equation (2) in the literature [22]. In the case of standard classification, a misclassification cost that objects whose true class is 1 (respectively 0) is classified to the class 0 (respectively 1) is 1. We turn to the ROC rule. For protected objects such that $S{=}0$, costs of misclassifying objects whose true class are 0 are kept to be 1, but those of misclassifying objects whose true class are 1 are increased to $t/(1-t)$. This means that if the protected individuals that should be favorably treated are treated unfavorably, misclassification costs are increased and are more heavily penalized, but costs for those who should be unfavorably treated are unchanged. Non-protected individuals are treated inversely. Costs of misclassifying objects whose true class are 0 and 1 are increased to $t/(1-t)$ and are unchanged, respectively. That is to say, non-protected individuals are more heavily penalized if those who should be unfavorably treated are favorably treated.

*3) Other methods for fairness-aware classification:* We here briefly review the other methods for fairness-aware classification. Kamiran et al. developed algorithms for learning decision trees for a fairness-aware classification task [13]. When choosing features to divide training examples at non-leaf nodes of decision trees, their algorithms evaluate the information gain regarding sensitive information as well as that about the target variable. Additionally, the labels at leaf nodes are changed so as to decrease the above CV score.

Kamishima et al. proposed a prejudice remover regularizer, which is mutual information between a sensitive feature and a class variable, $\mathrm{I}(Y;S)$, [8]. The regularization term imposes a fairness constraint, and is applied to the logistic regression model. Fukuchi et al. introduced another constraint, *$\eta$-neutrality*:

$$\frac{\Pr[Y=y, S=s]}{\Pr[Y=y]\Pr[S=s]} \leq 1+\eta, y \in \mathrm{Dom}(Y), s \in \mathrm{Dom}(S),$$

where $\eta$ is a hyper parameter, which balances the fairness and accuracy [10]. They further introduced a model to predict a sensitive feature from non-sensitive features, and examined regression tasks.

Zemel et al. proposed *learning fair representation*, which is a framework for classification adopting an approach of fairness-aware data publication [4]. They tried to obtain an intermediate representation that fulfills three constraints. The first constraint is to satisfy statistical parity, which is the unbiased status in terms of a sensitive feature. The second constraint is minimizing the distortion between the original data and the intermediate data, and the third is maximizing the accuracy of class prediction.

### F. Connections with Privacy-preserving Data Mining

FADM is closely related to privacy-preserving data mining [23], which is a technology for mining useful information without exposing individuals' private records. The privacy protection level is quantified by mutual information between the public and private realms [23, chapter 4]. Almost all the fairness indexes concern the dependency between $Y$ and $S$, and the dependence can be evaluated by the mutual information. Due to the similarity of these two uses of mutual information, the design goal of fairness-aware data mining can be considered the protection of sensitive information when exposing target information. Further, FADM has connection with a notion of $t$-closeness [24] in terms of the prevention of disclosing specific information when disclosing distinct information. Other concepts for privacy preservation can be exploited for the purpose of maintaining fairness. Relations between concepts of differential privacy [25] and differential fairness are discussed in [11]. Differential privacy is considered a special case of differential fairness whose loss function represents the distortion of query results. The effect of applying anonymization techniques to the influence between a target variable and a sensitive variable has been previously investigated [16].

On the other hand, fairness and privacy preservation are different in some points. In the case of fairness, the exposure of identity is occasionally not problematic, because the identity is already exposed in a credit or employment application case. The use of a random transformation is accepted for privacy-preservation, but it is occasionally problematic in the case of FADM. For example, if employment or admissions are determined randomly, it becomes difficult to explain the reason for rejection to applicants.

## III. Hypothetical Fair-Factorization

We reviewed methods for fairness-aware classification. Among these methods, the CV2NB method outperformed the others with respect to the degree of fairness. For example, the latent variable model that was proposed in the same article of CV2NB was clearly inferior, and the logistic regression with a prejudice remover [8] failed to attain a higher level of fairness. However, the reason for the superiority of the CV2NB method was unclear, because its fairness was enhanced by a somewhat heuristic post-processing technique, which obscures what statistical model was in fact learned.

To reveal the cause, we introduce a generative model that is similar to the CV2NB model. This model is built by applying hypothetical fair-factorization, which is a technique to impose a fairness constraint onto a generative classification model. After discussing its connections to CV2NB and ROC methods, we show experimental results on benchmark data sets to confirm that this model is inferior to the CV2NB model.

### A. A Hypothetical Fair-Factorization Technique

We start with our hypothetical fair-factorization. A standard generative classification model represents a joint distribution of a target variable and features, $\hat{\Pr}[Y, \mathbf{X}, S]$. An object is classified into a class whose posterior probability given the feature values, $\hat{\Pr}[Y|\mathbf{X}, S]$, is maximized. As is well known, the class that maximizes this posterior coincides with the class

that maximizes a joint distribution of a class and features because the posterior probability is proportional to the joint probability:

$$\hat{\Pr}[Y|\mathbf{X},S] = \frac{\hat{\Pr}[\mathbf{X},S|Y]\,\hat{\Pr}[Y]}{\hat{\Pr}[\mathbf{X},S]} \propto \hat{\Pr}[\mathbf{X},S|Y]\,\hat{\Pr}[Y]. \quad (3)$$

Consequently, all we have to do is to estimate this joint distribution.

We next focus on the statistical independence between a target variable, $Y$, and a sensitive feature, $S$, because many of the proposed fairness indexes measure the degree of this independence as described in section II-B. Formally, a fair estimated distribution satisfies this condition, $Y \perp\!\!\!\perp S$, which is equivalent to $\hat{\Pr}^\dagger[Y,S] = \hat{\Pr}^\dagger[Y]\,\hat{\Pr}^\dagger[S]$. We embed this condition in a generative classification model:

$$\begin{aligned}\hat{\Pr}^\dagger[Y,\mathbf{X},S] &= \hat{\Pr}^\dagger[Y,S]\,\hat{\Pr}^\dagger[\mathbf{X}|Y,S] \\ &= \hat{\Pr}^\dagger[Y]\,\hat{\Pr}^\dagger[S]\,\hat{\Pr}^\dagger[\mathbf{X}|Y,S]. \end{aligned} \quad (4)$$

Note that the statistical independence condition is imposed in the second line. We call this technique of decoupling $Y$ and $S$ in a generative model so as to make them mutually independent by *fair-factorization*. In particular, because this version of fair-factorization is applied to a distribution in the hypothesis space, we call it *hypothetical fair-factorization* to differentiate it from the actual version described in section V.

We then applied this hypothetical fair-factorization to a naive Bayes model. We abbreviate this model by a HFFNB model. As in the CV2NB model in equation (1), the HFFNB model assumes that non-sensitive features, $X^{(k)}, k = 1,\ldots,K$, are conditionally independent given $Y$ and $S$. We assume The HFFNB model further assumes the independence between $Y$ and $S$; namely, fair-factorization is applied to this. Consequently, the HFFNB model becomes

$$\hat{\Pr}^\dagger[Y,\mathbf{X},S] = \hat{\Pr}^\dagger[Y]\,\hat{\Pr}^\dagger[S]\prod_k \hat{\Pr}^\dagger[X^{(k)}|Y,S]. \quad (5)$$

It is very easy to derive the maximum likelihood estimators of this model from a training data set $\mathcal{D}$ if both $Y$ and $S$ are binary as in this paper. $\hat{\Pr}^\dagger[Y]$, $\hat{\Pr}^\dagger[S]$, and $\hat{\Pr}^\dagger[X^{(k)}|Y,S], k = 1,\ldots,K$ can be fitted separately. $\hat{\Pr}^\dagger[Y=1]$ can be estimated by $|\mathcal{D}[Y=1]|/|\mathcal{D}|$. $\mathcal{D}[S=1]$ is a set of all data in $\mathcal{D}$ such that $S=1$ as defined in section II-A. Likewise, $\hat{\Pr}^\dagger[S=1]$ can be estimated by $|\mathcal{D}[Y=1]|/|\mathcal{D}|$, and $\Pr^\dagger[X^{(k)}|Y{=}y,S{=}s]$ can be computed from a data set $\mathcal{D}[Y{=}y,S{=}s]$. Note that we adopt a Laplace smoothing technique to avoid the zero-counting problem in the later experiments.

### B. Connection with Other Fairness-aware Classification Techniques

Here, we discuss the connection of our HFFNB model with the CV2NB model and ROC rule. We begin with the CV2NB model. The two models, equation (1) and (5), are the same except for the independence between $Y$ and $S$ in the HFFNB model. No such independence is imposed in the CV2NB model, but a joint distribution $\hat{\Pr}[Y,S]$ is modified so that $Y$ and $S$ are independent by the algorithm in Figure 3. Let us look at this algorithm more closely. Lines 5-6 and 8-9 in the algorithm are designed so that the CV score of the resulting distribution approaches zero. Specifically, the number of protected individuals that are favorably treated is increased in line 5 and the number of those who are unfavorably treated is decreased in line 6. Lines 8-9 similarly adjust the numbers of non-protected individuals. The main loop of this algorithm exits at line 2 if the resultant CV score is close to zero. Therefore, the resulting distribution $\hat{\Pr}[Y,S]$ satisfies the independence condition between $Y$ and $S$ as in our HFFNB model.

However, the marginal distributions of $Y$ differ between the two models. It is easy to show that a marginal distribution of $Y$ equals $\hat{\Pr}^\dagger[Y]$, which is the first factor of equation (5), by integrating out the HFFNB model over $S$ and $\mathbf{X}$. Therefore, the marginal distribution of $Y$ is equal to the sample distribution of $Y$ over a training data $\mathcal{D}$. The modified algorithm of the CV2NB method is designed so that the resultant marginal distribution of $Y$ does not diverge so much from the corresponding sample distribution by the adjustment in line 3. However, because the marginal distribution of $Y$ is not considered in the stopping criterion in line 2, the resultant distribution of $Y$ does not generally equal the sample distribution of $Y$.

We now turn to Kamiran et al.'s ROC decision rule. We first describe Elkan's theorem 2 in the literature [22]. Given a Bayesian classifier whose prior is $b'$ and whose decision boundary is $p'$, when this prior is changed to $b$, how should we choose a new decision boundary, $p$, so as to make these two classifiers indicate the same decision? Elkan's theorem describes the relation:

$$p' = \frac{b'p(1-b)}{b - pb + b'p - bb'}. \quad (6)$$

In the case of our HFFNB model, an original prior $b' = \hat{\Pr}[Y|S]$ is changed to $b = \hat{\Pr}[Y]$ by applying fair-factorization. If the decision boundary of our HFFNB model is $p = 1/2$, the decision boundary for an original classifier that leads an equivalent classifier is

$$\begin{aligned} p' &= \frac{b'(1-b)}{b + b' - 2bb'} \\ &= \frac{\hat{\Pr}[Y|S](1-\hat{\Pr}[Y])}{\hat{\Pr}[Y] + \hat{\Pr}[Y|S] - 2\,\hat{\Pr}[Y]\,\hat{\Pr}[Y|S]}. \end{aligned} \quad (7)$$

Consequently, our HFFNB model is equivalent to changing decision boundaries in an original classifier. In this sense, the HFFNB method can be considered as a kind of ROC approach.

### C. Experiments

We here compared the performance of our HFFNB method and the CV2NB method on two benchmark data sets. The HFFNB method performed poorly in comparison with the CV2NB model. We analyze the reason for this poor performance in the next section.

*1) Data Sets:* We tested our HFFNB method and a CV2NB method on two benchmark data sets[1] used in [14]. The first is an adult data set (a.k.a. the census income data set) originally distributed at the UCI repository [26]. We refer to this data set as Adult. The target variable represents whether an individual's

[1] distributed at https://sites.google.com/site/conditionaldiscrimination/

income is high or low, and the sensitive feature represents the individual's gender. The number of data is 15,696, and the number of non-sensitive features is 12. All features are discrete.

The second set is the Dutch census data set, which we refer to as Dutch. The target variable represents whether an individual's profession is high income or low income, and the sensitive feature represents the individual's gender. The number of data is 60,420, and the number of non-sensitive features is 10. All features are discrete.

*2) Evaluation Indexes:* We performed five-fold cross-validation, and calculated the following evaluation indexes. Given a test data $(y, \mathbf{x}, s)$, a label for the data, denoted by $\hat{y}$, is inferred by the learned classifier. This process is repeated for all data in the set, and we obtained new data $\mathcal{T} = \{\hat{Y} = \hat{y}_i, Y = y_i, S = s_i\}_i^M$. A set of all the data in $\mathcal{T}$ that satisfy the condition cond is denoted by $\mathcal{T}[\mathrm{cond}]$. A sample distribution over $\mathcal{T}$ is denoted by $\tilde{\Pr}[\cdot]$. For example, $\tilde{\Pr}[Y, S] \sim |\mathcal{T}[Y = y, S = s]|/M$, $y \in \{0, 1\}$, $s \in \{0, 1\}$.

To evaluate the performance of fairness-aware classifiers, we have to examine how strictly a fairness constraint is satisfied as well as how accurately class labels are predicted. This is because there is a trade-off between accuracy and fairness. We used an accuracy measure to evaluate the prediction accuracy:

$$\mathsf{Acc} = \frac{|\mathcal{T}[Y = \hat{Y}]|}{M}. \tag{8}$$

The larger this accuracy is, the more accurately classes are predicted.

We use two indexes for the evaluation of fairness. The first index is a CV score in equation (2), but it was computed using sample distributions over $\mathcal{T}$:

$$\mathsf{CVS} = \tilde{\Pr}[Y{=}1|S{=}1] - \tilde{\Pr}[Y{=}1|S{=}0]. \tag{9}$$

If this CV score is zero, the target variable is perfectly independent from the sensitive feature.

The second index is a normalized prejudice index [8]. A prejudice index is defined as the mutual information between $\hat{Y}$ and $S$, $\mathrm{I}(\hat{Y}; S)$, which is computed using sample distributions over $\mathcal{T}$. Further, a normalized prejudice index is obtained by normalizing into the range $[0, 1]$:

$$\mathsf{NPI} = \frac{\mathrm{I}(\hat{Y}; S)}{\sqrt{\mathrm{H}(\hat{Y})\mathrm{H}(S)}}, \tag{10}$$

where $\mathrm{H}(\cdot)$ is an entropy function that is computed using sample distributions. Note that we used a natural logarithm for computing mutual information and entropy. The smaller this normalized prejudice index is, the fairer the decisions are.

*3) Experimental Results:* We compared four classification methods. Two of them were fairness-aware classifiers, HFFNB and CV2NB, and the other two were baseline methods, which were standard naive Bayes classifiers. The first baseline was a naive Bayes classifier that used both non-sensitive and sensitive features, denoted as NB. The second baseline was a naive Bayes classifier that used only non-sensitive features, denoted as NBns. We applied these four methods (HFFNB, CV2NB, NB, and NBns) to two benchmark data sets (Adult and Dutch) in section III-C1 and calculated three indexes (Acc, CVS, and NPI) in section III-C2. A $\Delta$ parameter of the CV2NB method was set to $0.01$ as in the original paper.

TABLE I. COMPARISON OF OUR HFFNB METHOD WITH THE CV2NB METHOD AND TWO BASELINES

(a) Adult data

| Methods | Acc | CVS | NPI |
|---|---|---|---|
| HFFNB | 0.828 | 0.129 | $1.52 \times 10^{-2}$ |
| CV2NB | 0.828 | $-0.003$ | $6.89 \times 10^{-6}$ |
| NB | 0.829 | 0.345 | $1.16 \times 10^{-1}$ |
| NBns | 0.836 | 0.278 | $7.62 \times 10^{-2}$ |

(b) Dutch data

| Methods | Acc | CVS | NPI |
|---|---|---|---|
| HFFNB | 0.810 | 0.312 | $7.17 \times 10^{-2}$ |
| CV2NB | 0.761 | $-0.003$ | $8.79 \times 10^{-6}$ |
| NB | 0.816 | 0.365 | $9.86 \times 10^{-2}$ |
| NBns | 0.789 | 0.162 | $1.90 \times 10^{-2}$ |

Experimental results are shown in Table I. We first focus on the two baseline methods, NB and NBns. Larger CVS values were observed for the NB method than for the NBns in both data sets, and similar results were observed in terms of the normalized prejudice indexes (NPI). This means that a fairer prediction was achieved by excluding the sensitive feature from models. However, both CVS and NPI were much larger than zero; this indicates that fully fair models could not be learned simply by eliminating a sensitive feature due to the red-lining effects.

We next compared our HFFNB model with two baseline methods. In the Adult case, the prediction accuracy for HFFNB was worse than those of the baseline methods. However, in the Dutch case, Acc for HFFNB was worse than that for NB, but was better than that for NBns. According to the two fairness measures, the HFFNB method successfully learned fairer models than the two baselines in the Adult case, but failed in the Dutch case. Generally speaking, fairness-aware models are designed so as to improve their fairness in exchange for a decrease in prediction accuracy. Unfortunately, our HFFNB method failed to obtain fully fair models especially in the Dutch case.

Finally, our HFFNB model was compared with the state-of-art CV2NB method. In terms of prediction accuracy, our HFFNB method was slightly better than the CV2NB method. However, the CV2NB learned almost perfectly fair models because both CVS and NPI measures were nearly zero. Our HFFNB method failed to obtain fully fair models in this comparison.

## IV. WHY DID THE HFFNB METHOD FAIL?

As observed in the experimental results of the previous section, our HFFNB model failed to learn fair models, though the model explicitly imposed the constraint of the independence between $Y$ and $S$. We hypothesized two reasons for this failure. The first reason is a model bias, which makes an estimated distribution different from a true distribution. This difference damages the fairness of the learned classifier. The second reason is the deterministic Bayes decision rule. Though class labels are in fact chosen according to a deterministic decision rule, our simple version assumes that labels are probabilistically decided.

TABLE II. NPI ON SYNTHETIC DATA TO CHECK THE INFLUENCE OF MODEL BIAS

| | HFFNB | CV2NB |
|---|---|---|
| $K$=5 | $1.02 \times 10^{-1}$ | $5.68 \times 10^{-4}$ |
| $K$=10 | $1.10 \times 10^{-1}$ | $9.60 \times 10^{-2}$ |
| $K$=50 | $1.28 \times 10^{-1}$ | $1.28 \times 10^{-1}$ |

### A. Model Bias

We first show how model bias damages the fairness. In classification with generative models, class labels are predicted based on an *estimated* distribution, $\hat{\Pr}[Y|\mathbf{X}, S]$. On the other hand, the objects to be classified are generated according to a *true* distribution, $\Pr[\mathbf{X}, S]$. As shown in Figure 2, the estimated distribution is generally different from a true distribution because an estimated distribution must lie in the model sub-space, but this limit does not apply for a true distribution. For example, in the HFFNB model, non-sensitive features $X^{(k)}, k = 1, \ldots, K$ are assumed to be conditionally independent, but this assumption is not generally the case for a true distribution. As a consequence, the joint distribution over $(Y, \mathbf{X}, S)$ diverges from a hypothetical fair-factorization generative model:

$$\hat{\Pr}[Y|\mathbf{X}, S] \Pr[\mathbf{X}, S] \neq \hat{\Pr}[Y] \hat{\Pr}[S] \hat{\Pr}[\mathbf{X}|Y, S]. \quad (11)$$

Therefore, when a joint distribution $\hat{\Pr}[Y, S]$ is obtained by integrating out $\mathbf{X}$ from a joint distribution $\hat{\Pr}[Y|\mathbf{X}, S] \Pr[\mathbf{X}, S]$, the resultant joint distribution $\hat{\Pr}[Y, S]$ fails to satisfy the fairness condition, $Y \perp\!\!\!\perp S$.

To check this hypothesis, we tested it on synthetic data. Synthetic data were generated from the following model, which is a variant of the naive Bayes model; thus, a true distribution would lie close to the model sub-space. Class labels and sensitive values were first generated from the following joint distribution:

| | $Y$=0 | $Y$=1 |
|---|---|---|
| $S$=0 | 0.3 | 0.1 |
| $S$=1 | 0.2 | 0.4 |

This distribution was designed so that protected objects had minority status, i.e., $\Pr[S{=}0] < \Pr[S{=}1]$, and protected objects were unfavorably treated, i.e., $\Pr[Y{=}1|S{=}0] < \Pr[Y{=}0|S{=}0]$, but non-protected objects were favorably treated, i.e., $\Pr[Y{=}1|S{=}1] > \Pr[Y{=}0|S{=}1]$. For each pair of $S = s$ and $Y = y$, $K$ binary non-sensitive features were independently generated according to binomial distributions whose parameters were chosen according to $\mathrm{Dirichlet}(\{0.7, 0.7\})$ a priori. Note that this model is a generative model of HFFNB without fair-factorization. We generated $10,000$ training data and $10,000$ test data while changing $K \in \{5, 10, 50\}$, and applied the HFFNB and CV2NB methods.

Experimental results are shown in Table II. We showed normalized prejudice indexes (NPIs) to check the influence of model bias to the fairness of the learned models. The NPI for the HFFNB and CV2NB methods differed considerably in the $K = 5$ case. As $K$ increased, the difference between NPIs for the two methods diminished. This would be because the model bias is reduced with the increase of $K$, and the true distribution tends to lie closer to the estimated distribution. As a result, the model bias would damage the fairness less in the case of HFFNB than in the case of CV2NB.

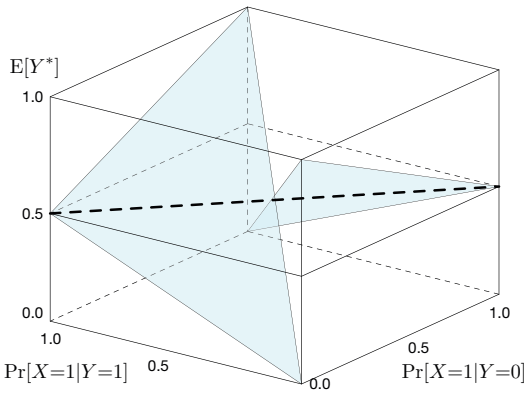


Fig. 4. The changes of the expectation of actual labels, $\mathrm{E}[Y^*]$

### B. Deterministic Decision Rules

We next discuss the influence of deterministic decision rules in determining class labels. The independence condition $Y \perp\!\!\!\perp S$ is satisfied if the distribution of actual class labels equals that induced from a generative model. However, this is not the case because actual labels, $y^*$, are deterministically chosen by the Bayes decision rule:

$$y^* = \arg\max_y \hat{\Pr}[Y = y|\mathbf{X} = x, S = s]. \quad (12)$$

We next examine how different the distribution of actual labels determined by a Bayes decision rule is from that induced from a generative model. For this purpose, we consider a very simple model with a binary class variable, $Y$, and one binary feature variable, $X$. The class prior is uniform, i.e., $\hat{\Pr}[Y{=}1] = 0.5$. Two other parameters, $\hat{\Pr}[X{=}1|Y{=}0]$ and $\hat{\Pr}[X{=}1|Y{=}1]$, are required for representing the joint distribution of $X$ and $Y$. In this case, $\mathrm{E}[Y]$ becomes a constant, $0.5$, if $Y$ follows the distribution induced from this model. We then consider the variable $Y^*$ to represent actual labels determined by equation (12). In Figure 4, we depict the variation of the expectation $\mathrm{E}[Y^*]$ according to the changes of $\hat{\Pr}[X{=}1|Y{=}0]$ and $\hat{\Pr}[X{=}1|Y{=}1]$. Surprisingly, the condition $\mathrm{E}[Y] = \mathrm{E}[Y^*]$ is satisfied only if $\hat{\Pr}[X{=}1|Y{=}0] + \hat{\Pr}[X{=}1|Y{=}1] = 1$ (depicted by the thick broken line in Figure 4). As a result, the two variables $Y$ and $Y^*$ behave differently at almost every point, and this difference damages the fairness.

## V. ACTUAL FAIR-FACTORIZATION

Above, we discussed two reasons why our hypothetical fair-factorization naive Bayes (HFFNB) performed poorly in the previous section. Here, we propose a modified version of the fair-factorization technique to eliminate the defects of the HFFNB model. This modified version is referred as the *actual fair-factorization naive Bayes* method (AFFNB for short).

### A. How to Modify a Hypothetical Fair-Factorization Model

Based on our findings in section IV, we modify the HFFNB model so as to fair-factorize the actual distribution. We call this

modified technique the *actual fair-factorization*. Hypothetical fair-factorization decouples a class variable and a sensitive feature of a distribution in the hypothesis space, $\hat{\Pr}[Y, \mathbf{X}, S]$. However, the actual distribution is deviated from this hypothetical distribution due to the two reasons as described in section IV. The first deviation is due to the model bias, and the second deviation is caused by the application of the deterministic decision rules. To fix the first deviation, we use the true distribution of inputs, $\Pr[\mathbf{X}, S]$, instead of the estimated distribution, $\hat{\Pr}^\dagger[\mathbf{X}, S]$. To fix the second deviation, we consider a distribution with the actual class label, $\hat{\Pr}^\dagger[Y^*, \mathbf{X}, S]$, instead of a distribution of a target variable, $\hat{\Pr}^\dagger[Y, \mathbf{X}, S]$. Hnce, the actual distribution generated by a prediction model becomes

$$\hat{\Pr}^\dagger[Y^*|\mathbf{X}, S] \Pr[\mathbf{X}, S]. \tag{13}$$

Let us recall the CV2NB model before applying post-processing:

$$\hat{\Pr}^\dagger[Y, \mathbf{X}, S] = \hat{\Pr}^\dagger[Y|S] \hat{\Pr}^\dagger[S] \hat{\Pr}^\dagger[\mathbf{X}|Y, S]. \tag{14}$$

To replace $Y$ with $Y^*$, we introduce parameters $q_s, s \in \{0, 1\}$ to control the ratios of actually generated class labels after the application of the deterministic decision rule. These parameters are used instead of parameters, $\hat{\Pr}^\dagger[Y{=}1|S{=}s], s \in \{0, 1\}$, but parameters $q_s$ cannot interpreted as probabilities any longer. Consequently, the ratio of data that actually labeled 1 becomes

$$\hat{\Pr}^\dagger[Y^*{=}1, \mathbf{X}, S{=}\,s]{=}q_s \hat{\Pr}^\dagger[S{=}s] \hat{\Pr}^\dagger[\mathbf{X}|Y^*{=}1, S{=}s]. \tag{15}$$

Using this model (15), we compute $\hat{\Pr}^\dagger[Y^*{=}1|S{=}s], s \in \{0, 1\}$, because these distributions are key for fairness in classification. They can be obtained by marginalizing equation (13) over $\mathbf{X}$ and dividing by $\Pr[S]$:

$$\hat{\Pr}^\dagger[Y^*{=}1|S{=}s]{=}\sum_{\mathbf{X}} \hat{\Pr}^\dagger[Y^*{=}1|\mathbf{X}, S{=}s] \Pr[\mathbf{X}|S{=}s]. \tag{16}$$

The marginalization over $\mathbf{X}$ together with the true distribution $\Pr[\mathbf{X}|S{=}s]$ can be approximated by the sample mean over the data set $\mathcal{D}[S{=}s]$. Equation (16) can be approximated by

$$\frac{1}{|\mathcal{D}[S{=}s]|} \sum_{(\mathbf{x}) \in \mathcal{D}[S=s]} \hat{\Pr}^\dagger[Y^*{=}1|\mathbf{X}{=}\mathbf{x}, S{=}s]. \tag{17}$$

$\hat{\Pr}^\dagger[Y^*{=}1|\mathbf{X}{=}\mathbf{x}, S{=}s]$ is the probability that an actual label becomes 1 given a specific data $(\mathbf{x}, s)$. This probability can be 0 or 1 because labels are deterministically assigned by the decision rule, and becomes 1 if the following condition is satisfied:

$$\hat{\Pr}^\dagger[Y^*{=}1|\mathbf{X}{=}\mathbf{x}, S{=}s] \geq \hat{\Pr}^\dagger[Y^*{=}0|\mathbf{X}{=}\mathbf{x}, S{=}s]$$

Using model (15), this condition is equivalent to

$$\frac{q_s \hat{\Pr}^\dagger[S{=}s] \hat{\Pr}^\dagger[\mathbf{X}{=}\mathbf{x}|Y^*{=}1, S{=}s]}{\hat{\Pr}^\dagger[\mathbf{X}{=}\mathbf{x}, S{=}s]} \geq \frac{(1 - q_s) \hat{\Pr}^\dagger[S{=}s] \hat{\Pr}^\dagger[\mathbf{X}{=}\mathbf{x}|Y^*{=}0, S{=}s]}{\hat{\Pr}^\dagger[\mathbf{X}{=}\mathbf{x}, S{=}s]},$$

$$q_s \geq \frac{\hat{\Pr}^\dagger[\mathbf{X}{=}\mathbf{x}|Y^*{=}0, S{=}s]}{\sum_{y \in \{0,1\}} \hat{\Pr}^\dagger[\mathbf{X}{=}\mathbf{x}|Y^*{=}y, S{=}s]}. \tag{18}$$

TABLE III. COMPARISON OF OUR AFFNB METHOD WITH CV2NB AND CV2NB METHODS

(a) Adult data

| Methods | Acc | CVS | NPI |
|---|---|---|---|
| AFFNB | 0.828 | $-0.002$ | $5.43 \times 10^{-6}$ |
| HFFNB | 0.828 | 0.129 | $1.52 \times 10^{-2}$ |
| CV2NB | 0.828 | $-0.003$ | $6.89 \times 10^{-6}$ |

(b) Dutch data

| Methods | Acc | CVS | NPI |
|---|---|---|---|
| AFFNB | 0.761 | $-0.002$ | $2.68 \times 10^{-6}$ |
| HFFNB | 0.810 | 0.312 | $7.17 \times 10^{-2}$ |
| CV2NB | 0.761 | $-0.003$ | $8.79 \times 10^{-6}$ |

We can obtain $\hat{\Pr}^\dagger[Y^*{=}1|S{=}\,s]$ together with equation (17):

$$\hat{\Pr}^\dagger[Y^*{=}1|S{=}\,s] = \frac{1}{|\mathcal{D}[S{=}s]|} \sum_{(\mathbf{x}) \in \mathcal{D}[S=s]} \mathrm{I}[\mathbf{x}, s], \tag{19}$$

where $\mathrm{I}[\mathbf{x}, s]$ is an indicator function that takes 1 if inequality (18) is satisfied; and 0 otherwise.

Now, all we have to do is tuning parameters in the model (15) so as to fit the model to the training data set. In terms of $\hat{\Pr}^\dagger[S]$ and $\hat{\Pr}^\dagger[\mathbf{X}|Y, S]$, parameters can be estimated simply counting the number of occurrences in the training data set, because these parameters can be interpreted as the corresponding probabilities. The remaining parameters, $q_s$, are tuned so that these distributions to satisfy the following fairness conditions:

- satisfying a fairness condition: $Y^* \perp\!\!\!\perp S$,
- preserving a distribution of $Y$: $\hat{\Pr}^\dagger[Y^*{=}1] = |\mathcal{D}[Y{=}1]|/N$, and
- preserving a distribution of $S$: $\hat{\Pr}^\dagger[S{=}1] = |\mathcal{D}[S{=}1]|/N$.

From these conditions $\hat{\Pr}^\dagger[Y^*{=}1|S{=}s]$ should equal to $|\mathcal{D}[Y{=}1]|/N$ for $s \in \{0, 1\}$. According to equation (19), $\hat{\Pr}^\dagger[Y^*{=}1|S{=}s]$ is a function of $q_s$. Hence, the parameter, $q_s$, should be optimized so that $\hat{\Pr}^\dagger[Y^*{=}1|S{=}s]$ well approximates $|\mathcal{D}[Y{=}1]|/N$. Note that while the model distribution $\hat{\Pr}^\dagger[Y|S]$ is fitted to $\Pr[Y]$ in the case of hypothetical fair-factorization, the actual distribution $\hat{\Pr}^\dagger[Y^*|S]$ is fit to $\Pr[Y]$ in the case of actual fair-factorization.

Parameters $q_s$ are numerically optimized by using a scholar optimizer, because equation (19) is not differentiable due to the discrete transformation. In our experiment, we used a Brent optimizer in the `scipy`[2] library. After computing the l.h.s. of equation (18) for each training data in $O(N)$ time, $q_s$ can be optimized in $O(\log N)$ time, because equation (19) is monotone in terms of $q_s$. Therefore, the total time complexity of the AFFNB is $O(N)$. Note that in the case of the CV2NB method, $O(N)$ are required for each iteration in the post-processing algorithm in Figure (3), because all training data must be classified to compute $disc$ in line 11. Therefore, the AFFNB method is much faster than the CV2NB method.

### B. Experimental Results

Finally, we compared this new AFFNB method with the HFFNB and CV2NB methods. Experimental conditions

[2]http://www.scipy.org/

were the same as described in section III-C3. We show the experimental results in Table III. The performance of our AFFNB method was dramatically improved in comparison with the HFFNB method. Our AFFNB method performed almost equally to the CV2NB method in both accuracy and fairness. Additionally, our AFFNB has a useful property that the CV2NB method does not have. As described in section II-E1, the CV2NB method may not preserve a distribution over $Y$, but our AFFNB method does, because such a constraint was explicitly imposed. This property is useful because in the context of admissions, for instance, it would be inconvenient if the number of admitted students changed.

In summary, the performance results of the CV2NB and AFFNB methods were very close. This indicates that the CV2NB method is designed to fair-factorize not a hypothetical distribution, but an actual distribution, as in the case of the AFFNB method. One distinction between the two methods is the explicit constraint to preserve the distribution over $Y$. It should be concluded that this is the reason why the CV2NB method performed better in terms of fairness. Indeed, the other fairness-aware classifiers, such as the latent variable model [3] or logistic regression with a prejudice remover [8], are designed to fair-factorize a hypothetical distribution. In this paper, we applied an actual fair-factorization technique to a naive Bayes classifier, but this technique can be applied to other classifiers. We plan to develop logistic regression and SVM classifiers with the actual fair-factorization technique.

## VI. Conclusion

In this paper, we first review fairness-aware classifiers, and then focus on why the CV2NB method can attain better fairness results than other methods. We theoretically and empirically show the reason by comparing a simple alternative naive-Bayes modified by a hypothetical fair-factorization technique. Based on our findings, we developed a modified version, an actual fair-factorization technique, and show that this technique drastically improved the performance. We plan to apply this actual fair-factorization technique in order to modify other classification methods, such as logistic regression.

## Acknowledgments

This work is supported by MEXT/JSPS KAKENHI Grant Number 16700157, 21500154, 23240043, 24500194, and 25540094.

## References

[1] D. Pedreschi, S. Ruggieri, and F. Turini, "Discrimination-aware data mining," in *Proc. of the 14th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, 2008, pp. 560–568.

[2] A. Romei and S. Ruggieri, "A multidisciplinary survey on discrimination analysis," *The Knowledge Engineering Review*, 2013, [FirstView Article].

[3] T. Calders and S. Verwer, "Three naive bayes approaches for discrimination-free classification," *Data Mining and Knowledge Discovery*, vol. 21, pp. 277–292, 2010.

[4] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, "Learning fair representations," in *Proc. of the 30th Int'l Conf. on Machine Learning*, 2013.

[5] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma, "Enhancement of the neutrality in recommendation," in *Proc. of the 2nd Workshop on Human Decision Making in Recommender Systems*, 2012, pp. 8–14.

[6] D. Gondek and T. Hofmann, "Non-redundant data clustering," in *Proc. of the 4th IEEE Int'l Conf. on Data Mining*, 2004, pp. 75–82.

[7] F. Kamiran, A. Karim, and X. Zhang, "Decision theory for discrimination-aware classification," in *Proc. of the 12th IEEE Int'l Conf. on Data Mining*, 2012, pp. 924–929.

[8] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma, "Fairness-aware classifier with prejudice remover regularizer," in *Proc. of the ECML PKDD 2012, Part II*, 2012, pp. 35–50, [LNCS 7524].

[9] L. Sweeney, "Discrimination in online ad delivery," *Communications of the ACM*, vol. 56, no. 5, pp. 44–54, 2013.

[10] K. Fukuchi, J. Sakuma, and T. Kamishima, "Prediction with model-based neutrality," in *Proc. of the ECML PKDD 2013, Part II*, 2013, pp. 499–514, [LNCS 8189].

[11] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," in *Proc. of the 3rd Innovations in Theoretical Computer Science Conf.*, 2012, pp. 214–226.

[12] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma, "Considerations on fairness-aware data mining," in *Proc. of the IEEE Int'l Workshop on Discrimination and Privacy-Aware Data Mining*, 2012, pp. 378–385.

[13] F. Kamiran, T. Calders, and M. Pechenizkiy, "Discrimination aware decision tree learning," in *Proc. of the 10th IEEE Int'l Conf. on Data Mining*, 2010, pp. 869–874.

[14] I. Žliobaitė, F. Kamiran, and T. Calders, "Handling conditional discrimination," in *Proc. of the 11th IEEE Int'l Conf. on Data Mining*, 2011.

[15] B. Berendt and S. Preibusch, "Exploring discrimination: A user-centric evaluation of discrimination-aware data mining," in *Proc. of the IEEE Int'l Workshop on Discrimination and Privacy-Aware Data Mining*, 2012, pp. 344–351.

[16] S. Hajian and J. Domingo-Ferrer, "A study on the impact of data anonymization on anti-discrimination," in *Proc. of the IEEE Int'l Workshop on Discrimination and Privacy-Aware Data Mining*, 2012, pp. 352–359.

[17] B. T. Luong, S. Ruggieri, and F. Turini, "k-NN as an implementation of situation testing for discrimination discovery and prevention," in *Proc. of the 17th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, 2011, pp. 502–510.

[18] D. Pedreschi, S. Ruggieri, and F. Turini, "Measuring discrimination in socially-sensitive decision records," in *Proc. of the SIAM Int'l Conf. on Data Mining*, 2009, pp. 581–592.

[19] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma, "Efficiency improvement of neutrality-enhanced recommendation," in *Proc. of the 3rd Workshop on Human Decision Making in Recommender Systems*, 2013, pp. 1–8.

[20] D. Gondek and T. Hofmann, "Non-redundant clustering with conditional ensembles," in *Proc. of the 11th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, 2005, pp. 70–77.

[21] F. Kamiran and T. Calders, "Data preprocessing techniques for classification without discrimination," *Knowledge and Information Systems*, vol. 33, pp. 1–33, 2012.

[22] C. Elkan, "The foundations of cost-sensitive learning," in *Proc. of the 17th Int'l Joint Conf. on Artificial Intelligence*, 2001, pp. 973–978.

[23] C. C. Aggarwal and P. S. Yu, Eds., *Privacy-Preserving Data Mining: Models and Algorithms*. Springer, 2008.

[24] B. C. M. Fung, K. Wang, B. R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," *ACM Computing Surveys*, vol. 42, no. 4, 2010.

[25] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. of the 3rd Theory of Cryptography Conference*, 2006, pp. 265–284, [LNCS 3876].

[26] A. Frank and A. Asuncion, "UCI machine learning repository," University of California, Irvine, School of Information and Computer Sciences, 2010, ⟨http://archive.ics.uci.edu/ml⟩.