Leave No One Out: Mitigating Subpopulation Shift via Leave-One-Out Upsampling

Anonymous Authors

Abstract

Empirical Risk Minimization (ERM) often fails on under-represented subpopulations. Weighted ERM with group labels mitigates this, but such annotations are often unavailable. We propose *Leave One-Out Influenced Sampling (LOUIS)*, a group-label-free framework that uses efficient LOO error approximations to identify difficult samples in **very early training steps**. These scores generate instance-level upsampling weights for a second stage of weighted ERM. Unlike prior work, LOUIS does not require a fully converged identifier model. On Waterbirds and ColorMNIST, LOUIS achieves performance competitive with group-annotated oracles, offering a practical solution for robust learning when group attributes are missing.

1. Introduction

Subpopulation shifts occur when attributes and classes are present in both source and target distributions, but the frequencies of the subclasses differ (Koh et al., 2021). A fundamental challenge of mitigating subpopulation shift is the reliance on predefined group labels to identify struggling subpopulations before training (Idrissi et al., 2022). The group labels demand discrete categories, which are often manually labelled. To date, few studies have proposed a methods that address this dependency. As such, samples can be left out of precise categorisation during training process.

Building on the insight linking Feldman (2020) and Feldman and Zhang (2020), rare, long-tail examples with memorization, we hypothesize that samples belonging to under-represented subpopulations have a high influence on the model's decision boundary, and increasing their representation improves generalization. We propose a method based on a first-order approximation of the Leave-One-Out (LOO) score, which requires no labelled groups during training. Large, over-parameterized deep learning models trained by stochastic gradient descent variants, tend to converge faster on typical examples than atypical examples (Toneva et al., 2019). As such, very early training (sub epoch, or 1-2 epoch) estimates of the LOO cross-validation (LOO-CV) score signal example typicality (Arpit et al., 2017), enabling detection of long-tail instances without attribute labels.

Our approach generalises group balancing baselines, which crucially *does not require any training-set group annotations*. Unlike methods requiring fully trained identifier models (Liu et al., 2021; Zhang et al., 2021), our approach is computationally efficient. We demonstrate on ColorMNIST, Waterbirds, and CelebA that upweighting minority examples via LOO scores improves worst-group performance.

2. Background

The Leave-One-Out (LOO) score, named after its relationship with LOO cross-validation (LOO-CV), defined by the probability of a single held-out observation (x_i, y_i) under a model fit to all

remaining data points. The most common application is measuring self-influence, defined as the change in the model's loss on a data point i when that same point is removed from the training set,

$$\mathrm{LOO}_i = \mathcal{L}(z_i, \theta^{\setminus i}) - \mathcal{L}(z_i, \theta) \tag{1}$$

where $\mathcal{L}(z, \theta*)$ is the point-wise loss for data point $z = (x_i, y_i)$ given model parameters $\theta*$. For a probabilistic model this corresponds to $p(y_i|f_i(\theta^{\setminus i}))$. (Nickl et al., 2023) introduced a framework approximates the effect from removal of sample i as the deviation in output, on the model's logit output $f_i(\theta)$ by the first order Taylor expansion, decomposing into the prediction variance $v_{i,t}$ and the prediction error $e_{i,t}$, for LOO score $\mathcal{I}_f(i)$,

$$\mathcal{I}_f(i) := f_i(\theta_t^{\setminus i}) - f_i(\theta_t) \approx \nabla f_i(\theta_t)^\top (\theta_t^{\setminus i} - \theta_t) \approx \underbrace{\nabla f_i\left(\theta_t\right)^\top \mathbf{H}_t^{-1} \nabla f_i\left(\theta_t\right)}_{=v_{i,t} \text{ (prediction variance)}} \underbrace{\left[\sigma\left(f_i(\boldsymbol{\theta})\right) - y_i\right]}_{=e_{i,t} \text{ (prediction error)}} \tag{2}$$

where θ_t is the model's parameter at step t, $\sigma(\cdot)$ is the final activation y_i is the label for the sample and H_t is the hessian of the loss function. The variance quantifies the uncertainty and the prediction error quantifies the deviation from the ground truth. The product $v_{i,t}e_{i,t}$ forms the perturbation factor, which provides an estimate via the deviation from the output. Re-arranging equation (2) yields an approximation $f_i(\theta_t^{\setminus i}) \approx f_i(\theta_t) + v_{it}e_{it}$, which motivates a per-instance "LOO score" that approximates the negative LOO-CV log likelihood as

$$\mathrm{LOO}_i(\theta_t) = -\log p\left(y_i \mid \sigma\left(f_i\left(\theta_t\right) + v_{it}e_{it}\right)\right). \tag{3}$$

3. Leave One Out Influenced Sampling (LOUIS)

Koh and Liang (2017) showed that difficult samples have outsized impacts decision boundaries. We hypothesise that these often belong to under-represented subpopulations. Leave-one-out cross validation provides an estimate of the test error for each particular *training instance*, which can be used as a proxy to determine an instance's atypicality, directly signalling subpopulation under-sampling. We therefore use LOO-CV negative log-likelihoods as per-instance weights for weighted empirical risk minimization (ERM), enabling an instance-level mitigation without discrete group labels. Our two stage method *Leave One Out Influenced Sampling (LOUIS)*, outlined in algorithm 1 utilises this.

Stage 1: Identification We first train a identifier model $\hat{f}_I$ using standard empirical risk minimization, over a set number of T training iterations for the identifier model, typically set in the early training iterations (Garima et al., 2020). Using parameters from the identifier model $\boldsymbol{\theta}_I^{(T)}$ after T iterations, we compute the per instance LOO scores in each iteration on the training dataset $\mathcal{D}_{tr}$, given by Equation 3.

For computational efficiency on the Hessian term on the variance, we used diagonal approximation of the inverse Hessian, from only the last layer of the model as introduced by (Koh and Liang, 2017), instead of the full Hessian outlined in (Nickl et al., 2023), rendering computational complexity from $O(n^2)$ to $O(n)$. The assumption supporting this approach is that early layers of a model enable fixed feature extraction whilst the last layer has the greatest impact on the model's output, supported by (Koh and Liang, 2017; Kirichenko et al., 2023). The implementation of the Hessian approximation is achieved using Laplace (Daxberger et al., 2021) and ADSL (Osawa et al., 2023) libraries in Python. All models are implemented in Pytorch.

Stage 2: Reweight Dataset and Retrain Using the normalized per-instance LOO scores $\{w_1, ..., w_N\}$ for $\mathcal{D}_{tr}$ from the previous stage as sampling probabilities, we construct a `WeightedRandomSampler`; each training batch is sampled with replacement from $\mathcal{D}_{tr}$. We then reinitialize the final model $\hat{f}_{tr}$ and optimizer, and minimize the weighted ERM objective, where $R(\theta)$ is

Algorithm 1 Leave No One Out Training

Stage 1: Identifier Model
Require: Training set $\mathcal{D}_{tr}$, identifier training iterations T.
Initialize identifier model parameters $\boldsymbol{\theta}_I$.
Train identifier model on $\mathcal{D}_{tr}$ for T early iterations (batches) to obtain $\boldsymbol{\theta}_I^{(T)}$.
Calculate per instance sampling weights w_i for sample (x_i, y_i) using an LOO score approximation with parameters $\boldsymbol{\theta}_I^{(T)}$.
Stage 2: Weighted Retraining
Initialize final model parameters $\boldsymbol{\theta}_{final}$ and optimizer.
Construct a sampler for $\mathcal{D}_{tr}$ with sampling probabilities $p_i \propto w_i$.
Train the final model on $\mathcal{D}_{tr}$ using the upsampled training set until convergence.
Return: Final trained parameters $\boldsymbol{\theta}_{tr}^{(E_{tr})}$.

the regularisation term and $\ell_i(\theta)$ is the per instance loss.

$$\theta_\star = \underset{\theta}{\operatorname{argmin}}\, R(\theta) + \sum_{i=1}^{N} w_i \ell_i(\theta). \tag{4}$$

We select T training iterations, batch size, learning rate, and weight decay for the identifier model by performing a search and evaluating the final model's worst-group accuracy on a labelled validation set. Small batch sizes were used for the training model $\hat{f}_{tr}$ to promote better generalization (Keskar et al., 2017).

4. Experiments

4.1. Results

We evaluate LOUIS on four standard datasets: ColorMNIST (Arjovsky et al., 2020; Sun et al., 2023), Waterbirds (Sagawa et al., 2020), CelebA (Liu et al., 2015; Sagawa et al., 2020), and CivilComments (Idrissi et al., 2022). For Waterbirds, CelebA, and CivilComments, we adopt the training schedules and default WILDS splits from (Idrissi et al., 2022). Across all experiments, the LOUIS pre-training stage saves checkpoints at 2^N batch intervals. To demonstrate the computational efficiency of LOUIS, all experiments were conducted on a single consumer-grade workstation equipped with an NVIDIA RTX 4090 (24GB) GPU and an Intel Core i7-13700K CPU. Primary results summarized in Table 1.

ColorMNIST. A synthetic binary classification dataset assigning MNIST digits to $\tilde{y} = 0$ $(0 \leq d < 5)$ or $\tilde{y} = 1$ $(5 \leq d \leq 10)$. Targets y match $\tilde{y}$ with 0.75 probability. Images are converted to RGB and assigned a color (red or green) that matches y with probability p_λ. We train at $\lambda = 0.05$ (highly correlated) and test across $\lambda \in [0, 1]$ in 0.05 increments to evaluate robustness to subpopulation shift. Both identifier and final models use the Adam optimizer with a batch size of 250. The Laplace approximation uses a prior standard deviation of 0.001.

Waterbirds. This dataset overlays bird images onto varying backgrounds (Welinder et al., 2010). The goal is to classify waterbirds versus landbirds despite spurious background correlations. We use a pretrained ResNet-50. The identifier model is trained with SGD (batch size 128, learning rate 1×10^{-3}, weight decay 1×10^{-4}, momentum 0.9). The final model uses a batch size of 4, learning rate 1×10^{-5}, and weight decay 1.0. LOUIS runs for 1-2 epochs.

CelebA. A dataset of over 200,000 celebrity faces (Liu et al., 2015). We classify the target attribute "blond" against the spurious attribute "male," as the blond attribute is rarely correlated with males in this dataset. We use a pretrained ResNet-50. The identifier model is trained with SGD (batch size 128, learning rate 1×10^{-3}, weight decay 0.1, momentum 0.9). The final model uses a batch size of 32, learning rate 1×10^{-5}, and weight decay 0.1. LOUIS runs for 1 epoch.

Table 1: **Performance results average accuracy (%)**. Results are mean ± std over 5 runs. Other values from (Idrissi et al., 2022). Note that LOUIS, RWG and JTT are pre-hoc methods, i.e. they do not require oracle group annotations at training. LOUIS outperforms all other such methods. Baselines for CivilComments-Fine are from (Qiao et al., 2025b); all others are our own implementation.

Method	CMNIST	Waterbirds	CelebA	CivilComments (Coarse)	CivilComments (Fine)
ERM	7.2 ± 1.3	85.5 ± 1.0	79.7 ± 3.7	61.3 ± 2.0	55.6 ± 0.6
JTT	88.9 ± 1.8	85.6 ± 0.2	75.6 ± 7.7	67.8 ± 1.6	69.3
LOUIS (Ours)	$\mathbf{96.2 \pm 0.7}$	$\mathbf{88.9 \pm 0.7}$	81.0 ± 1.4	$\mathbf{72.2 \pm 0.6}$	$\mathbf{72.0 \pm 1.9}$
RWG	$\mathbf{96.3 \pm 0.9}$	87.6 ± 1.6	84.3 ± 1.8	$\mathbf{72.0 \pm 0.5}$	-
RWY	–	86.1 ± 0.7	82.9 ± 2.2	67.5 ± 0.6	-
gDRO	–	87.1 ± 3.4	$\mathbf{86.9 \pm 1.1}$	69.9 ± 1.2	70.0 ± 2.0

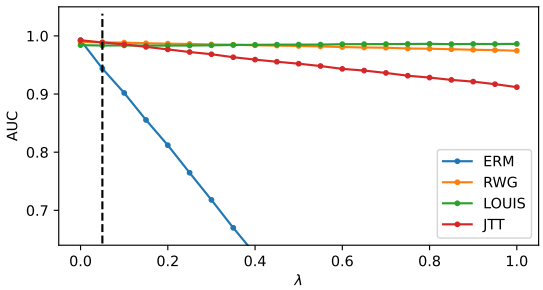

Figure 1: ColorMNIST AUC, evaluated on test sets with differing values of λ, which controls the mix of data sources (All models were trained on $\lambda = 0.05$.). LOUIS and RWG are more robust to shift than JTT and ERM. Hyperparameters were selected by WGA on a held-out validation set with group annotations.

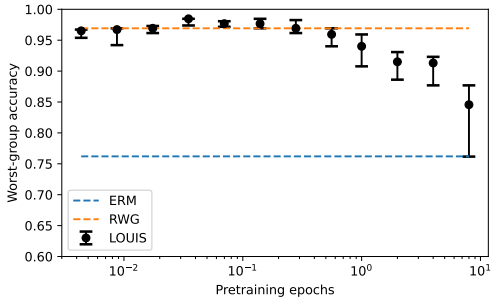

Figure 2: ColorMNIST worst-group validation accuracy vs. identifier pre-training steps (log scale). LOUIS consistently outperforms ERM, and with early stopping exceeds RWG—which uses training-set group annotations.

CivilComments. A toxicity classification dataset sourced from online forums. We consider both the "coarse" setting (Idrissi et al., 2022), utilizing a binary attribute denoting whether a comment explicitly mentions any of 8 demographic groups, as well as the original "fine" setting to test LOUIS' ability to perform with multiple attributes. We use a BERT model (Devlin et al., 2019) pretrained on BookCorpus. The learning rate is set to 1×10^{-3} for the LOUIS stage and 1×10^{-5} for the weighted ERM stage. LOUIS runs for 1 epoch.

For ColorMNIST, we additionally consider evaluate performance by considering the AUC-ROC of the classifier synthetic test datasets where the parameter λ has shifted. These results, in Figure 1, demonstrate the clear failure of ERM to handle significant subpopulation shift in this setting; LOUIS exhibits the least degradation of performance even as we approach $\lambda = 1.0$.

4.2. Evolution of LOO scores

The primary additional hyperparameter introduced by LOUIS, relative to ERM, is the number of batches (or epochs) for which the identifier model should be run in stage one. To investigate the robustness of the method to choice of this parameter, we consider a range of values, from a single batch up to 8 full epochs for ColorMNIST, in Figure 2, 2 epochs for Waterbirds, in Figure 7, and 5 epochs for CelebA and CivilComments (both Coarse and Fine). In Figure 3 we further analyse a

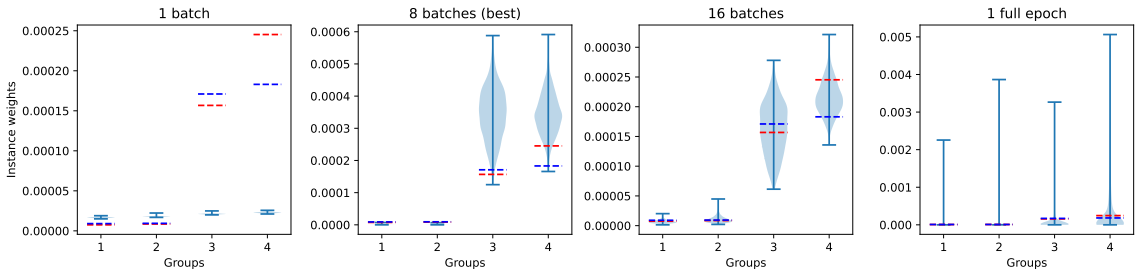

Figure 3: Evolution of per-instance weight distributions on ColorMNIST by ground-truth group (Groups 3–4 are minority). Red lines show RWG weights (based on group frequency); blue lines show JTT. Weights evolve from uniform (1 batch), distinguishing groups (8 steps), weights approaching RWG (16 steps), uniform except outliers uncorrelated with groups (1 epoch), and lead to a low effective sample size.

few specific stopping times, by plotting the distribution of instance weights grouped by ground-truth group annotation.

As a general trend, we observe that the earlier iterations of the identifier model produce better sample weight estimates, which occurs prior to full model convergence, However, there is additionally an indication of a "sweet spot", which is to be expected: early on in training, the model has nearly random weights and the overall LOO scores are roughly uniform (reflecting the low predictive power of the model); we see this in the leftmost panel of Fig. 3. As training continues, the model achieves high accuracy on the typical set; for ColorMNIST we can see from Fig. 2 this happens after only a handful of batches (far less than a full epoch). The two middle panels of Fig. 3 show this intermediate regime, in which the instance weights fairly closely reflect those found from RWG, and in which performance of the weighted ERM model is strong. Later on in training, we can see in the last panel of Fig. 3 that the model has converged on most of the training examples, meaning most instances have a very low LOO score; the weights are then dominated by a small number of training instances which leads to a low effective sample size and correspondingly poor performance in Stage 2.

This trend can consistently be observed on Waterbirds in Appendix Fig. 5, and in and CelebA, in Appendix Fig. 8, where, over the course of early training, the LOO scores of individual instances converges towards their group frequency, or other combinations of attributes within the dataset. For Waterbirds, we similarly observe in Fig. 7 that stopping Stage 1 after approximately one epoch performs well; additional pre-training of the identifier model degrades performance. The group-level breakdown of per-instance weights for Waterbirds after one epoch is shown in Fig. 6.

4.3. Discussion

We note that the effectiveness of first-epoch LOO estimates depends on model capacity: an overly expensive model may result in memorization which occurs before meaningful signals emerge, whereas underpowered models may fail to capture differences in sample difficulty. Although Nickl et al. (2023) note that diagonal Hessian approximations provide a less faithful estimate of LOO values, we observe that their curves still track the ground-truth influence reasonably well. As our scheme relies on relative instance weights rather than the absolute likelihood values, this approximation suffices in practice (see Figures 2 and 7).

5. Related Work

Methods Using Group Labels. Approaches with predefined group labels include importance weighting (Shimodaira, 2000; Sugiyama et al., 2007), invariant learning via domain adversarial

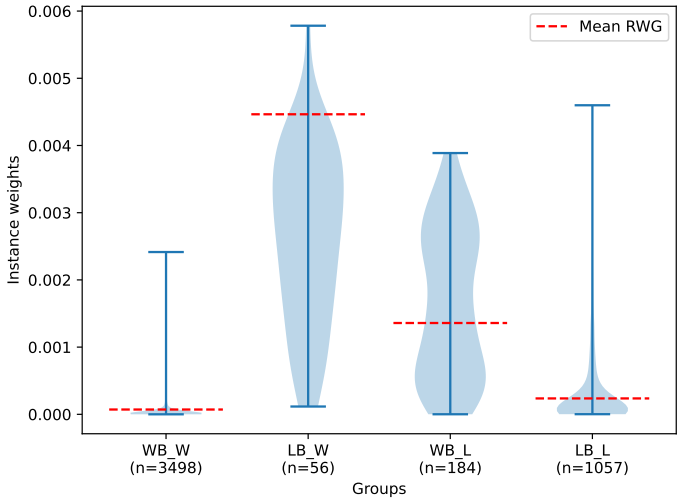

Figure 4: Example distribution of per-instance weights from identifier model after 1 epoch for Waterbirds (Resnet-50), where n is group frequency. RWG (based on inverse group frequency) weights are given as reference shown in red. The weights from LOO weights closely approximate RWG weights without the need for group labels at train time.

training (Ganin et al., 2016), and Distributionally Robust Optimization (Duchi and Namkoong, 2021; Namkoong and Duchi, 2016). GroupDRO (Sagawa et al., 2020) upweights groups with highest loss during training. However, these methods require exhaustive, accurate group labels—often unavailable in practice.

Methods Reducing Group Reliance. Recent work aims to mitigate subpopulation shifts without group annotations (Jain et al., 2024), as required in applications like galaxy classification (Dickinson et al., 2018) or tumour malignancy prediction (van den Goorbergh et al., 2022). Some approaches leverage unbiased validation sets for post-training adjustment (Kirichenko et al., 2023; Sun et al., 2023). Others infer sample importance through learning dynamics (Toneva et al., 2019) or use two-stage processes with identifier models to focus on difficult subpopulations (Liu et al., 2021; Qiao et al., 2025a; Zhang et al., 2021). While reducing reliance on training-set labels, these methods shift dependency to validation set quality or require partially trained models.

6. Future Work

We would like to remove the need to validation set for setting the number of batches required for the identifier model. It has been noted that early batch gradients are crucial for class differentiation, from which the negative log likelihood has been heavily correlated with the types of dataset imbalances, suggesting a potential heuristic for validation-free model selection. However, any metric could act as a proxy for out-of-distribution performance, with rigorous theoretical analysis and extensive empirical validation are required to establish its reliability and generality, and in general guarding against arbitrary distribution shifts is impossible.

7. Conclusion

We introduced LOUIS, a sampling method that uses LOO scores as instance-level weights to mitigate subpopulation shift without group labels. By identifying difficult examples during early training, LOUIS improves robustness on ColorMNIST and Waterbirds, outperforming strong baselines that rely on explicit group annotations. Future work will explore reducing the estimation's dependence on model capacity and refining the optimal stopping point for weight estimation.

References

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2020.

Devansh Arpit, Stanisław Jastrzundefinedbski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 233–242. JMLR.org, 2017.

Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux – effortless bayesian deep learning. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. URL https://arxiv.org/abs/1810.04805.

Hugh Dickinson, Lucy Fortson, Chris Lintott, Claudia Scarlata, Kyle Willett, Steven Bamford, Melanie Beck, Carolin Cardamone, Melanie Galloway, Brooke Simmons, William Keel, Sandor Kruk, Karen Masters, Mark Vogelsberger, Paul Torrey, and Gregory F. Snyder. Galaxy zoo: Morphological classification of galaxy images from the illustris simulation. *The Astrophysical Journal*, 853(2):194, feb 2018. doi: 10.3847/1538-4357/aaa250. URL https://doi.org/10.3847/1538-4357/aaa250.

John C. Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021. doi: 10.1214/20-AOS2004.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2020, page 954–959, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450369794. doi: 10.1145/3357713.3384290. URL https://doi.org/10.1145/3357713.3384290.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: discovering the long tail via influence estimation. NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. URL http://jmlr.org/papers/v17/15-239.html.

Garima, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Badr Youbi Idrissi, Martin Arjovsky, Mohammad Pezeshki, and David Lopez-Paz. Simple data balancing achieves competitive worst-group-accuracy. In *First Conference on Causal Learning and Reasoning*, 2022. URL https://openreview.net/forum?id=cDxT7WYhaD.

Saachi Jain, Kimia Hamidieh, Kristian Georgiev, Marzyeh Ghassemi, and Aleksander Madry. Better than balancing: Debiasing through data attribution. In *NeurIPS 2023 Workshop on Distribution Shifts: New Frontiers with Foundation Models*, 2024. URL https://openreview.net/forum?id=Agekm5fdW3.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=H1oyRlYgg.

Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Zb6c8A-Fghk.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1885–1894. JMLR.org, 2017.

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 Earnshaw, Imran Haque, Sara M Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5637–5664. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/koh21a.html.

Evan Z 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. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6781–6792. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/liu21f.html.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Hongseok Namkoong and John C Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/4588e674d3f0faf985047d4c3f13ed0d-Paper.pdf.

Peter Nickl, Lu Xu, Dharmesh Tailor, Thomas Möllenhoff, and Mohammad Emtiyaz Khan. The memory-perturbation equation: Understanding model's sensitivity to data. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=dqS1GuoG2V.

Kazuki Osawa, Satoki Ishikawa, Rio Yokota, Shigang Li, and Torsten Hoefler. Asdl: A unified interface for gradient preconditioning in pytorch. *CoRR*, abs/2305.04684, 2023. URL https://doi.org/10.48550/arXiv.2305.04684.

Rui Qiao, Zhaoxuan Wu, Jingtan Wang, Pang Wei Koh, and Bryan Kian Hsiang Low. Group-robust sample reweighting for subpopulation shifts via influence functions. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=aQj9Ifxrl6.

Rui Qiao, Zhaoxuan Wu, Jingtan Wang, Pang Wei Koh, and Bryan Kian Hsiang Low. Group-robust sample reweighting for subpopulation shifts via influence functions. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=aQj9Ifxrl6.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000. ISSN 0378-3758. doi: https://doi.org/10.1016/S0378-3758(00)00115-4. URL https://www.sciencedirect.com/science/article/pii/S0378375800001154.

Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(35):985–1005, 2007. URL http://jmlr.org/papers/v8/sugiyama07a.html.

Qingyao Sun, Kevin Patrick Murphy, Sayna Ebrahimi, and Alexander D'Amour. Beyond invariance: Test-time label-shift adaptation for addressing "spurious" correlations. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=9mJXDcr17V.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BJlxm30cKm.

Ruben van den Goorbergh, Maarten van Smeden, Dirk Timmerman, and Ben Van Calster. The harm of class imbalance corrections for risk prediction models: illustration and simulation using logistic regression. *Journal of the American Medical Informatics Association*, 29(9):1525–1534, 06 2022. doi: 10.1093/jamia/ocac093. URL https://doi.org/10.1093/jamia/ocac093.

Peter Welinder, Steve Branson, Tomáš Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, May 2010.

Michael Zhang, Nimit Sharad Sohoni, Hongyang R. Zhang, Chelsea Finn, and Christopher Ré. Correct-n-contrast: A contrastive approach for improving robustness to spurious correlations. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021. URL https://openreview.net/forum?id=Q41kl_DwS3Y.

A. Dataset Details

ColorMNIST Dataset. ColorMNIST is a synthetic dataset introduced in (Arjovsky et al., 2020). This is a binary classification problem, where MNIST digits with label d in the range $0 \leq d < 5$ are assigned a value $\tilde{y} = 0$, and $5 \leq d \leq 10$ are assigned $\tilde{y} = 1$. The classification targets $y \in \{0, 1\}$ are obtained by setting $y = \tilde{y}$ with probability 0.75. The normally greyscale images are converted to RGB; they are assigned either a color red ($z = 1$) or green ($z = 0$); they are assigned the color $z = y$ with probability p_λ.

Here we follow the implementation by (Sun et al., 2023), using spurious factor λ. All training runs were performed at $\lambda = 0.05$; for testing, we consider a range of λ from 0 to 1 at all increments of 0.05. At $\lambda = 0.5$, the image color is highly correlated with the target y; at greater lambdas, the correlation between the label and the color flips from few anti-correlated images to many. Group annotations consist of four groups corresponding to the possible (z, y) pairs.

Waterbirds Dataset. Waterbirds (Sagawa et al., 2020), is made from a composite image of birds (Welinder et al., 2010) overlaid on spuriously correlated backgrounds from the Places dataset (Welinder et al., 2010). Its goal is to classify the label — waterbird or landbird We have used the dataset implementation from the WILDS library with its default splits.

CelebA Dataset. CelebA (Sagawa et al., 2020; Liu et al., 2015), is a composite dataset of faces of celebrities, each with 40 labelled attributes. Under the implementation by (Sagawa et al., 2020), the model aims to classify the attribute "blond" as the label and "male" as the spurious attribute. dataset implementation from the WILDS library with its default splits.

CivilComments Dataset. A toxicity classification dataset sourced from online forums. We consider both the "coarse" setting (Idrissi et al., 2022), utilizing a binary attribute denoting whether a comment explicitly mentions any of 8 demographic groups, as well as the original "fine" setting to test LOUIS' ability to perform with multiple attributes. We use a BERT model (Devlin et al., 2019) pretrained on BookCorpus. The learning rate is set to 1×10^{-3} for the LOUIS stage and 1×10^{-5} for the weighted ERM stage. LOUIS runs for 1 epoch.

B. Additional figures from experiments

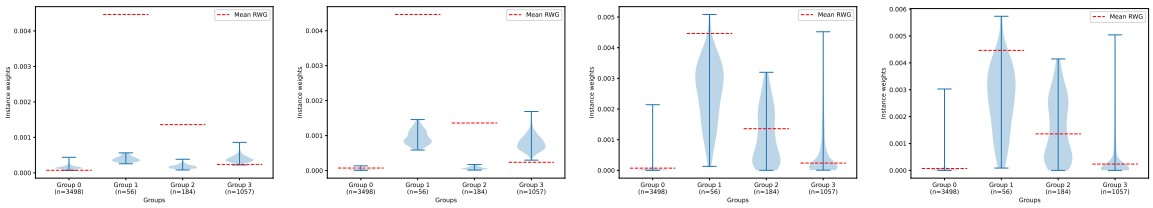

Figure 5: Evolution of the distribution of per-instance weights for Waterbirds within the first epoch, conditioned on ground-truth group assignments. Waterbird and landbird are abbreviated as WB and LB respectively, and that water and land are abbreviated as W and L. For reference, the per-group weights assigned by RWG (based solely on group frequency) are shown in red; Like ColorMNIST, instance weights are initially relatively uniform (after only one batch); after 8 batches, the instance weights start to converge towards towards RWG weights; at 32 steps, the weights approach that of RWG; at two batches after one epoch (the weights start to converge towards the number of samples in each subpopulation.

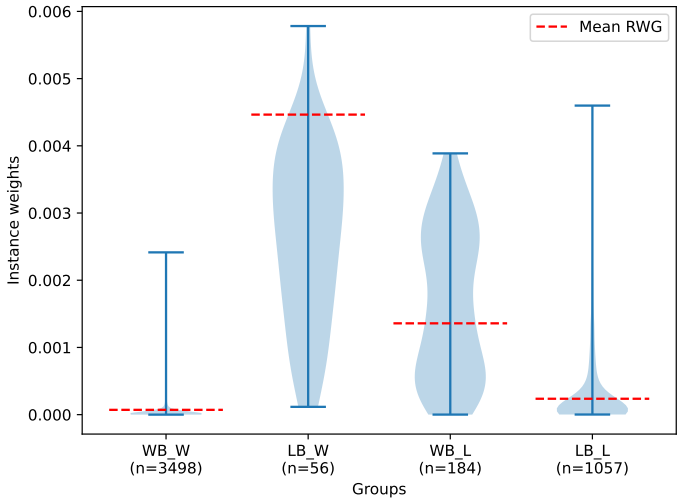

Figure 6: Example distribution of per-instance weights from identifier model after 1 epoch for Waterbirds (Resnet-50), where n is group frequency. RWG (based on inverse group frequency) weights are given as reference shown in red. The weights from LOO weights closely approximate RWG weights without the need for group labels at train time.

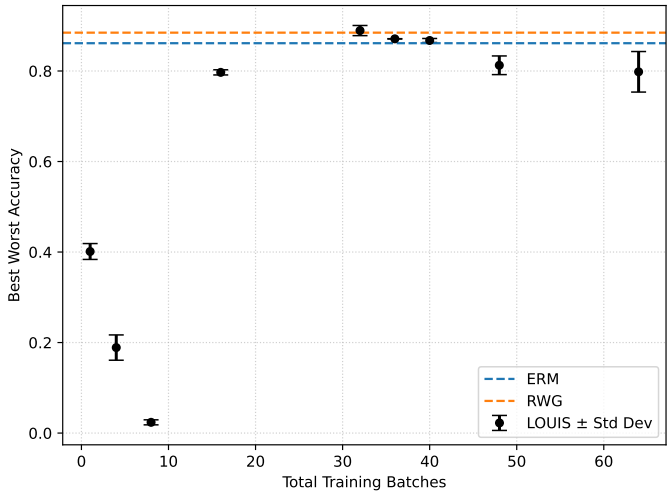

Figure 7: Performance of LOUIS on Waterbirds as a function of number of pretraining batches taken before computing LOO scores (2 epochs, 32 batches per epoch; averaged over 3 runs); Worst-group validation set accuracy for LOUIS consistently outperforms ERM and RWG with early stopping during training.

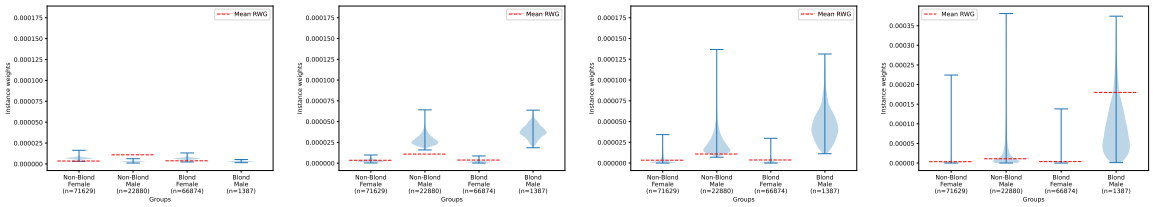

Figure 8: Evolution of the distribution of per-instance weights for CelebA within the first epoch, conditioned on ground-truth group assignments. Again, the per-group weights assigned by RWG (based solely on group frequency) are shown in red for reference; The number of batches are at the first batch, after 64 batches, after 128 batches and after 1024 batches respectively, where we see all groups approach RWG group frequency weights over the iterations within the first epoch.