

GroupMixNorm Layer for Learning Fair Models

Anubha Pandey^[0000–0002–4695–0947], Aditi Rai^[0009–0009–9298–7861], Maneet Singh, Deepak Bhatt^[0000–0003–3694–1315], and Tanmoy Bhowmik

AI Garage, Mastercard, India
 {anubha.pandey, aditi.rai, maneet.singh, deepak.bhatt}@mastercard.com,
 tantanmoy@gmail.com

Abstract. Recent research has identified discriminatory behavior of automated prediction algorithms towards groups identified on specific protected attributes (e.g., gender, ethnicity, age group, etc.). When deployed in real-world scenarios, such techniques may demonstrate biased predictions resulting in unfair outcomes. Recent literature has witnessed algorithms for mitigating such biased behavior mostly by adding convex surrogates of fairness metrics such as demographic parity or equalized odds in the loss function, which are often not easy to estimate. This research proposes a novel in-processing based *GroupMixNorm* layer for mitigating bias from deep learning models. The GroupMixNorm layer probabilistically mixes group-level feature statistics of samples across different groups based on the protected attribute. The proposed method improves upon several fairness metrics with minimal impact on overall accuracy. Analysis on benchmark tabular and image datasets demonstrates the efficacy of the proposed method in achieving state-of-the-art performance. Further, the experimental analysis also suggests the robustness of the GroupMixNorm layer against new protected attributes during inference and its utility in eliminating bias from a pre-trained network.

Keywords: Deep Learning · Ethics and fairness · Bias Mitigation

1 Introduction

Most AI algorithms process large quantities of data to identify patterns useful for accurate predictions. Such pipelines are mostly automated in nature without any human intervention, along with large data processing, high efficiency, and high accuracy. Despite the benefits of automated processing, current AI systems are marred with the challenge of biased predictions resulting in unfavourable outcomes. One of the most infamous examples of such behavior is that of an AI-based recruitment tool¹, which disfavoured applications from women because it was trained on resumes from the mostly male workforce. In order to rectify such biases and support advancement in society, we need models that generate fair results without any discrimination towards certain individuals or groups. To this effect, this research proposes a novel *GroupMixNorm* layer for learning an unbiased model for ensuring fair outcomes across different groups.

¹ <https://tinyurl.com/5apv7xeu>

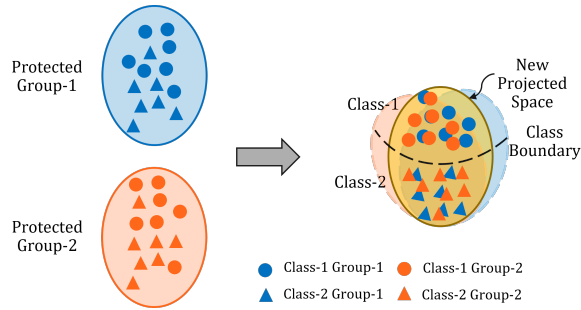


Fig. 1: The proposed GroupMixNorm layer projects the features of different classes and protected attributes onto a space which minimizes the distinction between the protected attributes, thus promoting a fairer classification model.

In the literature, research has focused on achieving fairness by introducing techniques at the pre-processing stage (transforming the input before feeding to the classification model) or the post-processing stage (transforming the output produced by the classification model). It is our hypothesis that these methods may not result in optimal accuracy, since they treat the classifier as a black box and focus on removing bias from the input representations or the output predictions only. Different from the above, *in-processing* techniques focus on learning bias-invariant models by incorporating additional constraints during training, thus resulting in more effective models [4]. Existing in-processing techniques mostly aim to solve a constraint optimization problem to ensure fairness [20,21,22] by introducing a penalty term in the loss function corresponding to the convex surrogates of the fairness objective like *demographic parity* or *equalized odds*. However, as observed in literature, it is challenging to formulate surrogates for different fairness constraints that is a reasonable estimate of the original [16].

In this research, we formulate the problem of bias mitigation as distribution alignment of several groups of the protected attribute (Fig. 1). The proposed *GroupMixNorm* layer is applied at the in-processing stage which promotes the model to learn unbiased features for classification. The formulation is motivated by the observation that Deep Learning based algorithms tend to explore the difference in the distribution among the groups of the protected attributes (e.g., male and female with similar features like age and education may have different salaries, thus resulting in different distributions) to lift the overall performance. The GroupMixNorm layer mixes the group-level feature statistics and transforms all the features in a training batch based on the interpolated group statistics. This enables the classifier to learn features invariant to the protected attribute. Further, transforming the data towards the interpolated groups regularizes the classifier and improves the generalizability at inference. Key highlights of this research are as follows:

- This research proposes a novel *GroupMixNorm* layer for learning fairer classification models. The proposed layer is applied at the architectural level

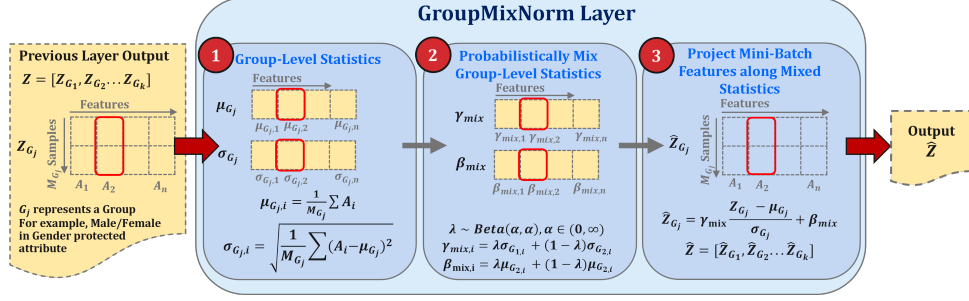


Fig. 2: The GroupMixNorm layer takes as input the previous layer’s output (\mathbf{Z}) along with each sample’s protected attribute. Group-level statistics are computed, followed by the probabilistic mixing and projection of the mini-batch features along the mixed statistics to obtain new features ($\hat{\mathbf{Z}}$).

and is an in-processing technique that focuses on distribution alignment of different groups during model training.

- GroupMixNorm operates at the feature level, thus making it flexible to be placed across various layers of a neural network-based model and fits well into the mini-batch gradient-based training. Experimental analysis suggests that with limited data, GroupMixNorm can be applied to mitigate the existing bias in classifiers as well, thus avoiding the need for re-training from scratch.
- The GroupMixNorm layer produces fairer results when evaluated for new groups at test time as well. We believe that the GroupMixNorm layer makes the model robust against distribution changes across sensitive groups, thus being able to generalize well for unseen groups at test time.
- The efficacy of the proposed approach has been demonstrated on different datasets (structured and unstructured), where it achieves improved performance while achieving multiple fairness constraints such as demographic parity, equal opportunity, and equalized odds simultaneously. For example, on the UCI Adult Income dataset [9], GroupMixNorm achieves an average precision of 0.77, while maintaining different fairness metrics below 0.03.

2 Related Work

Group fairness can be ensured in a machine learning system via *pre-processing*, *in-processing*, and *post-processing*. Pre-processing and post-processing methods consider the classifier as a black-box model, and try to mitigate bias from the input features or the classifier’s prediction. On the other hand, *in-processing* based bias mitigation techniques solve the constraint optimization problem for different fairness objectives. To ensure independence between the predictions and sensitive attributes, Woodworth et al. [20] regularize the covariance between them. Zafar et al. [21] minimize the disparity between the sensitive groups by regularizing the decision boundary of the classifier. Game theory based approaches [1,7] provide analytical solutions and theoretical guarantees for generalizability in

fair classifier but are limited by the scalability factor. Recent techniques [13,22] introduce an adversary network additional to the predictor network that predicts the sensitive label based on the classifier’s output, while other algorithms [2,3,17,24] learn unbiased representations through invariant risk minimization and attention-based feature learning. Research has also focused on eliminating superficial correlations and paying more attention on task related causal features [12,14]. Recently, Cheng et al. [5] utilize contrastive learning to minimize the correlation between sentence representations and biasing words, while mixup [19,23] techniques have proved to be effective in bias mitigation. For example, Chuang et al. [6] utilize mixup as a data augmentation strategy to improve the generalizability of the model while optimizing the fairness constraints, and Du et al. [8] utilize mixup for feature neutralization to remove the correlation between the sensitive information and class labels from the encoder feature.

Instead of focusing on optimizing surrogates of the fairness metrics, this research proposes a novel GroupMixNorm layer which operates at the *architectural* level of the classifier. GroupMixNorm focuses on learning unbiased representations which results in satisfying several fairness constraints across groups.

3 Proposed GroupMixNorm Layer

As discussed before, recent research has observed that deep learning models often tend to learn group-specific characteristics, making it easier to obtain a higher performance on the underlying classification task. As an ancillary effect, the learned group-specific features often also result in discriminative behavior towards specific groups based on the protected attribute. For example, a recruitment tool may learn features based on the gender of the applicant, resulting in unintended discrimination towards applicants from the under-represented group. In order to address the above limitation, the GroupMixNorm layer focuses on eliminating the difference between the group statistics during training.

As part of the GroupMixNorm layer, we normalize each group of a protected attribute in a batch separately to collect group specific statistics (i.e. for the gender attribute, normalize all male samples and female samples in a batch separately) and further take a probabilistic convex combination between the group-level statistics and apply across all the samples in a batch. This process ensures that any protected group related diversity is removed from the internal representation of a neural network and doesn’t allow the network to explore this information to lift the overall performance. The introduction of additional inductive bias in the network structure enforces it to learn invariant features pertaining to the protected attributes while training the network.

The GroupMixNorm layer is implemented as a plug-and-play module. It can be inserted between the fully connected layers of a neural network-based classifier during training (Algorithm 1). Let X , Y , and S be the input features, class labels, and protected attribute labels in a training batch, respectively. As illustrated in Fig. 2, let Z be an n dimensional representation obtained from the previous layer and A_i represent the feature along dimension i . We identify the groups G_j in a

Algorithm 1 GroupMixNorm Layer

Input: Z : Learned representation of the input batch obtained from the previous layer
 α, β : Hyper-parameters for the Beta distribution (default: 0.1)

Output: \hat{Z} : Transformed samples after the GroupMixNorm layer

- 1: **if** not in training mode **then**
- 2: **return** Z
- 3: **end if**
- 4: Compute μ_{G_j} and σ_{G_j} for a group G_j in a protected attribute
- 5: Sample mixing coefficient $\lambda \sim \text{Beta}(\alpha, \alpha)$
- 6: Compute γ_{mix} and β_{mix} as shown in Eq. 2
- 7: Normalize and transform all samples in a batch to compute \hat{Z}_{G_j} as shown in Eq. 3
- 8: $\hat{Z} = \{\hat{Z}_{G_j}\}_{j=1}^K$, where K is the number of groups identified in a protected attribute
- 9: **return** \hat{Z}

batch based on the protected attribute labels S , and calculate their respective mean ($\mu_{G_j,i}$) and variance ($\sigma_{G_j,i}$) along each dimension (step-1 of Fig. 2). Next we calculate the weighted average of mean $\gamma_{mix,i}$ and variance $\beta_{mix,i}$ along each dimension (Eq. 1), followed by concatenation to create a single vector (Eq. 2). As we mix statistics of two groups at a time, the mixing coefficient λ is sampled from a symmetric Beta distribution $\text{Beta}(\alpha, \alpha)$, for $\alpha \in (0, \infty)$. The hyper-parameters α controls the strength of interpolation.

Finally, we normalize all the samples by applying the calculated γ_{mix} and β_{mix} to each sample as shown in Eq. 3. For the ease of notation, we have considered two groups i.e. binary protected attributes. However, the proposed solution can easily be applied to non-binary protected attributes as well.

$$\gamma_{mix,i} = \lambda \sigma_{G_1,i} + (1 - \lambda) \sigma_{G_2,i}; \quad \beta_{mix,i} = \lambda \mu_{G_1,i} + (1 - \lambda) \mu_{G_2,i}; \quad (1)$$

$$\gamma_{mix} = [\gamma_{mix,1}, \dots, \gamma_{mix,n}]; \quad \beta_{mix} = [\beta_{mix,1}, \dots, \beta_{mix,n}] \quad (2)$$

$$\hat{Z}_{G_j} = \gamma_{mix} \frac{(Z_{G_j} - \mu_{G_j})}{\sigma_{G_j}} + \beta_{mix} \quad (3)$$

The updated features $\hat{Z} = [\hat{Z}_{G_1}, \hat{Z}_{G_2}]$ are then provided as input to the following layer of the neural network for further processing. The process of mixing group level statistics in a GroupMixNorm layer occurs in the feature space and has no learnable parameters. The GroupMixNorm layer is easy to implement and fits perfectly into mini-batch training. Further, it is turned off during inference, thus eliminating the need for protected attributes during inference. The training procedure of GroupMixNorm layer is shown in Algorithm 1.

4 Datasets and Experimental Details

The GroupMixNorm layer has been evaluated on two datasets with different fairness evaluation metrics and compared with state-of-the-art techniques. Details regarding the dataset protocols are as follows:

- **UCI Adult Dataset** [9] contains 50,000 samples with 14 attributes to describe each data point (individual) (e.g., gender, education level, age, etc.) from the 1994 US Census. The classification task is to predict the income of an individual. It’s a binary classification task, where class 1 represents salary $\geq 50K$ and class 0 represents salary $< 50K$. We select gender as the protected attribute for the fairness evaluation. The dataset is imbalanced such that only 24% of the samples belong to class 1, with only 15.13% female samples.
- **CelebA Dataset** [15] contains 200,000 celebrity faces with 40 binary attributes associated with each image. Following the literature [6,8], we select gender as the protected attribute and wavy hair attribute for the binary classification task. The dataset has 18.36% male samples as compared to female samples in the positive class.

4.1 Fairness Evaluation Metrics

The most widely used fairness metrics [18] are: Demographic Parity, Equal Opportunity, and Equalized Odds [10]. The metrics are elaborated in detail below, where Y (\hat{Y}) is actual (predicted) class label and S is protected attribute:

- **Demographic Parity Difference (DP)** suggests that the probability of favourable outcomes should be same for all the subgroups:

$$DPD = |P[\hat{Y} = 1|S = 1] - P[\hat{Y} = 1|S \neq 1]| \quad (4)$$

- **Equality of Opportunity Difference (EOP)** emphasises that there should be equal opportunities for all the subgroups having positive outcomes to have positive prediction i.e. true positive rates for all the groups should be same:

$$EOP = |P[\hat{Y} = 1|S = 1, Y = 1] - P[\hat{Y} = 1|S \neq 1, Y = 1]| \quad (5)$$

- **Equalized Odds Difference (EOD)** focuses on equalizing false positive rates along with the same true positive rates for all the subgroups:

$$EOD = |P[\hat{Y} = 1|S = 1, Y = 1] - P[\hat{Y} = 1|S \neq 1, Y = 1]| + |P[\hat{Y} = 1|S = 1, Y = 0] - P[\hat{Y} = 1|S \neq 1, Y = 0]| \quad (6)$$

For a fair algorithm, DP, EO and EOD values must be closer to 0.

4.2 Implementation Details

The GroupMixNorm layer has been implemented in the PyTorch framework on Ubuntu 16.04.7 OS with the Nvidia GeForce GTX 1080Ti GPU. For a fair comparison with existing literature, we have followed the same dataset pre-processing and protocols as the fair mixup approach [6]. For the Adult dataset, we use four fully connected layers with hidden dimension 50. Each layer except the last output layer is followed by SiLU activation and the proposed GroupMixNorm layer.

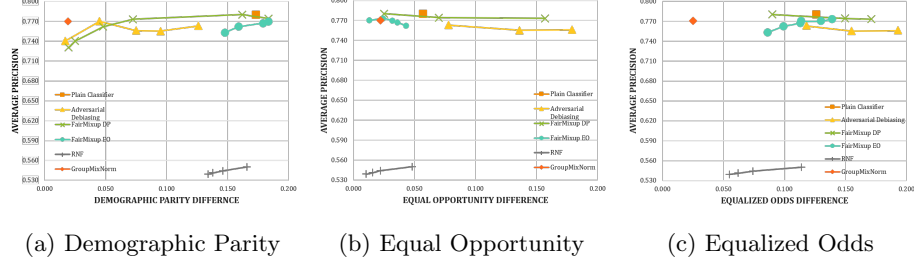


Fig. 3: **Fairness-AP trade-off curves** on the **Adult** dataset, where GroupMixNorm demonstrates improved performance. Results are obtained by varying the trade-off parameter as suggested in their respective publications: Adversarial Debiasing: [0.01 ~ 1.0], Fair Mixup DP: [0.1 ~ 0.7], Fair Mixup EO: [0.5 ~ 5.0], and RNF: [0.05, 0.015, 0.025, 0.035]. For a fair algorithm, it is desirable to have the AP closer to 1, and the fairness metrics (DP, EO, EOD) closer to 0.

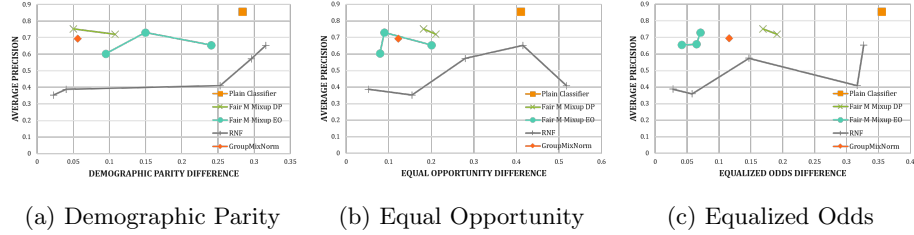


Fig. 4: **Fairness-AP trade-off curves** of GroupMixNorm layer and other comparative algorithms on the **CelebA** dataset, where the proposed approach achieves state-of-the-art performance. Results are obtained by varying the trade-off parameter as suggested in their respective publications: Fair M Mixup DP: [25, 50], Fair M Mixup EO: [1, 10, 50], RNF: [0.1, 0.5, 1, 5, 10].

The model is trained for 10 epochs with a 1000 batch size. For each epoch, the dataset is randomly split into 60-20-20 split of train, val, and test set, respectively. We select the best-performing model on the validation set across 10 independent runs and report the mean Average Precision and fairness metrics defined above. For the CelebA dataset, we use the ResNet-18 [11] model for feature extraction followed by two fully connected layers for the classification task. We apply SiLU activation and GroupMixNorm layer between the two FC layers. We use the original split of the dataset and train the model for 100 epochs with 128 batch size. Both the models are trained with the Adam optimizer with learning rate $1e-4$. In all experiments, mixing coefficient λ (Eqs. 1 and 2) is randomly sampled from $Beta(\alpha, \alpha)$. The value of α is empirically set to 0.1.

5 Results and Analysis

Figures 3–6 and Table 1 present the results and analysis of the GroupMixNorm layer and comparison with the state-of-the-art in-processing bias mitigation techniques. Detailed analysis is given in the following subsections:

5.1 Comparison with State-of-the-art Algorithms

Since the GroupMixNorm layer focuses on mitigating bias during the training process, comparison has been performed with algorithms that optimize fairness constraints during training: (i) Adversarial Debiasing [22], (ii) Fair Mixup: Fairness via Interpolation (Fair Mixup) [6], (iii) Fairness via Representation Neutralization (RNF) [8], and (iv) plain classifier. Fair Mixup uses two separate regularizing terms for optimizing the fairness metrics of Demographic Parity (DP) and Equal Opportunity (EO), and thus can solve for either DP or EO at a time. In this paper, we refer to these two variants of Fair Mixup as *Fair Mixup DP* and *Fair Mixup EO*. To calculate the DP, Chuang *et al.* [6] have computed the difference between the predicted probability across the protected groups. Similarly, for EO, Chuang *et al.* [6] compute class-wise difference between the predicted probability across protected groups. As part of this research, we have used the actual definitions of EO and DP for computing the fairness metrics (Eqs. 4–5). Parallely, the Representation Neutralization (RNF) technique [8] has shown the bias mitigation performance via two variants: (i) in model-1, proxy labels are generated for the protected attribute, while (ii) in model-2, ground-truth protected attribute labels are used. As part of this research, we have compared our results with their second variant (model-2), referred to as *RNF*.

For a fair comparison, we evaluate all the models under the same setting. Techniques such as Adversarial Debiasing, Fair Mixup, and RNF introduce a regularization term in the loss function to improve fairness via a hyper-parameter α that controls the trade-off between the average precision (AP) and fairness metrics (DP, EO, and EOD). We have reported the results on varying values of α as suggested in their respective papers. In our case, the GroupMixNorm layer is proposed towards architecture design and not the loss function, thus there is no such trade-off. Performance analysis on different datasets is as follows:

(a) Comparison on the UCI Adult Income Dataset: Fig. 3 shows the performance comparison on the UCI Adult dataset, where the GroupMixNorm layer produces fairer results as compared to other techniques across all fairness metrics (DP, EO, EOD) with minimal impact on average precision. Since Fair Mixup solves separate constraint optimizations to achieve lower DP and EO, it minimizes either DP or EO at a time. In terms of the fairness metrics, RNF produces fair results, however the average precision is relatively lower, thus making it unsuitable for the classification task.

(b) Comparison on the CelebA Dataset (Fig. 4): Consistent with the published manuscript [6], for Fair Mixup, comparison has been performed with the combination of manifold mixup [19] (Fair M Mixup DP and Fair M Mixup EO). Similar to the previous experiments, it is observed that either Fair M

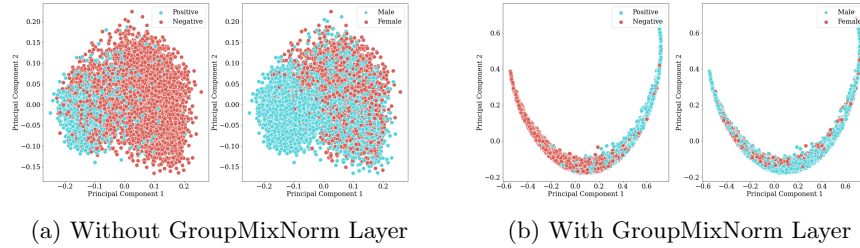


Fig. 5: PCA visualizations of the features for the MLP classifiers trained without and with GroupMixNorm. The left plots show the class distribution, and the right plots show the gender distribution (protected attribute). The model trained with GroupMixNorm demonstrates minimal distinction on the gender attribute.

Table 1: Cosine similarity between the learned weight parameters of C_{sens} and C_{cls} linear classifiers (the former is trained for predicting the sensitive attribute, while the later is trained for class label prediction). A lower score represents less biased models since lesser similarity is observed between the weight parameters.

Method	Cosine Similarity
Plain Classifier	0.205
RNF	0.075-0.2
GroupMixNorm	0.06

Mixup DP or Fair M Mixup EO achieves optimal performance at a time. Further, the RNF model produces fair results across fairness metrics, however achieves lower average precision. GroupMixNorm achieves comparable performance to the best performing model across all the metrics, while maintaining a high average precision, thus suggesting high utility for real-world applications.

5.2 Learned Representation Analysis

Experiments have been performed for (a) feature visualization and (b) auxiliary prediction task for understanding feature quality. The key findings are as follows: **(a) Feature Visualization:** Fig. 5 presents the 2D projections obtained by using the sigmoid kernel Principal Component Analysis (PCA). Fig. 5a presents the features learned by a biased MLP classifier (trained without GroupMixNorm layer), where the features appear both class and gender (protected attribute) discriminative. There is an overlap of male samples with the positive class samples, both lying majorly in the lower left side of the distribution. On the other hand, Fig. 5b shows features learned with the GroupMixNorm layer appear to be class discriminative, while not being gender discriminative. Further, both male and female samples are evenly distributed, thus preventing the model to get biased against a particular sensitive group.

(b) Auxiliary Prediction Task: Similar to Du *et al.* [8], we use an auxiliary prediction task to analyze the quality of the learned features. The objective is to analyze how well the model can reduce the correlation between the class labels

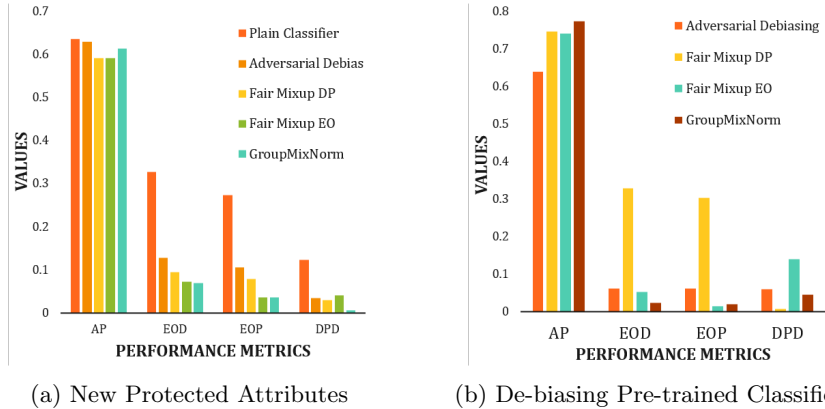


Fig. 6: Average precision and fairness metrics obtained by different techniques (a) when evaluated on new protected attributes and (b) for de-biasing a pre-trained classifier with limited training data. Experiments have been performed on the Adult Income dataset with race and gender as the protected attribute, respectively. GroupMixNorm presents improved performance across metrics.

and the sensitive attributes. To this effect, we train two linear classifiers C_{sens} and C_{cls} that take the representation vector as input and predicts class labels and sensitive attributes, respectively. Next, we compare the learned weight matrix of C_{sens} and C_{cls} using cosine similarity. A higher similarity would signify similar weights and thus higher correlation between the two tasks. Table 1 shows that our model has the least cosine similarity indicating that the classifier focuses more on task relevant information than sensitive information. It is important to note that the cosine similarity for the RNF model varies from 0.2 to 0.075, based on the fairness-accuracy trade-off parameter, while the GroupMixNorm layer based model achieves a cosine similarity of 0.06 only.

5.3 Generalizability to New Protected Groups

With time, as the data evolves, new sensitive groups often get introduced. For example, gender attribute values may change from binary to non-binary. A robust classification model must remain unbiased even with the introduction of additional sensitive groups during inference. In order to simulate this setup, the proposed solution was evaluated for new groups at test time without any re-training. Experiments were performed on the Adult Income dataset where data pertaining to two races (White and Black) was used for training, while the data from White, Black, and Others (Asian-Pac-Islander, Amer-Indian-Eskimo, Other) racial groups were used for testing. Fig. 6a presents the performance of the GroupMixNorm layer along with other comparative techniques, where GroupMixNorm is able to generalize well to unseen groups during inference by obtaining lower fairness metrics and a higher average precision.

5.4 Debias Pre-trained Model with Limited Data

Experiments have also been performed to analyze the effectiveness of the proposed GroupMixNorm layer to mitigate bias from a pre-trained biased classifier. We train an MLP classifier on the training partition of the Adult Income dataset, without the GroupMixNorm layer, and later fine-tune the model after plugging the proposed layer on the validation set. The validation set consists of only 20% samples of the entire dataset. For other techniques, we fine-tune the pre-trained biased classifier on the validation set with the respective methods. We evaluate the model for fairness on the Adult dataset with gender as the protected attribute. Fig. 6b presents the results obtained by the proposed GroupMixNorm layer as well as other comparative techniques. It can be observed that the proposed solution produces fairer results as compared to other algorithms across the different fairness metrics, while achieving the highest average precision. The experiment suggests that even with a small training set, the proposed GroupMixNorm can aid in eliminating bias from a pre-trained network.

6 Conclusion and Future Work

Learning bias-invariant models are the need of the hour for the research community. While existing research has focused on proposing novel solutions for learning unbiased classifiers, most of the techniques incorporate an additional term in the loss function for modeling the model fairness. We believe that it is often difficult to extrapolate the learnings of such an optimization function to the test set, especially under the challenging scenario of new protected attributes during evaluation. To this effect, this research proposes a novel *GroupMixNorm* layer, which promotes learning fairer models at the architectural level. GroupMixNorm is a distribution alignment strategy operating across the different protected groups, enabling attribute-invariant feature learning. Across multiple experiments, GroupMixNorm demonstrates improved fairness metrics while maintaining higher average precision levels, as compared to the state-of-the-art algorithms. Further analysis suggests high model generalizability to new protected attributes during evaluation, possibly due to the transformation of samples to interpolated groups resulting in model regularization during training. As an extension of this research, future research directions include studying the impact of GroupMixNorm on different convolution layers and extending the scope to evaluation on NLP datasets and tasks.

References

1. Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., Wallach, H.M.: A reductions approach to fair classification. In: ICML. vol. 80, pp. 60–69 (2018)
2. Ahuja, K., Shanmugam, K., Varshney, K.R., Dhurandhar, A.: Invariant risk minimization games. In: ICML. vol. 119, pp. 145–155 (2020)
3. Arjovsky, M., Bottou, L., Gulrajani, I., Lopez-Paz, D.: Invariant risk minimization. CoRR **abs/1907.02893** (2019)

4. Barocas, S., Hardt, M., Narayanan, A.: Fairness and Machine Learning. fairml-book.org (2019), <http://www.fairmlbook.org>
5. Cheng, P., Hao, W., Yuan, S., Si, S., Carin, L.: FairFil: Contrastive neural debiasing method for pretrained text encoders. In: ICLR (2021)
6. Chuang, C., Mroueh, Y.: Fair mixup: Fairness via interpolation. In: ICLR. Virtual Event, Austria, May 3-7, 2021 (2021)
7. Cotter, A. et al.: Training well-generalizing classifiers for fairness metrics and other data-dependent constraints. In: ICML. vol. 97, pp. 1397–1405 (2019)
8. Du, M., Mukherjee, S., Wang, G., Tang, R., Awadallah, A., Hu, X.: Fairness via representation neutralization. NeurIPS **34** (2021)
9. Dua, D., Graff, C.: UCI machine learning repository (2017), <http://archive.ics.uci.edu/ml>
10. Hardt, M., Price, E., Srebro, N.: Equality of opportunity in supervised learning. In: NeurIPS. pp. 3315–3323 (2016)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE CVPR. pp. 770–778 (2016)
12. Kilbertus, N. et al.: Avoiding discrimination through causal reasoning. In: NeurIPS. pp. 656–666 (2017)
13. Kim, B., Kim, H., Kim, K., Kim, S., Kim, J.: Learning not to learn: Training deep neural networks with biased data. In: IEEE CVPR. pp. 9012–9020 (2019)
14. Kusner, M.J., Loftus, J.R., Russell, C., Silva, R.: Counterfactual fairness. In: NeurIPS. pp. 4066–4076 (2017)
15. Liu, Z., Luo, P., Wang, X., Tang, X.: Large-scale celebfaces attributes (celeba) dataset. Retrieved August 15(2018), 11 (2018)
16. Manisha, P., Gujar, S.: FNNC: Achieving fairness through neural networks. In: IJCAI. pp. 2277–2283 (2020)
17. Singh, K.K., Mahajan, D., Grauman, K., Lee, Y.J., Feiszli, M., Ghadiyaram, D.: Don’t judge an object by its context: Learning to overcome contextual bias. In: IEEE/CVF CVPR. pp. 11067–11075 (2020)
18. Verma, S., Rubin, J.: Fairness definitions explained. In: International Workshop on Software Fairness. pp. 1–7 (2018)
19. Verma, V. et al.: Manifold mixup: Better representations by interpolating hidden states. In: ICML. vol. 97, pp. 6438–6447 (2019)
20. Woodworth, B.E., Gunasekar, S., Ohannessian, M.I., Srebro, N.: Learning non-discriminatory predictors. In: COLT. vol. 65, pp. 1920–1953 (2017)
21. Zafar, M.B., Valera, I., Gomez-Rodriguez, M., Gummadi, K.P.: Fairness constraints: Mechanisms for fair classification. In: AISTat. vol. 54, pp. 962–970 (2017)
22. Zhang, B.H., Lemoine, B., Mitchell, M.: Mitigating unwanted biases with adversarial learning. In: AAAI/ACM AIES. pp. 335–340 (2018)
23. Zhang, H., Cissé, M., Dauphin, Y.N., Lopez-Paz, D.: Mixup: Beyond empirical risk minimization. In: ICLR (2018)
24. Zunino, A. et al.: Explainable deep classification models for domain generalization. In: IEEE CVPRW. pp. 3233–3242 (2021)