Let Samples Speak: Mitigating Spurious Correlation by Exploiting the Clusterness of Samples

Supplementary Material

In Section 1, we detailed the experimental setups, datasets, and implementation frameworks utilized in our study, including preprocessing pipelines, and training method. Additionally, we present Class Activation Maps (CAMs) visualizations in Section 2 to validate the mitigation of spurious correlations. Furthermore, we provide theoretical proofs of theorems underpinning our method in Section 3.

1. Experiment Details

1.1. Datasets

- Waterbirds [7] is a dataset consisting of 4,795 training images combining bird photos from the CUB dataset [9] with background images from the Places dataset [12]. The task is to classify landbirds and waterbirds. The dataset is biased as most landbirds are shown with land backgrounds and most waterbirds with water backgrounds.
- CelebA [5] is a large-scale dataset with over 200,000 celebrity images, annotated with 40 attribute labels. The task is to classify gender, which is spuriously correlated with hair colors.
- MultiNLI [10] is a NLI dataset contains 433k sentence pairs, whose class labels are entailment, contradiction, or neutral. The spurious attribute is the presence of negation words in the second sentence due to the artifacts from the data collection process.
- CivilComments-WILDS [4] is a variant of the CivilComments dataset [1], which contains comments from the Civil Comments platform. The task is to classify comments as toxic or non-toxic, with demographic information annotated for eight identities (male, female, LGBTQ, Christian, Muslim, other religions, Black, White).
- CheXpert [3] is a chest X-ray dataset originating from the Stanford University Medical center containing over 200,000 images.

1.2. Implementation Details

In this section, we detail the training configuration. Codes and checkpoints will be released at https://github.com/davelee-uestc/nsf_debiasing.

Architectures & Frameworks We used the PyTorch implementation [6] of ResNet-50 [2] and the the HuggingFace implementation [11] of bert-base-uncased, both starting from pretrained weights. ResNet-50 for Waterbirds, CelebA and CheXpert, and BERT for MultiNLI and CivilComments.

Preprocessing of Image Dataset We apply an augmentation pipeline including random resized cropping with a target resolution of (224, 224), which resizes the image to the specified resolution while randomly selecting a scale between 70% and 100% of the original size and a random aspect ratio between 0.75 and 1.33, with bilinear interpolation. This is followed by a random horizontal flip, which randomly mirrors the image horizontally with $p = 0.5$.

The ERM Training We train the ERM models using the following hyperparameters: Waterbirds with a batch size of 32 and 100 epochs; CelebA and ChexPert with a batch size of 100 and 20 epochs each; CivilComments-WILDS and MultiNLI with a batch size of 16 and 10 epochs each. The learning rate is set to 3e-3 for image datasets and 5e-5 for text datasets.

Debiasing NSF uses the following hyperparameters: For the Waterbirds dataset, 10 steps are used for learning the transformation and 500 for fine-tuning the classifier. The CelebA dataset uses 350 steps for learning the transformation and 110 steps for fine-tuning. For MultiNLI, it's 1200 and 1500 steps, respectively. The CivilComments-WILDS dataset uses 300 and 100 steps.

2. Additional Experimental Results

2.1. More CAM Visualizations

The CAM (Class Activation Map) visualizations in Figure 1 illustrate the effect of biases in datasets such as CelebA, Waterbirds, and Chexpert.

The CelebA dataset is known to have biases related to hair color, which often correlates with gender classification tasks. In the CAM visualizations, ERM model highlights regions that correspond to hair color, indicating that the model relies on this bias for classification. The model debiased using the proposed method shows a more distributed activation across the face, indicating that the model is focusing more on the facial features rather than the biased hair color.

In the Waterbirds dataset, the background is often correlated with the type of bird, leading models relying on the background for classification rather than the bird itself. CAM visualizations using ERM highlight the background, whereas the visualization from the model debiased using the proposed method show focus on the bird.

The Chexpert dataset includes chest X-rays where the

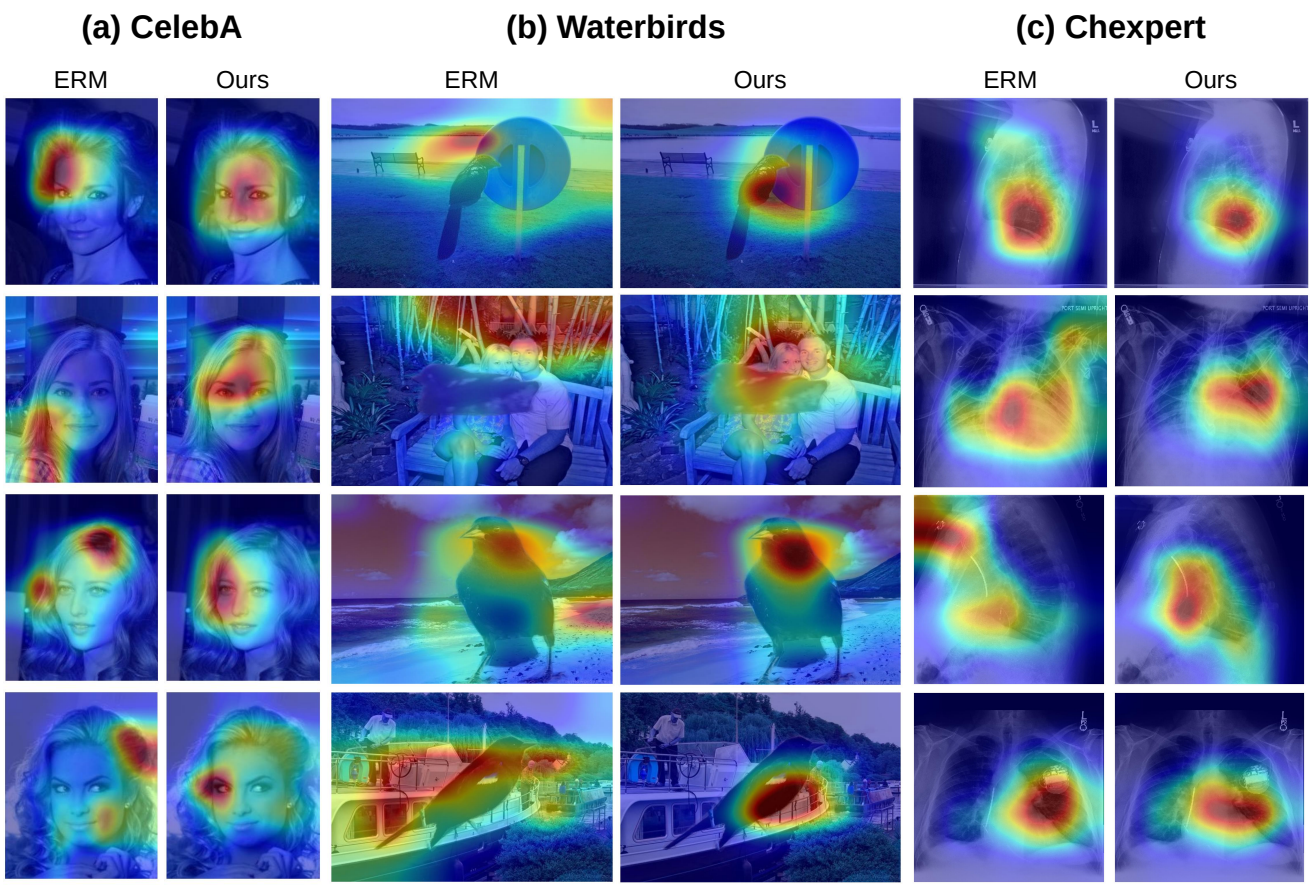

Figure 1. CAM on the Waterbirds, CelebA, and CheXpert Datasets, visualizing using GradCAM.

medical devices (like pacemakers) correlate with certain illness. In ERM models, CAM visualizations might highlight these medical devices, suggesting that the model uses them as shortcuts for classification. The model debiased using the proposed method focuses more on relevant areas of the chest X-ray.

The visualizations reveal how standard models tend to rely on biases (like background in Waterbirds, hair color in CelebA, and medical devices in Chexpert) for classification. In contrast, the model debiased using the proposed method mitigates these biases, leading to more accurate and fairer predictions.

3. Proof of Theorems

3.1. Data Distribution with Spurious Correlations

We adopt a data generation process from [8] to model the joint data distribution $(X_\rho, Y_\rho, A_\rho) \sim p_\rho$ under spurious correlation. The label $y \in Y_\rho$ follows the Uniform distribution over $\{1, -1\}$, the data point $\vec{x} = [Ba, y, \vec{\delta}] \in X_\rho$ and the spurious feature $a \in A_\rho$ are generated as follow:

$$a \sim \begin{cases} P(a = k|y = k) = \rho \\ P(a = -k|y = k) = 1 - \rho \end{cases}, \vec{\delta} \sim \mathcal{N}(\vec{0}, \vec{I}^{D-2}),$$

where $\mathcal{N}$ is the normal distribution, D is the dimension of $\vec{x}$, $\rho \in (0.5, 1)$, and $B \geq 1$ is scalar constants.

3.2. Proof of Theorem 1

$\forall(\vec{x}_i, y_i, \vec{a}_i) \in p_\rho$, the relative distance between $\vec{x}_i$ and its corresponding centroid $C_{y_i,\rho}$ and another closest class centroid $\vec{q}_i \in Q$ is

$$\vec{q}_i = \min_{u \neq y_i} \{(\vec{x}_i - C_u)^2\}. \tag{1}$$

$$d(\vec{x}_i, \rho) = (\vec{x}_i - C^\rho_{y_i})^2 - (\vec{x}_i - C^\rho_{q_i})^2 \tag{2}$$

$$\begin{aligned} d(\vec{x}_i, \rho) = (Ba_i - (2\rho - 1)By_i)^2 + (y_i - y_i)^2 + (\delta - 0)^2 \\ -[(Ba_i + (2\rho - 1)By_i)^2 + (y_i + y_i)^2 + (\delta - 0)^2] \end{aligned} \tag{3}$$

$$d(\vec{x}_i, \rho) = -4y_i^2[(2\rho-1)B^2\frac{a_i}{y_i} + 1]. \tag{4}$$

If $1-(2\rho-1)^2B^4 < 0$, then

$$\forall y_i = y_j, d(\vec{x}_i) \times d(\vec{x}_j) < 0 \iff \vec{a}_i \neq \vec{a}_j$$

Proof

$$d(\vec{x}_i, \rho) \times d(\vec{x}_j, \rho) = 16y_i^2y_j^2[(2\rho-1)B^2\frac{a_i}{y_i} + 1][(2\rho-1)B^2\frac{a_j}{y_j} + 1] \tag{5}$$

Since $y_i = y_j$,

$$d(\vec{x}_i, \rho) \times d(\vec{x}_j, \rho) = 16k^4[(2\rho-1)B^2\frac{a_i}{y_i} + 1][(2\rho-1)B^2\frac{a_j}{y_j} + 1] \tag{6}$$

Sufficiency $a_i \neq a_j, 1-(2\rho-1)^2B^4 < 0 \Rightarrow$,

$$d(\vec{x}_i, \rho) \times d(\vec{x}_j, \rho) = 16y_i^4[(2\rho-1)B^2 + 1][-(2\rho-1)B^2 + 1] \tag{7}$$

$$d(\vec{x}_i, \rho) \times d(\vec{x}_j, \rho) = 16y_i^4[1-(2\rho-1)^2B^4] < 0 \tag{8}$$

Necessity $\Leftarrow d(\vec{x}_i, \rho) \times d(\vec{x}_j, \rho) < 0, 1-(2\rho-1)^2B^4 < 0$,

$$d(\vec{x}_i, \rho) \times d(\vec{x}_j, \rho) = 16y_i^4[(2\rho-1)B^2\frac{a_i}{y_i} + 1][(2\rho-1)B^2\frac{a_j}{y_j} + 1] < 0 \tag{9}$$

$$\iff [(2\rho-1)B^2\frac{a_i}{y_i} + 1][(2\rho-1)B^2\frac{a_j}{y_j} + 1] < 0 \tag{10}$$

$$\xrightarrow{a_i = a_j} [(2\rho-1)B^2 + 1]^2 < 0 \tag{11}$$

Since a square cannot be negative, the assumption $a_i = a_j$ is false. Therefore, $a_i \neq a_j$.

3.3. Proof of Theorem 2

The conditional mean $C_k^\rho = \mathbb{E}[X_\rho | Y = k]$, also known as the centriod, can be estimated as

$$C_k^\rho = \frac{1}{\sum_{i=1}^N \mathbb{1}[\vec{y}_i = k]} \sum_{i=1}^N \mathbb{1}[\vec{y}_i = k] * \vec{x}_i, \tag{12}$$

Here, we want to estimate the unbiased conditional mean value of $\vec{x}$ in the true data distribution as $C_k = C_k^{0.5}$.

If $1-(2\rho-1)^2B^4 < 0$, then

$$C_k = \mathbb{E}(\frac{1}{2|U_k|}\sum_i^{|U_k|}\vec{u}_i + \frac{1}{2|V_k|}\sum_j^{|V_k|}\vec{v}_j)$$

where $U_k = \{\vec{x} \mid (\vec{x}, y) \in p_\rho, y = k, d(\vec{x}, \rho) > 0\}$, $V_k = \{\vec{x} \mid (\vec{x}, y) \in p_\rho, y = k\} \setminus U_k$, $\vec{u}_i \in U_k$, $\vec{v}_i \in V_k$.

Proof

$$\mathbb{E}(C_k - \mathbb{E}(C_k^{0.5})) = \mathbb{E}(\frac{1}{2|U_k|}\sum_i^{|U_k|}\vec{u}_i + \frac{1}{2|V_k|}\sum_j^{|V_k|}\vec{v}_j - \sum_{a \in A_{0.5}} \mathbb{E}[X_{0.5} \mid Y = k, A = a]\, P(A = a \mid Y = k)) \tag{13}$$

$$\mathbb{E}(C_k - \mathbb{E}(C_k^{0.5})) = \mathbb{E}(\frac{\bar{U}_k + \bar{V}_k}{2} - \frac{\mathbb{E}(X_{0.5} \mid A = -k, Y = k) + \mathbb{E}(X_{0.5} \mid A = k, Y = k)}{2}) \tag{14}$$

With Theorem 1, it has

$$\mathbb{E}(C_k - \mathbb{E}(C_k^{0.5})) = \frac{\mathbb{E}(U_k) + \mathbb{E}(V_k)}{2} - \frac{\mathbb{E}(U_k) + \mathbb{E}(V_k)}{2} = 0 \tag{15}$$

References

[1] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion Proceedings of the 2019 World Wide Web Conference*, pages 491–500, 2019. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 1

[3] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 590–597, 2019. 1

[4] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021. 1

[5] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015. 1

[6] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019. 1

[7] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2019. 1

[8] Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pages 8346–8356. PMLR, 2020. 2

[9] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010. 1

[10] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, 2018. 1

[11] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019. 1

[12] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2017. 1