Mitigating Simplicity Bias in Deep Learning for Improved OOD Generalization and Robustness

Bhavya Vasudeva [1] **Kameron Shahabi** [1] **Vatsal Sharan** [1]

Abstract

Neural networks are known to exhibit *simplicity bias* (SB) where they tend to prefer learning 'simple' features over more 'complex' ones, even when the latter may be more informative. SB can lead to the model making biased predictions which have poor out-of-distribution (OOD) generalization and robustness. To address this, we propose a framework that encourages the model to use a more diverse set of features to make predictions. We first train a simple model, and then regularize the conditional mutual information with respect to it to obtain the final model. We demonstrate the effectiveness of this framework in various problem settings and real-world applications, showing that it effectively addresses SB, and enhances OOD generalization, sub-group robustness and fairness. We complement these results with theoretical analyses of the effect of the regularization and its OOD generalization properties.

1. Introduction

Recent studies [53; 17; 36] investigating generalization in deep learning suggest that current techniques for training neural networks (NNs) tend to favor learning simple functions over complex ones. While this inductive bias has benefits in terms of preventing overfitting, it has been found that in the presence of multiple predictive features of varying complexity, NNs tend to be overly reliant on simpler features while ignoring more complex features that may be equally or more informative of the target [44; 35; 34]. This phenomenon has been termed *simplicity bias* (SB) and has several undesirable implications for robustness and out-of-distribution (OOD) generalization.

[1]Department of Computer Science, University of Southern California, Los Angeles, CA, USA. Correspondence to: Bhavya Vasudeva <bvasudev@usc.edu>.

Published at the ICML 2023 Workshop on Spurious Correlations, Invariance, and Stability. Honolulu, Hawaii, USA.

As an illustrative example, consider the Waterbirds dataset [42]. The objective here is to predict a bird's type (landbird vs. waterbird) based on its image (see Fig. 1 for an example). While features such as the background (land vs. water) are easier to learn, and can have significant correlation with the bird's type (since most images of landbirds are on a land background, and vice-versa), more complex features like the bird's shape are more predictive of its type. However, SB can cause the model to be highly dependent on simpler yet predictive features, such as the background in this case. A model which puts high emphasis on the background for solving this task is not desirable, since its performance may not transfer across different environments. A similar story arises in many tasks—Table 1 summarizes various datasets, where the target or task-relevant features (also known as invariant features), are more complex than surrogate features that are superficially correlated with the label (also known as spurious features). SB causes NNs to heavily rely on these surrogate features for predictions.

Dataset	Task-relevant/invariant feature	Surrogate/spurious feature
Waterbirds [42]	Bird type	Background
CelebA [33]	Hair colour	Gender
MultiNLI [51]	Reasoning	Negation words
CivilComments-WILDS [6]	Sentiment	Demographic attributes
Colored-MNIST [1]	Digit	Color
Camelyon17-WILDS [7]	Diagnoses	Hospital
Adult-Confounded [10]	Income	Demographic attributes

Table 1. Summary of the datasets we consider. Spurious features seem *simpler* than invariant features.

Several methods [1; 42; 31] have been proposed to address this problem of OOD generalization (see Appendix for discussion on related work). However, most of them require some knowledge about the spurious feature. This is because some knowledge of the different environments of interest or the underlying causal graph is in general necessary to decide whether a feature is spurious or invariant. To sidestep this, *in this work we take the viewpoint that features that are usually regarded as being spurious for the task are often simple and quite predictive* (as suggested by Table 1). If this hypothesis is true, alleviating simplicity bias can lead to better robustness and OOD generalization.

With improved robustness and OOD generalization as major end goals in mind, we develop a new framework to mitigate SB and encourage the model to utilize a more diverse set

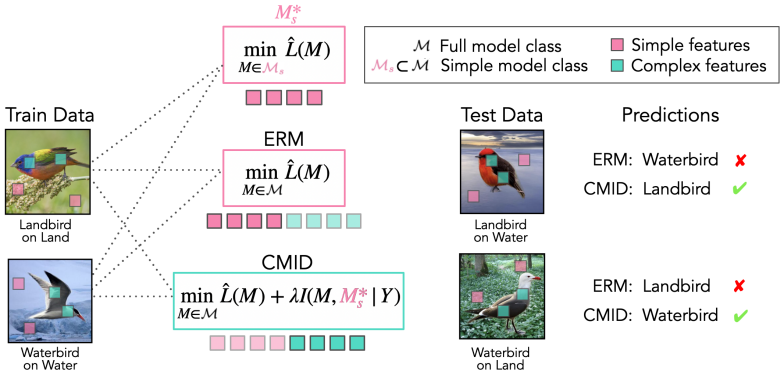

Figure 1. Summary of our approach. Models trained with ERM tend to use simple features (e.g., background) that do not generalize, while encouraging conditional independence with respect to a simple model increases reliance on complex features (e.g., shape) that generalize.

of features for its predictions. Our high-level goal is to ensure that the trained model M has minimal conditional mutual information $I(M; S|Y)$ with any simple, predictive feature S, conditioned on the label Y. To achieve this, we first train a simple model M_s^* on the task, with the idea that this model captures the information in simple, predictive features. Subsequently, to train the final model M, we add its conditional mutual information $I(M; M_s^*|Y)$ with the model M_s^* conditioned on the label Y as a regularizer to the usual empirical risk minimization (ERM) objective. With this regularization term, we incentivize the model to leverage additional, task-relevant features which may be more complex. We refer to our approach as *Conditional Mutual Information Debiasing or CMID* (see Fig. 1).

We demonstrate that our framework effectively mitigates simplicity bias, and achieves improved OOD generalization, sub-group robustness and fairness. Our approach leads to models which use more diverse features on certain tasks which have been previously used to measure SB. It achieves improved sub-group robustness and OOD generalization on several benchmark tasks, including in the presence of multiple spurious features. It also improves fairness and leads to predictions that are less dependent on protected attributes such as race or gender. In addition, *we prove theoretical results which provide insight and certain guarantees for our approach*. We analyze a Gaussian mixture model setup to understand the effect of our regularization and its ability to reduce dependence on spurious features. We also derive guarantees on OOD generalization in a causal learning framework. We present the experimental results in Section 4 and defer the theoretical results to the Appendix.

2. Spurious Features are Simple and Predictive

In this section, we show that surrogate features are generally 'simpler' than invariant features. First, we define *simple models/features* for a task as follows:

Definition 2.1 (Simple models and features)**.** Consider a task on which benchmark models which achieve high accuracy have a certain complexity (in terms of number of parameters, layers etc.). We consider models that have significantly lower complexity than benchmark models as *simple models*. Similarly, we consider features that can be effectively learned using simple models as *simple features*.

Definition 2.1 proposes a metric of simplicity which is task-dependent since it depends on the complexity of the models which get high accuracy on the task. We choose to not define quantitative measures in Definition 2.1 as those would be problem-dependent. As an example, for Waterbirds, deep networks such as ResNet-50 [23] achieve best results, so a shallow CNN can be considered as a simple model.

Dataset	Predict invariant feature		Predict surrogate feature	
	Train	Test	Train	Test
CMNIST	86.2 ± 0.2	86.6 ± 0.3	100	100
Waterbirds	60.5 ± 2.5	60.4 ± 2.4	79.6 ± 0.6	78.4 ± 0.6

Table 2. Comparison of performance for predicting invariant and surrogate features.

Importantly, we observe that simple models can still achieve good performance when the task is to distinguish between surrogate features even though they are not as accurate in predicting the invariant feature. Specifically, for CMNIST, we compare the performance of a linear model on color classification and digit classification on clean MNIST data. Similarly, for Waterbirds, we compare the performance of a shallow CNN on background classification (using images from the Places dataset [54]) and bird type classification (using segmented images of birds from the CUB dataset [49]). The results in Table 2 indicate that surrogate features for these datasets are simple features, since they can be predicted much more accurately by simpler models than invariant features.

Based on these observations, we define *spurious features* as follows. Operationally, our definition has the advantage that it does not require knowledge of some underlying causal graph or environment labels to identify spurious features.

Definition 2.2 (Spurious features)**.** *Spurious features* are simple features that are still reasonably correlated with the

target label.

In the presence of such features, SB causes the model to prefer such features over invariant ones that are more complex.

We also verify that when simple models are trained on the target tasks on these datasets, they tend to rely on these spurious features to make accurate predictions. Specifically, we consider the digit classification task using CMNIST data, where the correlation between the color and the label in the test set is 10%, and birdtype classification using Waterbirds data, where the test set consists of balanced groups. Table 3 shows that the test accuracy is close to the correlation between the spurious feature-based group label and the target label. This indicates that simple models trained on the target task utilize the spurious features to make predictions.

Dataset	Train	Test
CMNIST	84.9 ± 0.2	10.7 ± 0.3
Waterbirds	93.3 ± 0.5	54.9 ± 1.1

Table 3. Train and test performance of the simple model on the target task.

3. CMID: Learning in the Presence of Spurious Features

In this section, we outline our proposed approach to mitigate SB. It leverages the fact that simple models can capture surrogate features much better than invariant features (as demonstrated in Table 2).

Notation Let $Z = (X, Y)$ denote an input-label pair, where $X \in \mathbb{R}^d, Y \in \{0, 1\}$, sampled from some distribution $\mathcal{D}$, D denote a dataset with n samples, $\mathbb{E}_D$ denote the empirical mean over D, $\mathcal{X}$ denote the input space, $M(\theta) : \mathcal{X} \to [0, 1]$ denote a model, parameterized by θ (shorthand M). The predictions of the model are given by $\mathbb{1}[M(X) > 0.5]$. Subscripts $(\cdot)_s$ and $(\cdot)_c$ denote simple and complex, respectively. Let $\mathcal{M}$ denote the class of all models M, $\ell_M(Z) : \mathcal{X} \times \{0, 1\} \to \mathbb{R}$ denote a loss function, and $\sigma(x, T) = \dfrac{1}{1 + e^{-Tx}}$ denote the sigmoid function, with temperature parameter T.

With slight abuse of notation, et M and M denote the (binary) random variables associated with the predictions of model M on datapoints X_i's and the labels Y_i's, across all $i \in [n]$, respectively. Let $H(\cdot)$ denote the Shannon entropy of a random variable. $I(\cdot;\cdot)$ measures Shannon mutual information between two random variables, and the conditional mutual information (CMI) between the outputs of two models given the label is denoted by $I(M_1; M_2|Y)$.

The empirical risk minimizer (ERM) for class $\mathcal{M}$ is denoted as M^*, and is given by:

$$M^* := \text{ERM}(\mathcal{M}) = \arg\min_{M \in \mathcal{M}} \mathbb{E}_D \, \ell_M(Z).$$

We consider the class of simple models $\mathcal{M}_s \subset \mathcal{M}$ and describe our two-stage approach CMI Debiasing (CMID):

- First, learn a simple model M_s, which minimizes risk on the training data:

$$M_s^* = \text{ERM}(\mathcal{M}_s).$$

- Next, learn a complex model M_c by regularizing its CMI with M_s:

$$M_c = \arg\min_{M \in \mathcal{M}} \mathbb{E}_D \, \ell_M(Z_i) + \lambda \hat{I}_n(M; M_s^*|Y),$$

  where $\hat{I}_n(M; M_s^*|Y)$ denotes the estimated CMI over D, and λ is the regularization parameter.

Note that we penalize the CMI instead of mutual information. This is because both M_s^* and M are expected to have information about Y (e.g., in the Waterbirds dataset, both bird type and background type are correlated with the label). Hence, they will not be independent of each other, but are closer to being independent when conditioned on the label. We also note that we use conditional mutual information to measure dependence, instead of other measures such as enforcing orthogonality of the predictions. This is for the simple reason that mutual information measures all—potentially non-linear—dependencies between the random variables.

To estimate CMI, first consider the conditional (joint) distributions over n samples:

$$p(M = m|M = y) = \frac{\Sigma_{i \in [n]} \mathbb{1}[Y_i = y] \, \zeta(M(X_i), m)}{\Sigma_{i \in [n]} \mathbb{1}[Y_i = y]},$$

$$p(M = m, M_s = m'|M = y) = \frac{\Sigma_{i \in [n]} \mathbb{1}[Y_i = y] \, \zeta(M(X_i), m) \, \zeta(M_s(X_i), m')}{\Sigma_{i \in [n]} \mathbb{1}[Y_i = y]},$$

where $m, m', y \in \{0, 1\}$, $\zeta(M(X_i), 1) = \mathbb{1}[M(X_i) > 0.5]$, and $\zeta(M(X_i), 0) = 1 - \mathbb{1}[M(X_i) > 0.5]$.

Note that the CMI computed using these would not differentiable as these densities are computed by thresholding the outputs of the model. Since we want to add a CMI penalty as a regularizer to the ERM objective and optimize the proposed objective using standard gradient-based methods, we need a differentiable version of CMI. Thus, for practical purposes, we use an approximation of the indicator function $\mathbb{1}[x > 0.5]$, given by $\sigma(x - 0.5, T)$, where T determines the degree of smoothness or sharpness in the approximation.

This can be easily generalized for multi-class classification with C classes. In that case, $m, m', y \in \{0, \cdots, C - 1\}$, $M(X_i)$ is a C-dimensional vector with the m^{th} entry indicating the probability of predicting class m, and $\zeta(M(X_i), m) = \mathbb{1}[\arg\max_{j \in [C]} M_j(X_i) = m]$. To make this differentiable, we use the softmax function with temperature parameter T to approximate the indicator function.

Using these densities, the estimated CMI is:

$$\hat{I}_n(M, M_s|M) = \sum_y p(M=y_i) \sum_{m,m'} p(M=m, M_s=m'|M=y) \log\left[\frac{p(M=m, M_s=m'|M=y)}{p(M=m|M=y)p(M_s=m'|M=y)}\right]. \quad (1)$$

This estimate is differentiable, making it compatible with gradient-based methods. Therefore, we utilize it as a regularizer for the proposed approach.

4. Experiments

We show that CMID reduces SB, and yields improvements across a number of OOD generalization, robustness and fairness metrics. We note that past approaches usually specifically target one or two of these problem settings. Thus, we consider the most task-relevant methods for comparison.

4.1. Mitigating Simplicity Bias

Slab Data Slab data was proposed in [44] to model simplicity bias. Each feature is composed of k data blocks or slabs, as shown in Fig. 2. There is a simple feature along which the data is linearly separable, while features with more slabs are complex and involve a piece-wise linear model. The linear model is perfectly predictive, but the predictor using both types of features attains a much better margin, and generalizes better. Fig. 2 shows the decision boundary using ERM and the proposed approach. We see that CMID encourages the model to use both features and attain better margin.

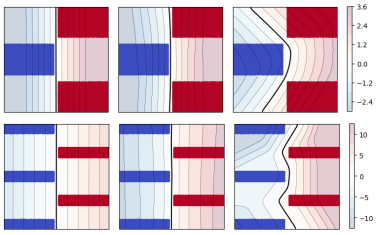

Figure 2. Results on slab data. Left: Linear model, Center: 1-hidden-layer NN-ERM, Right: 1-hidden-layer NN-CMID.

Texture vs Shape Bias on ImageNet-9 [19] showed that CNNs trained on ImageNet tend to make predictions based on image texture rather than image shape. To quantify this phenomenon, the authors designed the GST dataset, which contains synthetic images with conflicting shape and texture (e.g., image of a cat modified with elephant skin texture as a conflicting cue). The *shape bias* of a model on the GST dataset is defined as the number of datapoints for which the model correctly identifies shape compared to the total number of datapoints for which the model correctly identifies either shape or texture. We train a ResNet50 model on ImageNet-9 data, a subset of ImageNet organized into nine classes by [52] (more details in the Appendix) and observe that CMID (shape bias $\mathbf{51.8}$) mitigates the texture bias exhibited by ERM (shape bias $\mathbf{38.6}$) and encourages the model to rely on shape for predictions.

4.2. Better OOD Generalization

Synthetic: CMNIST and Color+Patch MNIST We present results on two variants of the MNIST dataset [12], using a binary digit classification task (< 5 or not). In the colored-MNIST data [1], color (red/green) is injected as a spurious feature, with environment-dependent label correlation $1 - p_e$. The train data has two environments with $p_e = 0.1, 0.2$ while the test data has $p_e = 0.9$, to test OOD performance. Further, it contains 25% label noise to reduce the predictive power of the task-relevant feature: digit shape. We also consider the color+patch MNIST data proposed in [3], which contains an additional spurious feature in the form of a 3×3 patch. The position of the patch (top left/bottom right) is correlated with the label, with the same p_e, but independent of the color. Table 4 shows that CMID gets competitive OOD performance with methods that require group knowledge, and has the lowest gap δ_{gap} between test performance on i.i.d and OOD samples, even in the presence of multiple spurious features.

Method	Group labels	Bias: Color			Bias: Color & Patch		
		Test (i.i.d) $p_e = 0.1$	Test (OOD) $p_e = 0.9$	δ_{gap}	Test (i.i.d) $p_e = 0.1$	Test (OOD) $p_e = 0.9$	δ_{gap}
ERM	No	88.6 ± 0.3	16.4 ± 0.8	-72.2	93.7 ± 0.3	14.0 ± 0.5	-79.7
IRM [1]	Yes	71.4 ± 0.9	66.9 ± 2.5	-4.5	93.5 ± 0.2	13.4 ± 0.3	-80.1
GDRO [42]	Yes	89.2 ± 0.9	13.6 ± 3.8	-75.6	92.3 ± 0.3	14.1 ± 0.8	-78.2
PI [5]	Yes	70.3 ± 0.3	70.2 ± 0.9	-0.1	85.4 ± 0.9	15.3 ± 2.7	-70.1
BLOOD [3]	Yes	70.5 ± 1.1	70.7 ± 1.4	0.2	68.3 ± 2.3	62.3 ± 3.3	-6.0
EIIL [10]	No	71.7 ± 1.6	62.8 ± 5.0	-8.9	65.3 ± 8.4	53.0 ± 5.6	-12.3
JTT [31]	No	72.2 ± 1.1	64.6 ± 0.56	-7.6	64.0 ± 2.7	56.2 ± 2.7	-7.8
CMID	No	69.2 ± 0.9	68.9 ± 0.9	-0.3	60.3 ± 2.7	59.4 ± 1.0	-0.9
Optimal	-	75	75	0	75	75	0

Table 4. Comparison on Colored MNIST and Color+Patch MNIST.

Medical: Camelyon17-WILDS Camelyon17-WILDS is a real-world medical image dataset of data collected from five hospitals [7; 28]. Three hospitals comprise the training set, one is the validation set and the third is the OOD test set. Images from different hospitals vary visually. The task is to predict whether or not the image contains tumor tissue, and the dataset is a well-known OOD generalization benchmark [3; 28]. Table 5 shows that CMID leads to higher average accuracies than existing group-based methods when evaluated on images from the test hospital.

Method	ERM	IRM [1]	GDRO [42]	PI [5]	BLOOD [3]	CMID
Train Acc	97.3 ± 0.1	97.1 ± 0.1	96.5 ± 1.4	93.2 ± 0.2	93.0 ± 1.8	92.9 ± 1.4
OOD Test acc	66.5 ± 4.2	59.4 ± 3.7	70.2 ± 7.3	71.7 ± 7.5	74.9 ± 5.0	76.1 ± 6.3

Table 5. Comparison on Camelyon17-WILDS.

Fairness: Adult-Confounded The Adult-Confounded dataset [10] is a variant of the UCI Adult dataset [38; 30]. It consists of four sensitive sub-groups based on binarized race and sex labels, and has confounded data where sub-group membership is predictive of the label on the training data but the correlation is reversed at test time. Therefore, it tests

whether the classifier makes biased predictions based on sub-group membership. Table 6 shows that compared to other methods CMID achieves superior OOD test performance with the least gap between train and test performance, indicating its low reliance on sensitive sub-group information.

Method	ERM	ARL [29]	EIIL [10]	CMID
Train Acc	92.7 ± 0.5	72.1 ± 3.6	69.7 ± 1.6	76.2 ± 2.2
OOD Test Acc	31.1 ± 4.4	61.3 ± 1.7	78.8 ± 1.4	78.8 ± 0.7
δ_{gap}	-61.6	-10.8	9.1	2.6

Table 6. Comparison on Adult-Confounded dataset.

4.3. Sub-group Robustness

We evaluate our approach on four benchmark classification tasks for robustness to spurious correlations, namely on Waterbirds, CelebA, MultiNLI and CivilComments-WILDS datasets (Table 1, details in the Appendix). Table 7 shows the average and worst-group accuracies for CMID and comparison with other methods [15; 37; 31] which do not use group information. GDRO [42], which uses group information, acts as a benchmark. We see that on three of these datasets, CMID competes with state-of-the-art algorithms that improve sub-group robustness. Interestingly, CMID seems paricularly effective on the two language-based datasets. We also note that CMID is not effective on CelebA images. We believe that this is because both the spurious feature (gender) and the invariant feature (hair color) for CelebA are of similar complexity. We explore this further in the Appendix.

Method	Waterbirds		CelebA		MultiNLI		CivilComments	
	Avg	Worst-grp	Avg	Worst-grp	Avg	Worst-grp	Avg	Worst-grp
ERM	97.3	72.6	95.6	47.2	82.4	67.9	92.6	57.4
CVaRDRO [15]	96.0	75.9	82.5	64.4	82.0	68.0	92.5	60.5
LfF [37]	91.2	78.0	85.1	77.2	80.8	70.2	92.5	58.8
JTT [31]	93.3	86.7	88.0	81.1	78.6	72.6	91.1	69.3
CMID	88.6	84.3	84.5	75.3	81.4	71.5	84.2	74.8
GDRO [42]	93.5	91.4	92.9	88.9	81.4	77.7	88.9	69.9

Table 7. Average and worst-group test accuracies on benchmark datasets for sub-group robustness.

4.4. Fairness Application: Bias in Occupation Prediction

The Bios dataset [11; 9] is a large-scale dataset of more than 300k biographies scrapped from the internet. The goal is to predict a person's occupation based on their bio. [9] formalize a notion of social norm bias (SNoB) which measures adherence to gender norms of the majority gender group for the occupation. They quantify this bias using $\rho(p_c, r_c)$, the Spearman rank correlation coefficient between p_c, the fraction of bios under occupation c mentioning 'she', and r_c, the correlation between occupation and gender predictions (details in the Appendix). A larger value of $\rho(p_c, r_c)$ indicates larger social norm bias, meaning that in male-dominated occupations the algorithm has a higher accuracy on bios that align with inferred masculine norms, and vice-versa.

Table 8 shows results on Bios data. We compare with a group fairness approach, Decoupled [16] that trains separate models for each gender, in order to mitigate gender bias. We see that CMID address SNoB bias better than ERM and Decoupled, achieving a lower $\rho(p_c, r_c)$ and improved accuracy.

Method	Accuracy	$\rho(p_c, r_c)$
ERM	0.95	0.66
Decoupled [16]	0.94	0.60
CMID	0.96	0.38

Table 8. Comparison on Bios data.

5. Conclusion and Discussion

We proposed a new framework (CMID) to mitigate simplicity bias, and showed that it yields improvements over ERM and many other previous approaches across a number of OOD generalization, robustness and fairness benchmarks. We note, however, that like any other regularization or inductive bias, CMID may not be effective for every task. For certain tasks, spurious features as defined by us in Section 2 may not actually be spurious, or the separation between the feature complexity of spurious and invariant features may not be as large. A natural direction of future work is to further explore the capabilities and limitations of our approach, and to also further understand its theoretical properties. More broadly, our work suggests auditing large models with respect to much simpler models can lead to improved properties of the larger models along certain robustness and fairness axes. It would be interesting to explore the power of similar approaches for other desiderata, and to understand its capabilities for fine-tuning large pre-trained models.

References

[1] Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization, 2020.

[2] Arpit, D., Jastrzębski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pp. 233–242. PMLR, 2017.

[3] Bae, J.-H., Choi, I., and Lee, M. BLOOD: Bi-level learning framework for out-of-distribution generalization, 2022. URL https://openreview.net/forum?id=Cm08egNmrl3.

[4] Bahng, H., Chun, S., Yun, S., Choo, J., and Oh, S. J. Learning de-biased representations with biased representations. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 528–539. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/bahng20a.html.

[5] Bao, Y., Chang, S., and Barzilay, R. Predict then

interpolate: A simple algorithm to learn stable classifiers. In *International Conference on Machine Learning*. PMLR, 2021.

[6] Borkan, D., Dixon, L., Sorensen, J., Thain, N., and Vasserman, L. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion Proceedings of The 2019 World Wide Web Conference*, 2019.

[7] Bándi, P., Geessink, O., Manson, Q., Van Dijk, M., Balkenhol, M., Hermsen, M., Ehteshami Bejnordi, B., Lee, B., Paeng, K., Zhong, A., Li, Q., Zanjani, F. G., Zinger, S., Fukuta, K., Komura, D., Ovtcharov, V., Cheng, S., Zeng, S., Thagaard, J., Dahl, A. B., Lin, H., Chen, H., Jacobsson, L., Hedlund, M., Çetin, M., Halıcı, E., Jackson, H., Chen, R., Both, F., Franke, J., Küsters-Vandevelde, H., Vreuls, W., Bult, P., van Ginneken, B., van der Laak, J., and Litjens, G. From detection of individual metastases to classification of lymph node status at the patient level: The camelyon17 challenge. *IEEE Transactions on Medical Imaging*, 38 (2):550–560, 2019. doi: 10.1109/TMI.2018.2867350.

[8] Chang, S., Zhang, Y., Yu, M., and Jaakkola, T. Invariant rationalization. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1448–1458. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/chang20c.html.

[9] Cheng, M., De-Arteaga, M., Mackey, L., and Kalai, A. T. Social norm bias: residual harms of fairness-aware algorithms. *Data Mining and Knowledge Discovery*, pp. 1–27, 2023.

[10] Creager, E., Jacobsen, J.-H., and Zemel, R. Environment inference for invariant learning. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 2189–2200. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/creager21a.html.

[11] De-Arteaga, M., Romanov, A., Wallach, H., Chayes, J., Borgs, C., Chouldechova, A., Geyik, S., Kenthapadi, K., and Kalai, A. T. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, pp. 120–128, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361255. doi: 10.1145/3287560.3287572. URL https://doi.org/10.1145/3287560.3287572.

[12] Deng, L. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[13] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

[14] Dixon, L., Li, J., Sorensen, J., Thain, N., and Vasserman, L. Measuring and mitigating unintended bias in text classification. 2018.

[15] Duchi, J. C., Hashimoto, T., and Namkoong, H. Distributionally robust losses against mixture covariate shifts, 2019.

[16] Dwork, C., Immorlica, N., Kalai, A. T., and Leiserson, M. Decoupled classifiers for group-fair and efficient machine learning. In Friedler, S. A. and Wilson, C. (eds.), *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pp. 119–133. PMLR, 23–24 Feb 2018. URL https://proceedings.mlr.press/v81/dwork18a.html.

[17] Frankle, J. and Carbin, M. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJl-b3RcF7.

[18] Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[19] Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness, 2022.

[20] Gururangan, S., Swayamdipta, S., Levy, O., Schwartz, R., Bowman, S., and Smith, N. A. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 107–112, New Orleans, Louisiana,

June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2017. URL https://aclanthology.org/N18-2017.

[21] Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pp. 3323–3331, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

[22] Hashimoto, T. B., Srivastava, M., Namkoong, H., and Liang, P. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, 2018.

[23] He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

[24] Hebert-Johnson, U., Kim, M., Reingold, O., and Rothblum, G. Multicalibration: Calibration for the (Computationally-identifiable) masses. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1939–1948. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/hebert-johnson18a.html.

[25] Jia, S., Meng, T., Zhao, J., and Chang, K.-W. Mitigating gender bias amplification in distribution by posterior regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2936–2942, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.264. URL https://aclanthology.org/2020.acl-main.264.

[26] Kamiran, F. and Calders, T. Data preprocessing techniques for classification without discrimination. *Knowl. Inf. Syst.*, 33(1):1–33, oct 2012. ISSN 0219-1377. doi: 10.1007/s10115-011-0463-8. URL https://doi.org/10.1007/s10115-011-0463-8.

[27] Kirichenko, P., Izmailov, P., and Wilson, A. G. Last layer re-training is sufficient for robustness to spurious correlations. In *International Conference on Machine Learning*, 2022.

[28] Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pp. 5637–5664. PMLR, 2021.

[29] Lahoti, P., Beutel, A., Chen, J., Lee, K., Prost, F., Thain, N., Wang, X., and Chi, E. H. Fairness without demographics through adversarially reweighted learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

[30] Leisch, F. and Dimitriadou, E. *mlbench: Machine Learning Benchmark Problems*, 2021. R package version 2.1-3.1.

[31] Liu, E. Z., Haghgoo, B., Chen, A. S., Raghunathan, A., Koh, P. W., Sagawa, S., Liang, P., and Finn, C. Just train twice: Improving group robustness without training group information. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6781–6792. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/liu21f.html.

[32] Liu, J., Hu, Z., Cui, P., Li, B., and Shen, Z. Heterogeneous risk minimization. In *International Conference on Machine Learning*, 2021.

[33] Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[34] Morwani, D., Batra, J., Jain, P., and Netrapalli, P. Simplicity bias in 1-hidden layer neural networks, 2023.

[35] Nakkiran, P., Kaplun, G., Kalimeris, D., Yang, T., Edelman, B. L., Zhang, F., and Barak, B. *SGD on Neural Networks Learns Functions of Increasing Complexity*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[36] Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. Deep double descent: Where bigger models and more data hurt. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1g5sA4twr.

[37] Nam, J., Cha, H., Ahn, S., Lee, J., and Shin, J. Learning from failure: De-biasing classifier from biased classifier. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*,

volume 33, pp. 20673–20684. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/eddc3427c5d77843c2253f1e799fe933-Paper.pdf.

[38] Newman, D., Hettich, S., Blake, C., and Merz, C. Uci repository of machine learning databases, 1998. URL http://www.ics.uci.edu/~mlearn/MLRepository.html.

[39] Park, J. H., Shin, J., and Fung, P. Reducing gender bias in abusive language detection, 2018.

[40] Pezeshki, M., Kaba, S.-O., Bengio, Y., Courville, A., Precup, D., and Lajoie, G. Gradient starvation: A learning proclivity in neural networks. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=aExAsh1UHZo.

[41] Rosenfeld, E., Ravikumar, P. K., and Risteski, A. The risks of invariant risk minimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=BbNIbVPJ-42.

[42] Sagawa*, S., Koh*, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

[43] Setlur, A., Dennis, D., Eysenbach, B., Raghunathan, A., Finn, C., Smith, V., and Levine, S. Bitrate-constrained dro: Beyond worst case robustness to unknown group shifts, 2023.

[44] Shah, H., Tamuly, K., Raghunathan, A., Jain, P., and Netrapalli, P. The pitfalls of simplicity bias in neural networks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

[45] Sohoni, N., Dunnmon, J. A., Angus, G., Gu, A., and Ré, C. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

[46] Sohoni, N. S., Sanjabi, M., Ballas, N., Grover, A., Nie, S., Firooz, H., and Ré, C. Barack: Partially supervised group robustness with guarantees, 2022.

[47] Utama, P. A., Moosavi, N. S., and Gurevych, I. Mind the trade-off: Debiasing NLU models without degrading the in-distribution performance. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8717–8729, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.770. URL https://aclanthology.org/2020.acl-main.770.

[48] Valle-Perez, G., Camargo, C. Q., and Louis, A. A. Deep learning generalizes because the parameter-function map is biased towards simple functions. *International Conference on Learning Representations*, 2019.

[49] Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[50] Wang, T., Zhao, J., Yatskar, M., Chang, K.-W., and Ordonez, V. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *International Conference on Computer Vision (ICCV)*, October 2019.

[51] Williams, A., Nangia, N., and Bowman, S. 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)*, pp. 1112–1122. Association for Computational Linguistics, 2018. URL http://aclweb.org/anthology/N18-1101.

[52] Xiao, K., Engstrom, L., Ilyas, A., and Madry, A. Noise or signal: The role of image backgrounds in object recognition, 2020.

[53] Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy8gdB9xx.

[54] Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

A. Related Work

Simplicity Bias in NNs. Several works [19; 2; 48; 18; 44; 35; 34; 40] show that NNs trained with gradient-based methods prefer learning solutions which are 'simple'. [19] show that CNNs make predictions which depend more on image texture than image shape. [44] create synthetic datasets (e.g. Slab data which we also consider) and show that in the presence of simple and complex features, NNs rely heavily on simple features even when both have equal predictive power. [35] show that the predictions of NNs trained by SGD can be approximated well by linear models during early stages of training. [34] show that 1-hidden layer NNs are biased towards using low-dimensional projections of the data for predictions.

OOD Generalization. Towards developing models which perform better in the real world, OOD generalization requires generalization to data from new *environments*. Environments are usually defined based on the correlation between some spurious feature and the label. Various methods aim to recover a predictor that is *invariant* across a set of environments. [1] develop the invariant risk minimization (IRM) framework where environments are known, while [10] propose environment inference for invariant learning (EIIL), to recover the invariant predictor, when the environments are not known. Predict then interpolate (PI) [5] and BLOOD [3] use environment-specific ERMs to infer groups based on the correctness of predictions.

Subgroup Robustness. In many applications, models should not only do well on average but also do well on subgroups within the data. Several methods [42; 37; 27; 46] have been developed to improve the worst-group performance of a model. One widely used approach is to optimize the worst case risk over a set of subgroups in the data [15; 42; 43], known as distributionally robust optimization (DRO). CVaRDRO [15] optimizes over all subgroups in the data, which is somewhat pessimistic, whereas GDRO [42] does this over a set of predefined groups. However, group knowledge may not always be available, various methods try to identify or infer groups and reweight minority groups in some way, when group labels are not available [37; 31; 45] or partially available [46]. Just train twice (JTT) [31] uses ERM to identify the groups based on correctness of predictions. Learning from failure (LfF) [37] simultaneously trains two NNs, encouraging one model to make biased predictions, and reweighting the samples it finds harder-to-learn (larger loss) to train the other model.

Fairness An essential aspect of trustworthy models is ensuring fairness in their predictions across subgroups based on sensitive demographic information within the data. Fairness interventions used when group information is available include: data reweighting to balance groups before training [26], learning separate models for different subgroups [16], and post-processing of trained models, such as adjusting prediction thresholds based on fairness-based metrics [21]. Some approaches focus on achieving fairness without group knowledge. Multicalibration [24] aims to learn a model whose predictions are calibrated for all subpopulations that can be identified in a computationally efficient way. [22] proposes a DRO-based approach to minimize the worst-case risk over distributions close to the empirical distribution to ensure fairness. [29] proposes adversarial reweighted learning (ARL), where an auxiliary model identifies subgroups with inferior performance and the model of interest is retrained by reweighting these subgroups to reduce bias.

De-biasing Methods Deep neural networks are known to exhibit unwanted biases. For instance, CNNs trained on image data may exhibit texture bias [19], and language models trained on certain datasets may exhibit annotation bias [20]. Several methods have been proposed to mitigate these biases. [4] introduce a framework to learn de-biased representations by encouraging them to differ from a reference set of biased representations. [47] propose a confidence regularization approach to encourage models to learn from all samples. Recent works also show that deep neural networks tend to amplify the societal biases present in training data [50; 25] and they propose domain-specific strategies to mitigate such amplification.

B. Theoretical Results

In this section, we analyze the effect of CMI regularization and show it leads to reduced dependence on spurious features in a Gaussian mixture model. We also show guarantees for OOD generalization for our approach in a causal learning framework. We present the proofs in Section C.

B.1. Effect of CMI Regularization for Gaussian Features

We consider data generated from the following Gaussian mixture model (refer to the left-most panel in Fig. 3 for an example of data generated from this distribution).

Assumption B.1. Let the label $y \sim \mathcal{R}(0.5)$, where $\mathcal{R}(p)$ is a $\{\pm 1\}$ random variable which is 1 with probability p. Consider

an *invariant* feature X_1 and a *spurious* feature X_2, with distributions:

$$X_1 \sim \mathcal{N}(y\mu_1, \sigma_1^2) \text{ and } X_2 \sim \mathcal{N}(a\mu_2, \sigma_2^2).$$

where $a \sim y\mathcal{R}(\eta)$ is a spurious attribute, with an unstable correlation with y, and $\mu_1, \mu_2 > 0, \eta > 0.5$. Assume $X_1 \perp X_2 | y$ and let $\mu_2' = (2\eta - 1)\mu_2$ and $\sigma_2'^2 = \sigma_2^2 + \mu_2^2 - \mu_2'^2$.

We consider linear models to predict y from the features X_1 and X_2. Let $\mathcal{M} = \{(w_1, w_2) : w_1, w_2 \in \mathbb{R}\}$ be all possible linear models and $\mathcal{M}_s = \{(0, w) : w \in \mathbb{R}\} \cup \{(w, 0) : w \in \mathbb{R}\}$ be a simpler model class which only uses one of the two features. We consider the mean squared error (MSE) loss, and the ERM solution is given by:

$$\text{ERM}(\mathcal{M}) = \arg\min_{\boldsymbol{w} \in \mathcal{M}} \mathbb{E}\,(w_1 X_1 + w_2 X_2 - y)^2.$$

Proposition B.2. *ERM($\mathcal{M}$) satisfies* $\frac{w_1}{w_2} = \frac{\mu_1}{\mu_2'} \frac{\sigma_2'^2}{\sigma_1^2}$. *When* $\frac{\mu_1}{\mu_2'} \frac{\sigma_2'^2}{\sigma_1^2} < 1$, *ERM($\mathcal{M}_s$)*$= \left[0, \frac{\mu_2'}{\sigma_2^2}\right]$ *(upto scaling).*

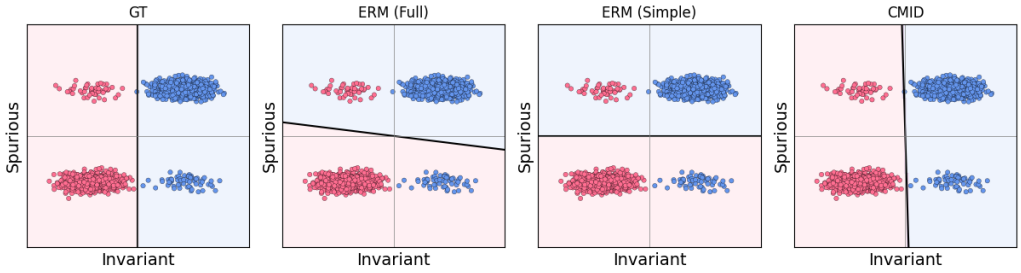

Figure 3. Results on synthetic Gaussian data generated as per Assumption B.1, with 2000 samples and $\mu_1 = \mu_2 = 5, \sigma_1 = 1.5, \sigma_2 = 0.5$. Left to right: Ground truth (GT) predictor, ERM with $\mathcal{M}$ as the class of linear models, ERM with $\mathcal{M}$ as the class of threshold functions, and ERM with CMI constraint, with $c = 0.01$.

We now consider the effect of CMI regularization. We assume that $\frac{\mu_1}{\mu_2'} \frac{\sigma_2'^2}{\sigma_1^2} < 1$, so as per Proposition B.2, in the first step we learn a simple model $w_2^* X_2$ which uses only X_2. We now consider the ERM problem but with a constraint on the CMI:

$$\text{ERM}_C(\mathcal{M}) = \arg\min_{\boldsymbol{w} \in \mathcal{M}} \mathbb{E}\,(w_1 X_1 + w_2 X_2 - y)^2 \text{ s.t. } I(w_1 X_1 + w_2 X_2; w_2^* X_2 | y) \leq \nu. \tag{2}$$

We show the following guarantee on the learned model.

Theorem B.3. *Let data be generated as per Assumption B.1. For* $\nu = 0.5 \log(1 + c^2)$ *for some* c*:*

1. When $\frac{\mu_1}{\mu_2'} \frac{\sigma_2'^2}{\sigma_1 \sigma_2} > \frac{1}{c}$, *the solution to (2) is the same as ERM($\mathcal{M}$), so* $\frac{w_1}{w_2} = \frac{\mu_1}{\mu_2'} \frac{\sigma_2'^2}{\sigma_1^2}$.

2. Otherwise, w_1 *is upweighted and the solution to (2) satisfies* $\frac{|w_1|}{|w_2|} = \frac{1}{c} \frac{\sigma_2}{\sigma_1}$.

Theorem B.3 suggests that for an appropriately small c, regularizing CMI with the simple model leads to a predictor which mainly uses the invariant feature. This is supported by experimental results on data drawn according to Assumption B.1, shown in Fig. 3.

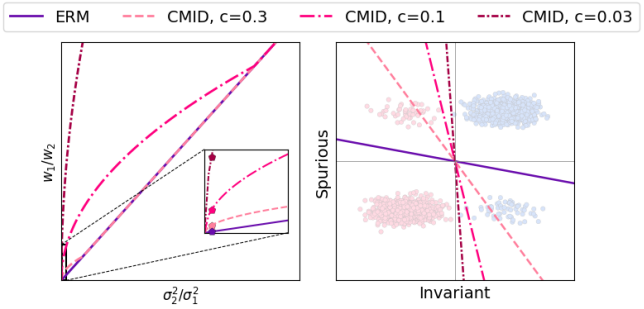

Figure 4. Effect of CMI regularization (wrt X_2) for different values of c (corresponding to regularization strength). Left: Lower values of c indicate stronger CMI regularization, resulting in more upweighting of w_1 wrt w_2. Inset shows a zoomed-in region and markers compare the three solutions when $\sigma_2^2/\sigma_1^2 = 1/6$. Right: Decision boundaries for predictors corresponding to the markers.

In Fig. 4 we visualize the relationship between $\frac{w_1}{w_2}$ and $\frac{\sigma_2^2}{\sigma_1^2}$ predicted in Theorem B.3 (assuming $\mu_1 = \mu_2$ and $\eta = 0.95$.) We see that a lower value of c promotes conditional independence with X_2 and upweighs w_1 more strongly. When $c \to 0$, $w_2 \to 0$.

B.2. OOD Generalization in a Causal Learning Framework

Following the setting in [1; 32], we consider a dataset $D = \{D^e\}_{e \in \mathcal{E}_{tr}}$, which is composed of data $D^e \sim \mathcal{D}_e^{n_e}$ gathered from different training environments $e \in \mathcal{E}_{tr}$, where e denotes an environment label, n_e represents the number of samples in e. $\mathcal{E}_{tr}$ denotes the set of training environments.

The problem of finding a predictor with good OOD generalization performance, can be formalized as:

$$\arg\min_{M \in \mathcal{M}} \max_{e \in \mathcal{E}} \mathbb{E}_D[\ell_M(Z)|e],$$

i.e., optimizing over the worst-case risk on all environments in set $\mathcal{E}$. Usually, $\mathcal{E} \supset \mathcal{E}_{tr}$, and hence, the data and label distribution can differ significantly for $e \in \mathcal{E}_{tr}$ and $e \in \mathcal{E} \setminus \mathcal{E}_{tr}$. This makes the OOD generalization problem hard to solve.

The invariant learning literature assumes the existence of invariant and variant features. In this section, we assume that the model of interest, say $M(X)$ is composed of a featurizer Φ and a classifier ω on top of it, *i.e.* $M(X) = \omega \circ \Phi(X)$. For simplicity, we omit the argument X and assume that learning a featurization includes learning the corresponding classifier, so we can write $M = \Phi$. Let E be a random variable sampled from a distribution on $\mathcal{E}$.

Definition B.4 (Invariant and Variant Predictors)**.** A feature map Φ is called *invariant* and is denoted by Φ if $Y \perp E|\Phi$, whereas it is called *variant* and is denoted by Ψ if $Y \not\perp E|\Psi$.

Several works [1; 32; 10] attempt to recover the invariant feature map by proposing different ways to find the maximally invariant predictor [8], defined as:

Definition B.5 (Invariance Set and Maximal Invariant Predictor)**.** The invariance set $\mathcal{I}$ with respect to environment set $\mathcal{E}$ and hypothesis class $\mathcal{M}$ is defined as:

$$\mathcal{I}_{\mathcal{E}}(\mathcal{M}) = \{\Phi : Y \perp E|\Phi\} = \{\Phi : H[Y|\Phi] = H[Y|\Phi, E]\}.$$

The corresponding maximal invariant predictor (MIP) of $\mathcal{I}_{\mathcal{E}}(\mathcal{M})$ is $\Phi^* = \arg\max_{\Phi \in \mathcal{I}_{\mathcal{E}}(\mathcal{M})} I(Y; \Phi)$.

The MIP is an invariant predictor that captures the most information about Y. Invariant predictors guarantee OOD generalization, making MIP the optimal invariant predictor (Theorem 2.1 in [32]).

As discussed in Section 1, most current work assumes that the environment labels e for the datapoints are known. However, environment labels typically are not provided in real-world scenarios. In this work, we don't assume access to environment labels and instead, we rely on another aspect of these features: are they simple or complex? We formalize this below:

Assumption B.6 (Simple and Complex Predictors)**.** The invariant feature comprises of complex features, *i.e.* $\Phi^* = [\Phi_c]$, where $\Phi_c \in \mathcal{M}_c$, *i.e.*, $\mathcal{M} \setminus \mathcal{M}_s$. The variant feature comprises of simple and complex features, *i.e.*, $\Psi^* = [\Psi_c, \Psi_s]$, where $\Psi_c \in \mathcal{M}_c$, $\Psi_s \in \mathcal{M}_s$ and $I(Y; \Psi_s) > 0$.

We consider the underlying causal model in [41] which makes the following assumption.

Assumption B.7 (Underlying Causal Model)**.** Given the model in Fig. 5, $I(\Phi, \Psi|Y) = 0$ and $I(\Psi_s, \Psi_c|Y) > I(\Psi_s, \Psi_c|Y, E)$.

The following simple result shows that our method finds the maximal invariant predictor, and thus generalizes OOD.

Theorem B.8. *Let ERM$(\mathcal{M}_s) = M_s^*$. Under Assumptions B.6 and B.7, the solution to the problem:*

$$\arg\min_{M \in \mathcal{M}} \mathbb{E}\, \ell_M(Z) \text{ s.t. } I(M; M_s^*|Y) = 0 \tag{3}$$

is $M = \Phi^*$, the maximal invariant predictor.

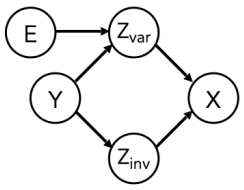

Figure 5. Our causal model. Latent variables Z_{inv} and Z_{var} correspond to invariant and variant features Φ^* and Ψ^* respectively.

C. Proofs for Section B

In this section, we present the proofs for the theoretical results in Section B.

C.1. Proof of Proposition B.2

Proof. We have:

$$\mathbb{E}\left(w_1X_1 + w_2X_2 - y\right)^2 = w_1^2(\sigma_1^2+\mu_1^2) + w_2^2(\sigma_2^2+\mu_2^2) + 1 - 2w_1\mu_1 - 2w_2\mu_2' + 2w_1w_2\mu_1\mu_2'.$$

Since $\arg\min_{\boldsymbol{w}\in\mathcal{M}} \mathbb{E}\left(w_1X_1 + w_2X_2 - y\right)^2$ is a convex problem, to find the minimizer we set gradients with respect to w_1 and w_2 to be 0. Subsequently, we solve the resulting set of equations. By taking the gradient and setting it to 0, we obtain the following set of equations:

$$2w_1(\sigma_1^2+\mu_1^2) - 2\mu_1 + 2w_2\mu_1\mu_2' = 0, \tag{4}$$

$$2w_2(\sigma_2^2+\mu_2^2) - 2\mu_2' + 2w_1\mu_1\mu_2' = 0. \tag{5}$$

From (5), $w_2 = \mu_2' \dfrac{1-w_1\mu_1}{\sigma_2^2+\mu_2^2}$. Substituting this in (4) and solving for w_1, we get:

$$w_1 = \frac{\mu_1}{\sigma_1^2}\left(\frac{1}{\frac{\mu_1^2}{\sigma_1^2}+\frac{\mu_2'^2}{\sigma_2'^2}+1}\right),$$

where $\sigma_2'^2 = \sigma_2^2+\mu_2^2-\mu_2'^2$. Using this, we get the expression for w_2:

$$w_2 = \frac{\mu_2'}{\sigma_2'^2}\left(\frac{1}{\frac{\mu_1^2}{\sigma_1^2}+\frac{\mu_2'^2}{\sigma_2'^2}+1}\right).$$

Thus, for ERM($\mathcal{M}$), $\frac{w_1}{w_2} = \frac{\mu_1}{\mu_2'}\frac{\sigma_2'^2}{\sigma_1^2}$.

Next, consider ERM over $\mathcal{M}_s$. If $w_2 = 0$, we use (4) and get $\left[\frac{\mu_1}{\sigma_1^2+\mu_1^2}, 0\right]$ as the solution, for which the loss value is $\frac{\sigma_1^2}{\sigma_1^2+\mu_1^2}$. If $w_1 = 0$, we use (5) and get $\left[0, \frac{\mu_2'}{\sigma_2^2+\mu_2^2}\right]$ as the solution, for which the loss is $\frac{\sigma_2'^2}{\sigma_2'^2+\mu_2'^2}$. When $\frac{\mu_1^2}{\mu_2'^2}\frac{\sigma_2'^2}{\sigma_1^2} < 1$, the latter has a smaller loss and thus, ERM($\mathcal{M}_s$) is $\left[0, \frac{\mu_2'}{\sigma_2^2}\frac{1}{1+\frac{\mu_2^2}{\sigma_2^2}}\right]$. □

C.2. Proof of Theorem B.3

Proof. Consider the constraint $I(w_1X_1 + w_2X_2; w_2^*X_2|y) \le \nu$. As we are working with continuous random variables in this setup, we employ differential entropy for our entropy computations. The entropy of a Gaussian random variable X with variance σ^2 is $H(X) = 0.5(\log(2\pi\sigma^2)+1)$. Using this and the definitions of CMI and conditional entropy, we have:

$$\begin{aligned} I(w_1X_1 + w_2X_2, w_2^*X_2|y) &= H(w_1X_1 + w_2X_2|y) - H(w_1X_1 + w_2X_2|y, w_2^*X_2) \\ &= H(w_1X_1 + w_2X_2|y) - H(w_1X_1|y) \\ &= \tfrac{1}{2}\log\left(\frac{w_1^2\sigma_1^2+w_2^2\sigma_2^2}{w_1^2\sigma_1^2}\right). \end{aligned}$$

Using $\nu = 0.5\log(1+c^2)$, the constraint becomes: $\frac{w_2^2\sigma_2^2}{w_1^2\sigma_1^2} \le c^2$. Thus, (2) reduces to solving:

$$\min_{w_1,w_2} \mathbb{E}\left(w_1X_1 + w_2X_2 - y\right)^2 \text{ s.t. } \frac{|w_2|\sigma_2}{|w_1|\sigma_1} \le c.$$

If $\frac{\mu_2'}{\mu_1}\frac{\sigma_1\sigma_2}{\sigma_2'^2} < c$, ERM($\mathcal{M}$) satisfies the constraint and serves as the solution to (2). Otherwise, since this is a convex optimization problem with an affine constraint, the constraint must be tight. Therefore, we determine the solution by finding ERM($\mathcal{M}$) subject to $\frac{|w_2|\sigma_1}{|w_1|\sigma_2} = c$. The solution is given by:

$$w_1 = \frac{\mu_1}{\sigma_1^2}\left(\frac{\left(1+\frac{\mu_2'\sigma_1}{\mu_1\sigma_2}c\right)}{\frac{\mu_1^2}{\sigma_1^2}+1+c^2\left(\frac{\mu_2^2}{\sigma_2^2}+1\right)+\frac{2c\mu_1\mu_2'}{\sigma_1\sigma_2}}\right), \quad w_2 = c\frac{\sigma_1}{\sigma_2}\frac{\mu_1}{\sigma_1^2}\left(\frac{\left(1+\frac{\mu_2'\sigma_1}{\mu_1\sigma_2}c\right)}{\frac{\mu_1^2}{\sigma_1^2}+1+c^2\left(\frac{\mu_2^2}{\sigma_2^2}+1\right)+\frac{2c\mu_1\mu_2'}{\sigma_1\sigma_2}}\right).$$

□

C.3. Proof of Theorem B.8

Proof. Using Assumption B.6, the class of simple models only contains variant predictors, so ERM$(\mathcal{M}_s) = \Psi_s$. Consequently, the constraint in (3) can be written as $I(M; \Psi_s|Y) = 0$.

Considering the set of candidate predictors for M, namely $\{0, \Phi_c, \Psi_s, \Psi_c\}$, we examine the CMI constraint for each. Using the definition of mutual information, we have $I(0, \Psi_s|Y) = 0$ and $I(\Psi_s, \Psi_s|Y) = H(\Psi_s|Y)$. According to the definition of *variant* predictor, $H(\Psi_s|Y) > H(\Psi_s|Y, E) \geq 0$.

From Assumption B.7, which states that the invariant and variant predictors are conditionally independent, we can deduce that $I(\Phi_c, \Psi_s|Y) = 0$. From Assumption B.7, we also have $I(\Psi_c, \Psi_s|Y) > I(\Psi_c, \Psi_s|Y, E) \geq 0$.

Using these results, the feasible set is $[0, \Phi_c]$, which corresponds to the invariance set $\mathcal{I}_E(\mathcal{M})$. Consequently, problem (3) is equivalent to finding $\arg\max_{\Phi \in \mathcal{I}_E(\mathcal{M})} I(Y; \Phi)$. The solution to this problem is Φ_c, which represents the MIP Φ^*. □

D. Experimental Settings

We begin by describing some common details and notation that we use throughout this section. As in the main text, we use λ to represent the regularization strength for CMI. To ensure effective regularization, we adopt an epoch-dependent approach by scaling the regularization strength using the parameter S. Specifically, we set $\lambda = \lambda_c\,(1 + t/S)$ at epoch t. The temperature parameter T is set as 12.5 throughout the experiments. Additionally, we use LR to denote the learning rate, BS to denote the batch size, and λ_2 to denote the weight decay parameter, which represents the strength of ℓ_2-regularization. When using the Adam optimizer, we employ the default values for momentum.

The experiments on Slab data, CMNIST and CPMNIST data and Adult-Confounded data were implemented on Google Colab. The ImageNet-9 experiments were run on an AWS G4dn instance with one NVIDIA T4 GPU. For experiments on the subgroup robustness datasets and the Camelyon17-WILDS data, we used two NVIDIA V100 GPUs with 32 GB memory each. We only used CPU cores for the Bios data experiments.

D.1. Mitigating Simplicity Bias Experiments

This section includes the details for the experiments showing that CMID mitigates simplicity bias where we use the slab data and the ImageNet-9 data.

D.1.1. Slab Data

Dataset We consider two configurations of the slab data, namely 3-Slab and 5-Slab. In both the cases, the first feature is linearly separable. The second feature has 3 slabs in the 3-Slab data and 5 slabs in the 5-Slab data. Both the features are in the range $[-1, 1]$. The features are generated by defining the range of the slabs along each direction and then sampling points in that range uniformly at random. The base code for data generation came from the official implementation of [44] available at `https://github.com/harshays/simplicitybiaspitfalls`. We consider 10^5 training samples and 5×10^4 test samples. In both the cases, the linear margin is set as 0.05. The 3-Slab data is 10-dimensional, where the remaining 8 coordinates are standard Gaussians, and are not predictive of the label. The slab margin is set as 0.075. The 5-Slab data is only 2-dimensional, and the slab margin is set as 0.14.

Training We consider a linear model for the simple model and following [44], a 1-hidden layer NN with 100 hidden units as the final model (for both ERM and CMID). Throughout, we use SGD with BS $= 500, \lambda_2 = 5 \times 10^{-4}$ for training. The linear model is trained with LR $= 0.05$, while the NN is trained with LR $= 0.005$.

For the 3-Slab data, the models are trained for 300 epochs. We consider $\lambda_c \in \{100, 150, 200\}$ and choose $\lambda_c = 150$ for the final result. For the 5-Slab data, the models are trained for 200 epochs and we use a 0.99 momentum in this case. We consider $\lambda_c \in \{1000, 2000, 2500, 3000\}$ and choose $\lambda_c = 3000$ for the final result. Note that we consider significantly high values of λ_c for this dataset compared to the rest because the simple model is perfectly predictive of the label in this case. This implies that its CMI with the final model is very small, and the regularization strength needs to be large in order for this term to contribute to the loss.

D.1.2. TEXTURE VS SHAPE BIAS ON IMAGENET-9

Dataset ImageNet-9 [52] is a subset of ImageNet with nine condensed classes that each consist of images from multiple ImageNet classes. These include dog, bird, wheeled vehicle, reptile, carnivore, insect, musical instrument, primate, and fish.

Calculating shape bias The GST dataset [19] consists of 16 shape classes. To interpret a prediction from a model trained on ImageNet-9 as a GST dataset prediction, we consider a subset of classes from both and use the mapping listed in Table 9. Specifically, to determine whether a model trained on ImageNet-9 predicts correctly on a GST image, we first determine which of the 5 ImageNet-9 classes from the Table has the highest probability based on the model's output, and then use the mapping to obtain the predicted GST class label. Thus, following the procedure detailed in https://github.com/rgeirhos/texture-vs-shape, the shape bias is calculated as:

ImageNet-9 Class	GST Dataset Classes
dog	dog
bird	bird
wheeled vehicle	bicycle, car, truck
carnivore	bear, cat
musical instrument	keyboard

Table 9. ImageNet-9 classes mapped to corresponding GST dataset classes.

$$\text{shape bias} = \frac{\text{number of correct shape predictions}}{\text{number of correct shape predictions} + \text{number of correct texture predictions}}.$$

Training We use ResNet50 pretrained on ImageNet data as the simple model, and train it on ImageNet-9 using Adam with LR $= 0.001$, BS $= 32$, $\lambda_2 = 10^{-4}$ for 10 epochs. We do not consider a simpler architecture and training from scratch since this pre-trained model already exhibits texture bias. For the final model, we consider the same model and parameters, except we use SGD with 0.9 momentum as the optimizer and $\lambda_2 = 0.001$ for both ERM and CMID and train for 10 epochs. Values of CMID specific parameters were $\lambda_c = 15, S = 10$. For tuning, we consider LR $\in \{10^{-5}, 10^{-4}, 10^{-3}\}$ for both the models and BS $\in \{16, 32\}$, $\lambda_2 \in \{0.0001, 0.001, 0.01\}$ and $\lambda_c \in \{0.5, 15, 25, 50\}$.

D.2. Better OOD Generalization Experiments

This section includes the details for the experiments showing that our approach leads to better OOD generalization. For this, we used CMNIST and CPMNIST, Camelyon17-WILDS and Adult-Confounded datasets.

D.2.1. CMNIST AND CPMNIST

Dataset Following [3], we use $25,000$ MNIST images (from the official train split) for each of the training environments, and the remaining $10,000$ images to construct a validation set. For both the test sets, we use the $10,000$ images from the official test split.

Training The details about model architecture and parameters for training the simple model with ERM and the final model with CMID, for both the datasets, are listed in Table 10. Following [3], the MLP has one hidden layer with 390 units and ReLU activation function. In both the cases, the simple model is trained for 5 epochs, while the final model is trained for 20 epochs. We choose the model with the smallest accuracy gap between the training and validation sets. For tuning, we consider the following values for each parameter: for the simple model, LR $\in \{0.005, 0.01, 0.05\}$, $\lambda_2 \in \{0.001, 0.005, 0.01\}$, and for the final model, LR $\in \{0.001, 0.005\}$, $\lambda_c \in [3, 8]$ and $S \in [3, 6]$, where lower values of S were tried for higher values of λ_c and vice-versa.

Dataset	Simple Model	Optimizer	LR	BS	λ_2	Final Model	Optimizer	LR	BS	λ_c	S
CMNIST	Linear	SGD	0.01	64	0.005	MLP	SGD	0.001	64	4	4
CPMNIST	Linear	SGD	0.01	64	0.001	MLP	SGD	0.005	64	5	3

Table 10. Training details for CMNIST and CPMNIST.

For the final results, we report the mean and standard deviation by averaging over 4 runs. For comparison, we consider the results reported by [3] for all methods, except EIIL [10] and JTT [31]. Results for EIIL are obtained by using their publicly available implementation for CMNIST data (available at https://github.com/ecreager/eiil), and incorporating the CPMNIST data into their implementation. The hyperparameter values in their implementation are kept the same. We implement JTT to obtain the results. We consider LR $\in \{0.001, 0.005, 0.01\}$ and the reweighting parameter [31] (for upweighting minority groups) $\lambda_{up} \in \{5, 10, 15, 20, 25\}$ for tuning.

D.2.2. CAMELYON17-WILDS

Dataset Camelyon17-WILDS [28] contains 96×96 image patches which may or may not display tumor tissue in the central region. We use the same dataset as Bae et al. [3], which includes $302,436$ training patches, $34,904$ OOD validation patches, and $85,054$ OOD test patches, where no two data-splits contain images from overlapping hospitals. We use the WILDS package, available at `https://github.com/p-lambda/wilds` for dataloading.

Training For the simple model, we train a `2DConvNet1` model (see Section D.4.2 for details) for 10 epochs. We use the Adam optimizer with LR $= 10^{-4}$, BS $= 32$, $\lambda_2 = 10^{-4}$. For the final model, we train a DenseNet121 (randomly initialized, no pretraining) for 5 epochs using SGD with 0.9 momentum with LR $= 10^{-4}$, BS $= 32$, $\lambda_2 = 0.01$ and $\lambda_c = 0.5$, $S = 10$. We use the same BS and λ_2 values as [3] for consistency. For tuning, we consider LR $\in \{10^{-5}, 10^{-4}, 10^{-3}\}$ for both the models and $\lambda_c \in \{0.5, 2, 5, 15\}$ for CMID. While [3] and [28] use learning rates 10^{-5} and 0.001, respectively, we found a learning rate of 10^{-4} was most suited for our approach. We select the model with the highest average accuracy on the validation set for the final results. We report the mean and standard deviation by averaging over 3 runs. For comparison, we use the results reported by [3].

D.3. Adult-Confounded

Dataset The UCI Adult dataset [38; 30], comprises $48,842$ census records collected from the USA in 1994. It contains attributes based on demographics and employment information and the target label a binarized income indicator (thresholded at $\$50,000$). The task is commonly used as an algorithmic fairness benchmark. [29; 10] define four sensitive sub-groups based on binarized sex (Male/Female) and race (Black/non-Black) labels: Non-Black Males (G1), Non-Black Females (G2), Black Males (G3), and Black Females (G4). They observe that each sub-group has a different correlation strength with the target label ($p(y = 1|G)$), and thus, in some cases, sub-group membership alone can be used to achieve low error rate in prediction.

Based on this observation, [10] create a semi-synthetic variant of this data, known as Adult-Confounded, where they exaggerate the spurious correlations in the original data. As G1 and G3 have higher values of $p(y = 1|\text{G})$ across both the splits, compared to the other sub-groups (see [10] for exact values), these values are increased to 0.94, while they are set to 0.06 for the remaining two sub-groups, to generate the Adult-Confounded dataset. In the test set, these are reversed, so that it serves as a worst-case audit to ensure that the model is not relying on sub-group membership alone in its predictions. Following [10], we generate the samples for Adult-Confounded dataset by using importance sampling. We use the original train/test splits from UCI Adult as well as the same sub-group sizes, but individual examples are under/over-sampled using importance weights based on the correlation on the original data and the desired correlation.

Training We use a linear model (with a bias term) as the simple model, and following [10] use an Adagrad optimizer throughout. We use BS $= 50$. The simple model is trained for 50 epochs, with LR $= 0.05$, $\lambda_2 = 0.001$. Following [10; 29], we use a two-hidden-layer MLP architecture for the final model, with 64 and 32 hidden units, respectively. It is trained with LR $= 0.04$, $\lambda_c = 4$, $S = 4$ for 10 epochs. We also construct a small validation set from the train split by randomly selecting a small fraction of samples $(5 - 50)$ from each sub-group (depending on its size) and then upsampling these samples to get balanced sub-groups of size 50. We choose the model with lowest accuracy gap between the train and validation sets. For tuning, we consider LR $\in \{0.01, 0.02, 0.03, 0.04, 0.05\}$ and $\lambda_c, S \in \{3, 4, 5\}$. For the final results, we report the mean and accuracy by averaging over 4 runs. For comparison, we reproduced the results for ERM from [10], and thus, consider the values reported in [10] for the three methods.

D.4. Sub-group Robustness Experiments

This section includes the details for the experiments showing that CMID enhances sub-group robustness, where we use four benchmark datasets: Waterbirds, CelebA, MultiNLI and CivilComments-WILDS.

D.4.1. DATASETS

We consider the following datasets for this task. We follow the setup in [42] for the first three and [28] for CivilComments.

Waterbirds Waterbirds is a synthetic dataset created by Sagawa *et al.* [42] consisting of bird images over backgrounds. The task is to classify whether a bird is a *landbird* or a *waterbird*. The background of the image *land background* or *water*

background, acts a spurious correlation.

CelebA CelebA is a synthetic dataset created by Liu *et al.* [33] containing images of celebrity faces. We classify the hair color as *blonde* or *not blonde*, which is spuriously correlated with the gender of the celebrity *male* or *female*, as done in Sagawa *et al.* [42; 31].

MultiNLI MultiNLI [51] is a dataset of sentence pairs consisting of three classes: entailment, neutral, contradiction. Pairs are labeled based on whether the second sentence entails, is neutral with, or contradicts the first sentence, which is correlated with the presence of negation words in the second sentence [42; 31].

CivilComments-WILDS CivilComments-WILDS is a dataset of online comments proposed by Borkan *et al.* [6]. The goal is to classify whether a comment is *toxic* or *non-toxic*, which is spuriously correlated with the mention of one or more of the following demographic attributes: male, female, White, Black, LGBTQ, Muslim, Christian, and other religion [39; 14]. Similar to previous work [28; 31], we evaluate over 16 overlapping groups, one for each potential label-demographic pair.

D.4.2. Model Architectures

In this section, we discuss the architectures we consider for the simple models for this task. For the two image datasets, a shallow 2D CNN is a natural choice for the simple model as 2D CNNs can capture local patterns and spatial dependencies in grid-like data. On the other hand, for the two text datasets with tokenized representations, we consider a shallow FCN or 1D CNN for the simple model. FCNs can capture high-level relationships between tokens by treating each token as a separate feature, while 1D CNNs can capture local patterns and dependencies in sequential data.

Figure 6. Left: `2DConvNet1` and Right: `2DConvNet2` architectures.

Figure 7. Left: `2FCN` and Right: `1DConvNet` architectures.

Next, we describe the details for the model architectures. Let F denote the filter size and C denote the number of output channels (for convolutional layers) or the output dimension (for linear/FC layers). Throughout, we use $F = 2$ for the average pooling layers. Fig. 6 shows the `2DConvNet1` and the `2DConvNet2` architecture, which were used as simple models for Waterbirds and CelebA, respectively. These were the only two architectures we considered for the 2DCNN on these datasets. In `2DConvNet1`, the 2D convolutional layers use $F = 7, C = 10$ and $F = 4, C = 20$, respectively, while $C = 2000$ for the FC layer. In the `2DConvNet2` architecture, $F = 5, C = 10$ for the 2D convolutional layer and $C = 500$ for the FC layer. Fig. 7 shows the `2FCN` and the `1DConvNet` architecture, which were used as simple models for MultiNLI and CivilComments-WILDS, respectively. For tuning, we considered these models as well as a 1DCNN with one less 1D convolutional layer than `1DConvNet` for both the datasets. In `2FCN`, the FC layers use $C = 100$ and $C = 25$, respectively. In the `1DConvNet` architecture, the 1D convolutional layers use $F = 7, C = 10$, $F = 5, C = 32$ and $F = 5, C = 64$, respectively, while $C = 500$ for the FC layer.

D.4.3. Training Details

We utilize the official implementation of [42] available at `https://github.com/kohpangwei/group_DRO` as baseline code and integrate our approach into it. Most hyperparameter values are kept unchanged, and we list the the important parameters along with model architectures for all the datasets in Table 11. For the simple models, we consider shallow 2D CNNs for the image datasets, and FCN and 1D CNN for the text data, as discussed in the previous section. In all cases, the simple model is trained for 20 epochs. For the final model, following [42], we use the Pytorch `torchvision`

implementation of ResNet50 [23] with pretrained weights on ImageNet data for the image datasets, and the Hugging Face `pytorch-transformers` implementation of the BERT `bert-base-uncased` model, with pretrained weights [13] for the language-based datasets.

Dataset	Simple Model	Optimizer	LR	BS	λ_2	Final Model	Optimizer	LR	BS	λ_2	λ_c	S	# epochs
Waterbirds	`2DConvNet1`	Adam	10^{-5}	32	10^{-4}	ResNet50	SGD	5×10^{-4}	128	10^{-4}	20	4	100
CelebA	`2DConvNet2`	Adam	10^{-5}	32	5×10^{-4}	ResNet50	SGD	3×10^{-4}	128	0.001	10	5	50
MultiNLI	`2FCN`	Adam	0.005	16	10^{-4}	BERT	AdamW	5×10^{-5}	32	0	75	10	5
CivilComments	`1DConvNet`	Adam	10^{-4}	16	10^{-4}	BERT	AdamW	10^{-5}	32	0.001	25	10	10

Table 11. Training details for sub-group robustness datasets.

Table 12 shows the values of LR, λ_c and S we consider for tuning for the final model. Following [42], we keep $\lambda_2 = 0$ for MultiNLI. For the rest, we consider $\lambda_2 \in \{0.0001, 0.0005, 0.001\}$. For results, we choose the model with the best worst-group accuracy on the validation set. For comparison, we consider the values reported in [31].

Dataset	LR	λ_c	S
Waterbirds, CelebA	$[1, 5] \times 10^{-4}$	$\{10, 15, 20, 25, 50, 75\}$	$\{4, 5, 6, 8, 10\}$
MultiNLI, CivilComments	$\{1, 2, 5\} \times 10^{-5}$	$\{10, 25, 50, 75\}$	$\{10\}$

Table 12. Values considered for tuning the parameters for the final model for sub-group robustness datasets.

D.5. Bias in Occupation Prediction Experiment

Social Norm Bias [9] considers the task of predicting a person's occupation based on their textual bio from the Bios dataset [11]. Based on this task, [9] formalizes a notion of social norm bias (SNoB). SNoB captures the extent to which predictions align with gender norms associated with specific occupations. In addition to gender-specific pronouns, these norms encompass other characteristics mentioned in the bios. They represent implicit expectations of how specific groups are expected to behave. [9] characterizes SNoB as a form of algorithmic unfairness arising from the associations between an algorithm's predictions and individuals' adherence to inferred social norms. They also show that that adherence to or deviations from social norms can result in harm in many contexts and that SNoB can persist even after the application of some fairness interventions.

To quantify SNoB, the authors utilize the Spearman rank correlation coefficient $\rho(p_c, r_c)$, where p_c represents the fraction of bios associated with occupation c that mention the pronoun 'she', and r_c measures the correlation between occupation predictions and gender predictions. The authors employ separate one-vs-all classifiers for each occupation and obtain the occupation prediction for a given bio using these classifiers. For gender predictions, they train occupation-specific models to determine the gender-based group membership (female or not) based on a person's bio, and use the predictions from these models. A higher value of $\rho(p_c, r_c)$ represents a larger social norm bias, which indicates that in male-dominated occupations, the algorithm achieves higher accuracy on bios that align with inferred masculine norms, and vice versa.

Training We use a version of the Bios data shared by the authors of [11]. We used the official implementation of [9], available at `https://github.com/pinkvelvet9/snobpaper`, to obtain results and for comparison purposes. In this implementation, they consider 25 occupations and train separate one-vs-all linear classifiers for each occupation based on word embeddings to make predictions. We directly used their implementation to obtain results for ERM and Decoupled [16] on the data. For our approach, we employed linear models for both the simple model and the final model. We directly regularized the CMI with respect to the ERM from their implementation. The final model was trained for 5 epochs using SGD with LR $= 0.1$, BS $= 128, \lambda_c = 5, S = 5$. We only tuned the LR for this case, considering values of 0.05 and 0.1.

E. CelebA: Invariant and Spurious Features have Similar Complexity

In our subgroup robustness experiment for CelebA (Table 7 in Section 4.3), we found that our method did not yield a significant improvement in worst-group accuracy. We investigate this further in this section. We show that for the CelebA dataset, the complexity of the invariant and surrogate features are actually quite similar. The experiment is similar to the experiments we did for CMNIST and Waterbirds in Section 2. We create a subset of the CelebA dataset by sampling an equal number of samples from all four subgroups. Table 13 presents a comparison of the results when predicting the invariant feature (hair color) and the surrogate feature (gender) using the simple model (`1DConvNet`). We observe that the performance for both tasks is comparable, suggesting that the features exhibit similar complexity.

We contrast these results with those for CMNIST and Waterbirds in Table 2. For CMNIST and Waterbirds, there was a significant difference in the accuracy to which the simple model could predict the invariant and surrogate feature. For CelebA, the difference is much smaller which suggests that spurious features are *not* simpler than invariant features for this dataset—explaining why our method is not as effective for it.

Predict invariant feature		Predict surrogate feature	
Train	Test	Train	Test
89.1 ± 1.5	84.3 ± 0.6	92.4 ± 2.4	88.3 ± 1.6

Table 13. Comparison between performance for predicting the simple feature and the complex feature on CelebA dataset.

F. Broader Impacts

Our proposed CMID approach effectively mitigates simplicity bias, improves OOD generalization, and enhances sub-group robustness and fairness across various datasets. As a result, it can enhance the trustworthiness of machine learning models by promoting robust and fair predictions. However, robustness and fairness desiderata can be complex and vary significantly from application to application. Though our method does not require much prior knowledge of the biases in the data or the training procedures (e.g. which features are spurious), we do not believe that this should be regarded as a reason to not properly investigate what biases could potentially be present. Suitable problem-dependent metrics still need be defined based on the requirements of the application, and models need to be rigorously evaluated on these metrics before being deployed in consequential scenarios.