LEARNING FAIR REPRESENTATIONS WITHOUT DEMOGRAPHICS

By

Xiaoxue Wang

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science – Master of Science

2022

ABSTRACT

LEARNING FAIR REPRESENTATIONS WITHOUT DEMOGRAPHICS

By

Xiaoxue Wang

Due to hard accessibility, real-world adoption of fair representation learning algorithms lacks the prior knowledge of the sensitive attributes that we wish to be fair with. To address the challenge in fairness without explicit demographics, our solution is based on the idea of maximally randomizing the representation while being as informative as possible about the target task. We operationalize this goal through the concept of maximizing the entropy of the learned representation. For this purpose, we propose two new avenues for entropy maximization in the absence of demographic information: intra-class and inter-class entropy maximization. For 1) intra-class entropy maximization, it maximizes the entropy of the non-target class predictions (excluding the probability of the ground truth class label for classification problems), thus encouraging the model to discard spurious correlations between the different target classes, and for 2) inter-class entropy maximization, it maximizes the entropy of the representation conditioned on the target label, thus encouraging randomization of the samples within each target class label and minimizing the leakage of potential demographic information in the representation. Quantitative and qualitative results of our Maximum Entropy method (MaxEnt) on COMPAS and UCI Adult datasets show that 1) our method can outperform the State-of-the-art (SOTA) Adversarially Reweighted Learning (ARL) method and will enhance the difficulty of extracting sensitive demographic information in representation without prior demographic knowledge 2) our method reaches a good trade-off between utility and fairness.

This thesis is dedicated to my parents, Hui Yang and Yong Wang

ACKNOWLEDGEMENTS

The years at MSU have been a great experience in my life. It is a great chance to remember and to thank my teachers, colleagues, friends, and family whose influence contributed to this thesis.

First and foremost, I would like to thank my advisor, Vishnu Naresh Boddeti, for his extraordinary support, patience, guidance, and funding my entire research on fairness and privacy. His patience on guiding and teaching has strengthened my research ability in the field of computer vision and deep learning. I deeply appreciate is enthusiasm for novel approaches and genuinely positive attitude towards science and my research progress.

I thank all of the members of the HAL lab for participating in my research and providing valuable feedback on my work at all times. I am very grateful for their friendship and support.

I thank my entire family for their love and support, especially my parents, Yong Wang, Hui Yang, my grandma, Taoxiu Li, and my boyfriend, Yuanda Wang.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

A common scenario that is often encountered during real-world adoption of fair representation learning algorithms is the *apriori* lack of knowledge of the sensitive attributes that we wish to be fair with respect to. This may be the case where the sensitive attributes are not readily available or are too sensitive to be accessible. The underlying premises for the feasibility of this learning problem are, (1) by only retaining the minimal information relevant to the target task it is feasible to be fair with respect to unknown sensitive attributes that are independent of target attributes $\boldsymbol{y}$, and (2) all other cases will lead to a trade-off between utility and fairness that will depend on the degree of dependence between the attributes.

Methods to address this scenario include using Distributionally Robust Optimization (DRO) [1], Adversarially Reweighted Learning (ARL) [2] and Blind Pareto Fairness (BPF) [3]. DRO has been shown to overfit outliers [2], while ARL overfits when the adversary is more expressive than a simple linear layer, and BPF assumes convexity of the classifier's loss and hypothesis class. And more importantly, these approaches (1) seek to make the target predictions $\hat{\boldsymbol{y}}$ fair and do not consider the problem of learning fair representations, and (2) operate on small-scale problems. Another class of approaches seeks to leverage proxies to compensate for the lack of demographic labels [4, 5].

To address the challenge and the same scenario, we propose a Maximum Entropy method on both intra-class and inter-class branches by learning a representation that considers the trade-off between the utility and the fairness without prior demographic knowledge. The inter-class branch aims to make the non-target predictions uniformly distributed to mitigate the sensitive information leakage, while the intra-class branch tries to maximize the entropy of the embedding given each class to realize the fairness. We adopt two real-world datasets, COMPAS [6] and UCI Adult [7] datasets to validate the effectiveness of our proposed method. We make the following observations that (1) our method can outperform the SOTA ARL method [2] and will enhance the difficulty of extracting sensitive demographic information in representation without prior demographic knowledge (2) our method reaches a good trade-off between utility and fairness.

CHAPTER 2

RELATED WORK

There has been an increasing line of work to address fairness without explicit demographics. Some papers [4, 8, 9] address this issue by using proxy variables to compute the protected population labels. These methods contrast with our assumptions by relying on a preconceived notion on what the protected demographics are (i.e., the protected demographics are known, but unobserved), since prior knowledge is needed to design useful proxy variables. However, these proxy methods have been proved that undesired bias will be introduced to further exacerbate disparities [10, 11], and proxy information might be hard to obtain for many applications.

The fairness notions can be grouped into three categories: (i) individual fairness [12–16] provides guarantees beyond protected attributes, but requires predefined similarity functions which may be hard or infeasible to design for real-world tasks; (ii) group fairness [17–19] learns a model that satisfies a certain notion of fairness across these groups (e.g., statistical parity, equality of opportunity), but it conflicts with notions of no-harm fairness as in the [20], where the benefits from some groups might be purposely harmed. (iii) fairness notions that aim to improve per-group performance, such as Pareto-fairness and Rawlsian Max-Min fairness, which are appropriate where quality of service is paramount [1–3, 20]. In this paper, we are focusing on (iii) per-group fairness.

There are three recent approaches that are the most similar to our objective (learning maximum target information while protecting unknown and unobserved sensitive information/demographics). One is distributionally robust optimization (DRO) [1], where the goal is to achieve minimax fairness for unknown populations of sufficient size. They minimize the risk of the worst-case group for the worst-case group partition, and uses results from distributional robustness that focus the attention of the model exclusively on the high-risk samples (i.e., their model reduces the tail of the risk distribution). However, they do not explicitly account for Pareto efficiency, meaning that their solution may be sub-optimal on the population segment that lies below their high-risk threshold, doing unnecessary harm. The second recent method is adversarially reweighted learning (ARL) [2], where the model is trained to reduce a positive linear combination of the sample errors. These weighting coefficients are proposed by an adversary (implemented as a neural network), with the goal of maximizing the weighted empirical error. The method focuses on computationally identifiable subgroups. However, their adversary model is restricted to linear models, and they do not provide an optimality guarantee on the adversary, nor do they pose a constraint on the computationally identifiable subgroups. The third recent method is Blind Pareto Fairness (BPF) [3], which does not rely on predefined notions of at-risk groups, neither at train nor at test time. It leverages no-regret dynamics to recover a fair minimax classifier that reduces worst-case risk of any potential subgroup of sufficient size, and guarantees that the remaining population receives the best possible level of service. However, this method is restricted in the strong convexity of the loss given classifier and the classifier hypothesis class. And the classifier in the experiment is restricted to the 1-hidden layer Multi-layer Perceptron (MLP).

CHAPTER 3

PROBLEM SETTING

This chapter addresses the problem setting. In order to achieve fairness, we will first dive into the precise mathematics formulation.

In this work, we will use the **Rawlsian Max-Min Fairness** as our fairness definition metric.

Definition 1. Rawlsian Max-Min Fairness: Suppose H is a set of hypotheses, and $U_{\mathcal{D}_s}(h)$ is the expected utility of the hypothesis h for the individuals in group s, then a hypothesis h^* is said to satisfy Rawlsian Max-Min fairness principle [20] if it maximizes the utility of the worst-off group, i.e., the group with the lowest utility.

$$h^* = \operatorname*{argmax}_{h \in \mathcal{H}} \min_{s \in \mathcal{S}} U_{\mathcal{D}_s}(h) \tag{3.1}$$

In our evaluation in Chapter 5, we use AUC as our utility metric, which aligns with the paper [2], and report the minimum utility over protected groups S as AUC(min).

Most utility metrics in traditional machine learning are not differentiable, thus convex loss functions are commonly used. The traditional Machine Learning task is to learn a model h that minimizes the loss over the training data $\mathcal{D}$.

$$h^*_{\text{avg}} = \operatorname*{argmin}_{h \in H} L_{\mathcal{D}}(h) \tag{3.2}$$

where $L_{\mathcal{D}}(h) = \mathbb{E}_{(x_i, y_i) \sim \mathcal{D}} \left[\ell(h(x_i), y_i)\right]$ for some loss function $\ell(\cdot)$ (e.g., cross entropy).

Therefore, we take the same perspective in turning Rawlsian Max-Min Fairness as given in the 3.1 in to a learning objective in 3.2. Replacing the expected utility when an appropriate loss function $L_{\mathcal{D}_s}(h)$ over the set of individuals in group s, we can formulate the fairness objective as:

$$h^*_{\max} = \operatorname*{argmin}_{h \in H} \max\nolimits_{s \in S} L_{\mathcal{D}_s}(h) \tag{3.3}$$

where $L_{\mathcal{D}}(h) = \mathbb{E}_{(x_i,y_i)\sim\mathcal{D}}\left[\ell(h(x_i), y_i)\right]$ for some loss function $\ell(\cdot)$ (e.g., cross entropy).

However, the prior demographic information s is not necessarily needed during training. We only need the protected group s information during evaluation. According to the objective function 3.3, there are two goals that we are going to achieve. One is to learn the target task well, and at the same time, protect the sensitive information leakage without prior demographic knowledge.

CHAPTER 4

APPROCH: MAXIMUM ENTROPY

In this chapter, we will introduce our Maximum Entropy method (MaxEnt). In information theory, the entropy of a random variable is a measurement of uncertainty. We maximize the uncertainty over the sensitive information to protect the privacy. We implement our Maximum Entropy method on two branches: inter-class branch and intra-class branch. For the inter-class branch, we maximize the entropy of non-target classes. For the intra-class branch, we maximize the entropy of class-conditional representations by making them as Gaussian distributions.

4.1 Inter-class MaxEnt: Maximizing Entropy of non-Target Class

Definition 2. Maximum entropy distribution over a finite (discrete) range is the Uniform Distribution [21].

Based on the definition 2, we make the non-target classes uniformly distributed to maximize the entropy. The reason why maximizing entropy of non-target prediction can protect the sensitive information can be illustrated in the toy example below. In the figure 4.1,

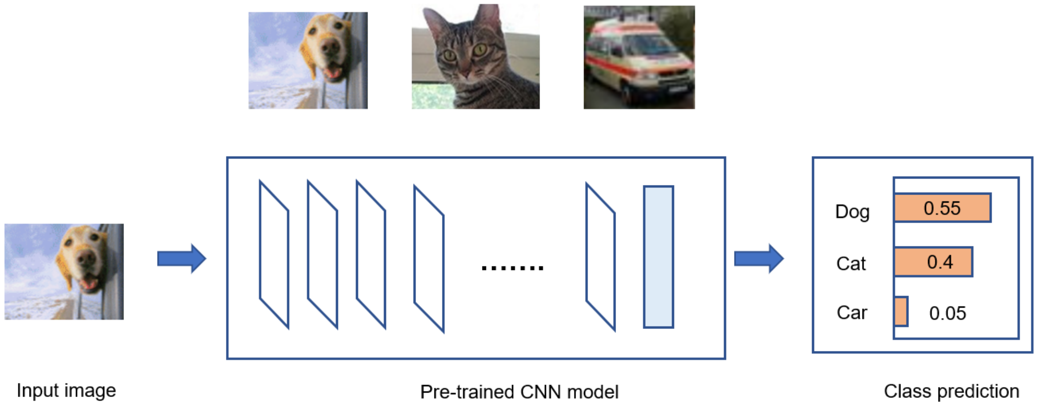


Figure 4.1: Inter-class MaxEnt toy example: Make the non-target classes uniformly distributed to maximize the entropy can protect the sensitive information while maintaining good target task performance.

for example, our target task is to classify the dog image from the dog, cat and car images [22]. At the mean time, we also have the sensitive labels in mind, which we do not use them during the training - animal and non-animal. From the output of class prediction, we can make the following observations from the results: we can predict the dog correctly as it has the highest prediction probability 0.55 compared to cat with 0.4 and car with 0.05. However, as cat and dog may have more similarities compared to car, so their predicted probabilities will be more similar, which will reveal that dog and cat may come from the same sensitive group - animal, and the remaining car will come from another sensitive group, non-animal. In this way, the sensitive information is leaked easily.

In order to mitigate this sensitive information leakage, we can make the non-target classes uniformly distributed. So the original probability triplet $P(input = dog) = 0.55, P(input = cat) = 0.4, P(input = car) = 0.05$ will be distributed as $P(input = dog) = 0.55, P(input = cat) = 0.225, P(input = car) = 0.225$. In this case, we can still correctly predict the input image, which is dog, while can not tell the relationship of the sensitive groups of the three anymore. In this way, we protect the sensitive information while maintaining good performance on target task by making the non-target classes uniformly distributed to maximize the entropy.

We further prove our idea with an example by using the real-world COMPAS dataset [6]. It has 11 features with 7215 data samples. The protected features are race and sex. The target label is recidivism prediction. The baseline we use is the vanilla group-agnostic Baseline used in [2], which performs standard Empirical Risk Minimization (ERM) with uniform weights. We use a set of AUCs as evaluation metrics. AUC(min) is the minimum AUC over all protected groups. AUC(macro-average) is the macro-average over all protected group AUCs, which is a weighted average. AUC(minority) is the AUC reported for the smallest protected group in the dataset. We choose AUC (area under the ROC curve) as our evaluation metric as it is robust to class imbalance. Also, it encompasses both False Positive Rate (FPR) and False Negative Rate (FNR), and is threshold agnostic.

Table 4.1: Main results: Inter-class Maximum Entropy branch vs Baseline

Dataset	Method	AUC avg	AUC macro-avg	AUC min	AUC minority
COMPAS	Baseline	0.748	0.730	0.674	0.774
	Inter-class branch	**0.748**	**0.733**	**0.693**	0.726

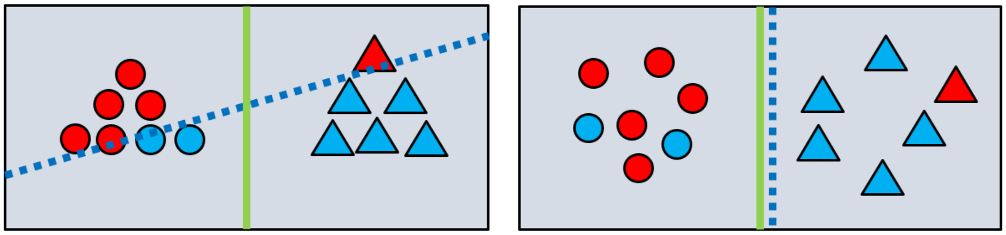
From table 4.1, we can make the following observations from the results that the overall AUCs by using the target loss and inter-class entropy loss will be larger than the baseline. And the gap between the AUCs will be smaller, which shows a good trade-off between utility and fairness. Though in COMPAS dataset, inter-class MaxEnt method has lower AUC (minority) compared to baseline method, it does not degrade our method, because AUC (minority) does not necessarily convey fairness. As AUC (minority) is denoted as the AUC from the minimum-size protected group, whereas in many cases minimum-size group does not belong to the weakness group, e.g. the rich and the poor group. So the other AUCs metrics are more convincing in fairness measurement. More detailed discussion is given in Chapter 6.

4.2 Intra-class MaxEnt: Maximizing Entropy of Class-Conditional Representations

Definition 3. Maximum entropy distribution with a variance and mean constraint is the Gaussian distribution [21].

Based on the definition 3, we make the class-conditional representations into Gaussian distributions to maximize the entropy. The reason why maximizing entropy of class-conditional representations can protect the sensitive information will be explained in the example below. In this intra-class MaxEnt toy example, we assume the shape (circle and triangle) represents the target classes, and the color (blue and red) represents sensitive classes. Green line is the decision boundary for target classes, while the blue line is the decision boundary for sensitive classes.

In figure 4.2, we make the following observations from the results that before making the class-conditional representations into Gaussian distributions, we can perfectly classify



(a) When class-conditional representations are not Gaussian

(b) When class-conditional representations are Gaussian

Figure 4.2: Intra-class MaxEnt toy example: Make the class-conditional representations into Gaussian distributions to maximize the entropy can protect the sensitive information while maintaining good target task performance.

the shape and the color with the green and blue decision boundaries in figure 4.2a. After making the class-conditional representations into Gaussian distributions as in figure 4.2b (where the shape of the representations are more round), we can still perfectly classify the shapes with the green decision boundary, but we can not perfectly classify the colors with the blue decision boundary anymore. In this way, based on the toy example, it is illustrated that we can protect the sensitive information by making the class-conditional representations into Gaussian distributions to maximize the entropy while maintaining good target task performance at the same time.

Next, We further prove our Intra-class MaxEnt idea with a linear example calling MaxEnt LDA.

In traditional Linear Discriminant Analysis (LDA) [23], the goal is to perform dimensionality reduction and preserve as much of the class discriminatory information as possible. So the ideal projected mapping is to achieve far-away means between each class, and the small variance within each class. So the objective function of traditional LDA is $\max\limits_{w} \frac{w^T S_1 w}{w^T S_2 w}, s.t. w^T w = 1$, where S_1 is the between-class scatter of the projected samples, and S_2 is the within-class scatter of the original samples/ feature vectors.

However, from figure 4.2b, we make the following observations from the results that the ideal representation/ embedding we want is to have the far-away means between each class, while the large variance within each class. So the objective function of the MaxEnt LDA (linear example of the MaxEnt method) can be extended from the traditional LDA, whose objective function is $\max_{w}(\alpha w^T S_1 w + +(1-\alpha) w^T S_2 w), s.t. \alpha \in [0,1], w^T w = 1$. In this toy example, we use 2-dimensional Gaussian as our toy data.

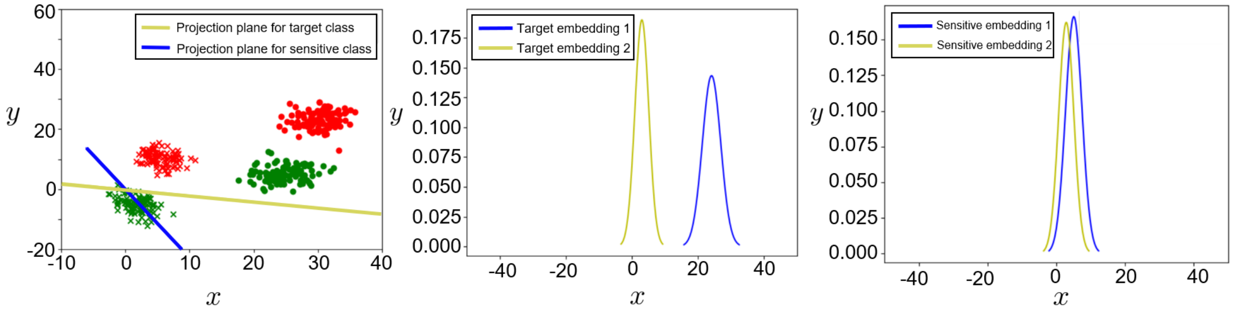


Figure 4.3: MaxEnt LDA

From figure 4.3, we can make the following observations from the results that when cleverly selecting α, we can find an ideal projection plane for the target class and the sensitive class so that we can perfectly classify the target class while not being able to classify the sensitive class by making the target representations well separated but the sensitive representations overlapped between each other.

4.3 Maximum Entropy (MaxEnt) Method

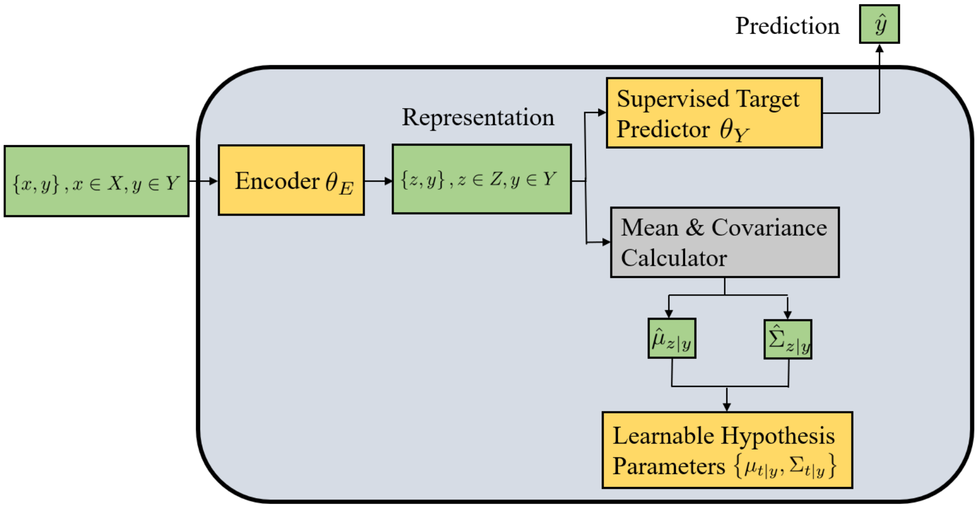

Prediction
$\hat{y}$
Representation
Supervised Target Predictor θ_Y
$\{x, y\}, x \in X, y \in Y$
Encoder θ_E
$\{z, y\}, z \in Z, y \in Y$
Mean & Covariance Calculator
$\hat{\mu}_{z|y}$
$\hat{\Sigma}_{z|y}$
Learnable Hypothesis Parameters $\{\mu_{t|y}, \Sigma_{t|y}\}$

Figure 4.4: MaxEnt Method Structure

Our goal is to learn a representation that achieves a good trade-off between utility and fairness. In order to reach a good utility, we learn a representation that can learn the target task well. And in order to achieve fairness, we use Maximum Entropy method on both inter-class and intra-class branch.

In order to learn the target task well, *encoderE* is trained to learn the representation parameterized with θ_E, which tries to maximize the likelihood of the target attribute, as measured by the target predictor parameterized with θ_Y.

$$\mathcal{L}_T(\theta_E, \theta_Y) = KL(p(y|x) \parallel q_Y(\hat{y}|e(x, \theta_E); \theta_Y) \tag{4.1}$$

In order to maximizing entropy of non-target classes to protect the sensitive information, the encoder and predictor will be trained with the inter-class Maximum Entropy loss $\mathcal{L}_E$.

$$\begin{aligned} \mathcal{L}_E(\theta_E, \theta_Y) &= KL(q_Y(\hat{y}|e(x, \theta_E); \theta_Y)_{\hat{y}\neq y} \parallel \tilde{U}_y) \\ \text{s.t.} \quad \tilde{U}_y &= \frac{1 - P(y)}{\#Y - 1} \end{aligned} \tag{4.2}$$

where $P(y)$ denotes the probability of the target class, and $\#Y$ denotes the number of the total target classes.

In order to make the class-conditional representations as Gaussian distributions, the learnable hypothesis parameters $\left\{\mu_{t|y}, \Sigma_{t|y}\right\}$ are trained. The mean and covriance calculator will be used to calculate the mean and covariance of the representations given each class represented with $\left\{\mu_{z|y}, \Sigma_{z|y}\right\}$. The learnable hypothesis parameters will be updated in an iterative manner until it converges with the calculated mean and covariance of the representations given each class. The learnable hypothesis parameters will be trained with the intra-class Maximum Entropy loss shown below:

$$\mathcal{L}_G(\theta_E, \mu_{z|y}, \Sigma_{z|y}) = \mathrm{E}\left[\parallel \mu_{z|y} - \hat{\mu}_{z|y} \parallel_2^2\right] + \mathrm{E}\left[\parallel \Sigma_{z|y} - \hat{\Sigma}_{z|y} \parallel_2^2\right] \tag{4.3}$$

As the class-conditional representations are learned to be Gaussian distributions, regularizations are needed. So, the regularization losses $\mathcal{L}_G$ and $\mathcal{L}_R$ are proposed to make the covariance elements of the class-conditional Gaussian distributions non-zero and isotropic.

$$\mathcal{L}_R(\theta_E, \Sigma_{z|y}) = \| \operatorname{diag}(\Sigma_{z|y}) - \operatorname{diag}(\hat{\Sigma}_{z|y}) \|_2^2 + \text{off-diag}(\hat{\Sigma}_{z|y}) \tag{4.4}$$

$$\mathcal{L}_S(\theta_E, \Sigma_{z|y}) = \mathrm{E}\left[\exp(-|\operatorname{diag}(\Sigma_{z|y})|) + \exp(-|\operatorname{diag}(\hat{\Sigma}_{z|y})|)\right] \tag{4.5}$$

In summary, the Maximum Entropy Method algorithm will be shown as follows:

Algorithm 4.1: Maximum Entropy Method

```
Require: Training data $x \in X$ with target labels $y \in Y$, initial encoder parameters $\theta_E$,
  initial target predictor parameters $\theta_Y$, initial learnable hypothesis parameters for each class-
  conditional representation $\left\{\mu_{t|y}, \Sigma_{t|y}\right\}$, trade-off parameters $\alpha_1, \alpha_2$ such that $0 \leq \alpha_1+\alpha_2 \leq 1$,
  small regularization parameter $\epsilon_1, \epsilon_2$ where $\epsilon_1, \epsilon_2 < 10^{-6}$, and learning rate $\eta_1, \eta_2$.
  $t \leftarrow 0$
  while not converge do
      $t \leftarrow t + 1$
      Compute target loss $\mathcal{L}_T(\theta_E, \theta_Y)$ // Using equation 5.3
      Compute inter-class MaxEnt loss $\mathcal{L}_E(\theta_E, \theta_Y)$ //Using equation 4.2
      Compute intra-class MaxEnt loss $\mathcal{L}_G(\theta_E, \mu_{z|y}, \Sigma_{z|y})$ //Using equation 4.3
      Compute the Gaussian regularization loss $\mathcal{L}_R(\theta_E, \Sigma_{z|y})$ //Using equation 4.4
      Compute the Gaussian singularity regularization loss $\mathcal{L}_S(\theta_E, \Sigma_{z|y})$ //Using equation 5.5
      $\mathcal{L}^t \leftarrow \alpha_1\mathcal{L}_T^t + \alpha_2\mathcal{L}_E^t + (1 - \alpha_1 - \alpha_2)\mathcal{L}_G^t + \epsilon_1\mathcal{L}_R^t + \epsilon_2\mathcal{L}_S^t$
      Compute the back-propagation error w.r.t. input $\frac{\partial \mathcal{L}^t}{\partial x_i^t}$ for each layer $i$
      Update the parameters $\theta_E^t$ by $\theta_E^{t+1} = \theta_E^t - \eta_1 \frac{\partial \mathcal{L}^t}{\partial \theta_E^t}$
      Update the parameters $\left\{\mu_{t|y}, \Sigma_{t|y}\right\}$ by $\mu_{t|y}^{t+1} = \mu_{t|y}^t - \eta_2 \frac{\partial \mathcal{L}^t}{\partial \mu_{t|y}^t}$ and $\Sigma_{t|y}^{t+1} = \Sigma_{t|y}^t - \eta_2 \frac{\partial \mathcal{L}^t}{\partial \Sigma_{t|y}^t}$
  end while
  return $\theta_E$ or $\left\{\mu_{t|y}, \Sigma_{t|y}\right\}$
```

CHAPTER 5

THEORETICAL ANALYSIS

This chapter shows the theoretical analysis of the proposed Maximum Entropy Method.

In order to theoretically prove the feasibility of our proposed Maximum Entropy method on inter-class and intra-class branch. The theorem 1 and theorem 2 are proposed.

Lemma 1. $p(s|z) = f_{ys}(z) k_{ys} p(y|z)$
Where $f_{ys}(z) = \frac{p(z|s)}{p(z|y)}$, and $k_{ys} = \frac{p(s)}{p(y)}$

Proof. As Y and S are dependent, so $P(S,Y)$ should exist. So,

$$\begin{aligned} P(S,Y) &= P(S|Y)P(Y) \\ &= P(Y|S)P(S) \\ \therefore P(S=s) &= \frac{P(S=s|Y=y)}{P(Y=y|S=s)} P(Y=y) \\ \therefore P(S=s) &= k_{ys} P(Y=y), k_{ys} = \frac{P(S=s|Y=y)}{P(Y=y|S=s)} \end{aligned} \tag{5.1}$$

Furthermore,

$$\begin{aligned} \because p(y|z) &= \frac{p(y,z)}{p(z)} = \frac{p(z|y)p(y)}{p(z)} \\ \therefore p(z) &= \frac{p(z|y)p(y)}{p(y|z)} \\ \therefore p(s|z) &= \frac{p(s,z)}{p(z)} = \frac{p(z|s)p(s)}{p(z)} \\ &= \frac{p(z|s)p(s)}{p(z|y)p(y)} p(y|z) \\ &= f_{ys}(z) k_{ys} p(y|z) \\ f_{ys}(z) &= \frac{p(z|s)}{p(z|y)} \\ k_{ys} &= \frac{p(s)}{p(y)} \\ f_{ys}(z) k_{ys} &= \frac{p(z|s)p(s)}{p(z|y)p(y)} = \frac{p(z,s)}{p(z,y)} \end{aligned} \tag{5.2}$$

□

Theorem 1. Maximizing $H(Z|Y)$ can either not change or maximize $H(S|Z)$

Proof.

$$\begin{aligned} H(S|Z) &= H(S,Z) - H(Z) \\ &= H(Z|S) + H(S) - H(Z) \end{aligned} \tag{5.3}$$

According to lemma 1, $p(s|z) = f_{ys}(z)k_{ys}p(y|z)$, where $f_{ys}(z) = \frac{p(z|s)}{p(z|y)}$, and $k_{ys} = \frac{p(s)}{p(y)}$, so,

$$\begin{aligned} H(Z|S) &= -\sum_{s\in S}\int_{z\in Z} p(s)p(z|s)\log\frac{p(z,s)}{p(s)}dz \\ &= -\sum_{s\in S}\int_{z\in Z} p(z,s)\log\frac{p(z,s)}{p(s)}dz \\ &= -\sum_{s\in S}\int_{z\in Z} p(s|z)p(z)\log\frac{p(s|z)p(z)}{p(s)}dz \\ &= -\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} f_{ys}(z)k_{ys}p(y|z)p(z)\log\frac{f_{ys}(z)k_{ys}p(y|z)p(z)}{p(s)}dz \\ &= -\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} k_{ys}f_{ys}(z)p(z,y)\log\frac{k_{ys}f_{ys}(z)p(z,y)}{p(s)}dz \\ &= -\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} k_{ys}f_{ys}(z)p(z,y)\log k_{ys}f_{ys}(z)dz \\ & -\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} k_{ys}f_{ys}(z)p(z,y)\log p(z,y)dz \\ & +\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} k_{ys}f_{ys}(z)p(z,y)\log p(s)dz \\ &= A_1 + A_2 + A_3 \end{aligned} \tag{5.4}$$

$$
\begin{aligned}
A_3 &= \sum_{s \in S} \sum_{y \in Y} \int_{z \in Z} k_{ys} f_{ys}(z) p(z,y) \log p(s) dz \\
&= \sum_{s \in S} \sum_{y \in Y} \int_{z \in Z} \frac{p(z,s)}{p(z,y)} p(z,y) \log p(s) dz \\
&= \sum_{s \in S} \int_{z \in Z} p(z,s) \log p(s) dz \\
&= \sum_{s \in S} \int_{z \in Z} p(z,s) dz \log p(s) \\
&= \sum_{s \in S} p(s) \log p(s) \\
&= -H(S)
\end{aligned}
\tag{5.5}
$$

For A_2, according to the **First Mean Value Theorem** [24] (shown in Theorem 3), we can rewrite A_2 as:

$$
\begin{aligned}
A_2 &= -\sum_{s \in S} \sum_{y \in Y} \int_{z \in Z} k_{ys} f_{ys}(z) p(z,y) \log p(z,y) dz \\
\because k_{ys} f_{ys} &= \frac{p(z,s)}{p(z,y)} = \frac{p(z|s)p(s)}{p(z|y)p(y)} continuous, -p(z,y) \log p(z,y) \geq 0 \\
\therefore &= M \sum_{s \in S} \sum_{y \in Y} \int_{z \in Z} -p(z,y) \log p(z,y) dz \\
&= M \sum_{y \in Y} \int_{z \in Z} -p(z,y) \log p(z,y) dz \\
&= MH(Z,Y)
\end{aligned}
\tag{5.6}
$$

For A_3, according to the **Chain rule** [25] (shown in Theorem 4), we can rewrite A_1 as:

$$\begin{aligned}
A_1 &= -\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} k_{ys}f_{ys}(z)p(z,y)\log k_{ys}f_{ys}(z)dz \\
&= -\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} \frac{p(z,s)}{p(z,y)}p(z,y)\log\frac{p(z,s)}{p(z,y)}dz \\
&= -\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} p(z,s)\log\frac{p(z,s)}{p(z,y)}dz \\
&= -D(p(z,s) \parallel p(z,y)) \\
&= -(D(p(s) \parallel p(y)) + D(p(z|s) \parallel p(z|y))) \\
&= C_1 - D(p(z|s) \parallel p(z|y)) \\
&= C_1 - \sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} p(z|s)\log\frac{p(z|s)}{p(z|y)}dz \\
&= C_1 - M_2\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} \log\frac{p(z|s)}{p(z|y)}dz \\
&= C_1 - M_2\sum_{s\in S}\sum_{y\in Y}\int_{z\in Z} (\log p(z|s) - \log p(z|y))dz \\
&= C_1 - M_2(\sum_{s\in S}\int_{z\in Z}\log p(z|s)dz - \sum_{y\in Y}\int_{z\in Z}\log p(z|y)dz) \\
&= C_1 - M_2(K_Y * 1 - K_S * 1) \\
&= C_2
\end{aligned} \tag{5.7}$$

Where K_Y is the total number of classes in Y, and K_S is the total number of classes in S. C_2 means a constant number.

We can rewrite equation 5.3 as:

$$
\begin{aligned}
H(S|Z) &= H(Z|S) + H(S) - H(Z) \\
&= A_1 + A_2 + A_3 + H(S) - H(Z) \\
&= C_2 + MH(Z,Y) - H(S) + H(S) - H(Z) \\
&= C_2 + MH(Z,Y) - H(Z) \\
\therefore H(S,Z) &= H(S|Z) + H(Z) \\
&= C_2 + MH(Z,Y) - H(Z) + H(Z) \\
&= MH(Z,Y) \\
\because H(Z,Y) &= H(Y|Z) + H(Z) \qquad (5.8) \\
&= H(Z) if H(Y|Z) = 0 \\
H(Z,S) &= H(S|Z) + H(Z) \\
&= H(S|Z) + H(Z,Y) \\
&\because H(S|Z) \neq 0 \\
\therefore H(Z,S) &= H(S|Z) + H(Z,Y) \geq H(Z,Y) \\
\therefore H(Z,S) &= MH(Z,Y) \geq H(Z,Y) \\
\therefore M \geq 1
\end{aligned}
$$

According to equation 5.8,

$$
\begin{aligned}
H(S|Z) &= H(S,Z) - H(Z) \\
&= MH(Z,Y) - H(Z) \\
&= M(H(Z|Y) + H(Y)) - H(Z) \qquad (5.9) \\
&= MH(Z|Y) - H(Z) + MH(Y), M \geq 1
\end{aligned}
$$

According to paper [26], the upper bound of $H(Z)$ is $C_3 + H(Z|Y)$ (shown in 5), where C_3 is a constant number.

$$
\begin{aligned}
H(Z|Y) &= -\sum_{y\in Y}\int_z p(z|y)p(y)\log p(z|y)dz \\
&= -\sum_{k=1}^{K}\int_z w_k p(z|y=k)\log p(z|y=k)dz \\
&= w_1H_1(Z) + w_2H_2(Z) + \cdots + w_KH_K(Z) \\
&= w_1(\frac{d}{2}\ln 2\pi e + \frac{1}{2}\ln|\Sigma_1|) + w_2(\frac{d}{2}\ln 2\pi e + \frac{1}{2}\ln|\Sigma_2|)+ \\
&\cdots + w_K(\frac{d}{2}\ln 2\pi e + \frac{1}{2}\ln|\Sigma_K|) \\
&= \frac{d}{2}\ln 2\pi e + \frac{1}{2}(w_1\ln|\Sigma_1| + \cdots w_K\ln|\Sigma_K|)
\end{aligned}
\tag{5.10}
$$

According to the Theorem 3 in the reference paper [26], we can get the upper bound of $H(Z)$ is that:

$$
\begin{aligned}
H(Z) &\le \sum_{k=1}^{K} w_k(-\log w_k + \frac{d}{2}\ln 2\pi e + \frac{1}{2}\ln|\Sigma_k|) \\
&= C + \frac{d}{2}\ln 2\pi e + \frac{1}{2}(w_1\ln|\Sigma_1| + \cdots w_K\ln|\Sigma_K|) \\
&= C + H(Z|Y)
\end{aligned}
\tag{5.11}
$$

According to equation 5.9, $H(S|Z) = MH(Z|Y) - H(Z) + MH(Y), M \ge 1$, so,

$$
\begin{aligned}
\frac{\partial H(S|Z)}{\partial|\Sigma_i|} &= M\frac{\partial H(Z|Y)}{\partial|\Sigma_i|} - \frac{\partial H(Z)}{\partial|\Sigma_i|}, M \ge 1 \\
&= (M-1)\frac{\partial H(Z|Y)}{\partial|\Sigma_i|}, M \ge 1 \\
&\because \frac{\partial H(Z|Y)}{\partial|\Sigma_i|} > 0, \\
&\therefore \frac{\partial H(S|Z)}{\partial|\Sigma_i|} = (M-1)\frac{\partial H(Z|Y)}{\partial|\Sigma_i|} \ge 0
\end{aligned}
\tag{5.12}
$$

□

Theorem 2. Making non-target classes uniformly distributed is equivalent to maximizing $H(Y|Z)$, and can maximize $H(S|Z)$.

Proof. Denote there are K target classes, and D_{KL} denotes the KL divergence between two probability distributions.

$$
\begin{aligned}
&D_{KL}(P(y \neq Y|Z) \parallel \frac{1-P(y=Y|Z)}{K-1}) \\
&= \int_{z\in Z} p(z) \sum_{y\notin Y} p(y \neq Y|z) \sum_{y\in Y} \log \frac{p(y\neq Y|z)}{\frac{1-p(y=Y|z)}{K-1}} dz \\
&= \sum_{y\in Y}\sum_{y\notin Y}\int_{z\in Z} p(z)p(y\neq Y|z)\log \frac{p(y\neq Y|z)}{\frac{1-p(y=Y|z)}{K-1}} dz \\
&= \sum_{y\in Y}\sum_{y\notin Y}\int_{z\in Z} p(z)p(y\neq Y|z)\log p(y\neq Y|z)dz- \\
&\sum_{y\in Y}\sum_{y\notin Y}\int_{z\in Z} p(z)p(y\neq Y|z)\log(1-p(y=Y|z))dz \\
&+\sum_{y\in Y}\sum_{y\notin Y}\int_{z\in Z} p(z)p(y\neq Y|z)\log(K-1)dz \\
&= B_1 + B_2 + B_3
\end{aligned}
\tag{5.13}
$$

$$
\begin{aligned}
B_3 &= \sum_{y\in Y}\sum_{y\notin Y}\int_{z\in Z} p(z)p(y\neq Y|z)\log(K-1)dz \\
&= \sum_{y\notin Y}\int_{z\in Z} p(z)p(y\neq Y_1|z)\log(K-1)dz + \cdots + \sum_{y\notin Y}\int_{z\in Z} p(z)p(y\neq Y_K|z)\log(K-1)dz \\
&= (\int_{z\in Z} p(z)p(y=Y_2|z)\log(K-1)dz + \cdots + \int_{z\in Z} p(z)p(y=Y_K|z)\log(K-1)dz) + \cdots \\
&+ (\int_{z\in Z} p(z)p(y=Y_1|z)\log(K-1)dz + \cdots + \int_{z\in Z} p(z)p(y=Y_{K-1}|z)\log(K-1)dz) \\
&= (K-1)\int_{z\in Z} p(z)p(y=Y_1|z)\log(K-1)dz + \cdots \\
&+ (K-1)\int_{z\in Z} p(z)p(y=Y_K|z)\log(K-1)dz \\
&= (K-1)\sum_{y\in Y}\int_{z\in Z} p(z)p(y|z)\log(K-1)dz \\
&= (K-1)\sum_{y\in Y}\int_{z\in Z} p(z,y)dz\log(K-1) \\
&= (K-1)\log(K-1) \\
&= C_4
\end{aligned}
\tag{5.14}
$$

Where C_4 is a constant.

$$\begin{aligned}
B_2 &= -\sum_{y \in Y} \sum_{y \notin Y} \int_{z \in Z} p(z) p(y \neq Y|z) \log\left(1 - p(y = Y|z)\right) dz \\
&= -\sum_{y \notin Y_1} \int_{z \in Z} p(z) p(y \neq Y_1|z) \log\left(1 - p(y = Y_1|z)\right) dz - \cdots \\
&- \sum_{y \notin Y_K} \int_{z \in Z} p(z) p(y \neq Y_K|z) \log\left(1 - p(y = Y_K|z)\right) dz \\
&= -\int_{z \in Z} p(z) p(y = Y_2|z) \log\left(1 - p(y = Y_1|z)\right) dz - \cdots \\
&- \int_{z \in Z} p(z) p(y = Y_K|z) \log\left(1 - p(y = Y_1|z)\right) dz - \cdots \\
&- \int_{z \in Z} p(z) p(y = Y_1|z) \log\left(1 - p(y = Y_1|z)\right) dz - \cdots \\
&- \int_{z \in Z} p(z) p(y = Y_{K-1}|z) \log\left(1 - p(y = Y_K|z)\right) dz \\
&= -\int_{z \in Z} p(z) (p(y = Y_2|z) + \cdots + p(y = Y_K|z)) \log\left(1 - p(y = Y_1|z)\right) dz - \cdots \\
&- \int_{z \in Z} p(z) (p(y = Y_1|z) + \cdots + p(y = Y_{K-1}|z)) \log\left(1 - p(y = Y_K|z)\right) dz \\
&= -\int_{z \in Z} p(z) (1 - p(y = Y_1|z)) \log\left(1 - p(y = Y_1|z)\right) dz - \cdots \\
&- \int_{z \in Z} p(z) (1 - p(y = Y_K|z)) \log\left(1 - p(y = Y_K|z)\right) dz
\end{aligned} \tag{5.15}$$

$$
\begin{aligned}
B_1 &= \sum_{y\in Y}\sum_{y\notin Y}\int_{z\in Z} p(z)p(y\neq Y|z)\log p(y\neq Y|z)dz \\
&= \sum_{y\notin Y_1}\int_{z\in Z} p(z)p(y\neq Y_1|z)\log p(y\neq Y_1|z)dz+\cdots \\
&+ \sum_{y\notin Y_K}\int_{z\in Z} p(z)p(y\neq Y_K|z)\log p(y\neq Y_K|z)dz \\
&= \int_{z\in Z} p(z)p(y=Y_2|z)\log p(y=Y_2|z)dz+\cdots+\int_{z\in Z} p(z)p(y=Y_K|z)\log p(y=Y_K|z)dz \\
&+\cdots+\int_{z\in Z} p(z)p(y=Y_1|z)\log p(y=Y_1|z)dz+\cdots \\
&+\int_{z\in Z} p(z)p(y=Y_{K-1}|z)\log p(y=Y_{K-1}|z)dz \\
&= (K-1)\int_{z\in Z} p(z)p(y=Y_1|z)\log p(y=Y_1|z)dz+\cdots \\
&+(K-1)\int_{z\in Z} p(z)p(y=Y_K|z)\log p(y=Y_K|z)dz \\
&= -(K-1)H(Y_1|Z)-\cdots-(K-1)H(Y_K|Z)
\end{aligned}
\tag{5.16}
$$

According to equations 5.14 - 5.16, we can have:

$$
\begin{aligned}
&D_{KL}(P(y \neq Y|Z) \parallel \frac{1 - P(y = Y|Z)}{K-1}) = B_1 + B_2 + B_3 \\
&= (K-1)\int_{z \in Z} p(z)p(y = Y_1|z) \log p(y = Y_1|z)dz + \cdots \\
&+ (K-1)\int_{z \in Z} p(z)p(y = Y_K|z) \log p(y = Y_K|z)dz \\
&- \int_{z \in Z} p(z)(1 - p(y = Y_1|z)) \log{(1 - p(y = Y_1|z))}dz - \cdots \\
&- \int_{z \in Z} p(z)(1 - p(y = Y_K|z)) \log{(1 - p(y = Y_K|z))}dz + C_4 \\
&= K\int_{z \in Z} p(z)p(y = Y_1|z) \log p(y = Y_1|z)dz + \cdots \\
&+ K\int_{z \in Z} p(z)p(y = Y_K|z) \log p(y = Y_K|z)dz \\
&- \int_{z \in Z} p(z)p(y = Y_1|z) \log p(y = Y_1|z)dz - \cdots \\
&- \int_{z \in Z} p(z)p(y = Y_K|z) \log p(y = Y_K|z)dz \\
&- \int_{z \in Z} p(z)(1 - p(y = Y_1|z)) \log{(1 - p(y = Y_1|z))}dz - \cdots \\
&- \int_{z \in Z} p(z)(1 - p(y = Y_K|z)) \log{(1 - p(y = Y_K|z))}dz + C_4 \\
&= -K * H(Y_1|Z) - \cdots - K * H(Y_K|Z) \\
&- \int_{z \in Z} p(z)p(y = Y_1|z) \log p(y = Y_1|z)dz - \cdots \\
&- \int_{z \in Z} p(z)p(y = Y_K|z) \log p(y = Y_K|z)dz \\
&- \int_{z \in Z} p(z)(1 - p(y = Y_1|z)) \log{(1 - p(y = Y_1|z))}dz - \cdots \\
&- \int_{z \in Z} p(z)(1 - p(y = Y_K|z)) \log{(1 - p(y = Y_K|z))}dz + C_4 \\
&= -K * H(Y_1|Z) - \cdots - K * H(Y_K|Z) \\
&- 2\int_{z \in Z} p(z)p(y = Y_1|z) \log p(y = Y_1|z)dz - \cdots \\
&- 2\int_{z \in Z} p(z)p(y = Y_K|z) \log p(y = Y_K|z)dz + C_4 \\
&= -K * H(Y_1|Z) - \cdots - K * H(Y_K|Z) + 2H(Y_1|Z) + \cdots + 2H(Y_K|Z) + C_4 \\
&= (2-K)H(Y_1|Z) + \cdots + (2-K)H(Y_K|Z) + C_4
\end{aligned}
\tag{5.17}
$$

In this way, we can rewrite the objective function 5.13 into:

$$\begin{aligned}
&min\left\{D_{KL}(P(y \neq Y|Z) \parallel \frac{1-P(y=Y|Z)}{K-1})\right\} \\
&= min\left\{(2-K)H(Y_1|Z) + \cdots + (2-K)H(Y_K|Z) + C_4\right\} \\
&= max\left\{(K-2)(H(Y_1|Z) + \cdots + H(Y_K|Z)) + C_4\right\} \\
&\because K \geq 2 \qquad (5.18) \\
&\therefore min\left\{D_{KL}(P(y \neq Y|Z) \parallel \frac{1-P(y=Y|Z)}{K-1})\right\} \\
&= max\left\{(K-2)(H(Y_1|Z) + \cdots + H(Y_K|Z)) + C_4\right\} \\
&\propto max\left\{H(Y_1|Z) + \cdots + H(Y_K|Z)\right\} = max\left\{H(Y|Z)\right\}
\end{aligned}$$

In this way, we can get that $min\left\{D_{KL}(P(y \neq Y|Z) \parallel \frac{1-P(y=Y|Z)}{K-1})\right\}$ is equivalent to $max\left\{H(Y|Z)\right\}$. Now we need to prove why $max\left\{H(Y|Z)\right\}$ can $max\left\{H(S|Z)\right\}$.

$$H(Y|Z) = -\sum_{y\in Y}\int_z p(z)p(y|z)\log p(y|z)dz \qquad (5.19)$$

According to equation Lemma 1, $p(s|z) = f_{ys}(z)k_{ys}p(y|z)$, so:

$$\begin{aligned}
H(S|Z) &= -\sum_{s\in S}\int_z p(z)p(s|z)\log p(s|z)dz \\
&= -\sum_{s\in S}\sum_{y\in Y}\int_z p(z)f_{ys}(z)k_{ys}p(y|z)\log f_{ys}(z)k_{ys}p(y|z)dz \\
&= -\sum_{s\in S}\sum_{y\in Y}\int_z p(z)f_{ys}(z)k_{ys}p(y|z)\log f_{ys}(z)k_{ys}dz- \qquad (5.20) \\
&\sum_{s\in S}\sum_{y\in Y}\int_z p(z)f_{ys}(z)k_{ys}p(y|z)\log p(y|z)dz \\
&= D_1 + D_2
\end{aligned}$$

Same as the proof in equation 5.7, $D_1 = A_1 = C_2$. And According to the First Mean Value

Theorem [24], we can get:

$$\begin{aligned} D_2 &= -\sum_{s\in S}\sum_{y\in Y}\int_z p(z)f_{ys}(z)k_{ys}p(y|z)\log p(y|z)dz \\ &= \sum_{s\in S}\sum_{y\in Y}\int_z \frac{p(z,s)}{p(z,y)} - p(z)p(y|z)\log p(y|z)dz \\ &\because \frac{p(z,s)}{p(z,y)} continuous, -p(z)p(y|z)\log p(y|z) \geq 0 \\ \therefore D_2 &= \sum_{s\in S}\sum_{y\in Y}\int_z \frac{p(z,s)}{p(z,y)} - p(z)p(y|z)\log p(y|z)dz \\ &= M_1 \sum_{y\in Y}\int_z -p(z)p(y|z)\log p(y|z)dz \\ &= M_1 H(Y|Z) \end{aligned} \tag{5.21}$$

In this way,

$$\begin{aligned} H(S|Z) &= -\sum_{s\in S}\int_z p(z)p(s|z)\log p(s|z)dz \\ &= D_1 + D_2 \\ &= C_2 + M_1 H(Y|Z) \\ &\because \frac{p(z,s)}{p(z,y)} \geq 0, \therefore M_1 \geq 0 \\ &\therefore \frac{dH(S|Z)}{dHH(Y|Z)} = M_1 \geq 0 \end{aligned} \tag{5.22}$$

So, increasing $H(Y|Z)$ can increase $H(S|z)$, so that $max\{H(Y|Z)\}$ can $max\{H(S|Z)\}$. □

Theorem 3. First Mean Value Theorem [24]:

Let f and g be two continuous functions on $[a,b]$ and assume g is non-negative, thus there exists some constant M such that: $\int_a^b f(x)g(x)dx = M\int_a^b g(x)dx$.

Theorem 4. Chain rule of KL divergence [25]:

$D(p(x,y) \parallel q(x,y)) = D(p(x) \parallel q(x)) + D(p(y|x) \parallel q(y|x))$

Proof.

$$\begin{aligned} D(p(x,y) \parallel q(x,y)) &= \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{q(x,y)} \\ &= \sum_x \sum_y p(x,y) \log \frac{p(x)p(y|x)}{q(x)q(y|x)} \\ &= \sum_x \sum_y p(x,y) \log \frac{p(x)}{q(x)} + \sum_x \sum_y p(x,y) \log \frac{p(y|x)}{q(y|x)} \qquad (5.23) \\ &= D(p(x) \parallel q(x)) + \sum_x p(x) \sum_y p(y|x) \log \frac{p(y|x)}{q(y|x)} \\ &= D(p(x) \parallel q(x)) + D(p(y|x) \parallel q(y|x)) \end{aligned}$$

□

Theorem 5. Upper bound of $H(Z)$ when Z is a mixture of multivariate Gaussians [23]: $H(Z) \leq \sum\limits_{k=1}^{K} w_k(-\log w_k + \frac{d}{2}\ln 2\pi e + \frac{1}{2}\ln|\Sigma_k|)$

Proof.

$$\begin{aligned} H(Z) &= -\int_z \sum_{k=1}^{K} w_k \mathcal{N}(z;\mu_k,\Sigma_k) \log (\sum_{k=1}^{K} w_k \mathcal{N}(z;\mu_k,\Sigma_k))dz \\ &= -\sum_{k=1}^{K} w_k \int_z \mathcal{N}(z;\mu_k,\Sigma_k) \log (w_k \mathcal{N}(z;\mu_k,\Sigma_k)(1+\epsilon_k))dz \\ &= -\sum_{k=1}^{K} w_k \int_z \mathcal{N}(z;\mu_k,\Sigma_k)(\log w_k \mathcal{N}(z;\mu_k,\Sigma_k) + \log (1+\epsilon_k))dz \qquad (5.24) \\ \epsilon_k &= \frac{\sum\limits_{i \neq j=1}^{K} w_j \mathcal{N}(z;\mu_j,\Sigma_j)}{w_i \mathcal{N}(z;\mu_i,\Sigma_i)} \\ &\log (1+\epsilon_i) \geq 0 \end{aligned}$$

□

CHAPTER 6

ABLATION STUDY AND EXPERIMENTAL RESULTS

6.1 Ablation study

6.1.1 Ablation study of target loss $\mathcal{L}_T$ and inter-class entropy loss $\mathcal{L}_E$

In this section, the ablation study of target loss $\mathcal{L}_T$ and inter-class entropy loss $\mathcal{L}_E$ are introduced. In this study, $\mathcal{L} = \alpha\mathcal{L}_T + (1-\alpha)\mathcal{L}_E, \alpha = \{0.0, 0.1, \cdots, 1.0\}$. And the data used in this section is Gaussian distributed.

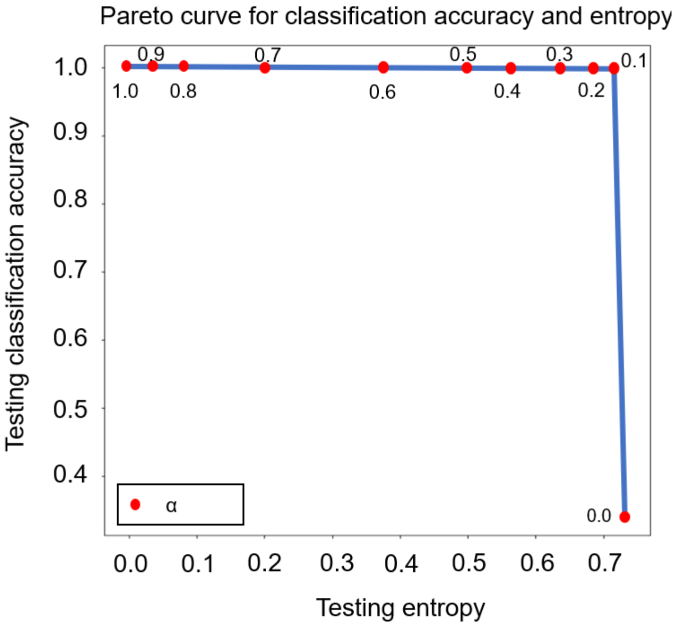


Figure 6.1: Curve of trade-off between utility and fairness vs α

In figure 6.1, the x-axis measures the fairness, which is denoted as the testing entropy. The larger the entropy, the better the fairness is achieved. And the y-axis measures the classification accuracy. The larger the classification accuracy value, the better the target task is learned. Each red dot denotes the value of the trade-off between the utility (classification accuracy) and the fairness (entropy) under different αs.

From figure 6.1, the result shows a Pareto Curve of utility and fairness, which means that by cleverly selecting the α, we can achieve a good trade-off between the utility and fairness. For example, in this scenario, when $\alpha = 0.1$, we can have the high classification accuracy (utility) and large entropy (fairness) at the same time.

We further display our results in a probability simplex to vividly visualize how target loss $\mathcal{L}_T$ and inter-class entropy loss $\mathcal{L}_E$ work in this ablation study.

Definition 4. Probability simplex: A probability simplex is a mathematical space where each point represents a probability distribution between a finite number of mutually exclusive events.

In this experiment, we use the 3-class Gaussian data. When $\mathcal{L} = \alpha\mathcal{L}_T + (1-\alpha)\mathcal{L}_E, \alpha = \{0.0, 0.1, \cdots, 1.0\}$ and α increases, we will focus more on the target prediction task instead of fairness, and vice versa. We can make the following observations from the results: when $\alpha = 0$, the probabilities for each class are all the same and all probabilities are uniformly distributed. The classification accuracy is 33%, which is randomly guess. While when we increase α, the points, which represents the probability triplets will move to the correct classification class areas and are less uniformly distributed. And when $\alpha = 1$, all probability points will be move to the correct classification class area.

We also show the entropy heat map on the probability simplex to mathematically prove the feasibility of our method. According to the figure 6.3, we can conclude that each area of the entropy value in the heat map on probability simplex after uniformity on the non-target classes will be larger than those before uniformity.

6.1.2 Ablation study of target loss $\mathcal{L}_T$ and intra-class entropy loss $\mathcal{L}_G$

In this section, the ablation study of target loss $\mathcal{L}_T$ and intra-class entropy loss $\mathcal{L}_G$ will be discussed. $\mathcal{L} = \alpha\mathcal{L}_T + (1-\alpha)\mathcal{L}_G + \epsilon\mathcal{L}_S, \alpha = \{0.0, 0.1, \cdots, 1.0\}, \epsilon\{0.00, 0.05\}$ in this study. The data used in this section is the 3-class Gaussian data. According to figure 6.4, we make

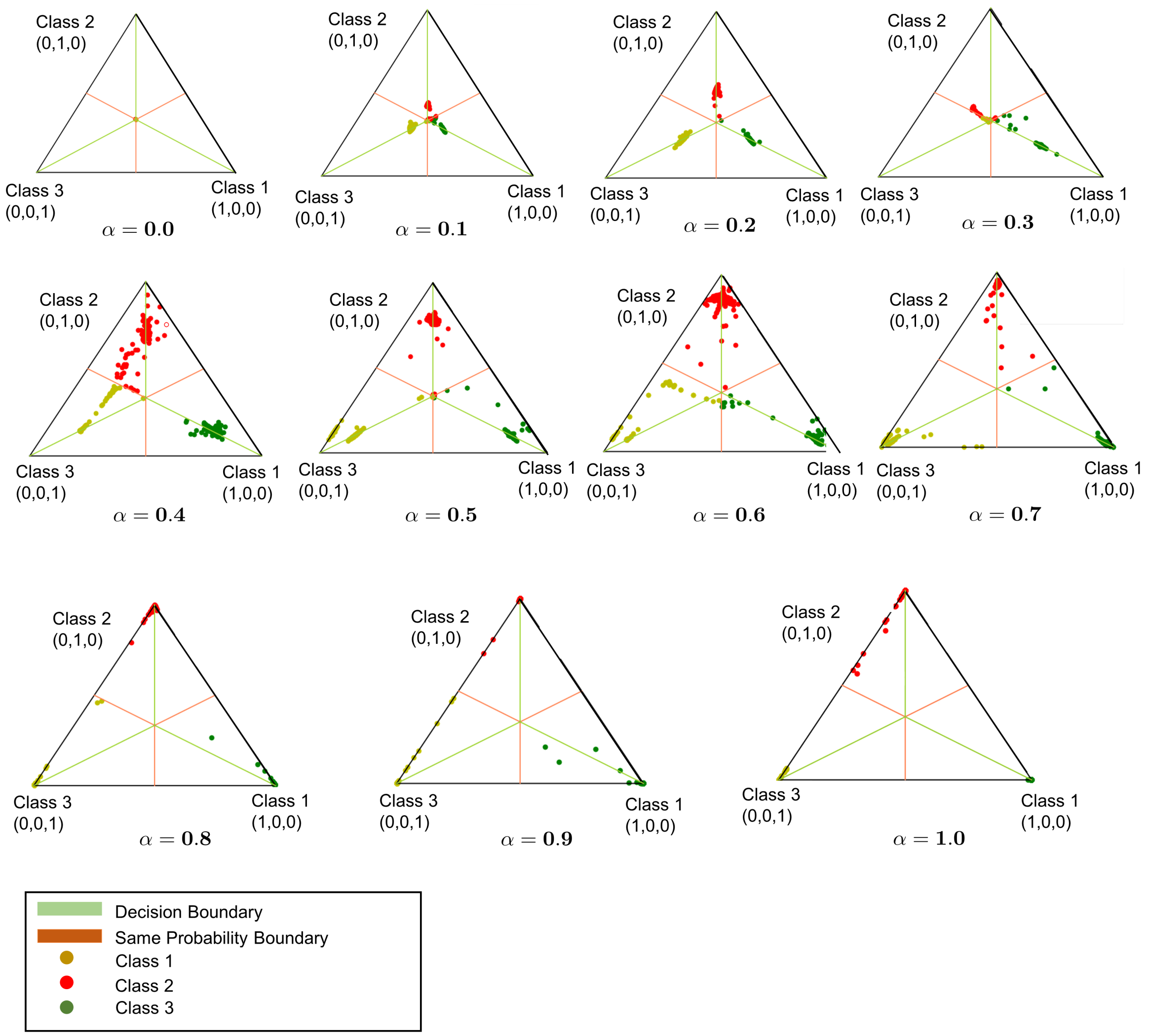


Figure 6.2: Probability simplex vs α

the following observations from the results that when $\alpha = 0.0$, we only care about whether the learnable hypothesis parameters and the calculated class-conditional representation parameters are the same or not, so all $\mathcal{N}_1, \mathcal{N}_2$ are overlapped, and the three classes can not be classified. While when increasing α, we are more focusing on the classification target task, so the class-conditional representations start to separate from each other. Until $\alpha = 1$, we only focus on the classification loss, so the 3-class representations are well separated, but the $\mathcal{N}_1, \mathcal{N}_2$ are not overlapped with each other. And the $\mathcal{N}_2$ returns to its initial parameters, that is why the 3 $\mathcal{N}_2$s are the same with $\mu_i = [0, 0]\,, \Sigma_i = [[1, 0], [0, 1]]\,, i = \{0, 1, 2\}$.

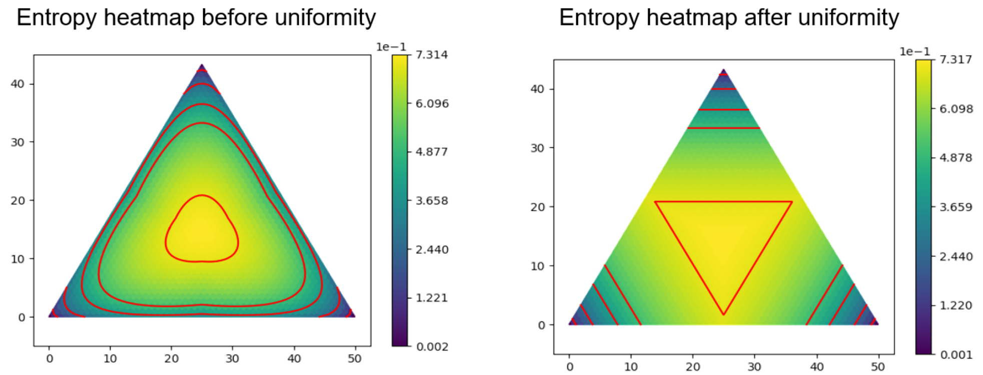
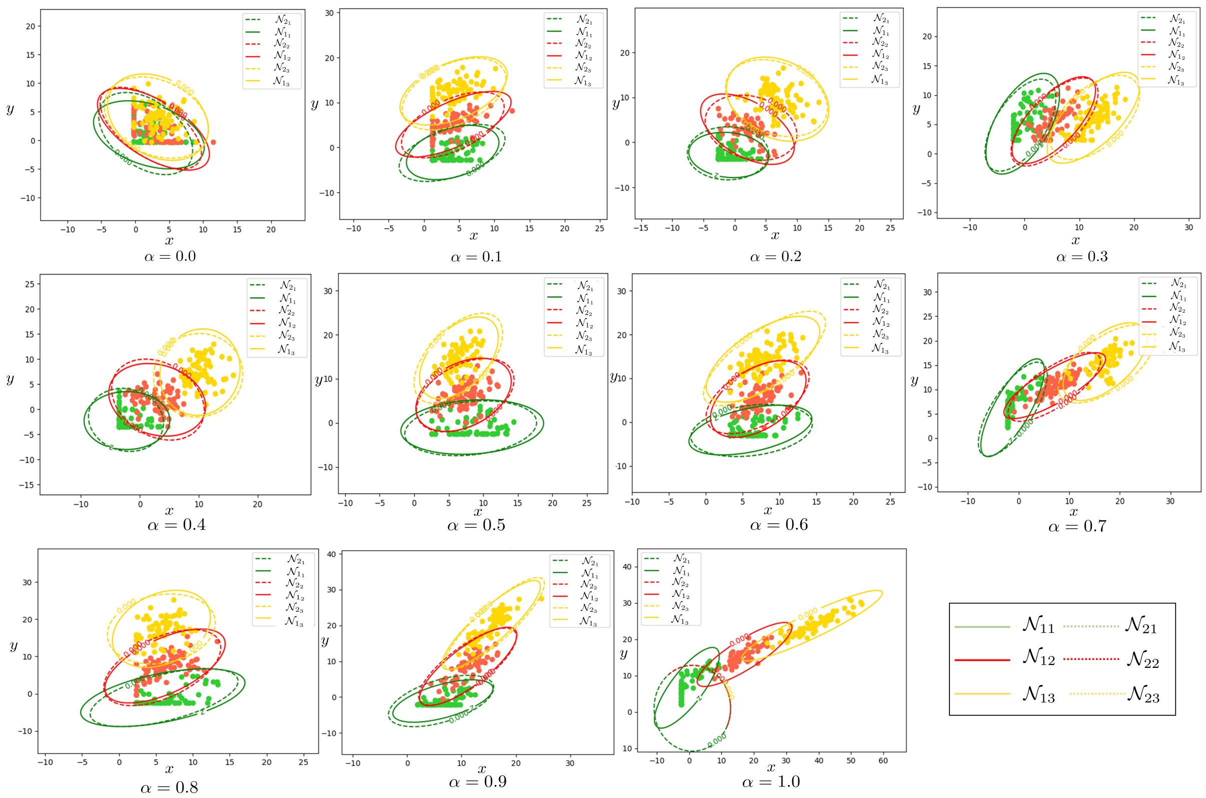


Figure 6.3: Entropy heat map on probability simplex



$\alpha = 0.0$ $\alpha = 0.1$ $\alpha = 0.2$ $\alpha = 0.3$

$\alpha = 0.4$ $\alpha = 0.5$ $\alpha = 0.6$ $\alpha = 0.7$

$\alpha = 0.8$ $\alpha = 0.9$ $\alpha = 1.0$

Figure 6.4: Class-conditional representations vs α

6.2 Experimental Results

6.2.1 Baselines and Implementation Details

Baselines: We consider baseline that is based on the Rawlsian Max-Min Fairness without using prior demographic information. Specifically, we consider two types of baselines: (1) the vanilla group-agnostic Baseline, which performs standard ERM with uniform weights, and (2) the ARL method [2].

Implementation Details: We use Optuna [27] to tune the hyperparameters. The target predictor is pre-trained for a binary-class classification task. And the initial class-conditional hypothesis mean-s and covariance-s are all set as the same.

6.2.2 Datasets and Evaluation Metrics

6.2.2.1 Datasets

We now demonstrate the effectiveness of our proposed MaxEnt approach through experiments over two real-world datasets well used in the fairness literature: (i) Adult UCI dataset [7]: income prediction (ii) COMPAS [6]: recidivism prediction.

1. COMPAS dataset [6]

   COMPAS is a landmark dataset to study algorithmic (un)fairness. This data was used to predict recidivism (whether a criminal will reoffend or not) in the USA. The tool was meant to overcome human biases and offer an algorithmic, fair solution to predict recidivism in a diverse population. However, the algorithm ended up propagating existing social biases and thus, offered an unfair algorithmic solution to the problem. In this dataset, a model to predict recidivism has already been fit and predicted probabilities and predicted status (yes/no) for recidivism have been concatenated to the original data. It has 11 features with 7215 data samples.The protected features are race and sex. The target label is recidivism prediction.

2. UCI Adult dataset [7]

   The Adult dataset is from the Census Bureau and the task is to predict whether a given adult makes more than $50,000 a year based attributes such as education, hours of work per week, etc. It has 14 features with 48842 data samples. The protected features are race and sex. The target label is income prediction.

6.2.2.2 Evaluation Metrics

Before using evaluation metrics, we first stratify the protected groups for evaluation. We use pairwise protected attributes to stratify groups. For example, in the case where we have sex and race as our sensitive attributes, we may have $\{\text{White}, \text{Black}\} \times \{\text{Male}, \text{Female}\}$ groups. So in the end, we will have 4 protected groups, which are subgroup 1-Black Male, subgroup 2-Black Female, subgroup 3-White Male and subgroup 4-White Female.

We use AUC as our evaluation metrics. We choose AUC (area under the ROC curve) as our evaluation metric as it is robust to class imbalance. Also, it encompasses both FPR and FNR, and is threshold agnostic. No prior sensitive information will be used during training but only used in evaluation. More specifically, we use a set of AUCs, which are AUC, AUC(min), AUC(macro-average) and AUC(minority). AUC is the AUC over all data. AUC(min) is the minimum AUC over all protected groups. AUC(macro-average) is the macro-average over all protected group AUCs, which is a weighted average. AUC(minority) is the AUC reported for the smallest protected group in the dataset.

Though AUC(minority) is used as one of the metrics, it does not fully measures the fairness. As higher AUC(minority) does not necessarily lead to better fairness. Instead, the gap between AUC(minority) and the AUC/AUC(macro-average) can measure the fairness. For example, in the case when $s \in S$ respectively denotes rich group and poor group in the world, rich group will definitely be the minority group, while they are not the disadvantage/weak group. But we still use this metric here as this is used in our baseline paper [2] for consistency.

6.2.3 Results

Table 6.1 reports results based on average performance across runs, with the best average performance highlighted in bold. According to the table 6.1, we make the following key observations:

MaxEnt improves worst-case performance: For both COMPAs and UCI Adult datasets, our proposed MaxEnt method outperforms the SOTA ARL method, and achieves best results

Table 6.1: Main results: MaxEnt vs ARL

Dataset	Method	AUC avg	AUC macro-avg	AUC min	AUC minority
COMPAS	Baseline	0.748	0.730	0.674	0.774
	ARL	0.743	0.727	0.658	0.785
	MaxEnt	**0.751**	**0.748**	**0.706**	0.730
UCI Adult	Baseline	0.898	0.891	0.867	0.875
	ARL	0.907	0.915	0.881	0.942
	MaxEnt	**0.919**	**0.923**	**0.906**	**0.943**

for AUC (min). Though in COMPAS dataset, MaxEnt method has lower AUC (minority) compared to ARL method, it does not degrade our method as AUC (minority) does not necessarily measure the fairness and is highly dependent on the dataset configureation and properties. As AUC (minority) is denoted as the AUC from the minimum-size protected group, whereas in many cases, minimum-size group does not belong to the weakness group, e.g. the small-size rich group vs the large-size poor group. So the other AUC metrics are more convincing in fairness measurement.

MaxEnt improves gap: We also conclude from the results that for both COMPAS and UCI Adult datasets, our proposed MaxEnt method outperforms the SOTA ARL method that gaps between each AUC metrics are smaller, which proves that our method mitigate the bias/differences between protected groups and can achieve the better fairness.

Utility-Fairness Trade-off: Furthermore, we observe that Min-Diff incurs a drop in overall AUC for UCI Adult dataset and COMPAS dataset. In contrast, as noted earlier MaxEnt in-fact shows an improvement in overall AUCs. This result shows that MaxEnt method achieves a better pareto allocation of overall and subgroup AUC performance. This is because the goal of MaxEnt is to realize a best trade-off between utility and fairness.

CHAPTER 7

CONCLUDING REMARKS

The main contribution of this paper is that it introduced an innovative Maximum Entropy Method for representation learning to mitigate sensitive information leakage from learned representations without prior demographic knowledge: 1) The fair representation learning can be learned without sensitive demographic information using Maximum Entropy Method under controlled correlations between target attributes and sensitive attributes, 2) and a good trade-off between utility and fairness can be reached using proposed Maximum Entropy Method. 3) The mathematics proof has also verified the availability of Maximum Entropy method. The main limitations are that: 1) More investigation and theoretic proofs are needed for complex sensitive-target attributes correlations. 2) More study on continuous sensitive/target attributes can be conducted. 3) Non-parametric method on representation learning might be employed instead of limiting the representation distribution only on the Gaussian distribution.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] T. Hashimoto, M. Srivastava, H. Namkoong, and P. Liang, “Fairness without demographics in repeated loss minimization,” in *International Conference on Machine Learning*, pp. 1929–1938, PMLR, 2018.

[2] P. Lahoti, A. Beutel, J. Chen, K. Lee, F. Prost, N. Thain, X. Wang, and E. H. Chi, “Fairness without demographics through adversarially reweighted learning,” *arXiv preprint arXiv:2006.13114*, 2020.

[3] N. L. Martinez, M. A. Bertran, A. Papadaki, M. Rodrigues, and G. Sapiro, “Blind pareto fairness and subgroup robustness,” in *International Conference on Machine Learning*, pp. 7492–7501, PMLR, 2021.

[4] Y. Zhang, “Assessing fair lending risks using race/ethnicity proxies,” *Management Science*, vol. 64, no. 1, pp. 178–197, 2018.

[5] H. Bahng, S. Chun, S. Yun, J. Choo, and S. J. Oh, “Learning de-biased representations with biased representations,” in *Proceedings of the 37th International Conference on Machine Learning* (H. D. III and A. Singh, eds.), vol. 119 of *Proceedings of Machine Learning Research*, pp. 528–539, PMLR, 13–18 Jul 2020.

[6] M. Barenstein, “Propublica’s compas data revisited,” 2019.

[7] D. Dua and C. Graff, “UCI machine learning repository,” 2017.

[8] M. N. Elliott, A. Fremont, P. A. Morrison, P. Pantoja, and N. Lurie, “A new method for estimating race/ethnicity and associated disparities where administrative records lack self-reported race/ethnicity,” *Health services research*, vol. 43, no. 5p1, pp. 1722–1736, 2008.

[9] M. Gupta, A. Cotter, M. M. Fard, and S. Wang, “Proxy fairness,” *arXiv preprint arXiv:1806.11212*, 2018.

[10] J. Chen, N. Kallus, X. Mao, G. Svacha, and M. Udell, “Fairness under unawareness,” *Proceedings of the Conference on Fairness, Accountability, and Transparency*, Jan 2019.

[11] N. Kallus, X. Mao, and A. Zhou, “Assessing algorithmic fairness with unobserved protected class using data combination,” *Management Science*, 2021.

[12] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, “Fairness through awareness,” in *Proceedings of the 3rd innovations in theoretical computer science conference*, pp. 214–226, 2012.

[13] P. Lahoti, K. P. Gummadi, and G. Weikum, “ifair: Learning individually fair data representations for algorithmic decision making,” in *2019 ieee 35th international conference on data engineering (icde)*, pp. 1334–1345, IEEE, 2019.

[14] P. Lahoti, K. P. Gummadi, and G. Weikum, "Operationalizing individual fairness with pairwise fair representations," *arXiv preprint arXiv:1907.01439*, 2019.

[15] H. Wang, N. Grgic-Hlaca, P. Lahoti, K. P. Gummadi, and A. Weller, "An empirical study on learning fairness metrics for compas data with human supervision," *arXiv preprint arXiv:1910.10255*, 2019.

[16] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, "Learning fair representations," in *International Conference on Machine Learning (ICML)*, 2013.

[17] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma, "Considerations on fairness-aware data mining," in *2012 IEEE 12th International Conference on Data Mining Workshops*, pp. 378–385, IEEE, 2012.

[18] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[19] T. Kamishima, S. Akaho, and J. Sakuma, "Fairness-aware learning through regularization approach," in *2011 IEEE 11th International Conference on Data Mining Workshops*, pp. 643–650, IEEE, 2011.

[20] J. Rawls, *Justice as fairness: A restatement.* Harvard University Press, 2001.

[21] K. Conrad, "Probability distributions and maximum entropy," 2005.

[22] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[23] S. Z. Li and A. Jain, eds., *LDA (Linear Discriminant Analysis)*, pp. 899–899. Boston, MA: Springer US, 2009.

[24] T. Riedel and P. K. Sahoo, *Mean value theorems and functional equations.* Singapore, Singapore: World Scientific Publishing, Oct. 1998.

[25] J. M. Joyce, *Kullback-Leibler Divergence*, pp. 720–722. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[26] M. F. Huber, T. Bailey, H. Durrant-Whyte, and U. D. Hanebeck, "On entropy approximation for gaussian mixture random vectors," in *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 181–188, 2008.

[27] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," 2019.