Pushing the Accuracy-Fairness Tradeoff Frontier with Introspective Self-play

Anonymous Author(s)
Affiliation
Address
email

Abstract

Improving the *accuracy-fairness frontier* of deep neural network (DNN) models is an important problem. Uncertainty-based active learning (AL) can potentially improve the frontier by preferentially sampling underrepresented subgroups to create a more balanced training dataset. However, the quality of uncertainty estimates from modern DNNs tend to degrade in the presence of spurious correlations and dataset bias, compromising the effectiveness of AL for sampling tail groups. In this work, we propose *Introspective Self-play (*ISP*)*, a simple approach to improve the uncertainty estimation of a deep neural network under dataset bias, by adding an auxiliary *introspection* task requiring a model to predict the bias for each data point in addition to the label. We show that ISP provably improves the *bias-awareness* of the model representation and the resulting uncertainty estimates. On two real-world tabular and language tasks, ISP serves as a simple "plug-in" for AL model training, consistently improving both the tail-group sampling rate and the final accuracy-fairness trade-off frontier of popular AL methods.

1 Introduction

Modern deep neural network (DNN) models are commonly trained on large-scale datasets [27, 84]. These datasets often exhibit an imbalanced long-tail distribution with many small population subgroups, reflecting the nature of the physical and social processes generating the data distribution [128, 33]. This imbalance in training data distribution, i.e., ***dataset bias***, prevents DNN models from generalizing equitably to the underrepresented population groups [40]. In response, the existing bias mitigation literature has focused on improving training procedures under a fixed and imbalanced training dataset, striving to balance performance between model accuracy and fairness (e.g., the average-case v.s. worst-group performance) [3, 72, 73]. Formally, this goal corresponds to identifying an optimal model $f \in \mathscr{F}$ that attains the *Pareto efficiency frontier* of the accuracy-fairness trade-off (e.g., see Figure 1), so that under the same training data $D = \{y_i, \mathbf{x}_i\}_{i=1}^n$, we cannot find another model $f' \in \mathscr{F}$ that outperforms f in both accuracy and fairness.

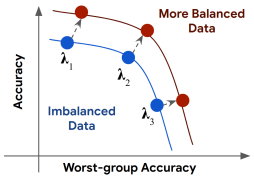

Figure 1: Example of accuracy-fairness frontier. Under a more balanced training data distribution, the model can attain a better accuracy-fairness frontier (**Red**) when compared to training under an imbalanced distribution (**Blue**) at every tradeoff level λ .

However, the limited size of the tail-group examples restricts the DNN model's worst-group performance, leading to a compromised accuracy-fairness frontier [125, 31]. In this work we ask: *Under a fixed learning algorithm, can we meaningfully push the model's accuracy-fairness frontier by improving the training data distribution using active learning?* Given a sampling model with *well-calibrated* uncertainty (i.e., the model uncertainty is well-correlated with generalization error), uncertainty-based AL has the promise to preferentially acquire tail-group examples from unlabelled data *without needing group annotations on the unlabelled set*, and reach a more balanced data distribution and improved accuracy-fairness performance of the final model[12]. However, recent work suggests that a DNN model's uncertainty estimate is less trustworthy under spurious correlations and distributional shift, potentially compromising the AL performance under dataset bias [80, 77, 67, 110].

Submitted to the Workshop on Distribution Shifts, 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

In this work, we introduce ***Introspective Self-play*** **(ISP)**, a simple training approach to improve a DNN model's uncertainty quality for underrepresented groups (Section 2). Using group annotations from the training data, ISP conducts *introspective training* to provably improve a DNN's representation and uncertainty quality for the tail groups. When group annotations are not available, ISP additionally estimates them using a cross-validation-based procedure. Under two challenging real-world tasks (census income prediction and toxic comment detection), we empirically validate the effectiveness of ISP in improving the performance of AL with a DNN model under dataset bias (Section 3).

2 Method

In this section, we introduce *Introspective Self-play (*ISP*)*, a simple training approach to improve model quality in representation learning and uncertainty quantification under dataset bias.

2.1 Introspective Training

We consider models of the form $p(y|\mathbf{x}) = \sigma\big(f_y(\mathbf{x})\big) = \sigma\big(\beta_y^\top h(\mathbf{x})\big)$, where $h : \mathscr{X} \to \mathbb{R}^D$ is a D-dimensional embedding function, $\beta_y \in \mathbb{R}^D$ the output weights, and $\sigma(\cdot)$ the activation function. Given model $f_y = \beta_y^\top h$, *introspective training* adds a bias head $f_b = \beta_b^\top h$ to the model, so it becomes a multi-task architecture $f = (f_y, f_b)$ with shared embedding $h(\cdot)$:

$$p(y|\mathbf{x}) = \sigma(f_y(\mathbf{x})),\ p(b|\mathbf{x}) = \sigma_{sigmoid}(f_b(\mathbf{x}));\ \text{where } (f_y, f_b) = \big(\beta_y^\top h + b_y,\ \beta_b^\top h + b_b\big). \tag{1}$$

Given examples $D = \{\mathbf{x}_i, y_i, g_i\}_{i=1}^n$, we generate the underrepresentation labels as $b_i = I(g_i \in \mathscr{B})$ and train the model with the target and underrepresentation labels (y_i, b_i) by minimizing a multi-task learning objective:

$$L((y_i, b_i), \mathbf{x}_i) = L(y_i, f_y(\mathbf{x}_i)) + L_b(b_i, f_b(\mathbf{x}_i)), \tag{2}$$

where L is the standard loss function for the task, and L_b is the cross-entropy loss. As a result, given training examples $\{\mathbf{x}_i\}_{i=1}^n$, *introspective training* (2) not only trains the model to predict the outcome y_i, but also instructs it to recognize its potential bias b_i by predicting whether $\mathbf{x}_i$ is from an underrepresented group.

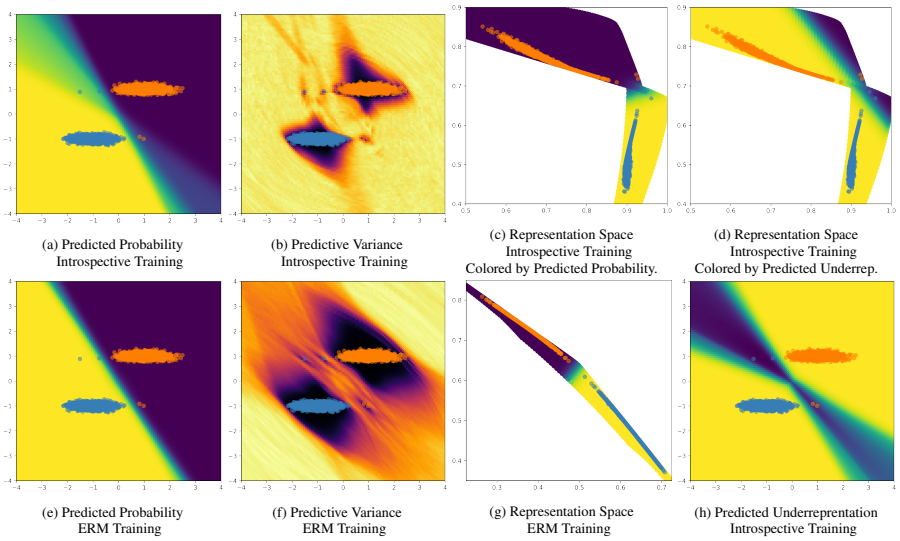

Figure 2: Prediction, uncertainty quantification, and representation learning behavior of introspective training v.s. ERM training in a binary classification task under severe group imbalance ($n = 5000$) [90]. Here, blue and orange indicates the two classes, and each class contains a minority group (the tiny clusters on the diagonal with $n < 5$) and a majority group (the large clusters on the off-diagonal). **Column 1-2** depicts the models' predictive probability and predictive uncertain surface in the data space. **Column 3** depicts the models' decision surface in the last-layer representation space, colored by the predictive probability of the target label. **Column 4** depicts the introspective-trained model's predicted bias probability in the representation space (fig. 2d) and in the data space (fig. 2h), colored by the predictive probability of the underrepresentation. Appendix D.1 described further detail.

Despite its simplicity, introspective training has a significant impact on the model's representation learning that is particularly important for quantifying uncertainty under dataset bias. Figure 2 illustrates this on a binary classification task under severe group imbalance [90], where we compare two dense ResNet ensemble models trained using the introspection objective (Equation (2)) v.s. the empirical risk minimization (ERM) objective, respectively. Comparing figures 2a and 2e, we observe

that the decision boundaries for the predicted label are similar between introspective training and ERM. However, the predictive variance (obtained via a Gaussian process (GP) layer [67]) exhibits sizable differences. In particular, the variance estimates for introspective training are uniformly high outside of the two clouds of underrepresented groups in the data. However, for ERM, the model confidence is high along the decision boundary, even in the unseen regions without training data. This is due to the fact that when training with ERM, the representation collapses in the direction that is not correlated with training label (i.e., parallel to decision boundary) and does not retrain any input information regarding the underrepresented groups in its representation (fig. 2g). However, with introspective training, the representations indeed are morphed to reflect the differences between the underrepresented examples and the majority group (as can be seen in figures fig. 2g vs fig. 2c), helping the model to better distinguish them in the representation space, and hence lead to improved uncertainty estimate in the neighborhood of underrepresented examples. Appendix D.1 contains further description.

Formally, introspective training induces the below guarantee on the model's *bias-awareness* in its hidden representation and uncertainty estimates (detailed discussion in Appendix B.1):

Proposition 1 (Introspective Training induces Bias-awareness)**.** *Denote $o_b(\mathbf{x}) = p(\mathbf{x}|b=1)/p(\mathbf{x}|b=0)$ the odds for $\mathbf{x}$ belongs to the underrepresented group $\mathscr{B}$. For a well-trained model $f = (f_y, f_b)$ that minimizes the introspective training objective (2), so that $p(b=1|\mathbf{x}) = \sigma(f_b(\mathbf{x}))$, we then have:*

- ***(Bias-aware Embedding Distance)*** *For two examples $(\mathbf{x}_1, \mathbf{x}_2)$, the embedding distance $||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2$ is lower bounded by (up to a scaling constant) the odds ratio of whether $\mathbf{x}_1$ belongs to the underrepresented groups versus that for $\mathbf{x}_2$: $||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2 \geq \frac{1}{||\beta_b||_2} \times \max\left(\log \frac{o_b(\mathbf{x}_1)}{o_b(\mathbf{x}_2)}, \log \frac{o_b(\mathbf{x}_2)}{o_b(\mathbf{x}_1)}\right)$, such that the distance between a pair of minority and majority examples $(\mathbf{x}_1, \mathbf{x}_2)$ is large due to the high values of the log odds ratio.*

2.2 Estimating underrepresentation via Cross-validated Self-play

In this section, we consider how to estimate the underrepresentation label b_i when it is absent, so that ISP can be applied to the setting where group annotations g_i is too expensive to obtain. We propose ***cross-validated self-play***, a simple method to estimate a model's generalization gap. Briefly, given training data D_{train} divided into K splits, we train a bootstrap ensemble of K models $\{f_k\}_{k=1}^{K}$ with ERM training, where each f_k sees a fraction of the training data (see Appendix Fig. 5). As a result, for each $(\mathbf{x}_i, y_i) \in D_{train}$, there exists a collection of in-sample predictions $\{f_{in,k'}(\mathbf{x}_i)\}_{k'=1}^{K_{in}}$ trained on data splits containing $(\mathbf{x}_i, y_i)$, and a collection of out-of-sample predictions $\{f_{out,k}(\mathbf{x}_i)\}_{k=1}^{K_{out}}$ trained on data splits not containing $(\mathbf{x}_i, y_i)$. Then, the ***self-play estimator*** of the model's generalization gap is

$$\hat{b}_i = \underbrace{\mathbb{E}_k[L(y_i, f_{out,k}(\mathbf{x}_i))]}_{\textit{estimated error}} - \underbrace{L(y_i, \bar{f}_{in}(\mathbf{x}_i))}_{\textit{estimated noise}} = \mathbb{E}_k[L(\bar{f}_{in}(\mathbf{x}_i), f_{out,k}(\mathbf{x}_i))]. \tag{3}$$

where $\bar{f}_{in}$ is the ensemble prediction based on in-sample predictors $f_{in,k'}$, the expectation $\mathbb{E}_k$ is taken with respect to the out-of-sample predictions [1]. Compared to the standard alternatives in the literature (e.g., single-model error $L(y, f)$ as in JTT), the *self-play* estimator $\hat{b}_i$ has the appealing property of controlling *noise* (by using $\bar{f}_{in}$) while more stably estimating *variance* (by using expectations over $\bar{f}_{out,k}$), thereby more stably estimating a model's generalization error due to dataset bias. Appendix C.2 contains detailed explanation in terms of the *noise*-*bias*-*variance* decomposition of model error [28].

Method Summary: Introspective Self-play. Combining the *self-play underrepresentation estimation* and *introspective training* together, we arrive at *Introspective Self-play* (ISP), a simple two-stage method that improves the representation quality and uncertainty estimates of a DNN for underrepresented population groups. ISP first (optionally) estimates underrepresentation labels using *cross-validated self-play* if the group annotation is not available, and then conducts *introspective training* to train the model to recognize its own bias while learning to predict the target label. For the unlabelled data to be sampled, the resulting model generates (1) predictive probability $p(y|\mathbf{x})$, (2) uncertainty estimates $\hat{v}(\mathbf{x})$ and (3) predicted probability for underrepresentation $p(b|\mathbf{x}) = \sigma(f_b(\mathbf{x}))$, offering a rich collection of active learning signals for downstream applications (Figure 3 & Algorithm 1).

[1] We apply early-stopping based on model's cross-validation error, so that $\bar{f}_{in}$ doesn't overfit to label noise [63, 97]. See "Practical Comments" paragraph of Appendix C.2.

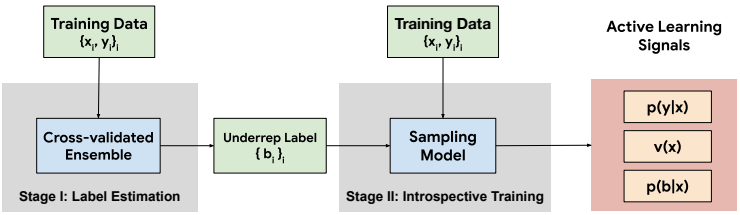

Figure 3: The two-stage *Introspective Self-play* (ISP) model.

3 Experiments

We consider two challenging real-world datasets: Census Income [59] and Toxicity Detection [11]. We demonstrate that for each task, ISP meaningfully improves the tail-group sampling rate and the accuracy-fairness performance of state-of-the-art AL methods. Appendix D.2 describes full experiment detail. Briefly, we consider two settings where group label is observed or unobserved on the labelled set (it is *never* observed on unlabelled set). We train AL models using ISP, using either the provided group labels as under-representation label $\hat{b}_i$ (**ISP-Identity**), or estimate it using the estimated generalization gap from cross-validated self-play (**ISP-Gap**). In each setting, we compare to popular training methods reweighting (**RWT**) [46] and Just Train Twice (**JTT**) [66] as well as random sampling and ERM baselines. For each AL model training method, we conduct 8 rounds of active learning until reaching half of the full dataset, so there's sufficient variation between the data collected by different AL models. To evaluate the final model's accuracy-fairness frontier given the data collected by a AL method, we perform reweighted training using a weighted objective $\sum_{(x,y)\notin\hat{\mathscr{B}}} L_{ce}(y, f(\mathbf{x})) + \lambda \sum_{(x,y)\in\hat{\mathscr{B}}} L_{ce}(y, f(\mathbf{x}))$, with the underrepresented group $\hat{\mathscr{B}}$ defined by either group label or by thresholding the estimated under-representation label $1_{\hat{b}_i > t}$, and tracing out the frontier of model's (accuracy, worst-group accracy) performances over a range of values for (λ, t).

Table 1: The tail-group sampling rate and final-model accuracy v.s. fairness performances under different AL model training methods. Here we show the best active learning signal for each task (i.e., variance for Census Income, and margin for toxicity detection). **Tail Sampling Rate**: The ratio between num. of sampled tail group examples (in final round) v.s. the total num. of tail group in population. **Combined Acc**: The combined accuracy-fairness score defined as (acc + worst-group acc)/2. It is proportional to the perimeter of the rectangle defined by a point on the accuracy-fairness curve.

AL Training Method	Group identity label in train set?	Census Income			Toxicity Detection		
		Tail Sampling Rate	Combined Acc.	Worst-group Acc.	Tail Sampling Rate	Combined Acc.	Worst-group Acc.
(Random)	✓	0.475	0.746	0.659	0.556	0.708	0.490
RWT [46]	✓	0.797	0.772	0.761	0.857	0.709	0.482
ISP-Identity (Ours)	✓	**0.907**	**0.785**	**0.796**	**0.905**	**0.719**	**0.506**
ERM	×	0.791	0.736	0.658	0.852	0.735	0.539
JTT [66]	×	**0.839**	0.752	0.695	0.866	0.747	0.571
ISP-Gap (Ours)	×	**0.839**	**0.770**	**0.788**	**0.867**	**0.759**	**0.597**

Table 1 shows sampling performance and the final-model fairness-accuracy performance of each AL model training method, and Figure 4 visualizes the full accuracy-fairness frontier of the final models.

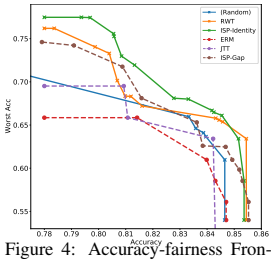

Figure 4: Accuracy-fairness Frontier for Census Income.

Our main conclusions are: **(1) Effectiveness of ISP training**: Compared to non-ISP baselines, we find ISP consistently improves a AL model's active learning (measured by tail-group sampling rate) and accuracy-fairness performance (measured by combined accuracy, which is defined as (accuracy + worst-group accuracy)/2). This advantage is seen in both settings where the group label is available or unavailable. In particular, in Figure 4, the final model from **ISP-Gap** (pink dashed line, trained on actively sampled data and using estimated underrepresentation label for final-model re-weighted training) almost dominates **Random** (blue solid line, trained on randomly sampled data and using true group label for final model training) despite not having access to true group label in the final reweighted training, highlighting the importance of the data distribution in the model's accuracy-fairness performance. **(2) Label Quality Matters**: Comparing the variants of ISP (Identity v.s. Gap) in Table 1, we see a clear impact of the quality of introspection signal to the performance of the AL model. For example, for active learning performance, we see that the sampling rate ISP-Identity is significantly better than ISP-Error. However, for toxicity detection where the group label suffers an under-coverage issue (i.e., the group definition excludes potentially identity-mention comments where raters disagree, see **Datasets** section of Appendix D.2), we see that ISP-Error in fact strongly outperforms ISP-Identity in accuracy-fairness performance. This validates the observation from previous literature on the failure mode of bias-mitigation methods when the available group annotation does not cover all sources of dataset bias, and speaks to the importance of high quality estimation methods that can detect underpresentation in the presence of unknown sources of bias [126]. Appendix D.3 contains further ablation analysis.

References

[1] Jacob D Abernethy, Pranjal Awasthi, Matthäus Kleindessner, Jamie Morgenstern, Chris Russell, and Jie Zhang. Active sampling for min-max fairness. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 53–65. PMLR, 17–23 Jul 2022.

[2] Ben Adlam and Jeffrey Pennington. Understanding double descent requires a fine-grained bias-variance decomposition. *Advances in neural information processing systems*, 33:11022–11032, 2020.

[3] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International Conference on Machine Learning*, pp. 60–69. PMLR, 2018.

[4] Sharat Agarwal, Sumanyu Muku, Saket Anand, and Chetan Arora. Does data repair lead to fair models? curating contextually fair data to reduce model bias. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3298–3307, 2022.

[5] Alexander Amini, Ava P Soleimany, Wilko Schwarting, Sangeeta N Bhatia, and Daniela Rus. Uncovering and mitigating algorithmic bias through learned latent structure. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 289–295, 2019.

[6] Hadis Anahideh, Abolfazl Asudeh, and Saravanan Thirumuruganathan. Fair active learning. *Expert Systems with Applications*, 199:116981, 2022.

[7] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[8] Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*, 2019.

[9] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *International Conference on Machine Learning*, pp. 528–539. PMLR, 2020.

[10] Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. Data decisions and theoretical implications when adversarially learning fair representations. *arXiv preprint arXiv:1707.00075*, 2017.

[11] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion proceedings of the 2019 world wide web conference*, pp. 491–500, 2019.

[12] Frédéric Branchaud-Charron, Parmida Atighehchian, Pau Rodríguez, Grace Abuhamad, and Alexandre Lacoste. Can active learning preemptively mitigate fairness issues? *arXiv preprint arXiv:2104.06879*, 2021.

[13] Jonathon Byrd and Zachary Lipton. What is the effect of importance weighting in deep learning? In *International Conference on Machine Learning*, pp. 872–881. PMLR, 2019.

[14] Tianle Cai, Ruiqi Gao, Jason Lee, and Qi Lei. A theory of label propagation for subpopulation shift. In *International Conference on Machine Learning*, pp. 1170–1182. PMLR, 2021.

[15] William Cai, Ro Encarnacion, Bobbie Chern, Sam Corbett-Davies, Miranda Bogen, Stevie Bergman, and Sharad Goel. Adaptive sampling strategies to construct equitable training datasets. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, pp. 1467–1478, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393522. doi: 10.1145/3531146.3533203.

[16] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32, 2019.

[17] Kaidi Cao, Yining Chen, Junwei Lu, Nikos Arechiga, Adrien Gaidon, and Tengyu Ma. Heteroskedastic and imbalanced deep learning with adaptive regularization. In *International Conference on Learning Representations*, 2020.

[18] Simon Caton and Christian Haas. Fairness in machine learning: A survey. *arXiv preprint arXiv:2010.04053*, 2020.

[19] Irene Chen, Fredrik D Johansson, and David Sontag. Why is my classifier discriminatory? *Advances in neural information processing systems*, 31, 2018.

[20] Yining Chen, Colin Wei, Ananya Kumar, and Tengyu Ma. Self-training avoids using spurious features under domain shift. *Advances in Neural Information Processing Systems*, 33:21061–21071, 2020.

[21] Pengyu Cheng, Weituo Hao, Siyang Yuan, Shijing Si, and Lawrence Carin. Fairfil: Contrastive neural debiasing method for pretrained text encoders. In *International Conference on Learning Representations*, 2020.

[22] Valeriia Cherepanova, Vedant Nanda, Micah Goldblum, John P Dickerson, and Tom Goldstein. Technical challenges for training fair neural networks. *arXiv preprint arXiv:2102.06764*, 2021.

[23] Jianfeng Chi, William Shand, Yaodong Yu, Kai-Wei Chang, Han Zhao, and Yuan Tian. Conditional supervised contrastive learning for fair text classification. *arXiv preprint arXiv:2205.11485*, 2022.

[24] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. *arXiv preprint arXiv:1909.03683*, 2019.

[25] Mark Collier, Basil Mustafa, Efi Kokiopoulou, Rodolphe Jenatton, and Jesse Berent. Correlated input-dependent label noise in large-scale image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1551–1560, 2021.

[26] Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *International Conference on Machine Learning*, pp. 2189–2200. PMLR, 2021.

[27] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

[28] Pedro Domingos. A unified bias-variance decomposition and its applications. In *17th International Conference on Machine Learning*, pp. 231–238, 2000.

[29] Mengnan Du, Subhabrata Mukherjee, Guanchu Wang, Ruixiang Tang, Ahmed Awadallah, and Xia Hu. Fairness via representation neutralization. *Advances in Neural Information Processing Systems*, 34:12091–12103, 2021.

[30] Michael Dusenberry, Ghassen Jerfel, Yeming Wen, Yian Ma, Jasper Snoek, Katherine Heller, Balaji Lakshminarayanan, and Dustin Tran. Efficient and scalable bayesian neural nets with rank-1 factors. In *International conference on machine learning*, pp. 2782–2792. PMLR, 2020.

[31] Sanghamitra Dutta, Dennis Wei, Hazar Yueksel, Pin-Yu Chen, Sijia Liu, and Kush Varshney. Is there a trade-off between fairness and accuracy? a perspective using mismatched hypothesis testing. In *International Conference on Machine Learning*, pp. 2803–2813. PMLR, 2020.

[32] Sebastian Farquhar, Yarin Gal, and Tom Rainforth. On statistical bias in active learning: How and when to fix it. In *International Conference on Learning Representations*, 2020.

[33] Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33: 2881–2891, 2020.

[34] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

[35] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International Conference on Machine Learning*, pp. 1183–1192. PMLR, 2017.

[36] Karan Goel, Albert Gu, Yixuan Li, and Christopher Re. Model patching: Closing the subgroup performance gap with data augmentation. In *International Conference on Learning Representations*, 2020.

[37] Umang Gupta, Aaron M Ferber, Bistra Dilkina, and Greg Ver Steeg. Controllable guarantees for fair outcomes via contrastive information estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7610–7619, 2021.

[38] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 297–304. JMLR Workshop and Conference Proceedings, 2010.

[39] Kimia Hamidieh, Haoran Zhang, and Marzyeh Ghassemi. Evaluating and improving robustness of self-supervised representations to spurious correlations. In *ICML 2022: Workshop on Spurious Correlations, Invariance and Stability*, 2022.

[40] Romana Hasnain-Wynia, David W Baker, David Nerenz, Joe Feinglass, Anne C Beal, Mary Beth Landrum, Raj Behal, and Joel S Weissman. Disparities in health care are driven by where minority patients seek care: examination of the hospital quality alliance measures. *Archives of internal medicine*, 167(12):1233–1239, 2007.

[41] Marton Havasi, Rodolphe Jenatton, Stanislav Fort, Jeremiah Zhe Liu, Jasper Snoek, Balaji Lakshminarayanan, Andrew Mingbo Dai, and Dustin Tran. Training independent subnetworks for robust prediction. In *International Conference on Learning Representations*, 2020.

[42] He He, Sheng Zha, and Haohan Wang. Unlearn dataset bias in natural language inference by fitting the residual. *arXiv preprint arXiv:1908.10763*, 2019.

[43] Max Hort, Zhenpeng Chen, Jie M Zhang, Federica Sarro, and Mark Harman. Bia mitigation for machine learning classifiers: A comprehensive survey. *arXiv preprint arXiv:2207.07068*, 2022.

[44] Neil Houlsby, Ferenc Huszar, Zoubin Ghahramani, and Mate Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

[45] Aapo Hyvarinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. *Advances in neural information processing systems*, 29, 2016.

[46] Badr Youbi Idrissi, Martin Arjovsky, Mohammad Pezeshki, and David Lopez-Paz. Simple data balancing achieves competitive worst-group-accuracy. In *Conference on Causal Learning and Reasoning*, pp. 336–351. PMLR, 2022.

[47] Pavel Izmailov, Sharad Vikram, Matthew D Hoffman, and Andrew Gordon Gordon Wilson. What are bayesian neural network posteriors really like? In *International conference on machine learning*, pp. 4629–4640. PMLR, 2021.

[48] Sangwon Jung, Sanghyuk Chun, and Taesup Moon. Learning fair classifiers with partially annotated group labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10348–10357, 2022.

[49] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *International Conference on Learning Representations*, 2019.

[50] 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*, pp. 9012–9020, 2019.

[51] Nayeong Kim, Sehyun Hwang, Sungsoo Ahn, Jaesik Park, and Suha Kwak. Learning debiased classifier with biased committee. In *ICML 2022: Workshop on Spurious Correlations, Invariance and Stability*, 2022.

[52] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. In *ICML 2022: Workshop on Spurious Correlations, Invariance and Stability*, 2022.

[53] Andreas Kirsch, Joost Van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32, 2019.

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

[55] Suraj Kothawade, Atharv Savarkar, Venkat Iyer, Ganesh Ramakrishnan, and Rishabh Iyer. Clinical: Targeted active learning for imbalanced medical image classification. In Ghada Zamzmi, Sameer Antani, Ulas Bagci, Marius George Linguraru, Sivaramakrishnan Rajaraman, and Zhiyun Xue (eds.), *Medical Image Learning with Limited and Noisy Data*, pp. 119–129, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-16760-7.

[56] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pp. 5815–5826. PMLR, 2021.

[57] Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. Fairness without demographics through adversarially reweighted learning. *Advances in neural information processing systems*, 33:728–740, 2020.

[58] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[59] Tai Le Quy, Arjun Roy, Vasileios Iosifidis, Wenbin Zhang, and Eirini Ntoutsi. A survey on datasets for fairness-aware machine learning. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, pp. e1452, 2022.

[60] Jungsoo Lee, Eungyeup Kim, Juyoung Lee, Jihyeon Lee, and Jaegul Choo. Learning debiased representation via disentangled feature augmentation. *Advances in Neural Information Processing Systems*, 34:25123–25133, 2021.

[61] Jungsoo Lee, Jeonghoon Park, Daeyoung Kim, Juyoung Lee, Edward Choi, and Jaegul Choo. Biasensemble: Revisiting the importance of amplifying bias for debiasing. *arXiv preprint arXiv:2205.14594*, 2022.

[62] Daniel Levy, Yair Carmon, John C Duchi, and Aaron Sidford. Large-scale methods for distributionally robust optimization. *Advances in Neural Information Processing Systems*, 33: 8847–8860, 2020.

[63] Mingchen Li, Mahdi Soltanolkotabi, and Samet Oymak. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *International conference on artificial intelligence and statistics*, pp. 4313–4324. PMLR, 2020.

[64] Yi Li and Nuno Vasconcelos. Repair: Removing representation bias by dataset resampling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9572–9581, 2019.

[65] Yunyi Li, Maria De-Arteaga, and Maytal Saar-Tsechansky. More data can lead us astray: Active data acquisition in the presence of label bias. *arXiv preprint arXiv:2207.07723*, 2022.

[66] 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 *International Conference on Machine Learning*, pp. 6781–6792. PMLR, 2021.

[67] Jeremiah Zhe Liu, Shreyas Padhy, Jie Ren, Zi Lin, Yeming Wen, Ghassen Jerfel, Zack Nado, Jasper Snoek, Dustin Tran, and Balaji Lakshminarayanan. A simple approach to improve single-model deep uncertainty via distance-awareness. *arXiv preprint arXiv:2205.00403*, 2022.

[68] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

[69] Francesco Locatello, Gabriele Abbati, Thomas Rainforth, Stefan Bauer, Bernhard Schölkopf, and Olivier Bachem. On the fairness of disentangled representations. *Advances in Neural Information Processing Systems*, 32, 2019.

[70] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019.

[71] Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. A simple baseline for bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[72] Natalia Martinez, Martin Bertran, and Guillermo Sapiro. Minimax pareto fairness: A multi objective perspective. In *International Conference on Machine Learning*, pp. 6755–6764. PMLR, 2020.

[73] Natalia L Martinez, Martin A Bertran, Afroditi Papadaki, Miguel Rodrigues, and Guillermo Sapiro. Blind pareto fairness and subgroup robustness. In *International Conference on Machine Learning*, pp. 7492–7501. PMLR, 2021.

[74] Kayo Matsushita, Kayo Matsushita, and Hasebe. *Deep active learning*. Springer, 2018.

[75] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6): 1–35, 2021.

[76] Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lucic. Revisiting the calibration of modern neural networks. *Advances in Neural Information Processing Systems*, 34:15682–15694, 2021.

[77] Yifei Ming, Hang Yin, and Yixuan Li. On the impact of spurious correlation for out-of-distribution detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 10051–10059, 2022.

[78] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684, 2020.

[79] Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

[80] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32, 2019.

[81] Sungho Park, Jewook Lee, Pilhyeon Lee, Sunhee Hwang, Dohyung Kim, and Hyeran Byun. Fair contrastive learning for facial attribute classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10389–10398, 2022.

[82] Andrija Petrović, Mladen Nikolić, Sandro Radovanović, Boris Delibašić, and Miloš Jovanović. Fair: Fair adversarial instance re-weighting. *Neurocomputing*, 476:14–37, 2022.

[83] David Pfau. A generalized bias-variance decomposition for bregman divergences. *Unpublished Manuscript*, 2013.

[84] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020.

[85] Ruggero Ragonesi, Riccardo Volpi, Jacopo Cavazza, and Vittorio Murino. Learning unbiased representations via mutual information backpropagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2729–2738, 2021.

[86] Piyush Rai, Avishek Saha, Hal Daumé III, and Suresh Venkatasubramanian. Domain adaptation meets active learning. In *Proceedings of the NAACL HLT 2010 Workshop on Active Learning for Natural Language Processing*, pp. 27–32, 2010.

[87] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54 (9):1–40, 2021.

[88] Esther Rolf, Theodora T Worledge, Benjamin Recht, and Michael Jordan. Representation matters: Assessing the importance of subgroup allocations in training data. In *International Conference on Machine Learning*, pp. 9040–9051. PMLR, 2021.

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

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

[91] Victor Sanh, Thomas Wolf, Yonatan Belinkov, and Alexander M Rush. Learning from others' mistakes: Avoiding dataset biases without modeling them. *arXiv preprint arXiv:2012.01300*, 2020.

[92] Burr Settles. Active learning literature survey. *Machine Learning*, 15(2):201–221, 1994.

[93] Amr Sharaf, Hal Daume III, and Renkun Ni. Promoting fairness in learned models by learning to active learn under parity constraints. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 2149–2156, 2022.

[94] Aili Shen, Xudong Han, Trevor Cohn, Timothy Baldwin, and Lea Frermann. Contrastive learning for fair representations. *arXiv preprint arXiv:2109.10645*, 2021.

[95] Nimit Sohoni, Jared Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *Advances in Neural Information Processing Systems*, 33:19339–19352, 2020.

[96] Nimit Sohoni, Maziar Sanjabi, Nicolas Ballas, Aditya Grover, Shaoliang Nie, Hamed Firooz, and Christopher Ré. Barack: Partially supervised group robustness with guarantees. *arXiv preprint arXiv:2201.00072*, 2021.

[97] Hwanjun Song, Minseok Kim, Dongmin Park, and Jae-Gil Lee. Prestopping: How does early stopping help generalization against label noise? In *ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning*, 2020.

[98] Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. Density ratio estimation: A comprehensive review (statistical experiment and its related topics). *RIMS Kokyuroku*, 1703: 10–31, 2010.

[99] Ki Hyun Tae and Steven Euijong Whang. Slice tuner: A selective data acquisition framework for accurate and fair machine learning models. In *Proceedings of the 2021 International Conference on Management of Data*, pp. 1771–1783, 2021.

[100] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11662–11671, 2020.

[101] Enzo Tartaglione, Carlo Alberto Barbano, and Marco Grangetto. End: Entangling and disentangling deep representations for bias correction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13508–13517, 2021.

[102] Damien Teney, Ehsan Abbasnejad, and Anton van den Hengel. Unshuffling data for improved generalization in visual question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1417–1427, 2021.

[103] Christopher J Tosh and Daniel Hsu. Simple and near-optimal algorithms for hidden stratification and multi-group learning. In *International Conference on Machine Learning*, pp. 21633–21657. PMLR, 2022.

[104] Dustin Tran, Jeremiah Liu, Michael W Dusenberry, Du Phan, Mark Collier, Jie Ren, Kehang Han, Zi Wang, Zelda Mariet, Huiyi Hu, et al. Plex: Towards reliability using pretrained large model extensions. *arXiv preprint arXiv:2207.07411*, 2022.

[105] Yao-Hung Hubert Tsai, Tianqin Li, Martin Q Ma, Han Zhao, Kun Zhang, Louis-Philippe Morency, and Ruslan Salakhutdinov. Conditional contrastive learning with kernel. In *International Conference on Learning Representations*, 2021.

[106] Yao-Hung Hubert Tsai, Martin Q Ma, Han Zhao, Kun Zhang, Louis-Philippe Morency, and Ruslan Salakhutdinov. Conditional contrastive learning: Removing undesirable information in self-supervised representations. *arXiv preprint arXiv:2106.02866*, 2021.

[107] Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*, 2019.

[108] Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 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, 2020.

[109] Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. Towards debiasing nlu models from unknown biases. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7597–7610, 2020.

[110] Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In *International conference on machine learning*, pp. 9690–9700. PMLR, 2020.

[111] Joost van Amersfoort, Lewis Smith, Andrew Jesson, Oscar Key, and Yarin Gal. On feature collapse and deep kernel learning for single forward pass uncertainty. *arXiv preprint arXiv:2102.11409*, 2021.

[112] Xuezhi Wang, Tzu-Kuo Huang, and Jeff Schneider. Active transfer learning under model shift. In *International Conference on Machine Learning*, pp. 1305–1313. PMLR, 2014.

[113] Florian Wenzel, Kevin Roth, Bastiaan S Veeling, Jakub Swiatkowski, Linh Tran, Stephan Mandt, Jasper Snoek, Tim Salimans, Rodolphe Jenatton, and Sebastian Nowozin. How good is the bayes posterior in deep neural networks really? *arXiv preprint arXiv:2002.02405*, 2020.

[114] Florian Wenzel, Jasper Snoek, Dustin Tran, and Rodolphe Jenatton. Hyperparameter ensembles for robustness and uncertainty quantification. *Advances in Neural Information Processing Systems*, 33:6514–6527, 2020.

[115] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

[116] Andrew G Wilson and Pavel Izmailov. Bayesian deep learning and a probabilistic perspective of generalization. *Advances in neural information processing systems*, 33:4697–4708, 2020.

[117] Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial intelligence and statistics*, pp. 370–378. PMLR, 2016.

[118] Sang Michael Xie, Ananya Kumar, Robbie Jones, Fereshte Khani, Tengyu Ma, and Percy Liang. In-n-out: Pre-training and self-training using auxiliary information for out-of-distribution robustness. In *International Conference on Learning Representations*, 2020.

[119] Da Xu, Yuting Ye, and Chuanwei Ruan. Understanding the role of importance weighting for deep learning. In *International Conference on Learning Representations*, 2020.

[120] Yadollah Yaghoobzadeh, Soroush Mehri, Remi Tachet des Combes, Timothy J Hazen, and Alessandro Sordoni. Increasing robustness to spurious correlations using forgettable examples. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 3319–3332, 2021.

[121] Yuzhe Yang and Zhi Xu. Rethinking the value of labels for improving class-imbalanced learning. *Advances in neural information processing systems*, 33:19290–19301, 2020.

[122] Jingzhao Zhang, Aditya Krishna Menon, Andreas Veit, Srinadh Bhojanapalli, Sanjiv Kumar, and Suvrit Sra. Coping with label shift via distributionally robust optimisation. In *International Conference on Learning Representations*, 2020.

[123] 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.

[124] Bowen Zhao, Chen Chen, Qi Ju, and Shutao Xia. Learning debiased models with dynamic gradient alignment and bias-conflicting sample mining. *arXiv preprint arXiv:2111.13108*, 2021.

[125] Han Zhao and Geoff Gordon. Inherent tradeoffs in learning fair representations. *Advances in neural information processing systems*, 32, 2019.

[126] Chunting Zhou, Xuezhe Ma, Paul Michel, and Graham Neubig. Examining and combating spurious features under distribution shift. In *International Conference on Machine Learning*, pp. 12857–12867. PMLR, 2021.

[127] Wei Zhu, Haitian Zheng, Haofu Liao, Weijian Li, and Jiebo Luo. Learning bias-invariant representation by cross-sample mutual information minimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15002–15012, 2021.

[128] Xiangxin Zhu, Dragomir Anguelov, and Deva Ramanan. Capturing long-tail distributions of object subcategories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 915–922, 2014.

A Additional Background

A.1 Recap: Notation and Problem Setup.

Dataset with subgroups: We consider a dataset D where each example $\{\mathbf{x}_i, y_i\}$ ($\mathbf{x}_i \in \mathscr{X}$ denotes the features and $y_i \in \mathscr{Y}$ the label) is associated with a discrete group label $g_i \in \mathscr{G} = \{1, \ldots, |\mathscr{G}|\}$.
Joint data distribution: We denote $\mathscr{D} = P(y, \mathbf{x}, g)$ as the joint distribution of the label, feature and groups, so that D above can be understood as a size-n set of i.i.d. samples from $\mathscr{D}$. Notice that this formulation implies a flexible noise model $P(y|\mathbf{x}, g)$ that depends on $(\mathbf{x}, g)$. It also implies a flexible group-specific distribution $P(y, \mathbf{x}|g)$, where the joint distribution of $(y, \mathbf{x})$ varies by group. Note however that we assume that the group label does not have additional predictive power beyond the features, i.e., we assume that $P(y|\mathbf{x}, g) = P(y|\mathbf{x})$.
Subgroup prevelance: We denote the prevalence of each group as $\gamma_g = E_{(y,\mathbf{x},g)\sim\mathscr{D}}(1_{G=g})$. As a result, the notion of *dataset bias* is reflected as the imbalance in group distribution $P(G) = [\gamma_1, \ldots, \gamma_{|\mathscr{G}|}]$ [88]. In the applications we consider, it is often feasible to identify a subset of *underrepresented* groups $\mathscr{B} \subset \mathscr{G}$ which are not sufficiently represented in the population distribution $\mathscr{D}$ and have $\gamma_g \ll \frac{1}{|\mathscr{G}|}$ [89, 90]. To this end, we also specify $\mathscr{D}^* = P(y, \mathbf{x}|g)P^*(g)$ an optimal distribution, where $P^*(g)$ is an ideal group distribution (i.e., uniform such that $P^*(g) = \gamma_g^* = \frac{1}{|\mathscr{G}|}$) so that all groups have sufficient representation in the data.
Loss function: We assume a loss function $L(y, \hat{y})$, that denotes the loss incurred when the predicted label is $\hat{y}$ while the actual label is y.
Hypothesis space: We consider learning a predictor from a hypothesis space $\mathscr{F}$ of functions $f : \mathscr{X} \mapsto \mathscr{Y}$. We assume that the hypothesis space is well-specified, i.e., that it contains the Bayes-optimal predictor $\tilde{y} : \mathscr{X} \to \mathscr{Y}$:

$$\tilde{y}(\mathbf{x}) = \arg\min_{y' \in \mathscr{Y}} E_{y\sim P(y|\mathbf{x})}(L(y, y')).$$

We require the model class $\mathscr{F}$ to come with certain degree of smoothness, so that the model $f \in \mathscr{F}$ cannot arbitrarily overfit to the noisy labels during the course of training. In the case of over-parameterized models, this usually implies $\mathscr{F}$ is subject to certain regularization that is appropriate for the model class (e.g., early stopping for SGD-trained neural networks) [63].

A.2 Disentangling model error under noise and bias

Given a dataset $D \sim \mathscr{D}$ and a loss function L , we consider learning the prediction function $f_D = \arg\min_{f\in\mathscr{F}} L(f, y|D)$, where $L(y, f|D) = \sum_{\{\mathbf{x}_i, y_i\}\in D} L(y_i, f(\mathbf{x}_i))$. Following the previous work [83], we denote the *ensemble predictor* $\bar{f} = \arg\min_{f\in\mathscr{F}} E_{D\sim\mathscr{D}}(L(f_D, f))$ over ensemble members f_D's, where each f_D is trained on a random draw of training dataset $D \sim \mathscr{D}$, and $\tilde{y}(\mathbf{x}) = \arg\min_{y'\in\mathscr{Y}} E_{y\sim P(y|\mathbf{x})}(L(y, y'))$ the (Bayes) optimal predictor. For test example $\{y_i, \mathbf{x}_i\}$, we can decompose the predictive error of a trained model $f_D(\mathbf{x})$ using a generalized bias-variance decomposition for Bregman divergence:

Proposition A.1 (Noise-Bias-Variance Decomposition under Bregman divergence [28, 83])**.** *Given a loss function of the Bregman divergence family, for a test example $\{y, \mathbf{x}\}$ the expected prediction loss $L(y, f_D(\mathbf{x}))$ of an empirical predictor f_D can be decomposed as:*

$$E_D\left[L(y, f_D(\mathbf{x}))\right] = \underbrace{E_D\left[L(y, \tilde{y}(\mathbf{x}))\right]}_{\textit{Noise}} + \underbrace{L(\tilde{y}(\mathbf{x}), \bar{f}(\mathbf{x}))}_{\textit{Bias}} + \underbrace{E_D\left[L(\bar{f}(\mathbf{x}), f_D(\mathbf{x}))\right]}_{\textit{Uncertainty}} \tag{4}$$

Given a fixed data distribution $\mathscr{D}$, the first term $E_D\left[L(y, f^*(\mathbf{x}))\right]$ quantifies the *irreducible noise* that is due to the stochasticity in the noisy observation y. The third term $E_D\left[L(\bar{f}(\mathbf{x}), f_D(\mathbf{x}))\right]$ quantifies the *variance* in the prediction, which can be due to variations in the finite-size data D, the stochasticity in the randomized learning algorithm $\mathscr{F} \times D \to f_D$, or the randomness in the initialization of an overparameterized model [2]. Finally, the middle term $L(\tilde{y}(\mathbf{x}), \bar{f}(\mathbf{x}))$ quantifies the *bias* between $\tilde{y}(\mathbf{x})$ (i.e., the "true label") and the ensemble predictor $\bar{f}$ learned from the empirical data $D \sim \mathscr{D}$. It is inherent to the specification of the model class and cannot be eliminated by ensembling, e.g., it can be caused by model misspecification, missing features, or regularization. To make the idea concrete, consider a simple example where we fit a ridge regression model $f(\mathbf{x}_i) = \beta^\top \mathbf{x}_i$ to the Gaussian observation data $y_i = \theta^\top \mathbf{x}_i + \varepsilon, \varepsilon \sim N(0, \sigma^2)$ under an imbalanced experiment design, where we have $|\mathscr{G}|$ treatment groups and n_g observations in each group. Here, $\mathbf{x}_i = [1_{g_i=1}, \ldots, 1_{g_i=|\mathscr{G}|}]$ is a $|\mathscr{G}| \times 1$

one-hot indicator of the membership of g_i for each group in $\mathscr{G}$, and $\theta = [\theta_1, \ldots, \theta_{|\mathscr{G}|}]$ is the true effect for each group. Then, under ridge regression, the noise-bias-variance decomposition for group g is $E_D(L(y, f_D)) = \sigma^2 + \frac{(\lambda\theta_g)^2}{(n_g+\lambda)^2} + \frac{\sigma^2 n_g}{(n_g+\lambda)^2}$, where the regularization parameter λ modulates a trade-off between the *bias* and *variance* terms.

A.3 Further Decomposition

Further Uncertainty Decomposition for Probabilistic Models As an aside, when the predictive model f_D is probabilistic (e.g., the model generates a posterior predictive distribution $P(f|D)$ rather than a point estimate f), the *variance* in Equation (4) is further decomposed as:

$$E_D\left[L(\bar{f}(\mathbf{x}), f_D(\mathbf{x}))\right] = \underbrace{E_D\left[L(\bar{f}(\mathbf{x}), \mu_D(\mathbf{x}))\right]}_{\textit{Ensemble Diversity}} + \underbrace{E_D E_{f\sim P(f|D)}\left[L(\mu_D(\mathbf{x}), f(\mathbf{x}))\right]}_{\textit{Posterior Variance}} \tag{5}$$

where $\mu_D(\mathbf{x}) = E_{f\sim P(f|D)}[f(\mathbf{x})]$ is the posterior mean, and $v_D(\mathbf{x}) = E_{f\sim P(f|D)}\left[L(\mu_D(\mathbf{x}), f(\mathbf{x}))\right]$ is the posterior variance of each ensemble member. As shown, comparing to an ensemble of deterministic models, the ensemble of probabilistic models provides additional flexibility in quantifying model uncertainty via the extra term of expected posterior variance.

Further Bias Decomposition for Minority Groups For the examples $\mathbf{x}$ coming from the under-represented groups with $\gamma_g \ll \frac{1}{|\mathscr{G}|}$, the bias term can be further decomposed into:

$$L(\tilde{y}(\mathbf{x}), \bar{f}(\mathbf{x})) = \underbrace{L(\tilde{y}(\mathbf{x}), \bar{f}^*(\mathbf{x}))}_{\textit{Bias, Model}} + \underbrace{\mathscr{E}(\bar{f}^*(\mathbf{x}), \bar{f}(\mathbf{x}))}_{\textit{Excess Bias, Data}}, \tag{6}$$

where $\bar{f}^* = \arg\min_{f\in\mathscr{F}} E_{D^*\sim\mathscr{D}^*}(L(f_{D^*}, f))$ is the optimal ensemble predictor based on size-n datasets D^* sampled from the optimal distribution $\mathscr{D}^*$ where all groups have equal representation. Here, $L(\tilde{y}(\mathbf{x}), \bar{f}^*(\mathbf{x}))$ is the bias inherent to the model class and cannot be eliminated by ensembling. It can be caused by model misspecification, missing features, or regularization. On the other hand, $\mathscr{E}(\bar{f}^*(\mathbf{x}), \bar{f}(\mathbf{x})) = L(\tilde{y}(\mathbf{x}), \bar{f}(\mathbf{x})) - L(\tilde{y}(\mathbf{x}), \bar{f}^*(\mathbf{x}))$ indicates the "excess bias" for the underrepresented groups caused by the imbalance in the group distribution $P(G)$ in the data-generation distribution $\mathscr{D}$.

To make the idea concrete, consider the ridge regression example from the previous section, where the noise-bias-variance decomposition for group g is $E_D(L(y, f_D)) = \sigma^2 + \frac{(\lambda\theta_g)^2}{(n_g+\lambda)^2} + \frac{\sigma^2 n_g}{(n_g+\lambda)^2}$, with the regularization parameter λ modulating a trade-off between the *bias* and *variance* terms (Appendix E). Consequently, for an underrepresented group with small size $\gamma_g \ll \frac{1}{|\mathscr{G}|}$, its predictive bias $\frac{(\lambda\theta_g)^{*2}}{(n_g+\lambda)^2}$ is exacerbated due to lacking sufficient statistical information to counter the regularization bias, incuring an excessive bias of $\mathscr{E}(\bar{f}^*(\mathbf{x}), \bar{f}(\mathbf{x})) \approx \frac{\lambda\theta_g}{n\gamma_g^*\gamma_g}(\gamma_g^* - \gamma_g)$ when compared to an optimal ensemble predictor $\bar{f}^*$ trained from a perfectly balanced size-n datasets with $\gamma_g^* = 1/|\mathscr{G}|$.

A.4 Modern uncertainty estimation techniques in deep learning

For a deep classifier $p(\mathbf{x}) = \sigma(f(\mathbf{x}))$ with logit function $f(\mathbf{x}) = \beta^\top h(\mathbf{x})$ and $h(\mathbf{x}) \in \mathbb{R}^M$ the last-layer hidden embeddings, the modern deep uncertainty methods quantifies model uncertainty by enabling it to generate random samples from a predictive distribution. That is, for a model trained on data $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, given a test data point $\mathbf{x}_{test}$, the model can return a size-K sample:

$$\{f_k(\mathbf{x}_{test})\}_{k=1}^K \sim P(f|\mathbf{x}_{test}, D).$$

For example, in Monte Carlo Dropout [34], the samples is generated by perturbing the dropout mask in the learned predictive function $f(\cdot) = \beta^\top h(\cdot)$'s embedding function $h(\cdot)$, while in Deep Ensemble [58], the sample comes directly from the multiple parallel-trained ensemble members. Finally, in a neural Gaussian process model [117, 67, 111], the samples are generated from a Gaussian process model using the hidden embedding function $h(\mathbf{x})$ as the input. For example, for classification problems, the predictive variance of the Gaussian process model $v(\mathbf{x}_{test}) = Var(f|\mathbf{x}_{test}, D)$ can be expressed as (Williams & Rasmussen [115], Chapter 3):

$$v(\mathbf{x}_{test}) = \mathbf{k}(\mathbf{x}_{test})_{1\times n}^\top \mathbf{V}_{n\times n}\mathbf{k}(\mathbf{x}_{test})_{n\times 1};$$

where $\mathbf{V}_{n\times n}$ is a fixed matrix computed from training data, and $\mathbf{k}(\mathbf{x}_{test}) = [k(\mathbf{x}_{test}, \mathbf{x}_1), \ldots, k(\mathbf{x}_{test}, \mathbf{x}_n)]$ is a vector of kernel distances between $\mathbf{x}_{test}$ and the training examples $\{\mathbf{x}_i\}_{i=1}^n$. The kernel function k is commonly defined to be a monotonic function of the hidden embedding distance, e.g., $k(\mathbf{x}_{test}, \mathbf{x}_i) = exp(-||h(\mathbf{x}_{test}) - h(\mathbf{x}_i)||_2^2)$ for the RBF kernel. As a result, the predictive uncertainty for a data points $\mathbf{x}_i$ is determined by the distance between $\mathbf{x}_{test}$ from the training data $\{\mathbf{x}_i\}_{i=1}^n$. Consequently, a DNN model's quality in representation learning has non-trivial impact on its uncertainty performance. Although first mentioned in the context of neural Gaussian process, this connection between the quality of representation learning and the quality of uncertainty quantification also holds for state-of-the-art techniques such as Deep Ensemble, as model averaging cannot eliminate the systematic errors in representation learning and consequently the issue in uncertainty quantification (for example, see Figure 6 and the corresponding ensemble uncertainty surface Figure 2f).

Neural Gaussian Process Ensemble In this work, to comprehensively investigate the effect of different uncertainty techniques, we should to use a Deep Ensemble of neural Gaussian process as our canonical model. That is, we parallel train K neural Gaussian process models $\{f_k\}_{k=1}^K$. Then, given a test data point $\mathbf{x}_{test}$, each ensemble member will return a predictive distribution with means $\{\mu_k(\mathbf{x})\}_{k=1}^K$ and variances $\{v_k(\mathbf{x})\}_{k=1}^K$. Then, we can generate model prediction as $\mathbb{E}_k[\mu_k(\mathbf{x})]$, and quantify uncertainty in one of the two ways:

$$\begin{aligned} \text{Ensemble Diversity}: &\quad Var_k(\mu_k(\mathbf{x})); \\ \text{Posterior Variance}: &\quad \mathbb{E}_k(v_k(\mathbf{x})), \end{aligned}$$

where Var_k, $\mathbb{E}_k$ are empirical means and variances over the ensemble members. As shown, they correspond to the two components of the total model variance under squared error introduced in A.3. We investigate the effectiveness of these two uncertainty signals in the experiments.

B Theoretical Analysis

B.1 Improving Representation Learning and Uncertainty Quantification Under Dataset Bias

Formally, introspective training induces the below guarantee on the model's *bias-awareness* in its hidden representation and uncertainty estimates:

Proposition B.1 (Introspective Training induces Bias-awareness)**.** *Denote $o_b(\mathbf{x}) = p(\mathbf{x}|b = 1)/p(\mathbf{x}|b = 0)$ the odds for $\mathbf{x}$ belongs to the underrepresented group $\mathscr{B}$. For a well-trained model $f = (f_y, f_b)$ that minimizes the introspective training objective (2), so that $p(b = 1|\mathbf{x}) = \sigma(f_b(\mathbf{x}))$, we then have:*

(I) ***(Bias-aware Hidden Representation)*** *The hidden representation $h(\mathbf{x})$ is aware of the likelihood ratio of whether an example $\mathbf{x}$ belongs to the underrepresented group $b = I(g \in \mathscr{B})$, i.e. $p(\mathbf{x}|b = 1)/p(\mathbf{x}|b = 0)$, such that:*

$$\beta_b^\top h(\mathbf{x}) + b_b = \log o_b(\mathbf{x}) + \log \frac{p(b = 1)}{p(b = 0)}. \tag{7}$$

(II) ***(Bias-aware Embedding Distance)*** *For two examples $(\mathbf{x}_1, \mathbf{x}_2)$, the embedding distance $||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2$ is lower bounded by (up to a scaling constant) the odds ratio of whether $\mathbf{x}_1$ belongs to the underrepresented groups versus that for $\mathbf{x}_2$:*

$$||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2 \geq \frac{1}{||\beta_b||_2} \times \max\Big(\log \frac{o_b(\mathbf{x}_1)}{o_b(\mathbf{x}_2)}, \log \frac{o_b(\mathbf{x}_2)}{o_b(\mathbf{x}_1)}\Big), \tag{8}$$

such that the distance between a pair of minority and majority examples $(\mathbf{x}_1, \mathbf{x}_2)$ is large due to the high values of the log odds ratio.

The proof is in Appendix F. Part (I) provides a consistency guarantee for the hidden representation $h(\cdot)$'s ability in expressing the likelihood of whether an example $\mathbf{x}$ belongs to the underrepresented group $\mathscr{B}$, i.e., *bias awareness*. The form of (7) is similar to the representation learning guarantee in the noise contrastive learning literature, as it shares the same underlying principle of encouraging feature diversity and disentanglement via contrastive comparison between groups [38, 98, 45] . Part (II) is a corollary of (7) and provides a direct guarantee on the model's learned embedding distance. It states that under introspective training, the model cannot discard important input features that are not predictive of the target label to a degree that it collapsed the representation of majority and minority examples together (i.e., making $||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2$ excessively small for two examples $(\mathbf{x}_1, \mathbf{x}_2)$ from

the majority and minority group, respectively), creating difficulty for identifying underrepresented groups in the feature space with uncertainty-based active learning. Empirically, we find the benefit of introspective training extends to other uncertainty-based active learning signals as well (e.g., margin and ensemble diversity, see Appendix D.3).

C Method Summary

C.1 Algorithm

Algorithm 1 *Introspective Self-play* (ISP)

Inputs: Training data $D_{train} = \{y_i, \mathbf{x}_i\}_{i=1}^{n}$; (Optional) Group annotation $G_{train} = \{g_i\}_{i=1}^{n}$;
Unlabelled data $D_{pool} = \{\mathbf{x}_j\}_{j=1}^{n'}$.

Output: Predicted probability $\{p(y|\mathbf{x}_j)\}_{j=1}^{n'}$; Bias probability $\{p(b|\mathbf{x}_j)\}_{j=1}^{n'}$; Predictive variance $\{v(\mathbf{x}_j)\}_{j=1}^{n'}$.

▷ Stage I: Label Generation

if $G_{train} \neq \emptyset$ **then**

  $B_{train} = \{b_i = I(g_i \in \mathscr{B})\}$; ▷ Make underrepresentation label using group annotation g_i.

else

  $\hat{B}_{train} = SelfPlayBiasEstimation(D_{train})$. ▷ Estimate underrepresentation label using Algorithm 2

▷ Stage II: Introspective Training

Train $\hat{f}$ on D_{train} with multi-task introspective objective $L((y_i, b_i), \mathbf{x}_i)$. ▷ Equation (2)

Evaluate $\hat{f}$ on $\mathbf{x}_j \in D_{pool}$ to generate sampling signals $\{p(y|\mathbf{x}_j), p(b|\mathbf{x}_j), v(\mathbf{x}_j)\}_{j=1}^{n'}$. ▷ Equation (1)

Algorithm 2 Underrepresentation Label Estimation via *Cross-validated Self-play*

Inputs: Training data $D_{train} = \{y_i, \mathbf{x}_i\}_{i=1}^{n}$.

Output: Estimate underrepresentation labels $\hat{B}_{train}$.

Train K-fold cross-validated ensemble $\{f_k\}_{k=1}^{K}$ with D_{train}.

Compute in-sample and out-of-sample ensemble predictions $\{f_{in,k}(\mathbf{x}_i)\}_{k=1}^{K_{in}}, \{f_{out,k}(\mathbf{x}_i)\}_{k=1}^{K_{out}}$ for all $\mathbf{x}_i \in D_{train}$.

Estimate underrepresentation labels as $\hat{B}_{train} = \{b_i = \mathbb{E}_k[L(\bar{f}_{in}(\mathbf{x}_i), f_{out,k}(\mathbf{x}_i))]\}_{i=1}^{n}$. ▷ (Equation (10))

C.2 Estimating Generalization Gap using Cross-validated Ensemble

A popular practice in the literature is to estimate dataset bias as the predictive error of a single (biased) model. That is, given a trained model f_D, prior work [24, 42, 78, 91, 66] estimates the underrepresentation label as the observed error $L(y_i, f_D(\mathbf{x}_i))$. To better understand this estimator for the generalization error of the underrepresented groups, Consider the noise-bias-variance decomposition (Domingos [28]) of the model error $L(y, f_D)$, which reveals, in the expectation of the random draws of the dataset $D \sim \mathscr{D}$:

$$\underbrace{E_D[L(y, f_D(\mathbf{x}))]}_{error} = \underbrace{E_D\big[L(y, \tilde{y}(\mathbf{x}))\big]}_{noise} + \underbrace{L(\tilde{y}(\mathbf{x}), \bar{f}(\mathbf{x}))}_{bias} + \underbrace{E_D\big[L(\bar{f}(\mathbf{x}), f_D(\mathbf{x}))\big]}_{variance}, \tag{9}$$

where $\tilde{y}(\mathbf{x}) = \arg\min_{y'} E_{y\sim P(y|\mathbf{x})}[L(y, y')]$ is the (Bayes) optimal predictor and $\bar{f}(\mathbf{x}) = \arg\min_f E_D[L(f, f_D(\mathbf{x}))]$ is the 'ensemble' predictor of the single models $\{f_D\}_{D\sim\mathscr{D}}$ trained from random data draws (see Appendix A.2 for a review). From (9), we see that for the purpose of estimating generalization error due to dataset bias, the naive estimator $\hat{b}_0 = L(y, f_D)$ based on single-model error suffers from two issues: (1) $\hat{b}_0$ conflates *noise* (typically arising from label noise or feature ambiguity) with the dataset bias signal we wish to capture, potentially leading to compromised quality in real datasets [57, 65]. (2) As $\hat{b}_0$ is calculated from a single model, its estimate of the *variance* term (an important component of generalization error [121]) is often not stable. This is exacerbated when $\hat{b}_0$ is computed from the training error, since model variance tends to be severely underestimated by

DNNs [66].[2] This observation motivates us to propose ***cross-validated self-play***, a simple method to estimate a model's generalization gap. Briefly, given training data D divided into K splits, we train a bootstrap ensemble of K models $\{f_k\}_{k=1}^{K}$ with ERM training, where each f_k sees a fraction of the training data (see Appendix Fig. 5). As a result, for each $(\mathbf{x}_i, y_i)$, there exists a collection of in-sample predictions $\{f_{in,k'}(\mathbf{x}_i)\}_{k'=1}^{K_{in}}$ trained on data splits containing $(\mathbf{x}_i, y_i)$, and a collection of out-of-sample predictions $\{f_{out,k}(\mathbf{x}_i)\}_{k=1}^{K_{out}}$ trained on data splits not containing $(\mathbf{x}_i, y_i)$. Then, the ***self-play estimator*** of the model's generalization gap is [3]

$$\hat{b}_i = \underbrace{\mathbb{E}_k[L(y_i, f_{out,k}(\mathbf{x}_i))]}_{\textit{estimated error}} - \underbrace{L(y_i, \bar{f}_{in}(\mathbf{x}_i))}_{\textit{estimated noise}} = \mathbb{E}_k[L(\bar{f}_{in}(\mathbf{x}_i), f_{out,k}(\mathbf{x}_i))]. \tag{10}$$

where $\bar{f}_{in}$ is the ensemble prediction based on in-sample predictors $f_{in,k'}$, the expectation $\mathbb{E}_k$ is taken with respect to the out-of-sample predictions, and we are estimating the Bayes optimal predictor $\tilde{y}$ using the in-domain prediction $\bar{f}_{in}$ (since the model class $\mathscr{F}$ is subject to suitable regularization, the $\bar{f}_{in}$'s do not arbitrarily overfit the noisy labels). Compared to the standard alternatives in the literature (e.g., single-model error $L(y, f_D)$), the *self-play* estimator $\hat{b}_i$ has the appealing property of controlling *noise* (by using $\bar{f}_{in}$) while better estimating *variance* (by using expectations over $\bar{f}_{out,k}$), thereby constituting a more informative signal for the underrepresented groups under dataset bias, label noise and feature ambiguity.

Practical Comments Note that due to its cross validation nature, the *self-play* bias estimator $\hat{b}_i$ estimates the generalization error of a weaker model (i.e., trained on a smaller data size $n_{cv} < n$). This is in fact consistent with the practice in the previous debasing literature, where the main model is trained on the error signals from weaker and more biased models [24, 42, 78].

Further, in the context of SGD-trained neural networks, it is important to properly estimate the $\bar{f}_{in}(\mathbf{x}_i)$ so it does not overfit to the training label, via early stopping [63, 68]. This is easy to do in the context of cross validation: during training, we collect the estimated bias $\hat{b}_{i,t}$ across the training epochs $t = 1, \ldots, T$, and select the optimal stopping point t as the first time the out-of-sample error $\mathbb{E}\big[L(y_i, f_{out,k}(\mathbf{x}_i))\big]$ stablizes. In practice, we specify the early-stopping criteria as when the running average (within a window $T' = 5$) of the cross validation error first stablizes below a threshold ε. This is to prevent the situation where the errors for some hard-to-learn examples keep oscillating throughout training and never stabilize.

C.3 Hyperparameters and Computational Complexity

Hyper-parameters The full ISP procedure contains 3 hyper-parameters: The (optional) *cross-validated self-play* in Stage I contains all three hyper-parameters: (1) the number of ensemble models K and (2) the number of examples n_{cv} to train each model. Both are standard to the bootstrap ensemble procedure, and we set them to $K = 10$ and $n_{cv} = n/K$ in this work to ensure the total computation complexity is comparable to training a single model on the full dataset. (3) the early stopping criteria ε for noise estimation (as discussed in the previous section C.2), we set it heuristically to $\varepsilon = 0.1$ in this work after visual inspection of the validation learning curves. The *introspective training* in Stage II does not contain additional hyperparameter other than the standard supervised learning parameters (e.g., learning rate and training epochs). We set these parameters based on a standard supervised learning hyperparameter sweep based on the full data.

Computation Complexity When the group annotation is available, the computation complexity of the ISP procedure (i.e., Stage II only) should be equivalent to the standard ERM procedure. On the other hand, the computational complexity of the full ISP procedure (Stage I + II) should be comparable to that of a standard two-stage debiasing method that trains multiple single models on the full dataset [109, 66, 120, 78, 26, 51].

[2] As an illustrative example, the generalization error of a ridge regression model under orthogonal design and group-specific noise is $E_D(L(y, f_D(\mathbf{x}_g))) = \sigma_g^2 + \frac{(\lambda\theta_g)^2}{(n_g+\lambda)^2} + \frac{\sigma^2 n_g}{(n_g+\lambda)^2}$, where σ_g is the noise level for group $g \in \mathscr{G}$, n_g is the sample size for group $g \in \mathscr{G}$, and λ is the ridge regularization parameter. See Appendix E for details.

[3] In this work, we use mean squared error $L(y, f) = \sqrt{(y - \sigma_{sigmoid}(f))^2}$ for the generalization gap computation, so that $\hat{b}_i \in [0, 1]$.

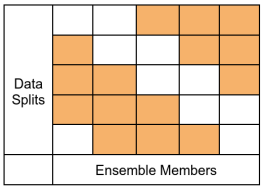

Figure 5: An example of 5-fold cross-validated ensemble. Each ensemble member received 60% of the data (highlighted in orange), and each data split receives 3 in-sample $f_{in,k}$ and 2 out-of-sample predictions $f_{out,k}$.

D Experiment Details and further discussion

D.1 2D Classification

We train a 10-member neural Gaussian process ensemble (as introduced in Appendix A.4), where each ensemble member is based on a 6-layer Dense residual network with 512 hidden units and pre-activation dropout mask (rate = 0.1). The model is trained using Adam optimizer (learning rate = 0.1) under cross entropy loss, and with a batch size 512 for 100 epochs. After training, each ensemble member returns a tuple of predicted label probability, predicted under-representation probability and predictive uncertainty $\{(p_k(y|\mathbf{x}), p_k(b|\mathbf{x}), v_k(\mathbf{x}))\}_{k=1}^{10}$, and we compute the ensemble's predicted probability surface as $\mathbb{E}_k[p(y|\mathbf{x})]$, predicted underrepresentation surface as $\mathbb{E}_k[p_k(b|\mathbf{x})]$, and the predictive uncertainty surface as $\mathbb{E}_k[v_k(y|\mathbf{x})]$, where $\mathbb{E}_k$ is the empirical average over the ensemble member predictions. The predictive uncertainty surface of individual members is shown in Figures 6-7.

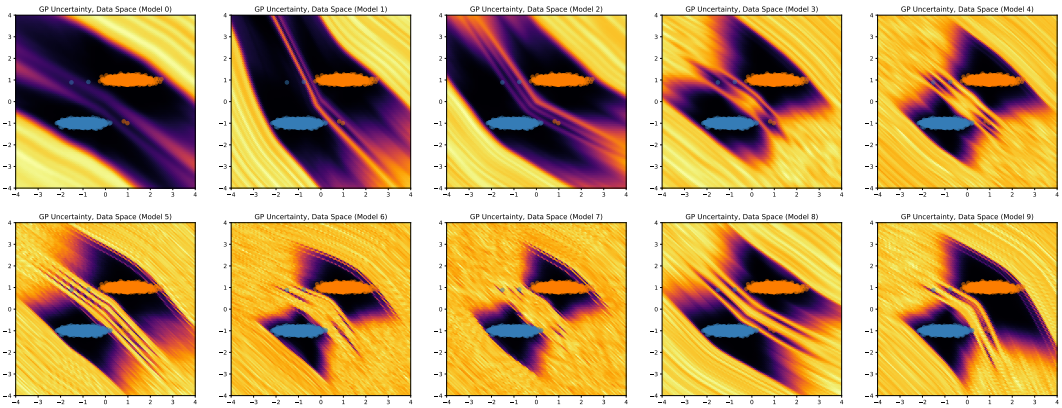

Figure 6: Uncertainty surface of individual ensemble members, ERM training

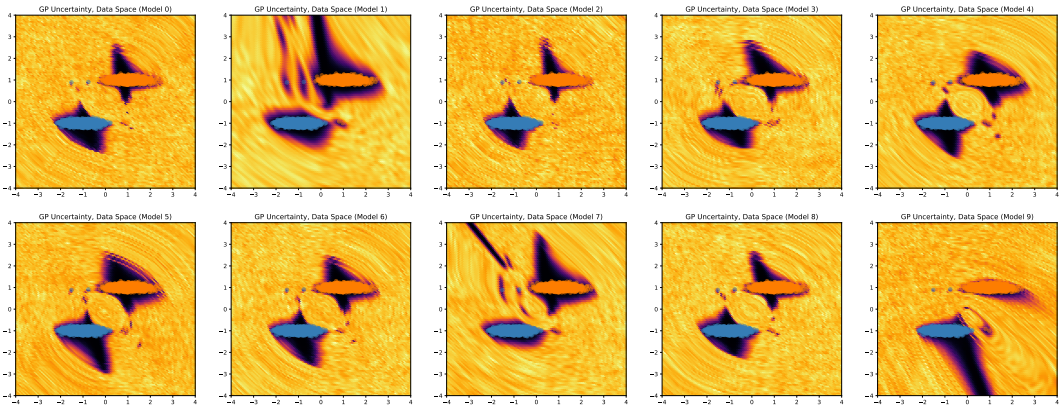

Figure 7: Uncertainty surface of individual ensemble members, introspective training.

As shown, compared to the ERM-trained model, the introspective-trained model generates similar label prediction decision $I(p(y|\mathbf{x}) > 0.5)$ (Figures 2a v.s. 2e), but with much improved uncertainty sur-

face (Figures 2b v.s. 2f). Specifically, we compute predictive variance using the standard Gaussian process variance formula $v(\mathbf{x}_{test}) = \mathbf{k}(\mathbf{x}_{test})^\top \mathbf{V}\mathbf{k}(\mathbf{x}_{test})$, where $\mathbf{k}(\mathbf{x}_{test}) = [k(\mathbf{x}_{test}, \mathbf{x}_1), \dots, k(\mathbf{x}_{test}, \mathbf{x}_n)]_{n\times 1}$ is a vector of kernel distances based on the embedding distances $||h(\mathbf{x}_{test}) - h(\mathbf{x}_i)||_2$ from the training data (Appendix A.4). As shown, the model uncertainty under ERM model are not sufficiently sensitive to directions in the data space that are irrelevant for making prediction decisions on the training data (i.e., the directions that are parallel to the decision boundary) (Figure 2f). As a result, it did not learn sufficiently diverse hidden features, leading to a significantly warped representation space that is extremely stretched out in the direction that is orthogonal to the decision boundary, and extremely compressed otherwise (Figure 2g). Consequently, the model cannot strongly distinguish the minority examples from the majority examples in the representation space, and can become overconfident even in unseen regions that was never covered by training data. This can be undesirable for uncertainty quantification under data bias, especially for the purpose of identifying underrepresented minority examples, where the distinguishing features between the minority and the majority examples are not predictive for the target label (e.g., the image background). This issue is further exacerbated in the single models (see Figure 6). In comparison, the uncertainty surface from an introspective-trained model does not suffer from this failure case. As shown in Figure 2b, the model is less inclined to become overconfident in unseen regions, especially in the neighborhood of the minority examples. Correspondingly in the representation space, the model learned more diverse features and is able to better distinguish the minority examples from the majority examples (Figure 2b). To understand how introspective training induces such improvement in model behavior, Figures (2g) and (2h) visualize the model's underrepresentation prediction $p(b|\mathbf{x})$ in the representation space and the data space, respectively. As shown, due to the need of predicting the underrepresented examples (i.e., "introspection") during training, the model has to learn hidden features that distinguishes the minority examples from the majority examples in its representation space, to a degree that they can be separated by a linear decision boundary in the last layer (Figure 2h). Consequently, the model naturally learns a more disentangled representation space through simple multi-task training, and is able to provide predicted bias probabilities $p(b|\mathbf{x})$ (Figure 2h) in addition to high-quality predictive uncertainty (Figure 2b) for the downstream active learning applications.

D.2 Tabular and Language Experiments

Datasets. We consider two challenging real-world datasets: Census Income [59] that contains 32,561 training records from the 1994 U.S. census survey. The task is to predict whether an individual's income is >50K, and the tail groups are female or non-white individuals with high income. We also consider Toxicity Detection [11] that contains 405,130 online comments from the CivilComments platform. The goal is to predict whether a given comment is toxic, and the tail groups are demographic identities $\times$ label class (male, female, White, Black, LGBTQ, Muslim, Christian, other religion) $\times$ (toxic, non-toxic) following Koh et al. [54]. More specifically, we use the U.S. Census Income data `adult` from the official UCI repository[4]. For the language task, we use the `CivilCommentsIdentity` from the TensorFlow Dataset repository[5]. For Census Income, we define the underrepresented groups as the union of (Female, High Income) and (Black, High Income); for Toxicity Detection, we define the underrepresented groups as the identity $\times$ label combination (male, female, white, black, LGBTQ, christian, muslim, other religion) $\times$ (toxic, non-toxic) (16 groups in total) as in [54]. For CivilComments, the identity annotation is a value between $(0,1)$ (it is the average rating among multiple raters), and we include an example into the underrepresented group only if the rating > 0.99 (i.e. all raters agree about the identity) following [54]. However, we do note that this leads to a under coverage of the group membership, as many comments with plausible identity mentions are not included into the group identity labels.

AL Baselines and Method Variations. For all tasks, we use a 10-member DNN ensemble $f = \{f_k\}_{k=1}^{10}$ as the AL model, and replace their last layers with a random-feature GP layer [67] in order to compute posterior variance (see Appendix A.4). We compare the impact of different training methods in two settings depending on whether the group identity label will be annotated in the labelled set (they are *never* available in the unlabelled set). When group label is available, we compare ISP-identity (i.e., ISP with group identity as training label $b_i = I(g_i \in \mathscr{B})$) to a group-

[4] https://archive.ics.uci.edu/ml/datasets/adult

[5] https://www.tensorflow.org/datasets/catalog/civil_comments

specific reweighting (RWT) baseline [46][6]. When the group label is not known, we consider **ISP-Gap** using the *self-play*-estimated generalization gap $\hat{b}_i = \mathbb{E}_k[L(\bar{f}_{in}(\mathbf{x}_i), f_{out,k}(\mathbf{x}_i))]$ as the representation label (i.e., Equation (10)), and compare it to an ensemble of Just Train Twice (JTT) which uses the ensemble training error $\hat{b}_i = \mathbb{E}_k[L(y_i, f_{in,k}(\mathbf{x}_i))]$ to determine the training set. We also compare to an ERM baseline which trains the AL models with a routine ERM objective, but uses error for the reweighted training of the final model. We consider other method combinations in the ablation study Appendix D.3).

AL Training Method	Group identity label in train set?	Training Mechanism	Underrepresentation Label b_i	Available Sampling Signal
(Random)	✓	-	Group Identity	Random
RWT [46]	✓	Reweighting	Group Identity	Margin / Diversity / Variance
ISP-Identity	✓	Introspection	Group Identity	Margin / Diversity / Variance / Predicted Underrep.
(ERM)	×	-	Train Error	Margin / Diversity / Variance
JTT [66]	×	Reweighting	Train Error	Margin / Diversity / Variance
ISP - Gap	×	Introspection	Generalization Gap	Margin / Diversity / Variance / Predicted Underrep.

Table 2: Training methods to be compared in the experiment study. Components proposed in this work are highlighted in red. The two baselines **(Random)** & **(ERM)** does not use underrepresentation label to train AL model, and only use it as reweighting signal for the reweighted training of the final model. For detailed definition of the sampling signals, see "Active Learning Signal" paragraph of Appendix D.2.

Active Learning Protocol. Figure 8 visualizes the experiment protocol. As shown, in each stage, we first (optionally) trains a cross validated ensemble to estimate the under-representation labels, where we split the data into 10 cross-validation splits, and train ensemble members on 1 split and predict the rest of the 9 splits. We then use the ensemble's in-sample and out-of-sample predictions to compute the underrepresentation label $\hat{b}_i$ (Equation (10)), and conduct introspective training (eq. (2)) to generate the final active sampling signals for 8 rounds to generate the final sampled data (red box). At the end of round 8, we estimate the underrepresentation label for the final sampled data, and send it to the final model for reweighted training to generate the full accuracy-fairness frontier. The sampling model is always a 10-member ensemble of neural Gaussian process (introduced in Appendix A.4), and the final model is always a single DNN with architecture identical to the sampling model (i.e., 2-layer Dense ResNet for census income and BERT$_{small}$ for toxicity detection).

For both tasks, we randomly sample as small subset as the initial labelled dataset (2,500 out of 32,561 total training examples for census income, and 50,000 out of total 405,130 examples for toxicity detection), and use the rest of the training set as the unlabelled set for active learning. For each sampling round, the AL model acquires 1,500 examples for census income, and 15,000 examples for the toxicity detection, so the total sample reaches roughly half of the total training set size after 8 rounds.

In the final model training, we use the standard re-weighting objective [66]:

$$\sum_{(x,y)\notin\hat{\mathscr{B}}} L_{ce}(y, f(\mathbf{x})) + \lambda \sum_{(x,y)\in\hat{\mathscr{B}}} L_{ce}(y, f(\mathbf{x}))$$

where $\hat{\mathscr{B}}$ is the set of underrepresented examples identified by the underrepresentation label, i.e., $(x_i, y_i) \in \hat{\mathscr{B}}$ if $\hat{b}_i > t$. We vary the thresholds t and the up-weight coefficient λ over a 2D grid ($t \in \{0.05, 0.1, 0.15, ..., 1.0\}$ and $log(\lambda) \in \{0., 0.5, 1, 1.5, \ldots, 10.\}$) to get a collection of model accuracy-fairness performances (i.e., accuracy v.s. worst-group accuracy), and use them to identify the Pareto frontier defined by this combination of data and reweighting signal.

Active Learning Signals. In this work, we consider four types of active sampling signals. Recall that the sampling model (neural Gaussian process ensemble) is a K-member ensemble that generates three predictive quantities: (1) label probability $\{p_k(y|\mathbf{x})\}_{k=1}^{10}$, (2) underrepresentation probability $\{p_k(b|\mathbf{x})\}_{k=1}^{10}$ and (3) predictive variance $\{v(\mathbf{x})\}_{k=1}^{10}$ (recall that $\mathbb{E}_k$ and Var_k are the empirical mean and variance).

- **Margin**: The gap between the highest class probability and the second highest class probability for the output label. In the binary prediction context, this is equivalent to $2 * |p(y|\mathbf{x}) - 0.5|$, i.e., the gap between the mean predicted probability and the null value of 0.5. We use the mean predictive label probability of the ensemble, which leads to:

$$Margin(\mathbf{x}) = 2 * |\mathbb{E}_k(p_k(y|\mathbf{x})) - 0.5|.$$

[6]Notice we choose RWT over other recently popular alternatives (e.g., Distributionally Robust Optimization (DRO)) since RWT corresponds better to JTT and outperforms DRO in difficult tasks such as CivilComments.

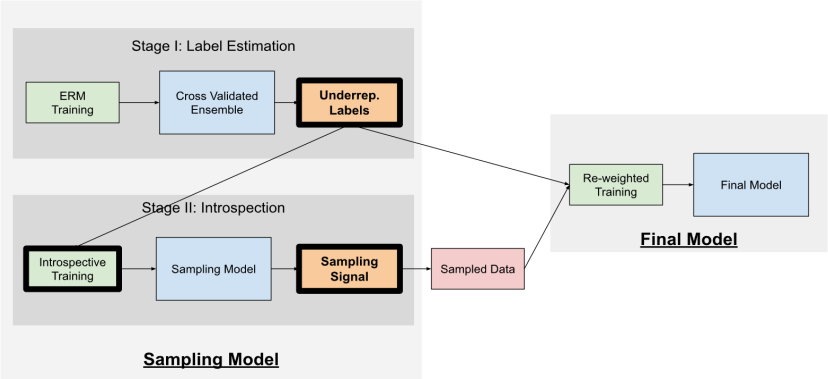

Figure 8: Experiment Protocol. Boxes with thick outlines (Underrepresentation Label, Introspective Training Method, Sampling Signal) indicates the experiment components where the methods differ.

- **Predicted Underrepresentation**: The mean predictive underrepresentation probability of the ensemble, which leads to:

$$Underrep(\mathbf{x}) = \mathbb{E}_k(p_k(b|\mathbf{x})).$$

- **Diversity**: i.e., Ensemble Diversity (introduced in Appendix A.4). The variance of label predictions:

$$Diversity(\mathbf{x}) = Var_k(p_k(y|\mathbf{x})).$$

- **Variance**: i.e., Predictive Variance (introduced in Appendix A.4). The mean of predictive variances:

$$Variance(\mathbf{x}) = \mathbb{E}_k(v_k(\mathbf{x})).$$

Model Architecture and Training Detail. For tabular experiments, we use a 2-layer Dense ResNet model with 128 hidden units and pre-activation dropout rate = 0.1, using a random-feature Gaussian process with hidden dimension 256 as the output layer [67] (In the preliminary experiments, we tried larger models with update to 6-layers and 1024 hidden units, and did not observe significant improvement). For language experiments, we used $\text{BERT}_{\text{small}}$ mode initialized from the official pre-trained checkpoint released at BERT GitHub page[107][7]. In each active learning round, we train the Dense ResNet model with Adam optimizer with learning rate 0.1, batch size 256 and maximum epoch 200; and train the BERT model with AdamW optimizer (learning rate 1e-5) for 6 epochs with batch size 16.

D.3 Further Ablation Analysis

In the main results above, we have (1) used the same underrepresentation label for both the AL-model introspective training and the final-model reweighted training, and (2) focused on the most effective active sampling signal under each task. In this section, we conduct ablations by decoupling the signal combinations along these two axes.

Impact of Data Distribution and Reweighting Signal to Accuracy-Fairness Frontier First, we investigate the joint impact of data distribution and reweighting signal on the final models' accuracy-fairness performance. We train the final model under data collected by different AL policy (Random v.s Margin v.s. Group Identity, etc), and perform reweighted training using different underrepresentation labels (Error v.s. Gap v.s. Group Identity) and compare to an ERM baseline without reweighted training (Table 3). As shown, holding the choice of reweighted signal constant and compare across data distributions (i.e., comparing across columns within each row), we observe that the data distribution in general has a non-trivial impact on the final model's accuracy-fairness performance. Specifically, under appropriate sampling signal, data collected by ISP-Gap (which has no access to true group identity label) can lead to model performance that is competitive with data collected by ISP-Identity (e.g., the third v.s. fourth columns). Comparing across reweighting signals within each dataset (i.e., compare across rows within each column), we see that all underrepresentation labels brings a meaningful improvement over the ERM baseline, with Group Identity bringing the most

[7] https://github.com/google-research/bert

significant improvement when it is of high quality (i.e., Census Income), and Gap bringing the most improvement when group annotation is imperfect (i.e., Toxicity Detection).

Table 3: Impact to final-model fairness-accuracy performance (measured by combined accuracy = acc + worst-group acc)/2) of the choice of reweighting signal (rows), across dataset collected by different active learning methods (columns). **Random**: data collected via random sampling. **Margin/Variance/Diversity**: data collected using introspective-trained AL model (with Gap as underrepresentation label) using the said sampling signal. **Group Identity**: data collected by introspective-trained AL model with group identity as introspection signal, using the best sampling signal for the task (Variance for census income and Margin for toxicity detection).

Final Model Reweighting Signal	AL Method, Census Income				AL Method, Toxicity Detection			
	Random	Margin	Variance	Group Identity	Random	Diversity	Margin	Group Identity
(ERM)	0.692	0.669	0.719	0.720	0.698	0.699	0.702	0.703
Error	0.706	0.683	0.750	0.743	0.758	0.761	0.744	0.752
Gap	0.692	0.694	0.770	0.777	**0.776**	**0.776**	**0.758**	**0.810**
Group Identity	**0.746**	**0.756**	**0.778**	**0.785**	0.711	0.701	0.705	0.713

Impact of Underrepresentation Label on Different Sampling Signals. Finally, we evaluate the choice of introspection signal on the sampling performance of a introspective-trained AL-model, under different types of sampling signals (Table 4). As this evaluation is computationally expensive (requiring multiple active learning experiments for all underrepresentation label v.s. sampling signal combinations), here we focus on the Census Income task. As shown, we observe the introspective training brings a consistent performance boost across different types of sampling signals (esp. when using Group Identity), highlighting the appeal of introspective training as a “plug-in" method that meaningfully boost the performance of a wide range of active learning methods. Interestingly, we also observe the “Predicted Underrep." (i.e., the underrepresentation prediction in $p(b|\mathbf{x})$ in Figure 3) is exceptionally effective when the group identity is available (tail sampling rate > 0.95) but underperforms classic active learning signals otherwise, indicating that the proper use of this signal depending on the availability of group labels.

Table 4: Impact to AL performance (measured by tail sampling rate) of the choice of introspection signal (rows) across different active learning methods (columns).

Underrep. Label	AL Method, Census Income			
	Margin	Diversity	Variance	Predicted Underrep.
Error	0.780	0.324	0.771	0.671
Gap	0.803	0.276	0.839	0.708
Group Identity	0.873	0.330	0.907	0.967

E Noise-bias-variance decomposition in Ridge Regression

Consider fitting a ridge regression model $f(\mathbf{x}_i) = \beta^\top \mathbf{x}_i$ to the Gaussian observation data $y_i = \theta^\top \mathbf{x}_i + \varepsilon, \varepsilon \sim N(0, \sigma^2)$ under an imbalanced experiment design, where we have $|\mathscr{G}|$ treatment groups and n_g observations in each group. . Here, $\mathbf{x}_i = [1_{g_i=1}, ..., 1_{g_i=|\mathscr{G}|}]$ is a $|\mathscr{G}| \times 1$ one-hot indicator of the membership of g_i for each group in $\mathscr{G}$, and $\theta = [\theta_1, \ldots, \theta_{|\mathscr{G}|}]$ is the true effect for each group. Then, under ridge regression, the noise-bias-variance decomposition for group g is $E_D(L(y, f_D)) = \sigma^2 + \frac{(\lambda\theta_g)^2}{(n_g+\lambda)^2} + \frac{\sigma^2 n_g}{(n_g+\lambda)^2}$, where the regularization parameter λ modulates a trade-off between the *bias* and *variance* terms. In Appendix E.3, we also treat the case of group-specific noise $\varepsilon_i \overset{indep}{\sim} N(0, \sigma_g^2)$.

E.1 Error Decomposition in a General Setting

We first derive the decomposition in a general setting with data $\{y_i, \phi_i\}_{i=1}^n$, where ϕ_i is the $d \times 1$ (fixed) features that follows a distribution $P(\phi)$. We consider a well-specified scenario where the data generation mechanism as:

$$y_i = \tilde{y}_i + \varepsilon, \quad \text{where} \quad \tilde{y}_i = \theta^\top \phi_i, \;\; \varepsilon \overset{iid}{\sim} N(0, \sigma^2),$$

and $\theta_{d\times 1} = [\theta_1, \ldots, \theta_{|\mathscr{G}|}]$ is the true coefficient. Under ridge regression, we fit a linear model $f(\mathbf{x}_i) = \phi_i^\top \beta$ to the data by minimizing the following squared loss objective:

$$||\mathbf{y}_{n\times 1} - \Phi_{n\times d}\beta_{d\times 1}||_2^2 + \lambda ||\beta||_2^2,$$

which gives rise to the following solution:

$$\hat{\beta} = (\Phi^\top \Phi + \lambda I_d)^{-1} \Phi^\top \mathbf{y}. \tag{11}$$

notice $\hat{\beta}$ is a random variable that depends on the data $\Phi_{n\times d} = [\phi_1^\top, \ldots, \phi_n^\top] \overset{iid}{\sim} P(\phi)$. Notice that under squared loss, the ensemble predictors $\bar{f} = argmin_f E_\Phi[(f - \hat{\beta}^\top \mathbf{x}_i)^2]$ is simply the mean of individual predictors, i.e, $\bar{f} = E_\Phi(\hat{\beta}^\top \mathbf{x}_i) = \bar{\beta}^\top \mathbf{x}_i$, where $\bar{\beta} = E_\Phi(\hat{\beta})$.

Consequently, given a new observation $\{y, \phi\}$, the noise-bias-variance decomposition of $\hat{\beta}$ under squared loss is:

$$\begin{aligned} E[(y - \Phi\hat{\beta})^2] &= E_y[(y - \tilde{y})^2] + (\tilde{y} - \bar{\beta}^\top \phi_i)^2 + E_\Phi[\bar{\beta}^\top \phi_i - \hat{\beta}^\top \phi_i]^2 \\ &= \underbrace{\sigma^2}_{Noise} + \underbrace{\phi_i^\top [\theta - \bar{\beta}][\theta - \bar{\beta}]^\top \phi_i}_{Bias} + \underbrace{\phi_i^\top Var(\hat{\beta}) \phi_i}_{variance}. \end{aligned} \tag{12}$$

As shown, to obtain a closed-form expression of the decomposition, we need to first derive the expressions of $Bias(\hat{\beta}) = [\theta - \bar{\beta}]$ and $Var(\hat{\beta})$. Under the expression of the ridge predictor (11), we have:

$$\begin{aligned} Bias(\hat{\beta}) &= [\theta - \bar{\beta}] \\ &= \theta - E[(\Phi^\top \Phi + \lambda I_d)^{-1} \Phi^\top \Phi \theta] \\ &= E[I - (\Phi^\top \Phi + \lambda I_d)^{-1} \Phi^\top \Phi]\theta \\ &= \lambda * E[(\Phi^\top \Phi + \lambda I_d)^{-1}]\theta; \\ Var(\hat{\beta}) &= E(Var(\hat{\beta}|\Phi)) + Var(E(\hat{\beta}|\Phi)), \end{aligned}$$

with

$$\begin{aligned} E(Var(\hat{\beta}|\Phi)) &= E[(\Phi^\top \Phi + \lambda I_d)^{-1} \Phi^\top Var(\mathbf{y}) \Phi (\Phi^\top \Phi + \lambda I_d)^{-1}] \\ &= \sigma^2 E[(\Phi^\top \Phi + \lambda I_d)^{-1} \Phi^\top \Phi (\Phi^\top \Phi + \lambda I_d)^{-1}] \\ &= \sigma^2 * E[(\Phi^\top \Phi + \lambda I_d)^{-1} - \lambda (\Phi^\top \Phi + \lambda I_d)^{-2}]. \\ Var(E(\hat{\beta}|\Phi)) &= E[S\theta\theta^\top S^\top] - E[S]\theta\theta^\top E[S^\top] \end{aligned}$$

where $S = (\Phi^\top \Phi + \lambda I_d)^{-1} \Phi^\top \Phi$.

As shown, the above expression depends on the random-matrix moments $E[(\Phi^\top\Phi+\lambda I_d)^{-1}]$, $E[(\Phi^\top\Phi+\lambda I_d)^{-2}]$, $E[(\Phi^\top\Phi+\lambda I_d)^{-1}\Phi^\top\Phi]$ and $E[S\theta\theta^\top S^\top]$.

E.2 Error Decomposition under Orthogonal Design

The above moments are in general difficult to solve due to the involvement of matrix inverse and product within the expectation. However, a closed-form expression is possible under an orthogonal design where $\phi_i = [1_{g_i=1},\ldots,1_{g_i=|\mathscr{G}|}]$ is the one-hot vector of treatment group memberships. Then, denote $Diag[z_g]$ the diagonal matrix with diagonal elements z_g and $[z_{gg'}]_{gg'}$ the full matrix whose (g,g') element is $z_{gg'}$, we have:

$$\begin{aligned}
\Phi^\top\Phi &= diag[n_g] \\
E[(\Phi^\top\Phi+\lambda I_d)^{-1}] &= diag[\frac{1}{n_g+\lambda}], \\
E[(\Phi^\top\Phi+\lambda I_d)^{-2}] &= diag[\frac{1}{(n_g+\lambda)^2}], \\
E[(\Phi^\top\Phi+\lambda I_d)^{-1}\Phi^\top\Phi] &= diag[\frac{n_g}{n_g+\lambda}],
\end{aligned}$$

and

$$E[(\Phi^\top\Phi+\lambda I_d)^{-1}\Phi^\top\Phi\theta\theta^\top\Phi^\top\Phi(\Phi^\top\Phi+\lambda I_d)^{-1}] = [\frac{n_g n_g'}{(n_g+\lambda)(n_g'+\lambda)}\theta_g\theta_{g'}]_{gg'}.$$

We are now ready to derive the full decomposition (13), without loss of generality, we assume ϕ_i belongs to group g. Then:

$$\begin{aligned}
\phi_i^\top Bias(\hat\beta) &= \lambda * \phi_i^\top E[(\Phi^\top\Phi+\lambda I_d)^{-1}]\theta = \frac{\lambda}{n_g+\lambda}\theta_g; \\
\phi_i^\top Var(\hat\beta)\phi_i &= \sigma^2 * \phi_i^\top E[(\Phi^\top\Phi+\lambda I_d)^{-1} - \lambda(\Phi^\top\Phi+\lambda I_d)^{-2}]\phi_i; \\
&= \frac{\sigma^2}{n_g+\lambda} - \frac{\lambda\sigma^2}{(n_g+\lambda)^2} = \frac{\sigma^2 n_g}{(n_g+\lambda)^2}.
\end{aligned}$$

Consequently, we have the noise-bias-variance decomposition in (13) as:

$$\begin{aligned}
\textit{Noise}: &\quad \sigma^2; \\
\textit{Bias}: &\quad ||\phi_i^\top Bias(\hat\beta)||_2^2 = \frac{(\lambda\theta_g)^2}{(n_g+\lambda)^2}; \\
\textit{Uncertainty}: &\quad \phi_i^\top Var(\hat\beta)\phi_i \quad = \frac{\sigma^2 n_g}{(n_g+\lambda)^2}.
\end{aligned}$$

E.3 Error Decomposition under Orthogonal Design and Heterogeneous Noise

We now consider the case where $y_i \sim N(\theta^\top\phi_i, \sigma_g^2)$ follows a normal distribution with group-specific noise. Using the same decomposition as in E.1, we see that:

$$\begin{aligned}
E[(y-\Phi\hat\beta)^2] &= E_y[(y-\tilde y)^2] + (\tilde y - \bar\beta^\top\phi_i)^2 + E_\Phi[\bar\beta^\top\phi_i - \hat\beta^\top\phi_i]^2 \\
&= \underbrace{\sigma_g^2}_{Noise} + \underbrace{\phi_i^\top[\theta-\bar\beta][\theta-\bar\beta]^\top\phi_i}_{Bias} + \underbrace{\phi_i^\top Var(\hat\beta)\phi_i}_{variance}.
\end{aligned} \tag{13}$$

As shown, the nature of the bias and variance decomposition in fact does not change, and the noise component is now the group-specific variance σ_g^2. Therefore, by following the same derivation as in

Appendix E.2, we have:

$$
\begin{aligned}
\textit{Noise}: \quad & \sigma_g^2; \\
\textit{Bias}: \quad & ||\phi_i^\top Bias(\hat{\beta})||_2^2 = \frac{(\lambda\,\theta_g)^2}{(n_g+\lambda)^2}; \\
\textit{Uncertainty}: \quad & \phi_i^\top Var(\hat{\beta})\phi_i \quad = \frac{\sigma^2 n_g}{(n_g+\lambda)^2}.
\end{aligned}
$$

F Proof of Proposition B.1

Through introspective training, there is a guarantee on a model's bias-awareness based on its hidden representation and uncertainty estimates. At convergence, a well-trained model $f = (f_y, f_b)$ should satisfy the property that $p(b=1|x) = \sigma(f_b(\mathbf{x}))$.

(I) ***(Bias-aware Hidden Representation)*** We denote the odds for $\mathbf{x}$ belonging to the underrepresented group $\mathscr{B}$ as $o_b(\mathbf{x}) = p(\mathbf{x}|b=1)/p(\mathbf{x}|b=0)$. Using Bayes' theorem, we derive the following:

$$
\begin{aligned}
p(b|\mathbf{x}) &= \sigma(\beta^T h(\mathbf{x}) + \beta_0) \\
log\,\frac{p(b=1|\mathbf{x})}{p(b=0|\mathbf{x})} &= \beta^T h(\mathbf{x}) + \beta_0 \\
log\,\frac{p(\mathbf{x}|b=1)p(b=1)}{p(\mathbf{x}|b=0)p(b=0)} &= \beta^T h(\mathbf{x}) + \beta_0 \\
\beta^T h(\mathbf{x}) + \beta_0 &= log\,P(\mathbf{x}|b=1) - logP(\mathbf{x}|b=0) + log\frac{p(b=1)}{p(b=0)} \\
\beta^T h(\mathbf{x}) + \beta_0 &= log\,o_b(\mathbf{x}) + log\frac{p(b=1)}{p(b=0)} \qquad (14)
\end{aligned}
$$

Hence, the hidden representation is aware of the likelihood ratio of whether an example $\mathbf{x}$ belongs to the underrepresented group, and the last-layer bias β_0 corresponds to the marginal likelihood ratio of the prevalence of the underrepresented groups $p(b=1)/p(b=0)$.

(II) ***(Bias-aware Embedding Distance)*** Next, we examine the embedding distance between two examples $(\mathbf{x}_1, \mathbf{x}_2)$, i.e., $||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2$.

The Cauchy-Schwarz inequality states that for two vectors $\mathbf{u}$ and $\mathbf{v}$ of the Euclidean space, $|\langle \mathbf{u}, \mathbf{v}\rangle| \leq ||\mathbf{u}||\,||\mathbf{v}||$. Hence, the distance between two embeddings can be expressed as $\beta^T[h(\mathbf{x}_1) - h(\mathbf{x}_2)] \leq ||\beta||_2\,||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2$. Using this property and Equation (14), we derive the following:

$$
\begin{aligned}
\beta^T h(\mathbf{x}_1) - \beta^T h(\mathbf{x}_2) &= log\,o_b(\mathbf{x}_1) - log\,o_b(\mathbf{x}_2) \\
\beta^T[h(\mathbf{x}_1) - h(\mathbf{x}_2)] &= log\,o_b(\mathbf{x}_1) - log\,o_b(\mathbf{x}_2) \\
log\,o_b(\mathbf{x}_1) - log\,o_b(\mathbf{x}_2) &\leq ||\beta||_2\,||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2 \\
\frac{1}{||\beta||_2}[log\,o_b(\mathbf{x}_1) - log\,o_b(\mathbf{x}_2)] &\leq ||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2 \\
\frac{1}{||\beta||_2} log\frac{o_b(\mathbf{x}_1)}{o_b(\mathbf{x}_2)} &\leq ||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2 \qquad (15)
\end{aligned}
$$

Since the above inequality is invariant to the relative position of $(\mathbf{x}_1, \mathbf{x}_2)$, we also have: $\frac{1}{||\beta||_2} log\frac{o_b(\mathbf{x}_2)}{o_b(\mathbf{x}_1)} \leq ||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2$, which implies:

$$
||h(\mathbf{x}_1) - h(\mathbf{x}_2)||_2 \geq \frac{1}{||\beta||_2} * max(log\frac{o_b(\mathbf{x}_1)}{o_b(\mathbf{x}_2)}, log\frac{o_b(\mathbf{x}_2)}{o_b(\mathbf{x}_1)}).
$$

As shown, the distance between the hidden embeddings $h(\mathbf{x}_1)$, $h(\mathbf{x}_2)$ is lower-bounded by the log-odds ratio that a given example is in the underrepresented group. With this guarantee on the model's

learned embedding distance, we expect the hidden features to be more diverse than when trained on the main task alone, since it needs to sufficient features to distinguish the underrepresented-group examples from those of the majority in the hidden space.

G Related Work

G.1 Supervised and semi-supervised learning under dataset bias

In recent years, there has been significant interest in studying robust generalization for long-tail population subgroups under dataset bias. The literature is vast and encompasses topics including fairness, debiasing, long-tail recognition, spurious correlation, distributional (i.e., domain or subpopulation) shift, etc. In the following, we focus on notable and recent work that is highly relevant to the ISP approach, and refer to works such as Caton & Haas [18], Mehrabi et al. [75], Hort et al. [43] for an exhaustive survey.

Majority of the fairness and debiasing work focuses on the supervised learning setting, where the model only have access to a fixed and imbalanced dataset. Among them, the earlier work operated under the assumption that the source of dataset bias is completely known, and the group annotation is available for every training example. Then these group information is use to train a robust model by modify components of the training pipeline (e.g., training objective, regularization method, or composition of training data). For example, Levy et al. [62], Sagawa et al. [89], Zhang et al. [122] proposes minimizing the worst-group loss via DRO; Teney et al. [102], Idrissi et al. [46], Byrd & Lipton [13], Xu et al. [119] studies the effect of group-weighted loss in model's fairness-accuracy performance, and REx [56] minimizes a combination of group-balanced and worst-case loss. Further, the recent literature has also seen sophisticated neural-network loss that modifies gradient for the tail-group examples. For example, LDAM [16] proposes to modify group-specific logits by an offset factor that is associated with group size, and equalization loss [100] uses a instance-specific mask to suppress the "discouraging gradients" from majority groups to the rare groups. On the regularization front, the examples include *Invariant Risk Minimization* (IRM) [7] that appends a group-balanced loss with a gradient norm penalty. *Heteroskedastic Adaptive Regularization* (HAR) [17] imposes Lipschitz regularization in the neighborhood of tail-group examples. There also exists a large collection of work imposing other types of fairness constraints. Finally, the third class of methods modifies the composition of the training data by enriching the number of obsevations in the tail groups, this includes Sagawa et al. [90], Idrissi et al. [46] that study the impact of resampling to the worst-group performance, and Goel et al. [36] that generates synthetic examples for the minority groups. In the setting where the group information is available, our work proposes a novel approach (introspective training) that has both a theoretical guarantee and is empirically competitive than reweighted training.

On the other hand, there exist a separate stream of work that allows for partial group annotation, i.e., the types of bias underlying a dataset is still completely known, but the group annotation is only available for a subset of the data. Most work along this direction employs semi-supervised learning techniques (e.g, confidence-threshold-based pseudo labeling), with examples include Spread Spurious Attribute (SSA) [78], BARACK [96] and Fair-PG [48]. This setting can be considered as a special case of ISP where we use group information as the underrepresentation label to train the $p(b|\mathbf{x})$ predictor. However, our goal is distinct that we study the efficacy of this signal as an active learning policy, and also investigate its extension in the case where the label information is completely unobserved in the experiments Appendix D.3.

G.2 Estimating dataset bias for model debiasing

In the situation where the source of dataset bias is not known and the group annotation is unavailable, several techniques has been proposed to estimate proxy bias labels for the downstream debiasing procedures. These methods roughly fall into three camps: clustering, adversarial search, and using the generalization error from a biased model.

For clustering, GEORGE [95] and CNC [123] proposed estimating group memberships of examples based on clustering the last hidden-layer output. For adversarial search, REPAIR [64], ARL [57], EIIL [26], BPF [73], FAIR [82], Prepend [103] estimate the likelihood for an example to be biased using an adversarial weighting model, which is trained by maximizing certain learning risk.

Estimating bias label using the error from a biased model is by far the most popular technique. These include *forgettable examples* [120], *Product of Experts* (PoE) [24, 91], DRiFt [42] and *Confidence*

Regularization (CR) [109, 108] that uses errors from a separate class of weak models that is different from the main model; *Neutralization for Fairness* (RNF) [29] and *Learning from Failure* (LfF) that trains a bias-amplified model of the same architecture using generalized cross entropy (GCE); and *Just Train Twice (JTT)* that directly uses the error from a standard model trained from cross entropy loss.

Notably, there also exists several work that estimates bias label using ensemble techniques, this includes *Gradient Alignment* (GA) [124] that identifies the tail-group (i.e., bias-conflicting) examples based on the agreement between two sets of epoch ensembles, *Bias-conflicting Detection* (BCD) [61] that uses the testing error of a biased deep ensemble trained with GCE, and *Learning with Biased Committee* (LWBC) uses the testing error of a bootstrap ensemble.

To this end, our work proposes a novel *self-play estimator* (Equation (10)) that uses bootstrap ensembles to estimate the *generalization gap* due to dataset bias. *self-play estimator* has the appealing property of better controlling for label noise while more stably estimating model variance, addressing two weaknesses of the naive predictive error estimator used in the previous works.

G.3 Representation learning under dataset bias

Situated in the fairness literature, the earlier work in debiased representation learning has focused on techniques to eliminate the information of spurious features (e.g., protected attributes) from the model representation. This include adversarial training [10, 50, 85, 127], regularization [9, 101], contrastive learning: [94, 81, 21] and its conditional variants [37, 105, 106, 23]. However, some later works questions the necessity and the sufficiency of such approaches. For example, some work shows that careful training of the output head along is sufficient to yield improved performance in fairness and bias mitigation [49, 29, 52], and Cherepanova et al. [22] shows that models with fair feature representations do not necessarily yield fair model behavior.

At the meantime, a separate stream of work explores the opposite direction of encouraging the model to learn diverse hidden features. For example, Locatello et al. [70, 69] establish a connection between the notion of feature disentanglement and fairness criteria, showing that feature disentanglement techniques may be a useful property to encourage model fairness when sensitive variables are not observed. However, such techniques often involves specialized models (e.g., VAE) which restricts the broad applicability of such approaches. Some other work explores feature augmentation techniques to learn both invariant and spurious attributes, and use them to debias the output head [60]. Finally, a promising line of research has been focusing on using self-supervised learning to help the model avoid using spurious features in model predictions [20, 118, 14, 39]. Our work follows this latter line of work by proposing novel techniques to encourage model to learn diverse features that is *bias-aware*, but with a distinct purpose of better uncertainty quantification.

G.4 Active learning under dataset bias

In recent years, the role of training data in ensuring the model's fairness and bias-mitigation performance has been increasing noticed. Notably, [19] presented some of the earlier theoretical and empirical evidence that increasing training set size along is already effective in mitigating model unfairness. Correspondingly, under the assumption that the *group information in the unlabelled set is fully known*, there has been several works that studies group-based sampling strategies and their impact on model behavior. For example, Rai et al. [86], Wang et al. [112] shows group-based active sampling stratgy improves model performance under domain and distributional shifts, and Abernethy et al. [1] proves a guarantee for a worst-group active sampling strategy's ability in helping the SGD-trained model to convergence to a solution that attains min-max fairness. A second line of research focuses on designing better active learning objectives that incorporates fairness constraints, e.g., *Fair Active Learning (FAL)* [6] and *PANDA* [93]. Agarwal et al. [4] introduce a data repair algorithm using the coefficient of variation to curate fair and contextually balanced data for a protected class(es). Finally, there exists few active learning works formulating the objective of their method as optimizing a fairness-aware objective. For example, *Slice Tuner* Tae & Whang [99] proposes adaptive sampling strategy based on per-group learning curve to minimize fairness tradeoff, performs numeric optimization. Cai et al. [15] which formalized the fairness learning problem as an min-max optimization objective, however their did not conduct further theoretical analysis of their objective, but instead proposed a per-group sampling algorithm based predicted model error using linear regression. In comparison, our proposed method (ISP) does not require group information from the unlabelled set.

On the other hand, there exists active re-sampling methods that do not require the knowledge of group information in the unlabelled set. For example, Amini et al. [5] learns the data distribution using a VAE model under additional supervision of class / attribute labels, and then perform IPW sampling with respect to learned model. REPAIR [64] that estimates dataset bias using prediction error of a weak model, and then re-train model via e.g., sample re-weighting based on the estimated bias. The bias estimation method used in this work is analogous to that of the JTT, which we compare with in our work. A work close to our direction is Branchaud-Charron et al. [12], which shows DNN uncertainty (i.e., BatchBALD with Monte Carlo Dropout [53]) helps the model to achieve fairness objectives in active learning on a synthetic vision problem. Our empirical result confirms the finding of Branchaud-Charron et al. [12] on realistic datasets, and we further propose techniques to improve the vanilla DNN uncertainty estimators for more effective active learning under dataset bias.

As an aside, a recent work Farquhar et al. [32] studies the statistical bias in the estimation of active learning objectives due to the non-i.i.d. nature of active sampling. This is separate from the issue of dataset bias (i.e., imbalance in data group distribution) which we focus on in this work.

G.5 Uncertainty estimation with DNNs

In recent years, the probabilistic machine learning (ML) literature has seen a plethora of work that study enabling calibrated predictive uncertainty in DNNss. Given a model f, the probabilistic DNN model aims to learn a predictive distribution for the model function f, such that given training data $D = \{(y_i, \mathbf{x}_i)\}_{i=1}^n$ and a testing point $\mathbf{x}_{test}$, the model outputs a predictive distribution $f(\mathbf{x}_{test}) \sim P(f|\mathbf{x}_{test}, D)$ rather than a simple point prediction. To this end, the key challenge is to learn a predictive distribution (implicitly or explicitly) during the SGD-based training process of DNN, generating calibrated predictive uncertainty without significantly impacting the accuracy or latency when compared to a deterministic DNN.

To this end, the classic works focus on the study of Bayesian neural networks (BNNs) [79], which took a full Bayesian approach by *explicitly* placing priors to the hidden weights of the neural network, and performance MCMC or variance inference during learning. Although theoretically sound, BNN are delicate to apply in practice, with its performance highly dependent on prior choice and inference algorithm, and are observed to lead to suboptimal predictive accuracy or even poor uncertainty performance (e.g., under distributional distribution shift) [113, 47]. Although there exists ongoing works that actively advancing the BNN practice (e.g., [30]). On the other hand, some recent work studies computationally more approaches that *implicitly* learn a predictive distribution as part of deterministic SGD training. Notable examples include Monte Carlo Dropout [34] which generates predictive distribution by enabling the random Dropout mask during inference, and ensemble approaches such as Deep Ensemble [58] and their later variants [71, 114, 41] that trains multiple randomly-initialized networks to learn the modes of the posterior distribution of the neural network weights [116]. Although generally regarded as the state-of-the-art in deep uncertainty quantification, these methods are still computationally expensive, requiring multiple DNN forward passes at the inference time.

At the meantime, a more recent line of research avoids probablistic inference for the hidden weights altogether, focusing on learning a scalable probabilistic model (e.g., Gaussian process) to replace the last dense layer of the neural network [110, 111, 67, 25]. A key important observation in this line of work is the role of hidden representation quality in a model's ability in obtaining high-quality predictive uncertainty. In particular, Liu et al. [67], Van Amersfoort et al. [110] suggests that this failure mode in DNN uncertainty can be caused by an issue in representation learning known as *feature collapse*, where the DNN over-focuses on correlational features that help to distinguish between output classes on the training data, but ignore the non-predictive but semantically meaningful input features that are important for uncertainty quantification. [77] also observed that DNN exhibits particular modes of failure in out-of-domain (OOD) detection in the presence of dataset bias. Later, Tran et al. [104], Minderer et al. [76] suggests that this issue can be partially mitigated by large-scale pre-traininig with large DNNs, where larger pre-trained DNN's tend to exhibit stronger uncertainty performance even under spurious correlation and subpopulational shift. In this work, we confirm this observation in the setting of dataset bias in Figure 2), and propose simple procedures to mitigate this failure mode in representation learning without needing any change to the DNN model, and illustrates improvement even on top of large-scale pre-trained DNNs (BERT).

Deep uncertainty methods in active learning. Active learning with DNNs is an active field with numerous theoretical and applied works, we refer to [74, 87] for comprehensive survey, and only

mention here few notable methods that involves DNN uncertainty estimation techniques. Under a classification model, the most classic approach to uncertainty-based active learning is to use the predictive distribution's entropy, confidence or margin as the acquisition policy [92]. Notice that in the binary classification setting, these three acquisition policy are rank-equivalent since they are monotonic to the distance between $max[p(\mathbf{x}), 1 - p(\mathbf{x})]$ and the null probability value of 0.5. On the other hand, *Batch Active learning by Diverse Gradient Embeddings (BADGE)* [8] proposes to blend diversity-based acquisition policy into uncertainty-based active learning, by applying k-means++ algorithm to the gradient embedding of the class-specific logits (which quantifies uncertainty). As a result, *BADGE* may also suffer from the pathogloy in model representation under dataset bias, which this work is attempt to address.

Finally, [44] has proposed a information-theoretic policy Bayesian active learning by disagreement (BALD), which measures the mutual information between data points and model parameters and is adopted in the deep uncertainty literature [35, 53, 55]. However, stable estimation of mutual information can be delicate in practice, and we leave the investigation of these advanced acquisition policy under dataset bias for future work.