Boosting worst-group accuracy without any group annotations

Anonymous Author(s)
Affiliation
Address
email

Abstract

Despite having good average test accuracy, classification models can have poor performance on subpopulations that are not well represented in the training set. In this work we introduce a method to improve prediction accuracy on underrepresented groups that does not require any group labels for training or validation, unlike existing approaches. We provide a sound empirical investigation of our procedure and show that it recovers the worst-group performance of methods that use oracle group annotations.

1 Introduction

Machine learning models have been remarkably successful on a broad range of prediction task, when their performance is measured with respect to the average error on a holdout i.i.d. test. However, good performance on average does not guarantee that the error is indeed low on the various subpopulation of the data distribution. In applications where the accuracy on the subpopulations matters (e.g. instances where fair prediction is critical), other metrics than average validation error are more informative.

One common way for such subpopulations to emerge is when a prediction model relies on features spuriously correlated with the target variable to make predictions [Sagawa et al., 2020a, Goel et al., 2020, Geirhos et al., 2020]: such a model produces incorrect outputs on samples for which the spurious correlation does not hold, leading to subpopulations (groups) which are consistently predicted incorrectly. In Sagawa et al. [2020b] the authors show that overparameterized predictors obtained via empirical risk minimization (ERM) have poor performance on groups that cannot be predicted correctly using the spurious feature, if they are underrepresented in the training set. We argue that in scenarios with group imbalance, ERM does not arrive at a solution with good worst-group performance even for infinite training data. This suggests that we may need to optimize a different training objective in order to achieve good low worst-group accuracy.

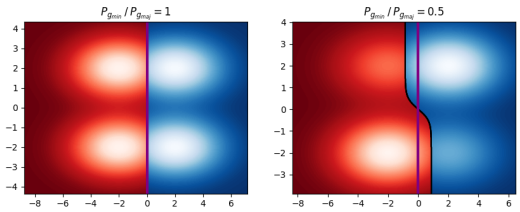


Figure 1: Decision Boundary of the Bayes classifer (**black**) aligns with the best worst-group classifier (**violet**) only for balanced subgroups (left). For imbalanced groups the Bayes classifier has low worst group performance (right).

A number of approaches have been proposed in order to alleviate the poor worst-group performance of ERM. Notably, all of them assume some level of access to labeled data with group annotations. Group distributionally robust optimization (GDRO) Sagawa et al. [2020a], reweighting or resampling Sagawa et al. [2020a,b], Menon et al. [2021] use a training set with group labels to improve worst-

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

group accuracy. Alternatively, Just Train Twice (JTT) Liu et al. [2021], Learning from Failure (LFF) Nam et al. [2020] only require group labels for the validation set to perform hyperparameter tuning. A few approaches Sohoni et al. [2020], Creager et al. [2021] achieve good worst-group performance without any group annotations, but they avoid the model selection step altogether and adopt the hyperparameters used by other works, which tune them using group labels.

While these approaches provide evidence that the poor worst-group performance of ERM can be overcome, they require collecting at least some group labels. This task is often challenging, as it is not always clear what the spurious features are (e.g. artifacts of an Xray machine) [Minderer et al., 2020]. Even if we know the protected attributes on which a model should not rely when making a prediction, it might not be possible to collect them during the data acquisition process. For instance, public or private institutions are often not allowed to hold records on sensitive information such as gender, race, religion etc. In these situations when group annotations are not available, the algorithms mentioned above cannot be applied out-of-the-box. In previous work Liu et al. [2021] the importance to overcome this label dependence is also emphasised.

Contributions. In this work we set out to address this issue and propose a two-stage procedure, similar to JTT, but for which we can perform training and model selection in the absence of group labels for either the training or the validation data. In the first stage we select a particularly biased predictor from a heavily regularized function class. Similarly to bootstrap methods Breiman [1996], we seek to correct the mistakes made by the biased predictor in the second stage. Since we do not rely on validation data with group labels, we devise a proxy criterion for model selection and show that it correlates well with worst group accuracy. We show that our method significantly outperforms ERM and achieves similar performance to approaches that require group annotations on a variety of prediction tasks for image and tabular data sets.

2 Problem setting

In this section we describe the setting that we assume throughout our work. We adopt the data model described in Sagawa et al. [2020b] which assumes that samples are drawn from a mixture of groups, i.e. $x \sim P_X = \sum_{g \in \mathcal{G}} P[X|G=g]P_g$ with $\mathcal{G} = \{g_1, ..., g_G\}$ the finite set of groups. We consider classification problems in which all the samples from one group are assigned the same ground truth label and a class can comprise several groups. Elements from different groups differ by features that do not reflect the concept of the prediction task, these features are called spurious.

In Figure 1 we illustrate the setting for the simple case of 2D binary data with 2 groups per class and isotropic Gaussian group conditional distributions. If the groups are balanced (i.e. $P_{g_i} = P_{g_j}, \forall i, j \in [1, G]$), then the Bayes classifier f_{Bayes} has simultaneously both optimal average and worst-group error. However, if some groups are significantly more represented in the training distribution than others (i.e. $P_{g_i} >> P_{g_j}$ for some $i, j \in [1, G]$), then f_{Bayes} is optimal with respect to the average error, but does not have the best error among all possible classifiers on the minority groups (Figure 1 right). This relationship between the worst-group and the average accuracy for scenarios with group imbalance is reminiscent of other well-studied trade-offs such as the one involving the standard and the adversarial robust risk [Tsipras et al., 2019]. This argument indicates that, even given infinite training data and with a well-specified hypothesis class, ERM would still arrive at a solution with subpar worst-group performance. Instead, the worst-group optimal classifier aligns with the Bayes classifier on a reweighted distribution. Prior works propose implicit (e.g. GDRO [Sagawa et al., 2020a]) and explicit (e.g. generating minority group samples [Goel et al., 2020], subsampling majority groups [Menon et al., 2021],[Sagawa et al., 2020b]) ways to "alter" the training distribution and demonstrate their efficacy. However, these approaches rely on access to group annotations for the training set. While this stringent requirement is relaxed in Liu et al. [2021],Nam et al. [2020],Creager et al. [2021] to only using the group labels of the validation set for model selection.

In this work we propose a two-stage procedure, similar to boosting algorithms, that optimizes a reweighted objective without any group annotations, which ultimately allows it to learn a classifier with improved worst group accuracy.

3 Proposed method

We now provide an overview of the two stages of the procedure that we propose. Our algorithm is summarized in Algorithms 1, 2.

First stage. We begin by training a biased classifier, with the goal of using its error set (i.e. the set of validation points on which it predicts incorrect outputs) as a stand-in for the minority groups. JTT

Algorithm 1: First stage

Input : Train set S, Validation set V, Ensemble size K
Result: Train error sets $\{\xi_{S_i}\}_{i=1}^K$; Val error sets $\{\xi_{V_i}\}_{i=1}^K$
$HP \leftarrow$ Do *hyperparameter tuning* with 2-fold CV on S
for *i in K* **do** `// train K models`
  $S_T, S_V \leftarrow$ *Split, 0.5*(S)
  $f_s \leftarrow$ *train one epoch from scratch* (S_T, HP)
  $\xi_{S_i}, \xi_{V_i} \leftarrow$ *get error sets* (f_s, S_V, V)
return $\{\xi_{S_i}\}_{i=1}^K,\ \{\xi_{V_i}\}_{i=1}^K$

Algorithm 2: Second Stage

Input : Train set S, Validation set V, Train error sets $\{\xi_{S_i}\}_{i=1}^K$; Val error sets $\{\xi_{V_i}\}_{i=1}^K$
Result: Core Feature Model f_c
$f_c \leftarrow$ *Initialize*(W)
for *k in max epochs* **do** `// train core model`
  $f_c \leftarrow$ *train one epoch* $\left(f_c, S_T, \{\xi_{S_i}\}_{i=1}^K\right)$
  for ξ *in* $\{\xi_{V_i}\}_{i=1}^K$ **do** `// compute metric`
    $acc_i \leftarrow$ *Acc*(f_c, ξ)
  early stop on Metric$(mean(\{acc_i\}_{i=1}^K))$
return f_c

relies on a similar procedure for its first stage, but depends on selecting the hyperparameters of the biased classifier using a validation set with group annotations. Instead, we avoid the need for group labels and build on the observation by [Nam et al., 2020] that the loss on majority samples is reduced first during NN training. We use and amplify this effect by limiting training time to one epoch and using the average accuracy for model selection. We perform 2-fold cross validation to collect the samples with incorrect predictions from the entire training data.

Second stage. Once we have selected the subset of the training data on which the biased estimator makes mistakes, we use the standard procedure of up-weighting the train error set examples to correct these mistakes. Once again, the key difference between our approach and JTT lies with how we perform hyperparameter tuning. In particular, instead of using the worst-group accuracy measured on an annotated validation set, we propose a proxy criterion for model selection. Intuitively, we could use the error set of the validation data as a substitute for the minority group and use the accuracy on this set to select hyperparameter values. However, numerous majority samples make their way into the error set. In order to reduce their effect, we use the error sets from K different biased predictors and compute the average error set accuracy for model selection:

$$f^* = \arg\max_{f \in \mathcal{F}} \left(\frac{1}{K} \sum_{i=1}^{K} \frac{1}{|\xi_i|} \sum_{(x,y) \in \xi_i} \mathbb{1}[f(x) = y] \right)$$

This criterion can be interpreted as a "soft" set intersection over all error sets: samples that occur in several of the error sets are weighted more than samples that only appear in a few error sets. Mistakes on majority samples are mostly random effects and do not generalize to different models. Mistakes on minority points are a result of the inductive bias and thus present in most models. This effect is enhanced by explicitly adding the intersections of the error sets to the procedure.

4 Experiments

We now show empirical results obtained with our procedure and provide insight into the role played by each of the stages.

Data sets. We highlight that one inconspicuous pitfall when developing new methodology is inadvertent overfitting, that is tailoring a method to particular data sets. In order to prevent this, we only use two data sets for designing and analyzing our approach (i.e. Corrupted MNIST and Waterbirds), and then run it on a number of other new data sets from various domains, ranging from tabular data (e.g. the Adult data set) to natural images (e.g. CelebA). We stress that we do not change our approach in any way in order improve performance on the unseen data sets.

Baselines. There is no previous work that tackles the challenge of worst subgroup performance without using any subgroup information. Thus, we compare our method to baselines with different degree of required group label information: ERM w/o group labels, ERM tuned with a validation set with group annotations (ERM WG), JTT and GDRO. Detailed information on the tuning of the baselines can be found in Appendix D.

Main results. The results of our empirical analysis are summarized in Table 1. We point out that our method consistently outperforms the ERM baseline, the only other method that does not use any group annotations for training or model selection. Moreover, our approach often achieves a

worst-group accuracy comparable to JTT, which has the advantage of oracle group label knowledge for the validation data.

Table 1: Average and worst-group test accuracy for methods that use group labels to varying degrees.

	Tuning	Corrupt-MNIST Avg	Corrupt-MNIST Wg	Waterbirds Avg	Waterbirds Wg	CelebA Avg	CelebA Wg	Color MNIST Avg	Color MNIST Wg	Adult Avg	Adult Wg	Poverty Avg	Poverty Wg
No group labels	ERM	99.8	88.0	97.9	79.2	95.4	45.0	99.8	84.0	84.5	7.5	88.8	48.4
	Ours	99.6	**93.6**	96.5	**84.8**	85.9	**82.1**	99.4	**96.4**	73.7	**63.8**	84.0	**51.8**
Val group labels	ERM WG	99.7	81.4	97.6	86.7	93.1	77.8	99.7	82.0	78.9	61.2	87.7	51.5
	JTT	99.5	**95.0**	93.3	**86.7**	88.0	**81.1**	98.3	**94.8**	77.8	**63.3**	64.5	**60.5**
Train & val group labels	GDRO	96.5	96.0	89.6	89.4	94.3	93.4	98.2	93.1	76.8	71.4	73.5	67.5

4.1 Ablation studies

In this section we break down our approach and give insights into the different parts of the algorithm.

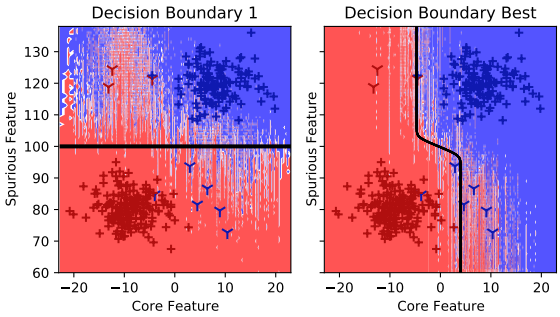


Figure 2: Prediction regions of a classifier (background) after one epoch (left) and at convergence (right). The black line shows the classifier that relies on the spurious feature (left) and the Bayes classifier (right).

Biased classifier. We being with the first stage and show that neural networks (NNs) early stopped after one epoch are indeed good at selecting minority points in their error set. This point is illustrated in Figure 2 where we train LeNet5 to perform even/ odd digit classification on the Color MNIST dataset. We project the input space into a two-dimensional representation where one of the coordinates correlates with the core features, and the other with the spurious features (see Appendix C) for more details on how we obtained the projection). As shown in Figure 2 our biased classifier tuned for average accuracy after one epoch will be more aligned with a predictor that solely relies on spurious features, meaning that it will produce incorrect predictions primarily on the minority samples. Conversely, a classifier trained to convergence will recover the Bayes optimal predictor that does not make mistakes uniformly over the minority groups, leading to error sets that do not contain representative minority samples.

Minority sample selection. Furthermore, even though our final goal is not to correctly identify minority samples, we show in Appendix F how the precision and recall of our approach's error set compare to those of JTT. In particular, we note that with our procedure for hyperparameter tuning we generally achieve a larger precision, while JTT, having the advantage of the group labels on the validation set, manages to obtain higher recall. We stress however that, as both JTT and our method show, an imperfect error set (i.e. with low precision and/or low recall) does not necessarily mean low worst-group accuracy at the end of the second stage.

Proxy metric. For the second stage we analyze the relationship between the proxy criterion that we propose for hyperparameter selection and the worst subgroup validation accuracy. For corruption MNIST (correlation 0.971) and waterbirds (correlation 0.948) our proxy metric is indeed highly correlated with the actual metric of interest (see Appendix G for further results and visualizations). Therefore, our approach selects in general hyperparameter values that are good for worst-group performance.

5 Discussion and future work

Our current approach relies on a number of implicit assumptions. For instance, when selecting the error set in the first stage, data with large amounts of label noise could potentially render our method infeasible. Even though this is not the case for CelebA, which contains a fair amount of labels noise, we leave a more thorough investigation on ways to prevent the catastrophic effects of noise for future work. Moreover, we implicitly assume that we know which classes contain minority groups, which helps especially with model selection in the second stage of our method. Relaxing this assumption is another interesting direction for future research. In summary, we introduce a novel strong baseline for worst-group accuracy which requires no group label information. We hope that our results will encourage more research in this direction of crucial practical importance.

References

Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning, 2021.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix Wichmann. Shortcut learning in deep neural networks. *ArXiv*, abs/2004.07780, 2020.

Karan Goel, Albert Gu, Yixuan Li, and Christopher Ré. Model patching: Closing the subgroup performance gap with data augmentation, 2020.

Liangxiao Jiang and Chaoqun Li. Scaling up the accuracy of decision-tree classifiers: A naive-bayes combination. *J. Comput.*, 6(7):1325–1331, 2011.

Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9012–9020, 2019.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts, 2021.

Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed H. Chi. Fairness without demographics through adversarially reweighted learning, 2020.

Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL `http://yann.lecun.com/exdb/mnist/`.

Yann LeCun et al. Lenet-5, convolutional neural networks.

Evan Zheran Liu, Behzad Haghgoo, Annie S. Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information, 2021.

Aditya Krishna Menon, Ankit Singh Rawat, and Sanjiv Kumar. Overparameterisation and worst-case generalisation: friend or foe? In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=jphnJNOwe36`.

Matthias Minderer, Olivier Bachem, Neil Houlsby, and Michael Tschannen. Automatic shortcut removal for self-supervised representation learning. In *ICML*, 2020.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: Training debiased classifier from biased classifier, 2020.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization, 2020a.

Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations, 2020b.

Nimit S. Sohoni, Jared A. Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems, 2020.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *arXiv: Machine Learning*, 2019.

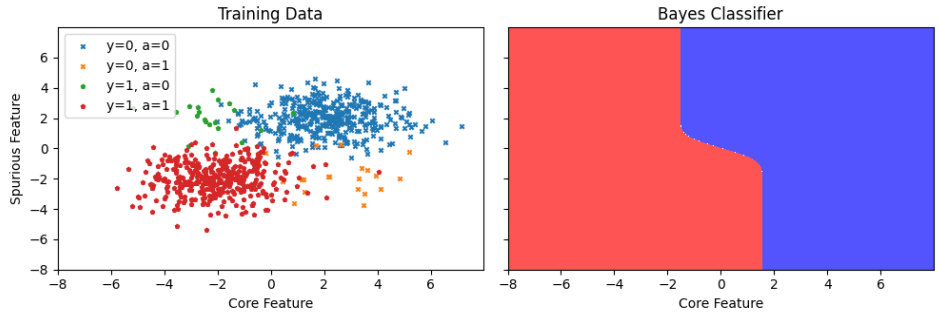


Figure 3: Synthetic data & Bayes classifier from the Gaussian mixture model with minority subgroups.

A Optimality in the presence of subgroups

In this section we conceptually and theoretically analyse a data model for the subgroup performance challenge. In this discussion we have two goals: First, analyse fundamental performance limits of the Bayes classifier to subgroup performance for certain data generating processes. Second, analyse to what extend this fundamental limit is relevant for the behaviour of over-parametrized models and in that light contribute a different perspective to the question of memorization and feature learning raised in Sagawa et al. [2020b] and Menon et al. [2021].

To achieve this goals, we introduce and parametrize an explicit example of the data generating process in form of a Gaussian mixture model (GMM). For this GMM we demonstrate that the Bayes classifier has low subgroup performance; show that an over-parametrized random feature model learns a decision rule that is very similar to the Bayes classifier in the Gaussian mixture setting with rare subgroups.

B Explicit data model

Following this general description of the data generating process we define a concrete parametric example. We take the data generating model used in Sagawa et al. [2020b] to get sample data for experiments on implicit memorization. In this case each conditional distribution is a Gaussian with diagonal co-variance matrix. This "Gaussian mixture with minority subgroups" is defined as the following.

Definition B.1 (Gaussian Mixture with Minority Subgroups)**.** *The data of our Gaussian mixture with minority subgroups $x \in \mathbb{R}^d$ with $d \in 2\mathbb{Z}^+$ is distributed as a mixture of four Gaussian with each of them distributed according to $p_x(x|s,c) \sim \mathcal{N}(\mu_{s,c}, \Sigma)$. Of the d dimensions $d/2$ are spurious features $x_s \in \mathbb{R}^{d/2}$ and $d/2$ are core features $x_c \in \mathbb{R}^{d/2}$. The mean values $\mu_{s,c}$ are dependent on the core $c \in \{0,1\}$ and the spurious category ($s \in \{0,1\}$). For simplicity of calculation we assume symmetry around $\vec{0}$ with the centers $\mu_{0,0} = [\vec{k}, \vec{k}]$, $\mu_{1,1} = [-\vec{k}, -\vec{k}]$, $\mu_{1,0} = [-\vec{k}, \vec{k}]$, $\mu_{0,1} = [\vec{k}, -\vec{k}]$. The variance $\Sigma = \operatorname{diag} \vec{\sigma}_s, \vec{\sigma}_c$ is a diagonal matrix with the spurious variance $\sigma_s < \sigma_c$. The class conditional distribution are mixtures of two of the groups each:*

$$
\begin{aligned}
p(x|y=0) &= p_x(x|a=1,c=0)p(a=1|c=0) + p_x(x|a=0,c=0)p(a=0|c=0) \\
p(x|y=1) &= p_x(x|a=1,c=1)p(a=1|c=1) + p_x(x|a=0,c=1)p(a=0|c=1)
\end{aligned} \tag{1}
$$

We define that each class label contains a minority group $0 < p(a=1|c=0) \ll p(a=0|c=0)$ and $0 < p(a=0|c=1) \ll p(a=1|c=1)$.

For this model it is simple to calculate the Bayes classifier and analyse its properties. Using Bayes rule we have

$$
C^{\text{Bayes}}(x) = \arg\max_{y \in \{0,1\}} p(y|x) = \arg\max_{y \in \{0,1\}} p(x|y)p(y) \tag{2}
$$

Table 2: Performance of overparametrized models in comparison to the Bayes Classifier in the Gaussian Mixture with minority subgroup dataset

	Bayes		Overparametrized	
Minority group ratio	Average	Worst	Average	Worst
0.5	0.843	0.836	0.798	0.766
0.7	0.858	0.693	0.814	0.645
0.9	0.928	0.445	0.905	0.425
0.99	0.991	0.076	0.989	0.027

In this simple case of binary classification, the classifier can be rewritten as an inequality.

$$C^{\text{Bayes}}(x) = \begin{cases} \{0\} & \text{if } p(x|y=0)p(y=0) > p(x|y=1)p(y=1) \\ \{1\} & \text{if } p(x|y=0)p(y=0) < p(x|y=1)p(y=1) \\ \{0,1\} & \text{if } p(x|y=0)p(y=0) = p(x|y=1)p(y=1) \end{cases} \tag{3}$$

The worst subgroup performance for this case transforms to:

$$\min_{g \in \mathcal{G}} \mathbb{E}_{(x,y)\sim P_g}\left[C^{\text{Bayes}}(x) = y\right] = \min_{c\in\{0,1\}, a\in\{0,1\}} \int_{x\in\mathbb{R}^d} C^{\text{Bayes}}(x) p(x|c,a)\, dx \tag{4}$$

The integration cannot be solved analytically for Gaussians, but numerical approximations are simple especially for low dimensions. In high dimensions one can use Monte Carlo simulations to get a good approximation of the integral. We compute numerical values for $d = 2$ through numerical double integration computation. In the first step we analyse the worst group and average accuracy values that Sagawa et al. [2020b] observed for overparametrized models. While they attributed low worst group performance to the overparametrized models, we hypothesize that the accuracy values are not a model phenomenon but correspond to properties of the underlying data distribution and the optimization objective. To have exact results we carry out the analysis for $d = 2$. The analysis is to change the relative probability of minority points in the range from $[0.5, 0.99]$ and analyzing different values of variance for σ_c. The resulting graph is displayed in (4). The general tendency aligns with Sagawa et al. [2020b] who described that increasing label spurious feature correlation leads to a sharp drop in worst subgroup performance of overparametrized models. We see the same change for the Bayes classifier (compare 4). For the variance analysis overparametrized behaviour and Bayes classifier align. In addition it is interesting to look at the development of the accuracy based purely on the core feature vs the overall accuracy.

Additionally, we compute numerical approximations through sampling for the exact settings from Sagawa et al. [2020b] with $d = 200$ and evaluate the over-parametrized random feature model from their work (a logistic regression on a random feature model of size 3500) in terms of worst subgroup and average accuracy. We then compare it to the Bayes classifer. We find that the over-parametrized model almost attains the performance of the Bayes classifier in terms of worst subgroup as well as average accuracy (compare 2). Consequently, we conclude that worst subgroup performance can be a consequence of Bayes Error minimization on the data distribution.

The experiments and analysis demonstrate that in the synthetic setting of Sagawa et al. [2020b] the Bayes Classifier has low worst subgroup performance and that the low worst subgroup performance of overparametrized models is not due to problematic model behaviour but due to the underlying data distribution. If improving worst subgroup performance is the target, it is important to realize that there usually is a trade-off with average accuracy. Thus, the optimization target needs to be shifted away from the cross entropy loss as a proxy for the Bayes Error to achieve good worst subgroup performance.

C Mapping images to spurious-core representation

In a general setting it is computationally infeasible to analyse distributional properties over image spaces. Especially, if the data is not generated with full control over the manifold. However, we utilize our assumptions on the underlying data distribution and limited control over the data generating process to get a 2D- representation of core and spurious features for some image datasets. This

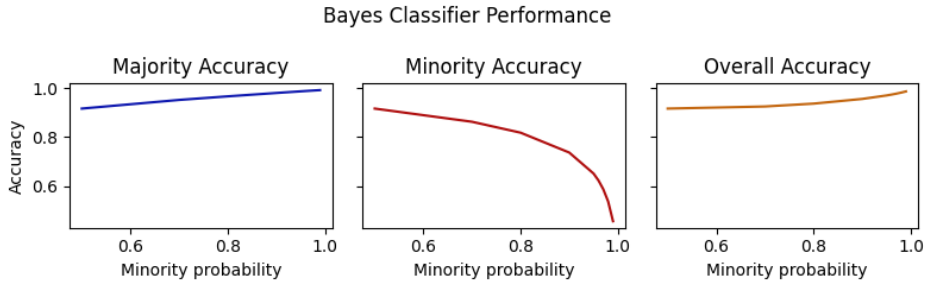


Figure 4: Bayes optimal classifier accuracy for different label spurious factor correlations. The minority accuracy is strongly affected by the correlation while the majority accuracy is less affected. The overall accuracy increases with an increase in label spurious feature correlation.

analysis is only feasible if core and spurious feature can be manipulated separately, this is the case for the worst subgroup data-sets of color-MNIST and waterbirds. To facilitate the analysis it is necessary to generate a mapping from the image domain x to the core feature x_c and the spurious feature x_s which are assumed to be one dimensional. Given this mapping it is possible to estimate the distributions $p(x_c)$ and $p(x_s)$, visualize the Bayes Classifier, compute its accuracy values and compare it with visualization and accuracies of CNN classification models. Furthermore, it is possible to get an intuition of our models behaviour in the first stage.

C.1 Identifying Core and Spurious Feature Distribution

First, we train a mapping f from the core features to a one dimensional representation. The closer f represents the ranking of the ground truth conditional distribution $p(y|x_c)$ meaning $p(y=1|x_c^1) > p(y=1|x_c^2) \implies f(x_c^1) > f(x_c^2)$ of the labels given the input, the better. In practice, we take the training images from MNIST, add a neutral background color to all images and train a LeNet5 LeCun et al. to minimize the cross entropy loss on the odd / even classification task. For waterbirds, we take the cropped bird images and place them on a black background. We take the non-normalized output of the last layer to get the mapping

$$f : [0,1]^{\text{dim}_1 \times \text{dim}_2} \to \mathbb{R} \quad f(x) = x_c \tag{5}$$

Second, for color-MNIST we define a bijection between the spurious feature color and a one dimensional space by replacing the varying color channel with a value between 60 and 120.

$$g^{-1} : [60, 120] \to [0, 255]^3 \quad g^{-1}(\lambda) = [100, \lambda, 100] \tag{6}$$

For waterbirds we repeat the first step, but instead of having a classifier learning the core feature, this time the classifier learns the spurious feature. We take only the background images of water and land and train a classifier to classify land and water correctly with a mapping:

$$g : [0,1]^{\text{dim}_1 \times \text{dim}_2 \times 3} \to \mathbb{R} \quad g(x) = x_s \tag{7}$$

Third, we combine this two mappings and create a complete mapping of the input images to a 2d representation with the spurious feature and the core feature on different axis.

$$h : [0,1]^{\text{dim}_1 \times \text{dim}_2 \times 3} \to (\mathbb{R},\ [0,1]) \quad h(x) = \begin{pmatrix} f(x) \\ g(x) \end{pmatrix} = \begin{pmatrix} x_c \\ x_s \end{pmatrix} \tag{8}$$

Fourth, we create an artificial grid of images by combining the input factors. For color-MNIST it is simple to add a different background color to the images from the validation set. For waterbirds we repeat the original data creation pipeline from Sagawa et al. [2020a] and place the birds in the centre of the landscapes of the background. We assume that x_c and x_s remain identical when we combine the inputs. For color-MNIST this assumption is satisfied. For waterbirds some issues arise when the bird in the foreground covers important parts of the background, or the background itself contains birds or birdlike objects. However, this issue arises rarely.

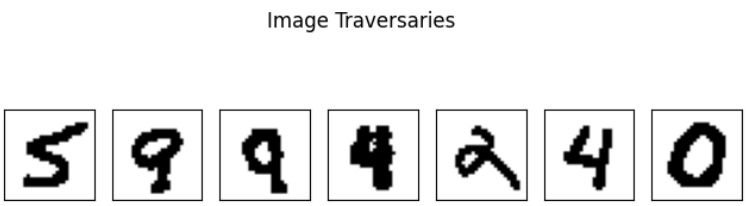


Figure 5: Image traversal for MNIST from clearly odd numbers to clearly even numbers with a ambiguous region in the middle

C.2 Analysis of the Representation Quality

The analysis of the mapping can be conducted somewhat similar to the analysis of disentanglement features in VAE. However, here images are not created but sorted in a learned representation space. The first method is to create a traversal starting with images that have a very low expression of the core feature over ambiguous examples to very high expression of the core feature. We can visually check if the hierarchy matches with the human perception. The second method is to check for model consistency. Namely, how strong do models learned with different initialization differ in their recovered distribution.

The traversal in 5 shows that the middle image is clearly ambiguous and the intended number could be even (4) or odd (9). And the numbers to the right and left look less stereotypical and clearly written than the ones at the corners. We check three model random seeds and compute the pairwise pearson correlation for the data points conditioned on the ground truth class label. This is important as for us it is not only relevant if the models learn the a similar classification rule, but also if the feature expression within each class carries some ground truth. We find significant correlation of above 80%. The results for these consistency check appear promising and indicate that the learned mapping can indeed be used for qualitative and with some uncertainty even quantitative analysis.

D Hyperparameter Tuning

D.1 Baselines:

For the ERM baselines we tune over learning rate, batch size, weight decay and if the dataset has class imbalance over reweighted and standard cross entropy loss. For the average baselines we select the best average accuracy at the best early stopping time with respect to average accuracy on the validation set.

For the ERM baselines with worst group early stopping we use the same tuning procedure. The only adaptation is that we select the model and early stopping time based on the best worst subgroup accuracy on a validation set. It is important to recognize that the high worst group accuracy is not consistent over runs with different seeds. In this setting high worst subgroup accuracy arises as a random deviation from the optimization target. Thus, hyperparameter search to a certain extent is the search over the best seed and parameter configuration. However, the good worst group performance generalizes to test data.

For the JTT baselines with worst group early stopping we have a larger grid in addition to the standard optimizer parameters, learning rate, batch size and weight decay there are the method specific selection epoch and upweight factor. Selection epoch is the epoch from which training mistakes of the first stage are selected for upweighting in the second stage. The up weight factor is the weighting that these samples receive during training of the second stage. We select the best model at best early stopping time based on worst subgroup performance on the validation set.

For the GDRO baselines with worst group early stopping we perform a grid search that in addition for the standard optimizer parameters, learning rate, batch size and weight decay selects the best value

for generalization adjustment. This factor for up-weighting the group loss relative to the square root of inverse group size has a large impact on model performance.

D.2 Ours:

For our method hyperparameter tuning is carried out in two stages. In the first stage we tune for overall accuracy on the validation set with models training for one epoch. The grid consists of different values for learning rate, batch size, optimizer and weight decay. We carry out hyperparameter tuning in a two fold cross validation because each model in our final ensemble is also trained on half of the data.

In the second stage we tune model selection and early stopping for the worst group average error set accuracy. In addition to the standard parameters of learning rate, weight decay and batch size, we tune for the best up weighting factor of the error set samples.

E Label Noise/ No subgroups

By design our procedure assumes a minority group in each label category. If this assumption is violated, the error set for the class without minority subgroups will consist of samples that are not classified correctly after one epoch for other reasons than the misleading spurious feature. In the smooth intersection proxy the samples classified wrongly by all the ensemble models will dominate the metric. Optimizing for this metric will inevitable lead to poor average performance and no improvement on the worst subgroups of the other label classes.

The violation of the "minority in each class assumption" can be identified by a sharp decrease in average accuracy to achieve proxy values close to or below random. If this violation is detected in one class the proxy metric should be limited to the other classes.

If there are minority subgroups in the class uniform label noise is less of a problem. The error set will be a mixture of minority and label noise samples. The label noise samples are in general impossible to optimize for and will thus cause the minority accuracy signal to be more noisy.

The boosting and upweighting of the label noise samples during training seem to be even less of a problem. In Celeba, where we almost exclusively upweight label noise for the class of non-blond people the achieved overall and average accuracy are still high.

F Precision Recall

Our procedure is targeted at achieving high majority accuracy in the first stage. Thus, avoiding many majority points in the error set and achieving high precision. However, for some datasets one epoch is enough for well optimized classifiers to pick up on the core signal as well and classify minority points correctly reducing the recall.

		Class 0		Class 1	
		Precision	Recall	Precision	Recall
Poverty	Ours	0.560	0.100	0.478	0.593
	JTT	0.268	0.387	0.69	0.541
CelebA	Ours	-	-	0.177	0.166
	JTT	-	-	0.09	0.94
Waterbirds	Ours	0.394	0.684	0.236	0.839
	JTT	0.27	0.85	0.191	0.875

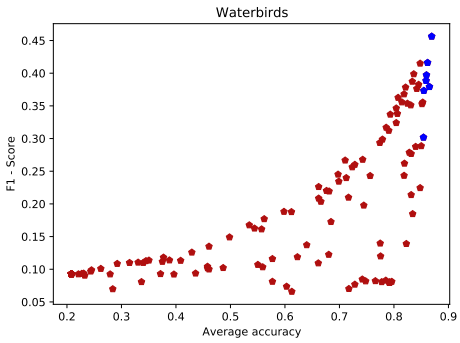


Figure 6: Waterbirds: F1 score for the identification of minority points with different first stage hyperparameters. The 5% highest overall accuracy runs are marked in blue.

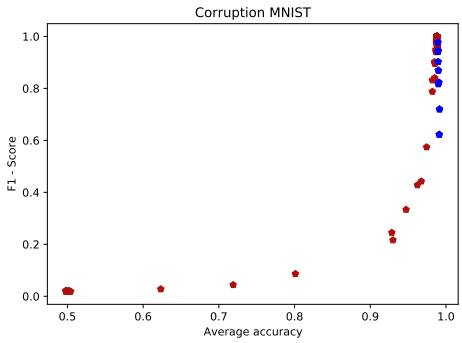


Figure 7: Corruption MNIST: F1 score for the identification of minority points with different first stage hyperparameters. The 5% highest overall accuracy runs are marked in blue.

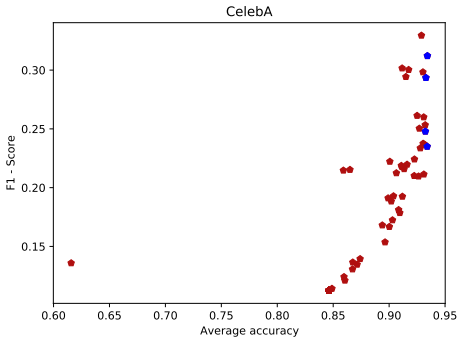


Figure 8: CelebA: F1 score for the identification of minority points with different first stage hyperparameters. The 5% highest overall accuracy runs are marked in blue.

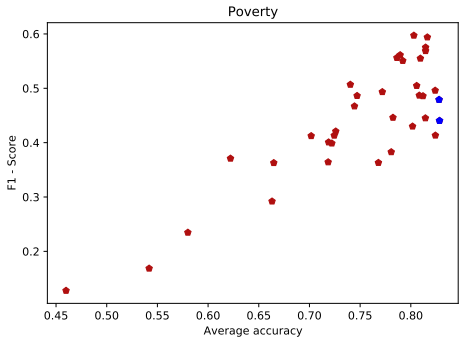


Figure 9: Poverty: F1 score for the identification of minority points with different first stage hyperparameters. The 5% highest overall accuracy runs are marked in blue.

G Proxy Metric

For the datasets analyzed we can see a high correlation between the average error set accuracy and the ground truth worst group metric. Furthermore, we highlight that part of the deviation can be attributed to random effects, as the validation set contains few minority points. This effect is shown in the left panel of the images for CelebA, corruption MNIST and Waterbirds, where the test sets contain considerably more samples from the minority groups. The relation of the worst group accuracy computed on few samples in the validation set to the worst group accuracy computed on the larger test set is visualized.

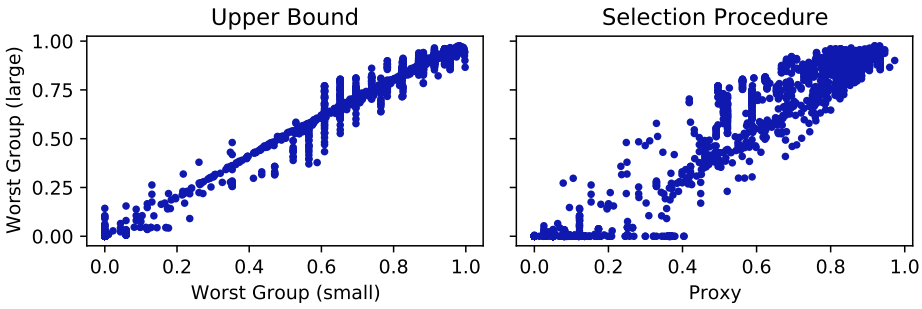


Figure 10: Corruption MNIST: Worst-group accuracy on the validation data vs. ground truth worst-group accuracy on the test data (left). Average error set accuracy vs. ground truth worst-group on the test data (right)

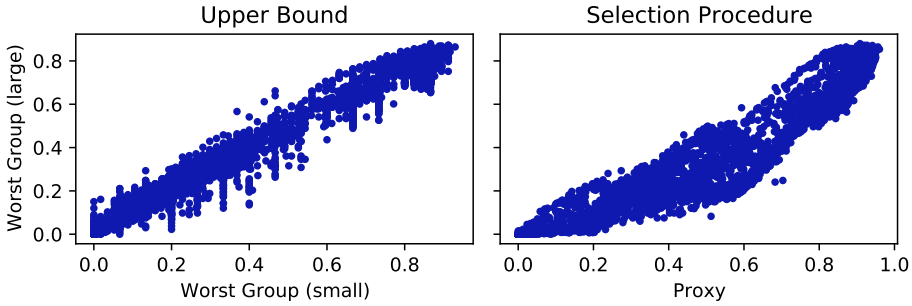


Figure 11: Waterbirds: Worst-group accuracy on the validation data vs. ground truth worst-group accuracy on the test data (left). Average error set accuracy vs. ground truth worst-group on the test data (right)

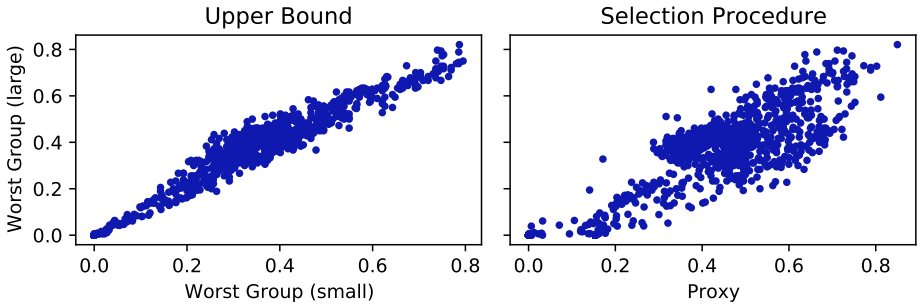
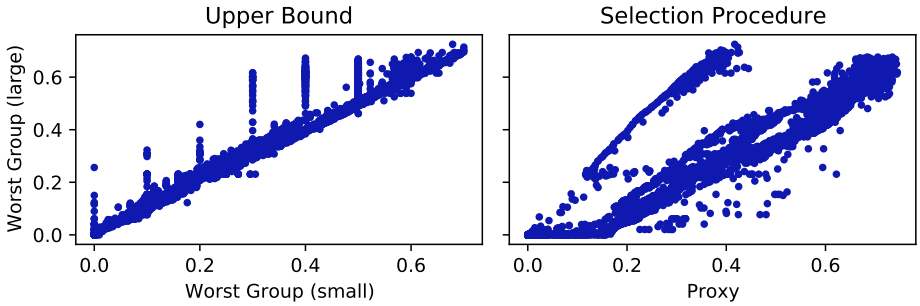


Figure 12: CelebA: Worst-group accuracy on the validation data vs. ground truth worst-group accuracy on the test data (left). Average error set accuracy vs. ground truth worst-group on the test data (right)

For the adult dataset, the test and validation set do not differ significantly in number of minority samples. Thus, the comparison is a visualization of the difference between two samples of equal size.

In general the visualization appears differently from the ones above. This can be attributed to the fact that the adult dataset contains four groups per class.

Figure 13: Adult: Worst-group accuracy on the validation data vs. ground truth worst-group accuracy on the test data (left). Average error set accuracy vs. ground truth worst-group on the test data (right)



H Datasets

Waterbirds: Designed by Sagawa et al. [2020a] to anaylze ERM behaviour with respect to spurious and core features the waterbirds dataset is a binary classification task of birds in to land birds and water birds. The dataset is synthetically created by extracting these birds from their original fotos and putting them in front of water or land backgrounds. The spurrious feature is the background with waterbirds in front of water in 95% of their occurence and landbirds in front of land in 95% of their occurence. While the classifier has to learn features related to coloring and form of the bird to correctly classify the images, the easy shortcut is to follow the color and structure of the background. The official publication comes with train, validation and test split and the experiments follow these official splits with one adaptation. The official validation and test sets are balanced with respect to subgroups for accurate estimation of worst subgroup performance. For our approach a iid validation set is necessary that was created by splitting the test set in half randomly and undersampling the subgroup samples from the one half to fit the training distribution. The other half is our new, reduced test set. To ensure comparability no data augmentation is used and the images are rescaled and cropped to 224 x 224 pixels. Afterwards, the images are normalized with the mean and standard deviation values of the imagenet pretraining.

CelebA The CelebA dataset Dua and Graff [2017] is a collection of face images of celebrities annotated with a variety of attributes, face landmark points and many more. Following Sagawa et al. [2020a] we take the classification of haircolor into blonde and non-blonde as an example for the worst subgroup question. The spurious feature in this situation is the annotated gender. In contrast to waterbirds there is only one rarley occuring subgroup: blond males. The easy feature to pick up is the gender (hair length) and the core feature hair colour is harder to learn for the classifier. To ensure comparability no data augmentation is used. The images are cropped to have a square format and then rescaled to the 224 x 224 pixels resolution. The last preprocessing step is a normalization with the image net mean and standard deviation values.

Poverty The poverty dataset Koh et al. [2021] is a collection of 8 channel satelite imagery from various regions in Africa with annotated information regarding country and if the picture is taken in an urban/ rural area. While the original task is to predict a continuous poverty index and generalize over country related domains, we create a new target and a new train, val, test split. We bin the continuous index into the classes "wealthy" and "poor" by taking 0 as a decision boundary and discarding images with a value between [-0.1,0.1] and try to classify images into these categories. Furthermore, we identify the urbanization label as spurious feature and analyse classification performance on the subgroups. The train, val, test splits are generated iid from all available data with 50% of the data used for training. No data augmentation and no transformation of the inputs is applied.

C-Mnist The corruption biased mnist dataset was developed on the basis of the work in Goel et al. [2020] that combine the standard MNIST datset LeCun and Cortes [2010] and the brightness corruption of Corrupted MNIST. The classification task is to tell if a digit is even or odd and the

spurious feature is having or not having a corruption. In comparison to the previous dataset we made a few adoptions: we changed the corruption type to brightness as it reduces variation of the spurrious feature and intensifies the subgroup problem, we reduced the fraction of subgroup samples from 5% to 1% to make the task more difficult and increase the overall amount of samples. The official publication of mnist and corruption mnist has a train test split and we only use data from the training split for training and split the testing data into validation, iid validation and testing. No data augmentation and no transformation of the inputs is applied.

Color-Mnist The colour biased mnist dataset is created based on the idea of Kim et al. [2019]. They color the mnist letters with different colors while leaving the background unaffected. In our adaptation we add color to the background and pose the binary classification task of even and odd numbers. The colored background is created as a spurious feature with two base colors with some random random color noise, with frequent color and even/odd combinations and rare once. In the experiments we vary the label, color correlation and sometimes use monocolor MNIST for demonstrative experiements. The official publication of MNIST mnist has a train test split and we only use data from the training split for training and split the testing data into validation, iid validation and testing. No data augmentation and no transformation of the inputs are applied.

Adult The Adult dataset is taken from Jiang and Li [2011] and the preprocessing and subgroup definition was replicated from Lahoti et al. [2020]. The dataset is a tabular dataset extracted from a census in Amerca. The binary prediction tasks is to estimate if the individuals yearly income is above or below 50.000 USD based on the given characteristics such as education and martial status. We define the gender and race to be protected attributes and define the following subgroups per class:

$$\mathcal{G} = \{\{\text{male,non-black}\}, \{\text{female,non-black}\}, \{\text{male,black}\}, \{\text{female,black}\}\} \tag{9}$$