What You See is What You Get: Principled Deep Learning via Distributional Generalization

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Having similar behavior at train-time and test-time—what we call a "What You See Is What You Get (WYSIWYG)" property—is desirable in machine learning. However, models trained with standard stochastic gradient descent (SGD) are known to not capture it. Their behaviors such as subgroup performance, or adversarial robustness can be very different during training and testing. We show that Differentially-Private (DP) training provably ensures the high-level WYSIWYG property, which we quantify using a notion of Distributional Generalization (DG). Applying this connection, we introduce new conceptual tools for designing deep-learning methods by reducing generalization concerns to optimization ones: to mitigate unwanted behavior at test time, it is provably sufficient to mitigate this behavior on the train datasets. By applying this novel design principle, which bypasses "pathologies" of SGD, we construct simple algorithms that are competitive with SOTA in several distributional robustness applications, significantly improve the privacy vs. disparate impact tradeoff of DP-SGD, and mitigate robust overfitting in adversarial training. Finally, we also improve on known theoretical bounds relating DP, stability, and distributional generalization.

1 *What You See is What You Get* Generalization: What, Why, and How?

Much of machine learning (ML), both in theory and in practice, operates under two assumptions. First, we have independent and identically distributed (i.i.d.) samples. Second, we care only about a single averaged scalar metric (error, loss, risk). Under these assumptions, we have mature methods and theory: Modern learning methods excel when trained on i.i.d. data to directly optimize a scalar loss, and there are many theoretical for reasoning about *generalization* which explain when does optimization of a scalar on the train dataset translates to similar values of this scalar at test time.

The focus on scalar metrics such as average error, however, misses many theoretically, practically, and socially relevant aspects of model performance. For example, models with small *average* error often have high error on salient minority subgroups [1, 2]. In general, ML models are applied to the heterogeneous and long-tailed data distributions of the real world [3]. Attempting to summarize their complex behavior with only a single scalar misses many rich and important aspects of learning.

These issues are compounded for modern overparameterized networks, as their nuanced test-time behavior is not reflected at train time. This presents an obstacle for algorithm design, because interventions which alter a network's properties on its training data do not always transfer to the test time. For example, consider the setting of *importance sampling*: suppose we know that a certain subgroup of inputs is underrepresented in the training data compared to the test distribution

Submitted to the ML Safety Workshop at the 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute

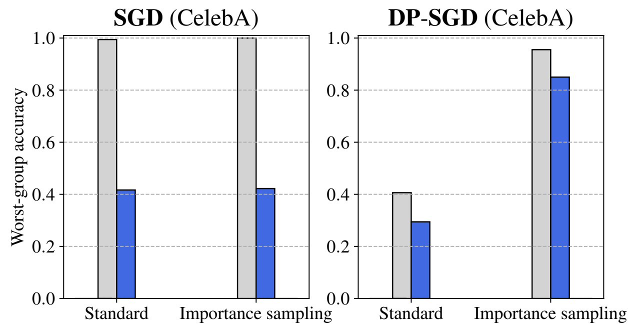


Figure 1: **Differential privacy ensures the desired behavior of importance sampling on test data.** The train and test accuracy of ResNets on CelebA, evaluated on the worst-performing ("male, blond") subgroup. *Left:* Standard SGD has a large generalization gap on this subgroup, and Importance Sampling (IS) has little effect. *Right:* DP-SGD provably has small generalization gap on all subgroups, and IS improves subgroup performance as intended. See App. B for details.

(breaking the i.i.d. assumption). For underparameterized models, we can simply upsample this underrepresented group to account for the distribution shift [see, e.g., 4]. This approach, however, is known to empirically fail for overparameterized models [5]. Because "what you see" (on the training data) is not "what you get" (at test time), we cannot make principled train-time interventions to affect test-time behaviors. This issue extends beyond importance sampling. For example, theoretically principled methods for distributionally robust optimization (e.g. Namkoong and Duchi [6]) fail for overparameterized deep networks, and require ad-hoc modifications [7].

We develop a theoretical framework which (1) sheds light on these existing issues, and (2) leads to improved practical methods in privacy, fairness, and distributional robustness. The core object in our framework is what we call the "What You See Is What You Get" (WYSIWYG) property. A training procedure with the WYSIWYG property does *not* exhibit the "pathologies" of standard stochastic gradient descent (SGD): all test-time behaviors will be expressed on the training data as well, and there will be "no surprises" in generalization.

What You See Is What You Get (WYSIWYG) as a Design Principle. The WYSIWYG property is desirable for two reasons. The first is diagnostic: as there are "no surprises" at test time, all properties of a model at test time are already evident on the training data. It cannot be the case, for example, that a WYSIWYG model has small disparate impact on the training data, but large disparate impact at test time. The second reason is algorithmic: to mitigate *any* unwanted test-time behavior, it is sufficient to mitigate this behavior on the training data. This means that algorithm designers can be concerned only with achieving desirable behavior at train time, as the WYSIWYG property guarantees it holds at test time too. In practice, this enables the usage of many theoretically principled algorithms which were developed in the underparameterized regime to also apply in the modern overparameterized (deep learning) setting. For example, we find that interventions such as importance sampling, or algorithms for distributionally robust optimization, which fail without additional regularization, work exactly as intended with WYSIWYG (See Fig. 1 for an illustration).

Formalizing WYSIWYG using Distributional Generalization. As WYSIWYG is a high-level conceptual property, we have to formalize it to use in practice. We do so using the notion of *Distributional Generalization* (DG), as introduced by Nakkiran and Bansal [8], Kulynych et al. [9]. A training algorithm *generalizes in expectation* in the classical sense if the values of loss on the training dataset and at test time are close on average [10]:

$$|\mathop{\mathbb{E}}_{\theta,S,z\sim S}\ell(z;\theta_S) - \mathop{\mathbb{E}}_{\theta,S,z\sim\mathcal{D}}\ell(z;\theta_S)| \leq \delta, \tag{1}$$

where θ_S is the parameter vector of the model obtained by training on the dataset $S \sim \mathcal{D}^n$, i.i.d. sampled from the data distribution $\mathcal{D}$. Distributional generalization is an extension of this standard concept that considers not only loss, but any other bounded test function $\phi(z;\theta) \in [0,1]$. Specifically, by saying that a model *distributionally* generalizes we mean that for *all* such test functions ϕ, their values in training and test are close on average:

$$\forall \phi : \quad |\mathop{\mathbb{E}}_{\theta,S,z\sim S} \phi(z;\theta_S) - \mathop{\mathbb{E}}_{\theta,S,z\sim \mathcal{D}} \phi(z;\theta_S)| \leq \delta. \tag{2}$$

This fact captures the high-level idea of the *"What You See is What You Get"* (WYSIWYG) guarantee for a large class of useful behaviors of machine learning models. Some example behaviors are:

- *Subgroup accuracy:* $\phi(z;\theta) = \mathbb{1}\{z \in G\} \cdot \ell(z;\theta)$, for some subgroup $G \subset \mathbb{D}$.
- *Robustness to corruptions:* $\phi(z;\theta) = \ell(A(z);\theta)$, where $A(x)$ is a possibly randomized transformation that distorts the example, e.g., by adding Gaussian noise.
- *Adversarial robustness:* $\phi(z;\theta) = \ell(A_\theta(z);\theta)$, where $A_\theta(z)$ is an adversarial example, e.g. generated using the PGD attack [11].
- *Counterfactual fairness:* $\phi((x,y);\theta) = f_\theta(\mathrm{CF}(x)) - f_\theta(x)$, where $\mathrm{CF}(x)$ is a counterfactual version of x [12].

Achieving DG in Practice. Our key observation is that distributional generalization (DG) is formally implied by *differential privacy* (DP) [13, 14]). The spirit of this observation is not novel: DP training is known to satisfy much stronger notions of generalization (e.g., *robust generalization*, see App. C for more details), and stability than standard SGD [15–18]. We show that a similar connection holds for the notion of distributional generalization, and prove (and improve) tight bounds relating DP, stability, and DG. In particular, we show that if a training procedure satisfies DP, it also satisfies the following DG guarantee:

Proposition 1.1. *A training algorithm satisfying (ϵ, δ)-DP also satisfies δ'-DG with:*

$$\delta' = \frac{\exp(\epsilon) - 1 + 2\delta}{\exp(\epsilon) + 1}. \tag{3}$$

This guarantees the WYSIWYG property for any method that is differentially-private, including DP-SGD on deep neural networks [19]. We detail these results in App. D.

2 Example Applications of WYSIWYG Training

We demonstrate how DG can be a useful design principle in three concrete settings. First, we show that we can mitigate disparate impact of DP training [20, 21] by leveraging importance sampling. Second, we study the setting of distributionally robust optimization [e.g., 7, 22]. We show how ideas from DP can be used to construct heuristic optimizers, which do not formally satisfy DP, yet empirically exhibit DG. Our heuristics lead to competitive results with SOTA algorithms in five datasets in the distributional robustness setting. Third, we show that the same heuristic optimizer also is capable of reducing the overfitting of adversarial loss in adversarial training [23]. Next, we provide the concise summary of the application settings and results, and defer the details to App. B.

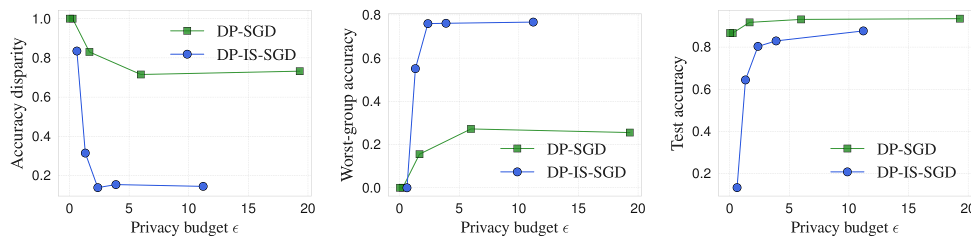
2.1 Mitigating Disparate Impact of DP

First, we consider applications in which learning presents privacy concerns, e.g., in the case that the training data contains sensitive information. Using training procedures that satisfy DP is a standard way to guarantee privacy in such settings. Training with DP, however, is known to incur *disparate impact* on the model accuracy: some subgroups of inputs can have worse test accuracy than others. For example, Bagdasaryan et al. [20] show that using DP-SGD—a standard algorithm for satisfying DP [19]—in place of regular SGD causes a significant accuracy drop on "darker skin" faces in models trained on the CelebA dataset of celebrity faces [24], but a less severe drop on "lighter skin" faces. Our goal is to mitigate such disparate impact.

For this, we propose the DP-IS-SGD algorithm (see App. A), which is a variant of standard DP-SGD [19] with importance sampling. Fig. 2 shows that DP-IS-SGD achieves lower disparity at the same privacy budget compared to standard DP-SGD, with a mild impact on test accuracy on CelebA.

2.2 Group-Distributional Robustness

Next, we consider a setting of *group-distributionally robust optimization* [e.g., 7, 22]. If in the standard learning approach we want to train a model that minimizes *average* loss, in this setting, we



(a) Accuracy disparity (lower is better) (b) Worst-group accuracy (higher is better) (c) Test accuracy (higher is better)

Figure 2: **Importance Sampling Mitigates Disparate Impact of DP-SGD at the Cost of Accuracy.** The accuracy disparity of the models trained with DP-SGD and DP-IS-SGD on CelebA. Adding importance sampling (IS) mitigates disparate impact at most privacy budgets in this setting. We set $\delta = 1/2n$, where n is the dataset size.

Table 1: **Our noisy-gradient algorithms produce competitive results compared to counterparts with ℓ_2 regularization.** The table shows the worst-group accuracy of each algorithm. Baselines are in the top rows; our algorithms are in the bottom. For gDRO-ℓ_2-SOTA, we show avg. $\pm$ std. over five runs from Idrissi et al. [25]. For CelebA, we show avg. $\pm$ std. over three random splits.

	CelebA	UTKFace	iNat.	Civil.	MNLI
SGD-ℓ_2	73.0 ± 2.2	86.3	41.8	57.4	67.9
IS-SGD-ℓ_2	82.4 ± 0.5	85.8	70.6	64.3	70.4
IW-SGD-ℓ_2	**89.0** ± 0.9	86.5	67.6	65.7	68.1
gDRO-ℓ_2	84.5 ± 0.8	85.2	67.3	67.3	75.9
gDRO-ℓ_2-SOTA	86.9 ± 0.5	—	—	69.9 ± 0.5	**78.0** ± 0.3
DP-IS-SGD	86.0 ± 0.8	82.5	51.4	70.4	72.3
IS-SGD-n	84.9 ± 1.0	85.5	**71.0**	**71.9**	70.8
IW-SGD-n	**88.5** ± 0.4	**88.5**	70.9	69.9	69.7
gDRO-n	83.3 ± 0.5	87.5	56.4	71.3	**78.0**

want to minimize the *worst-case (highest) group loss*. This objective can be used to mitigate fairness concerns such as those discussed previously, as well as to avoid learning spurious correlations [7].

Unlike the previous application, in this setting, we do not require privacy of the training data. We use training with DP as a *tool* to ensure the generalization of the worst-case group loss.

Inspired by our theoretical results, we propose a relaxation of DP-IS-SGD: gradient noise regularization method. We observe that the gradient noise, in general, has similar or slightly better performance compared to its non-noisy counterparts. This showcases that in terms of learning distributionally robust models, *noisy gradient can be potentially a more effective regularizer than the currently standard ℓ_2 regularizer*. We also find that DP-IS-SGD improves on baseline methods or even achieves similar SOTA performance on several datasets. This is surprising, as DP tends to deteriorate performance. This suggests that distributional robustness and privacy might not be incompatible goals. Moreover, DP can be a useful tool even when privacy is not required.

2.3 Mitigating Robust Overfitting

Finally, we consider the setting of robustness to test-time adversarial examples through adversarial training [26]. A common way to train robust models in this sense in image domains is to minimize *robust (adversarial) loss*. Rice et al. [23] observed that adversarially trained models exhibit "robust overfitting": higher generalization gap of robust loss than that of the regular loss. In this application, we similarly aim to use a relaxed version of training with DP as a tool to ensure generalization of robust loss, thus mitigate robust overfitting.

To verify this, we adversarially train models on the CIFAR-10 [27] dataset with varying levels of the noise magnitude. Fig. 8 (in Appendix G.6) shows that our proposed approach decreases the generalization gap of robust accuracy by more than $3\times$ to less than 10 p.p.

References

[1] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*. PMLR, 2018.

[2] Allison Koenecke, Andrew Nam, Emily Lake, Joe Nudell, Minnie Quartey, Zion Mengesha, Connor Toups, John R Rickford, Dan Jurafsky, and Sharad Goel. Racial disparities in automated speech recognition. *Proceedings of the National Academy of Sciences*, 117(14):7684–7689, 2020.

[3] Grant Van Horn and Pietro Perona. The devil is in the tails: Fine-grained classification in the wild. *arXiv preprint arXiv:1709.01450*, 2017.

[4] Arthur Gretton, Alex Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, and Bernhard Schölkopf. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4):5, 2009.

[5] Jonathon Byrd and Zachary Chase Lipton. What is the effect of importance weighting in deep learning? In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, 2019.

[6] Hongseok Namkoong and John C. Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 2016.

[7] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

[8] Preetum Nakkiran and Yamini Bansal. Distributional generalization: A new kind of generalization. *arXiv preprint arXiv:2009.08092*, 2020.

[9] Bogdan Kulynych, Mohammad Yaghini, Giovanni Cherubin, Michael Veale, and Carmela Troncoso. Disparate vulnerability to membership inference attacks. *Proceedings on Privacy Enhancing Technologies*, 2022(1), 2022.

[10] Shai Shalev-Shwartz, Ohad Shamir, Nathan Srebro, and Karthik Sridharan. Learnability, stability and uniform convergence. *The Journal of Machine Learning Research*, 11, 2010.

[11] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[12] Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 2017.

[13] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*. Springer, 2006.

[14] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4), 2014.

[15] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. Generalization in adaptive data analysis and holdout reuse. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, 2015.

[16] Rachel Cummings, Katrina Ligett, Kobbi Nissim, Aaron Roth, and Zhiwei Steven Wu. Adaptive learning with robust generalization guarantees. In *Proceedings of the 29th Conference on Learning Theory, COLT 2016, New York, USA, June 23-26, 2016*, volume 49 of *JMLR Workshop and Conference Proceedings*, 2016.

[17] Raef Bassily, Kobbi Nissim, Adam D. Smith, Thomas Steinke, Uri Stemmer, and Jonathan R. Ullman. Algorithmic stability for adaptive data analysis. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, 2016.

[18] Thomas Steinke and Lydia Zakynthinou. Reasoning about generalization via conditional mutual information. In *Conference on Learning Theory*, pages 3437–3452. PMLR, 2020.

[19] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016.

[20] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. Differential privacy has disparate impact on model accuracy. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 2019.

[21] David Pujol, Ryan McKenna, Satya Kuppam, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. Fair decision making using privacy-protected data. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 189–199, 2020.

[22] Weihua Hu, Gang Niu, Issei Sato, and Masashi Sugiyama. Does distributionally robust supervised learning give robust classifiers? In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, 2018.

[23] Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pages 8093–8104. PMLR, 2020.

[24] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, 2015.

[25] Badr Youbi Idrissi, Martin Arjovsky, Mohammad Pezeshki, and David Lopez-Paz. Simple data balancing achieves competitive worst-group-accuracy. *arXiv preprint arXiv:2110.14503*, 2021.

[26] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[27] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[28] Boel Nelson and Jenni Reuben. Sok: Chasing accuracy and privacy, and catching both in differentially private histogram publication. *arXiv preprint arXiv:1910.14028*, 2019.

[29] Kamalika Chaudhuri and Nina Mishra. When random sampling preserves privacy. In *Annual International Cryptology Conference*. Springer, 2006.

[30] Ninghui Li, Wahbeh Qardaji, and Dong Su. On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy. In *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, 2012.

[31] Jinshuo Dong, Aaron Roth, and Weijie J Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019.

[32] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(3), 2011.

[33] Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, 2016.

[34] Arvind Neelakantan, Luke Vilnis, Quoc V Le, Ilya Sutskever, Lukasz Kaiser, Karol Kurach, and James Martens. Adding gradient noise improves learning for very deep networks. *arXiv preprint arXiv:1511.06807*, 2015.

[35] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017.

[36] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alexander Shepard, Hartwig Adam, Pietro Perona, and Serge J. Belongie. The inaturalist species classification and detection dataset. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 2018.

[37] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text, classification. *arXiv preprint arXiv:1903.04561*, 2019.

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

[39] Ron Kohavi et al. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *Kdd*, volume 96, pages 202–207, 1996.

[40] Depeng Xu, Wei Du, and Xintao Wu. Removing disparate impact on model accuracy in differentially private stochastic gradient descent. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1924–1932, 2021.

[41] Amartya Sanyal, Yaxi Hu, and Fanny Yang. How unfair is private learning? *arXiv preprint arXiv:2206.03985*, 2022.

[42] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. On the compatibility of privacy and fairness. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, 2019.

[43] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *International Conference on Machine Learning*, pages 4615–4625. PMLR, 2019.

[44] Jingzhao Zhang, Aditya Menon, Andreas Veit, Srinadh Bhojanapalli, Sanjiv Kumar, and Suvrit Sra. Coping with label shift via distributionally robust optimisation. *arXiv preprint arXiv:2010.12230*, 2020.

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

[46] 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.

[47] 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*, 32:1567–1578, 2019.

[48] John C Duchi, Peter W Glynn, and Hongseok Namkoong. Statistics of robust optimization: A generalized empirical likelihood approach. *Mathematics of Operations Research*, 2021.

[49] 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*, pages 6781–6792. PMLR, 2021.

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

[51] Chaojian Yu, Bo Han, Li Shen, Jun Yu, Chen Gong, Mingming Gong, and Tongliang Liu. Understanding robust overfitting of adversarial training and beyond. In *International Conference on Machine Learning*, pages 25595–25610. PMLR, 2022.

[52] Olivier Bousquet and André Elisseeff. Stability and generalization. *The Journal of Machine Learning Research*, 2, 2002.

[53] Sofya Raskhodnikova, Adam Smith, Homin K Lee, Kobbi Nissim, and Shiva Prasad Kasiviswanathan. What can we learn privately. In *Proceedings of the 54th Annual Symposium on Foundations of Computer Science*, 2008.

[54] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Leon Roth. Preserving statistical validity in adaptive data analysis. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, 2015.

[55] Yu-Xiang Wang, Jing Lei, and Stephen E Fienberg. Learning with differential privacy: Stability, learnability and the sufficiency and necessity of erm principle. *The Journal of Machine Learning Research*, 17(1), 2016.

[56] Quan Geng, Wei Ding, Ruiqi Guo, and Sanjiv Kumar. Optimal noise-adding mechanism in additive differential privacy. In *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, 2019.

[57] Rina Foygel Barber and John C Duchi. Privacy and statistical risk: Formalisms and minimax bounds. *arXiv preprint arXiv:1412.4451*, 2014.

[58] Larry Wasserman and Shuheng Zhou. A statistical framework for differential privacy. *Journal of the American Statistical Association*, 105(489), 2010.

[59] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The composition theorem for differential privacy. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, 2015.

[60] Zhiqi Bu, Jinshuo Dong, Qi Long, and Weijie J Su. Deep learning with gaussian differential privacy. *Harvard data science review*, 2020(23), 2020.

[61] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019.

[62] Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, Graham Cormode, and Ilya Mironov. Opacus: User-friendly differential privacy library in PyTorch. *arXiv preprint arXiv:2109.12298*, 2021.

[63] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL `https://doi.org/10.1038/s41586-020-2649-2`.

[64] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[65] The pandas development team. pandas-dev/pandas: Pandas, February 2020. URL `https://doi.org/10.5281/zenodo.3509134`.

[66] 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*. PMLR, 2021.

[67] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016.

[68] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019.

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

[70] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019.

[71] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE, 2017.

[72] Bargav Jayaraman and David Evans. Evaluating differentially private machine learning in practice. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 1895–1912, 2019.

[73] Jiaxiang Liu, Simon Oya, and Florian Kerschbaum. Generalization techniques empirically outperform differential privacy against membership inference. *arXiv preprint arXiv:2110.05524*, 2021.

[74] Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Ruslan Salakhutdinov, and Kamalika Chaudhuri. A closer look at accuracy vs. robustness. In *Proceedings of Neural Information Processing Systems (NeurIPS)*, 2020.

Algorithm 1 DP-IS-SGD (DP Importance Sampling SGD)

Input: Dataset S, loss $\ell(z;\theta)$, initial parameters θ_0, learning rate η, maximal gradient norm C, noise parameter σ, number of epochs T, sampling rate $\bar{p}$, group probabilities $(q_1, \ldots, q_m)$.

for $t = 1, \ldots, T$ **do**

Sample batch $S_t \leftarrow$ $\mathsf{Sample}_{p(\cdot)}(S)$, with sampling probabilities $p(z) \triangleq \bar{p}/m \cdot q_{g(z)}$

$\tilde{g}_t \leftarrow \frac{1}{|S_t|} \sum_{z \in S_t} \underbrace{1/\max\{1,\ C^{-1} \cdot \|\nabla_\theta \ell(z;\theta)\|_2\}}_{\text{Gradient clipping}} \cdot \nabla_\theta \ell(z;\theta) + \underbrace{\sigma C \cdot \mathcal{N}(0, I)}_{\text{Gradient noise}}$

$\theta_t \leftarrow \theta_{t-1} + \eta \cdot \tilde{g}_t$

The highlighted parts indicate the differences with respect to DP-SGD. We obtain DP-SGD as a special case when we have a single group with $q = 1$ (implying $p(z) = \bar{p}$).

A Algorithms which Distributionally Generalize

In this section, we construct algorithms for the applications in Sec. 2. Our approach follows the blueprint: First, we apply a principled algorithmic intervention that ensures desired behavior on *the training dataset* (e.g., importance sampling). Second, we modify the resulting algorithm to additionally ensure DG, which guarantees that the desired behavior generalizes to the *test data*.

A.1 DP Training with Importance Sampling

Our first algorithm, DP-IS-SGD (Algorithm 1), is a version of DP-SGD [19] which performs importance sampling. DP-IS-SGD is designed to mitigate disparate impact while retaining DP guarantees. The standard DP-SGD samples data batches using *uniform Poisson subsampling:* Each example in the training set is chosen into the batch according to the outcome of a Bernoulli trial with probability $\bar{p} \in [0, 1]$. To correct for unequal representation and the resulting disparate impact, we use *non-uniform Poisson subsampling*: Each example $z \in S$ has a possibly different probability $p(z)$ of being selected into the batch, where $p(z)$ does not depend on the dataset S otherwise, and is bounded: $0 \leq p(z) \leq p^* \leq 1$. We denote this subsampling procedure as $\mathsf{Sample}_{p(\cdot)}(S)$.

We assume that we know to which group any $z = (x, y)$ belongs, denoted as $g(z)$, e.g., the group is one of the features in x. We choose $p(z)$ to satisfy two properties. First, to increase the sampling probability for examples in minority groups: $p(z) \propto 1/q_{g(z)}$. Second, to keep the average batch size equal to $\bar{p} \cdot n$ as in standard DP-SGD. In the rest of the paper, we assume that the group probabilities $(q_1, \ldots, q_m)$ are known, but it is possible to estimate them in a private way using standard methods [28]. We present DP-IS-SGD in Algorithm 1, along with its differences to the standard DP-SGD.

DP Properties of DP-IS-SGD. Uniform Poisson subsampling is well-known to amplify the privacy guarantees of an algorithm [29, 30]. For example, Li et al. [30] show that if an algorithm $\theta(S)$ satisfies (ϵ, δ)-DP, then $\theta \circ \mathsf{Sample}_{\bar{p}}(S)$ provides approximately $(O(\bar{p}\epsilon), \bar{p}\delta)$-DP for small values of ϵ. We show in App. E that non-uniform Poisson subsampling provides the same amplification guarantee with $\bar{p} = p^*$, where p^* is the maximum value of $p(\cdot)$.

As this guarantee is independent of the internal workings of $\theta(S)$, it is loose. For DP-SGD, one way of computing tight privacy guarantees of subsampling is using the notion of *Gaussian differential privacy* (GDP) [31]. GDP is parameterized by a single parameter μ. If an algorithm $\theta(S)$ satisfies μ-GDP, one can efficiently compute a set of (ϵ, δ)-DP guarantees also satisfied by $\theta(S)$ [31]. We show that we can use any GDP-based mechanism for computing the privacy guarantee of DP-SGD to obtain the privacy guarantees of DP-IS-SGD in a black-box manner:

Proposition A.1. *Let us denote by $\mu(\bar{p}, \sigma, C, T)$ (see Algorithm 1) a function that returns a μ-GDP guarantee of DP-SGD. Then, DP-IS-SGD satisfies a GDP guarantee $\mu(p^*, \sigma, C, T)$.*

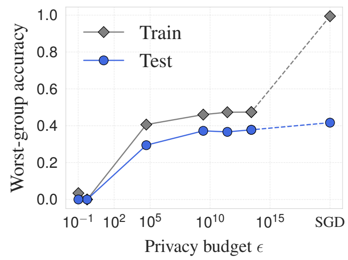


Figure 3: **Privacy induces DG.** Train/test worst-case group accuracies as a function of privacy parameter ϵ of DP-SGD on CelebA (x axis). Increasing privacy reduces the generalization gap.

A.2 Group-DRO with Noisy Gradients

We showed that DP-IS-SGD enjoys theoretical guarantees for both DP and DG. However, DP models often have lower test accuracy compared to standard training [32]. This can be an unnecessary disadvantage in settings where privacy is not required, such as in group-distributional robustness. Thus, we explore non-DP algorithms which do not come with theoretical guarantees on DG, but are inspired by our theory, and satisfy good empirical DG in practice.

DP-SGD uses gradient clipping (line 5 in Algorithm 1) and gradient noise (lines 7–8). Individually, these are used as *regularization methods* for improving stability and generalization [33, 34], thus possibly improving DG in practice. Following this, we relax DP-IS-SGD to only use addition of noise to the gradient as a regularizer. This sacrifices privacy in exchange for practical performance. Specifically, we apply *gradient noise* to three standard algorithms for achieving group-distributional robustness: importance sampling (IS-SGD), importance weighting (IW-SGD) [4], and gDRO [7]. This results in the following variations: IS-SGD with noisy gradient (IS-SGD-n), IW-SGD with noisy gradient (IW-SGD-n), and gDRO with noisy gradient (gDRO-n). See Appendix F for more details.

B Experiments

We empirically study the distributional generalization in real-world applications.

Datasets. We use the following datasets with group annotations: CelebA [24], UTKFace [35], iNaturalist2017 (iNat) [36], CivilComments [37], MultiNLI [7, 38], and ADULT [39]. For every dataset, each example belongs to one group (e.g., CelebA) or multiple groups (e.g., CivilComments). For example, in the CelebA dataset, there are four groups: “blond male”, “male with other hair color”, “blond female”, and “female with other hair color”. Additionally, we use the CIFAR-10 [27] dataset for the adversarial-overfitting application. We present more details on the datasets, their groups, and used model architectures in App. G.

B.1 Enforcing DG in Practice

We empirically confirm that a training procedure with DP guarantees also has a bounded DG gap.

In practice, it is not possible to compute the exact DG gap. As a proxy in applications which concern subgroup performance in this section, and App. B.2 and B.3, we use the difference between train-time and test-time worst-group accuracy. This (1) follows the empirical approach by Nakkiran and Bansal [8] which proposes to estimate the train-test gap using a finite set of test functions, and (2) measures the aspect of distributional generalization that is relevant to our applications. We provide more details on this choice of the proxy measure in App. G.2.

We train a model on CelebA using DP-SGD for different levels of privacy ϵ. Fig. 3 shows that the gap between training and testing worst-group accuracy increases as the level of privacy gets smaller, which is consistent with our theoretical bounds. In App. G.3 we also explore how regularization methods which do not necessarily formally imply DG, can empirically improve DG.

B.2 Disparate Impact of Differentially Private Models

We evaluate DP-IS-SGD (Algorithm 1), and demonstrate that it can mitigate the disparate impact in realistic settings where both privacy and fairness are required.

Fig. 2 shows the accuracy disparity, test accuracy, and worst-case group accuracy, as a function of the privacy budget ϵ. The models are trained with DP-SGD and DP-IS-SGD. When comparing DP-SGD and DP-IS-SGD with the same or similar ϵ, we observe that DP-IS-SGD achieves lower disparity on all datasets. However, this comes with a drop in average accuracy. On CelebA, for example, with $\epsilon \in [2, 12]$, DP-IS-SGD has around 8 p.p. lower test accuracy than DP-SGD. At the same time, the disparity drop ranges from 40 p.p. to 60 p.p., which is significantly higher than the accuracy drop. We observe similar results on UTKFace. On iNat, however, although DP-IS-SGD decreases disparity, the overall test accuracy suffers a significant hit. This is likely because the minority subgroup is extremely small, and importance-sampling are poorly behaved for very small groups. Details for UTKFace and iNat are in App. G.4.

In summary, we find that DP-IS-SGD can achieve lower disparity at the same privacy budget compared to standard DP-SGD, with mild impact on test accuracy.

Comparison to DP-SGD-F [40]. DP-SGD-F is a variant of DP-SGD which dynamically adapts gradient-clipping bounds for different groups to reduce the disparate impact. We did not manage to achieve good overall performance of DP-SGD-F on the datasets above. In App. G.4, we compare it to DP-IS-SGD on the ADULT dataset (used by Xu et al. [40]), finding that DP-IS-SGD obtains lower disparity for the same privacy level, yet lower overall accuracy.

B.3 Group-Distributionally Robust Optimization

We investigate whether our proposed versions of standard algorithms with Gaussian gradient noise (App. A.2) can improve group-distributional robustness. To do so, we evaluate empirical DG using worst-group accuracy as a proxy for DG gap as in App. B.1, following the evaluation criteria in prior work [7, 25]. State-of-the-art (SOTA) methods apply ℓ_2 regularization and early-stopping to achieve the best performance. We compare three baselines with ℓ_2 regularization, IS-SGD-ℓ_2, IW-SGD-ℓ_2, and gDRO-ℓ_2 to our noisy-gradient variations as well as DP-IS-SGD. We use the validation set to select the best-performing regularization parameter and epoch (for early stopping) for each method. See App. G.5 for details on the experimental setup.

Tab. 1 shows the worst-group accuracy of each algorithm on five datasets. When comparing IS-SGD, IW-SGD, and gDRO with their noisy counterparts, we observe that the noisy versions in general have similar or slightly better performance compared to non-noisy counterparts. For instance, IS-SGD-n improves the SOTA results on CivilComments dataset. This showcases that in terms of learning distributionally robust models, *noisy gradient can be potentially a more effective regularizer than the currently standard ℓ_2 regularizer*. We also find that DP-IS-SGD improves on baseline methods or even achieves similar SOTA performance on several datasets. For instance, on CelebA and MNLI, DP-IS-SGD achieves better performance than IS-SGD-ℓ_2, and achieves comparable performance to SOTA. This is surprising, as DP tends to deteriorate performance. This suggests that distributional robustness and privacy might not be incompatible goals. Moreover, DP can be a useful tool even when privacy is not required.

B.4 Mitigating Robust Overfitting

As in the previous section, we expect that a modification of a standard projected gradient-descent method for adversarial training [11]—with added Gaussian gradient noise (App. A.2)—improves the generalization behavior of adversarial training.

To verify this, we adversarially train models on the CIFAR-10 dataset with varying levels of the noise magnitude. We provide more details on the setup in App. G.6. Fig. 8 shows that in standard adversarial training without noise the gap between robust training accuracy and robust test accuracy is large at approximately 30 p.p., which is consistent with the prior observations of Rice et al. [23]. By injecting noise into the gradient, our proposed approach decreases the generalization gap of robust accuracy by more than $3\times$ to less than 10 p.p. Surprisingly, in our experiments, training with gradient noise achieves both a small adversarial accuracy gap *and* better adversarial test accuracy compared to standard adversarial training, when using a small noise magnitude ($\sigma = 0.0005$). These experimental results demonstrate how WYSIWYG can be a useful design principle in practice.

C Related Work

DP and Strong Generalization. DP is known to imply a stronger than standard notion of generalization, called *robust generalization*[1] [16, 17]. Robust generalization can be thought as a high-probability counterpart of DG: generalization holds with high probability over the train dataset, not only on average over datasets. We focus on our notion of DG for both conceptual and theoretical simplicity. Other than robust generalization, our connection between DP and DG can also be derived from weaker generalization bounds that rely on information-theoretic measures [18].

Disparate Impact of DP. Bagdasaryan et al. [20], Pujol et al. [21] have shown that ensuring DP in algorithmic systems can cause error disparity across population groups. Xu et al. [40] proposed a variant of DP-SGD for reducing disparate impact. We compare our method to DP-SGD-F in App. G.4. In another line of related work, Sanyal et al. [41], Cummings et al. [42] show fundamental trade-offs between performance and DP training. As our theoretical results concern generalization, not performance, our results do not contradict these theoretical trade-offs.

Group-Distributional Robustness. Group-distributional robustness aims to improve the worst-case group performance. Existing approaches include using worst-case group loss [7, 43, 44], balancing majority and minority groups by reweighting or subsampling [5, 25, 45], leveraging generative models [46], and applying various regularization techniques [7, 47]. Although some work [7, 47] discusses the importance of regularization in distributional robustness, they have not explored potential reasons for this (e.g. via the connection to generalization). Another line of work studies how to improve group performance without group annotations [48–50], which is a different setting from ours as we assume the group annotations are known.

Robust Overfitting. Rice et al. [23], Yu et al. [51] have shown that adversarially trained models tend to overfit in terms of robust loss. Rice et al. [23] proposed to use regularization to mitigate overfitting, but the noisy gradient has not been explored for this. We showed that the WYSIWYG framework can serve as an alternative direction for mitigating and explaining this issue.

D Details on Theory

The connections between privacy, stability, and generalization are well-known. In particular, stability of the learning algorithm—its non-sensitivity to limited changes in the training data—implies generalization [10, 52]. In turn, differential privacy implies strong forms of stability, thus ensuring generalization through the chain Privacy $\Rightarrow$ Stability $\Rightarrow$ Generalization [15, 53–55].

Let us formally define differential privacy:

Definition D.1 (Differential Privacy [13, 14])**.** An algorithm $\theta(S)$ is (ϵ, δ)-differentially private (DP) if for any two *neighbouring datasets*—differing by one example—S, S' of size n, for any subset $K \subseteq \Theta$ it holds that $\Pr[\theta(S) \in K] \leq \exp(\epsilon) \Pr[\theta(S') \in K] + \delta$.

DP mathematically encodes a notion of plausible deniability of the inclusion of an example in the dataset. However, it can also be thought as a strong form of stability [54]. As such, DP implies other notions of stability.

We consider the following notion, which has been studied in the literature under multiple names and contexts. In the context of privacy, it is equivalent to $(0, \delta)$-differential privacy, and has been called additive differential privacy [56], and total-variation privacy [57]. In the context of learning, it has been called total-variation (TV) stability [17]. We take this last approach and refer to it as TV stability:

Definition D.2 (TV Stability)**.** An algorithm $\theta(S)$ is δ-TV stable if for any two *neighbouring datasets* S, S' of size n, for any subset $T \subseteq \Theta$ it holds that $\Pr[\theta(S) \in K] \leq \Pr[\theta(S') \in K] + \delta$.

[1] Unrelated to "robust overfitting" in adversarial training.

It is easy to see that (ϵ, δ)-DP immediately implies δ'-TV stability with:

$$\delta' = \exp(\epsilon) - 1 + \delta \tag{4}$$

From Classical to Distributional Generalization. Similarly to the classical generalization, one way to achieve distributional generalization is through strong stability:

Theorem D.3. *Suppose that the training algorithm is δ-TV stable. Then, the algorithm satisfies δ-DG.*

We refer to App. E for the proofs of this and all other formal statements in the rest of the paper.

As DP implies TV-stability, by Theorem D.3 we have that DP also implies DG. We show that DP algorithms enjoy a significantly stronger stability guarantee than previously known, which means that the DG guarantee that one obtains from DP is also stronger.

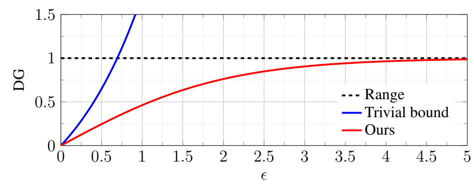


Figure 4: Bound on TV stability (therefore DG) from DP, assuming $\delta = 0$. x axis: ϵ level of DP. y axis: δ-level of TV stability/DG.

Proposition D.4. *Suppose that the algorithm is (ϵ, δ)-DP. Then, the algorithm satisfies δ'-TV stability with:*

$$\delta' = \frac{\exp(\epsilon) - 1 + 2\delta}{\exp(\epsilon) + 1}.$$

We show that our bound is tight in App. E.

Stronger Distributional Generalization Guarantees. Although DG immediately implies generalization for all bounded properties, it is possible to obtain tighter bounds from TV stability. For example, directly applying δ-DG to the *subgroup loss* property yields a bound that decays with the size of the subgroup: accuracy on very small subgroups is not guaranteed to generalize well. In **??** we show that TV stability in fact implies "subgroup DG", which guarantees that the accuracy on even small subgroups generalizes well in expectation. As another example, in **??** we show that TV stability also ensures the generalization of calibration properties of the learning algorithm.

E Proofs

E.1 TV-Stability implies Distributional Generalization

Proof of Theorem D.3. First, observe that the following distributions are equivalent as the dataset is an i.i.d. sample:

$$\begin{aligned} \Pr_{\substack{S\sim\mathcal{D}^n \\ z\sim S}}[\phi\big(z; \theta(S)\big)] &\equiv \Pr_{\substack{S\sim\mathcal{D}^{n-1} \\ z\sim\mathcal{D}}}[\phi\big(z; \theta(S \cup \{z\})\big)], \\ \Pr_{\substack{S\sim\mathcal{D}^n \\ z\sim\mathcal{D}}}[\phi\big(z; \theta(S)\big)] &\equiv \Pr_{\substack{S\sim\mathcal{D}^{n-1} \\ z\sim\mathcal{D} \\ z'\sim\mathcal{D}}}[\phi\big(z'; \theta(S \cup \{z\})\big)]. \end{aligned} \tag{5}$$

It is thus sufficient to analyze the equivalent distributions instead. By the post-processing property of differential privacy, for any dataset $S \in \mathbb{D}^{n-1}$, any two examples $z, z' \in \mathbb{D}$, and any set $K \subseteq \{0, 1\}$:

$$\Pr[\phi\big(z; \theta(S \cup \{z\})\big) \in K] \leq \Pr[\phi\big(z; \theta(S \cup \{z'\})\big) \in K] + \delta,$$

as datasets $S \cup \{z\}$ and $S \cup \{z'\}$ are neighbouring. Taking the expectation of both sides over $z, z' \sim \mathcal{D}$ and $S \sim \mathcal{D}^{n-1}$, we get:

$$\begin{aligned} \Pr_{\substack{S\sim\mathcal{D}^{n-1} \\ z\sim\mathcal{D}}}[\phi\big(z; \theta(S \cup \{z\})\big) \in K] &\leq \Pr_{\substack{S\sim\mathcal{D}^{n-1} \\ z\sim\mathcal{D} \\ z'\sim\mathcal{D}}}[\phi\big(z; \theta(S \cup \{z'\})\big) \in K] + \delta \\ &= \Pr_{\substack{S\sim\mathcal{D}^{n-1} \\ z\sim\mathcal{D} \\ z'\sim\mathcal{D}}}[\phi\big(z', \theta(S \cup \{z\})\big) \in K] + \delta, \end{aligned} \tag{6}$$

where the last equality is simply renaming of the variables for convenience. Note that analogously we also can obtain a symmetric bound:

$$\Pr_{\substack{S\sim\mathcal{D}^{n-1}\\ z\sim\mathcal{D}\\ z'\sim\mathcal{D}}}[\phi\big(z',\theta(S\cup\{z\})\big)\in K]\leq \Pr_{\substack{S\sim\mathcal{D}^{n-1}\\ z\sim\mathcal{D}}}[\phi\big(z;\theta(S\cup\{z\})\big)\in K]+\delta, \tag{7}$$

The total variation between these two distributions is bounded:

$$\begin{aligned} &d_{\mathsf{TV}}\Big(\Pr_{\substack{S\sim\mathcal{D}^{n-1}\\ z\sim\mathcal{D}}}[\phi\big(z;\theta(S\cup\{z\})\big)], \Pr_{\substack{S\sim\mathcal{D}^{n-1}\\ z\sim\mathcal{D}\\ z'\sim\mathcal{D}}}[\phi\big(z',\theta(S\cup\{z\})\big)]\Big)\\ &=\sup_{K\subseteq\mathsf{range}(\phi)}\Big|\Pr_{\substack{S\sim\mathcal{D}^{n-1}\\ z\sim\mathcal{D}}}[\phi\big(z;\theta(S\cup\{z\})\big)\in K]-\Pr_{\substack{S\sim\mathcal{D}^{n-1}\\ z\sim\mathcal{D}\\ z'\sim\mathcal{D}}}[\phi\big(z',\theta(S\cup\{z\})\big)\in K]\Big|\leq\delta,\end{aligned}$$

where the last inequality is by Eq. (7). Using the equivalences in Eq. (5) we can see that:

$$d_{\mathsf{TV}}\Big(\Pr_{\substack{S\sim\mathcal{D}^n\\ z\sim S}}[\phi\big(z;\theta(S)\big)], \Pr_{\substack{S\sim\mathcal{D}^n\\ z\sim\mathcal{D}}}[\phi\big(z;\theta(S)\big)]\Big)=\Big|\mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n\\ z\sim S}}[\phi(z;\theta(S)]-\mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n\\ z\sim\mathcal{D}}}[\phi(z;\theta(S)]\Big|\leq\delta,$$

which is the sought result. □

E.2 Tight Bound on TV-Stability from DP

To prove Proposition D.4, we make use of the hypothesis-testing interpretation of DP [58]. Let us define the hypothesis-testing setup and the two types of errors in hypothesis testing. For any two probability distributions P and Q defined over $\mathbb{D}$, let $\phi:\mathbb{D}\to\{0,1\}$ be a *hypothesis-testing decision rule* that aims to tell whether a given observation from the domain $\mathbb{D}$ comes from P or Q.

Definition E.1 (Hypothesis-testing FPR and FNR)**.** Without loss of generality, the *false-positive error rate* α_ϕ (FPR, or type I error rate), and the *false-negative error rate* β_ϕ (FNR, or type II error rate) of the decision rule $\phi:\mathbb{D}\to[0,1]$ are defined as the following probabilities:

$$\begin{aligned}\alpha_\phi &\triangleq \Pr_{z\sim P}[\phi(z)=1]=\mathop{\mathbb{E}}_P[\phi],\\ \beta_\phi &\triangleq \Pr_{z\sim Q}[\phi(z)=0]=1-\mathop{\mathbb{E}}_Q[\phi].\end{aligned} \tag{8}$$

A well-known result due to Le Cam provides the following relationship between the trade-off between the two types of errors and the total variation between the probability distributions:

$$\alpha_\phi+\beta_\phi\geq 1-d_{\mathsf{TV}}(P,Q). \tag{9}$$

DP is known to provide the following relationship between FPR and FNR of any decision rule:

Proposition E.2 (Kairouz et al. [59])**.** *Suppose that an algorithm $\theta(S)$ satisfies (ϵ,δ)-DP. Then, for any decision rule $\phi:\mathbb{D}\to[0,1]$:*

$$\begin{aligned}\alpha_\phi+\exp(\epsilon)\,\beta_\phi &\geq 1-\delta,\\ \exp(\epsilon)\,\alpha_\phi+\beta_\phi &\geq 1-\delta.\end{aligned} \tag{10}$$

We can now prove Proposition D.4:

Proof. Consider a hypothesis-testing setup in which we want to distinguish between the distributions $\theta(S)$ and $\theta(S')$. Let us sum the two bounds in Eq. (10):

$$(\exp(\epsilon)+1)(\alpha_\phi+\beta_\phi)\geq 2(1-\delta)\implies \alpha_\phi+\beta_\phi\geq\frac{2-2\delta}{\exp(\epsilon)+1}. \tag{11}$$

Let us take the optimal decision rule ϕ^*. In this case, the bound in Eq. (9) holds exactly:

$$d_{\mathsf{TV}}(\theta(S),\theta(S'))=1-(\alpha_{\phi^*}+\beta_{\phi^*}).$$

Combining this with Eq. (11), we get:

$$d_{\mathsf{TV}}(\theta(S), \theta(S')) \le 1 - \frac{2 - 2\delta}{\exp(\epsilon) + 1} = \frac{\exp(\epsilon) - 1 + 2\delta}{\exp(\epsilon) + 1}.$$

□

Next, we show that the upper bound is tight up to δ:

Proposition E.3. *There is an algorithm $\theta(S)$ satisfying $(\varepsilon, 0)$-DP, such that $d_{\mathsf{TV}}(\theta(S), \theta(S')) = \frac{\exp(\varepsilon)-1}{\exp(\varepsilon)+1}$ for two neighbouring datasets S and S'.*

Proof. Consider two distributions P_0 and P_1 on a set $\{0, 1\}$, with $P_0(\{0\}) = P_1(\{1\}) = \gamma$ for some γ to be chosen later, and $P_0(\{1\}) = P_1(\{0\}) = 1 - \gamma$. Those two distributions satisfy $d_{\mathsf{TV}}(P_0, P_1) = 1 - 2\gamma$, as well as the closeness condition appearing in the definition of $(\varepsilon, 0)$-DP

$$\forall T, \Pr_{z \sim P_0}(z \in T) \le \exp(\varepsilon) \Pr_{z \sim P_1}(z \in T),$$

with $\exp(\varepsilon) = \frac{1-\gamma}{\gamma}$. Expressing now TV-distance in terms of ε, we get $d_{\mathsf{TV}}(P_0, P_1) = \frac{\exp(\varepsilon)-1}{\exp(\varepsilon)+1}$. With those distributions in hand, it is easy to provide a mechanism $\theta : \{0, 1\} \to \{0, 1\}$ satisfying the desired property: on the input 0, it generates output according to distribution P_0, and on the input 1, it generates output according to distribution P_1. □

E.3 Privacy Analysis of DP-IS-SGD

First, we present a loose analysis of the privacy guarantees of non-uniform Poisson subsampling.

Lemma E.4. *Suppose that $\theta(S)$ satisfies (ϵ, δ)-DP and $\mathsf{Sample}(S)$ is a Poisson sampling procedure where each of the sampling probabilities p_i depend on the element z_i (but do not depend on the set S otherwise) and is guaranteed to satisfy $p_i \le p^*$. Then $\theta \circ \mathsf{Sample}$ satisfies $(\ln(1 - p^* + p^* e^\epsilon), p^*\delta)$-DP. For small ϵ this can be bounded by $(\mathcal{O}(p^*\epsilon), p^*\delta)$-DP.*

Proof of Lemma E.4. Consider two neighboring datasets S and $S' = S \cup \{z_0\}$ for some $z_0 \notin S$. We wish to show that for any set K, we have

$$\Pr(\theta(\mathsf{Sample}(S')) \in K) \le (1 - p + pe^\epsilon) \Pr(\theta(\mathsf{Sample}(S)) \in K) + p\delta$$

and symmetrically for S and S'. We will only prove first of those inequalities, as the second is analogous.

Note that with probability $p_0 \le p$ the element z_0 is included in $\mathsf{Sample}(S')$ and we have $\mathsf{Sample}(S') = \{z_0\} \cup \mathsf{Sample}(S)$, otherwise the element z_0 is not included, and conditioned on z_0 not being included $\mathsf{Sample}(S')$ has the same distribution as $\mathsf{Sample}(S)$. Therefore,

$$\Pr(\theta(\mathsf{Sample}(S')) \in K) = p_0 \Pr(\theta(\{z_0\} \cup \mathsf{Sample}(S)) \in K) + (1 - p_0) \Pr(\theta(\mathsf{Sample}(S)) \in K). \tag{12}$$

Now for each realization $\mathsf{Sample}(S) = \tilde{S}$, we have $\Pr(\theta(\{z_0\} \cup \tilde{S}) \in K) \le e^\epsilon \Pr(\theta(\tilde{S}) \in K) + \delta$ by the assumed DP guarantee of the algorithm $\theta(S)$. We can average over all possible subsets $\tilde{S}$ to get

$$\begin{aligned}
\Pr(\theta(\{z_0\} \cup \mathsf{Sample}(S)) \in K) &= \sum_{\tilde{S}} \Pr(\mathsf{Sample}(S) = \tilde{S}) \Pr(\theta(\{z_0\} \cup \tilde{S}) \in K) \\
&\le \sum_{\tilde{S}} \Pr(\mathsf{Sample}(S) = \tilde{S})(e^\epsilon \Pr(\theta(\tilde{S}) \in K) + \delta) \\
&= e^\epsilon \Pr(\theta(\mathsf{Sample}(S)) \in K) + \delta.
\end{aligned}$$

Plugging this back to the inequality (12), we get

$$\begin{aligned}
\Pr(\theta(\mathsf{Sample}(S')) \in K) &\le p_0(e^\epsilon \Pr(\theta(\mathsf{Sample}(S)) \in K) + \delta) + (1 - p_0) \Pr(\theta(\mathsf{Sample}(S)) \in K) \\
&\le (1 - p^* + p^* e^\epsilon) \Pr(\theta(\mathsf{Sample}(S)) \in K) + p^*\delta.
\end{aligned}$$

Finally, when $\epsilon \le 1$ we have $e^\epsilon \le (1+2\epsilon)$, and therefore $(1-p^* + p^* e^\epsilon) \le 1 + 2\epsilon p^* \le e^{2\epsilon p^*}$. $\square$

For the tight privacy analysis of non-uniform Poisson subsampling, we make use of the notion of f-privacy:

Definition E.5 (f-Privacy Dong et al. [31])**.** An algorithm $\theta(S)$ satisfies f-privacy if for any two neighbouring datasets S, S' the following holds:

$$\tau(\theta(S), \theta(S')) \ge f,$$

where $\tau(P, Q)$ is a trade-off function between the FPR and FNR of distinguishing tests (see App. E.2):

$$\tau(P,Q)(\alpha) = \inf_{\phi: \mathbb{D} \to [0,1]} \{\beta_\phi : \alpha_\phi \le \alpha\}, \tag{13}$$

and $f(\alpha) \in [0, 1]$ is a convex, continuous, non-increasing function.

Bu et al. [60] show that uniform Poisson subsampling (see App. A.1) provides the following privacy amplification:

Proposition E.6 (Bu et al. [60])**.** *Suppose that $\theta(S)$ satisfies f-privacy, and* $\mathsf{Sample}(S)$ *is a uniform Poisson sampling procedure with sampling probability $\bar{p}$. The composition $\theta \circ \mathsf{Sample}(S)$ satisfies f'-privacy with $f' = \bar{p} f + (1-\bar{p})\mathsf{Id}$, where $\mathsf{Id}(\alpha) = 1 - \alpha$ is the trade-off function that corresponds to perfect privacy.*

We show that a similar result holds for non-uniform Poisson subsampling:

Lemma E.7. *Suppose that $\theta(S)$ satisfies f-privacy, and $\mathsf{Sample}(S)$ is a non-uniform Poisson sampling procedure, where the sampling probabilities p_i depend on the element z_i (but do not depend on the set S otherwise) and each is guaranteed to satisfy $p_i \le p^*$. The composition $\theta \circ \mathsf{Sample}(S)$ satisfies f'-privacy with $f' = p^* + (1-p^*)\mathsf{Id}$.*

To show this, we adapt the proof Proposition E.6, and make use of the following lemma:

Lemma E.8 (Bu et al. [60])**.** *Let $\{P_i\}_{i \in I}$ and $\{Q_i\}_{i \in I}$ be two collections of probability distributions on the same sample space for some index set I. Let $(\lambda_i)_{i \in I} \in [0,1]^{|I|}$ be a collection of numbers such that $\sum_{i \in I} \lambda_i = 1$. If $\tau(P_i, Q_i) \ge f$ for all $i \in I$, then for any $p \in [0, 1]$:*

$$\tau\left(\sum_i \lambda_i \cdot P_i,\ \sum_i (1-p) \cdot \lambda_i \cdot P_i + \sum_i p \cdot \lambda_i \cdot Q_i\right) \ge pf + (1-p)\mathsf{Id}.$$

Proof of Lemma E.7. We can think of the result of the subsampling procedure as outputting a binary vector $\vec{b} = (b_1, \ldots, b_n) \in \{0,1\}^n$, where each bit b_i indicates whether an example $z_i \in S$ was chosen in the subsample or not. We denote the resulting subsample as $S_{\vec{b}} \subseteq S$. By definition of Poisson subsampling, each bit b_i is an independent sample $b_i \sim \mathsf{Bern}(p_i)$. Let us denote by $\lambda_{\vec{b}}$ the joint probability of $\vec{b}$. The composition $\theta(S) \circ \mathsf{Sample}(S)$ can be expressed as a mixture distribution:

$$\theta(S) \circ \mathsf{Sample}(S) = \sum_{\vec{b} \in \{0,1\}^n} \lambda_{\vec{b}} \cdot \theta(S).$$

Analogously, for a neighbouring dataset $S' \triangleq S \cup \{z_0\}$, with the sampling probability p_0 corresponding to z_0, we have:

$$\theta(S) \circ \mathsf{Sample}(S) = \sum_{\vec{b} \in \{0,1\}^n} p_0 \cdot \lambda_{\vec{b}} \cdot \theta(S'_{\vec{b}} \cup \{z_0\}) + \sum_{\vec{b} \in \{0,1\}^n} (1-p_0) \cdot \lambda_{\vec{b}} \cdot \theta(S_{\vec{b}}).$$

Applying Lemma E.8, we get f_0-privacy with $f_0 = p_0 f + (1-p_0)\mathsf{Id}$. Applying to an arbitrary other $z_0 \in \mathbb{D}$, we potentially get the worst-case privacy guarantee for the highest sampling probability, i.e., $f = p^* f + (1-p^*)\mathsf{Id}$. $\square$

Proposition A.1 is immediate from Lemma E.7 by the fact that GDP is a special case of f-privacy.

F Additional Details on Algorithms

We define q_g as the probability of group g, and m as the number of groups.

IS-SGD. The weight for group g is $w_g = {}^1/_{m \cdot q_g}$. Let g_i be the group that the i-th example belongs to. We then sample (with replacement) from the training set with the i-th example having a w_{g_i} chance of being sampled until we have b examples, where b is the batch size. Finally, for each mini-batch, we optimize the standard cross-entropy loss with the sampled examples.

IW-SGD. The weight for group g is $w_g = {}^1/_{m \cdot q_g}$. We optimize the following loss function:

$$w_g \cdot \ell(f_\theta(x), y),$$

where $\ell(\cdot, \cdot)$ is the cross-entropy loss and $(x, y) \in S$ drawn uniformly random drawn from the dataset, and g is the group to which (x, y) belongs.

G Additional Experiment Details

G.1 Details on Datasets, Software, and Model Training

Table 2: The number of examples in each subgroup for CelebA.

	training	validation	testing
not blond, female	71629	8535	9767
not blond, male	66874	8276	7535
blond, female	22880	2874	2480
blond, male	1387	182	180

Table 3: The number of examples in each subgroup for UTKFace.

	training	validation	testing
male, White	3919	454	1105
male, Black	1700	181	437
male, Asian	1115	157	303
male, Indian	1594	190	477
male, Others	563	61	136
female, White	3316	384	902
female, Black	1606	188	414
female, Asian	1302	158	399
female, Indian	1230	152	333
female, Others	655	75	202

Table 4: The number of examples in each subgroup for iNat.

	training	validation	testing
Actinopterygii	2112	195	312
Amphibia	14531	1242	1930
Animalia	5362	491	737
Arachnida	4838	461	660
Aves	191773	17497	26251
Chromista	435	52	55
Fungi	6148	575	883
Insecta	96894	8648	13013
Mammalia	26724	2475	3624
Mollusca	7627	693	1057
Plantae	159843	14653	22117
Protozoa	309	25	37
Reptilia	33404	2983	4494

Table 5: The number of examples in each subgroup for CivilComments.

	training	validation	testing
Non-toxic, Identity	94895	15759	46185
Non-toxic, Other	143628	24366	72373
Toxic, Identity	18575	3088	9161
Toxic, Other	11940	1967	6063

Table 6: The number of examples in each subgroup for MNLI.

	training	validation	testing
Contradiction, No negation	57498	22814	34597
Contradiction, Negation	11158	4634	6655
Entailment, No negation	67376	26949	40496
Entailment, Negation	1521	613	886
Neutral, No negation	66630	26655	39930
Neutral, Negation	1992	797	1148

Table 7: The number of examples in each subgroup for ADULT.

	training	validation	testing
Female, income$\leq$50k	11763	911	1749
Male, income$\leq$50k	18700	1373	2659
Female, income$>$50k	1444	105	220
Male, income$>$50k	8093	611	1214

All algorithms are implemented in `PyTorch`[2] [61]. For DP-related utilities, we use `opacus`[3] [62]. Other packages, including `numpy` [4] [63], `scipy` [5] [64], `tqdm` [6], and `pandas` [7] [65], are also used. For gDRO [7], we use the implementation from `wilds` [66]. We use Nvidia 2080ti, 3080, and A100 GPUs. Our experiments required approximately 400 hours of GPU time.

Datasets. For CelebA and CivilComments, we follow the training/validation/testing split in Koh et al. [66]. For UTKFace and iNat, we randomly split the data into 17000/2000/4708 and 550000/50000/75170 for training/validation/testing. For MNLI, we use the same training/validation/testing split in Sagawa et al. [7]. For Adult [39], we randomly split the data into 35000/3000/5842 for training/validation/testing. Tab. 2 to 7 show the dataset statistics on each group.

All the datasets are publicly available for non-commercial use. In our work, we adhere to additional rules regulating the use of each dataset. All datasets other than iNat could potentially contain personally identifiable information, and are likely collected without consent, to the best of our knowledge. They are all, however, collected from manifestly public sources, such as public posts on social media. Thus, we consider the associated privacy risks low.

The data also contain offensive material (e.g., explicitly in the case of CivilComments dataset). We consider the associated risks of reproducing the offensive behavior low, as we use the datasets only to evaluate our theoretical and theoretically-inspired results.

Models. Similar to previous work [7], we use the ImageNet-1k pretrained ResNet50 [67] from `torchvision` for CelebA, UTKFace, and iNat, and use the pretrained BERT-Base [68] from `huggingface` [69] for CivilComments and MNLI.

For ADULT, we follow the setup in [40] and use logistic regression with standard optimization, and DP-based training methods. We fix the batch size to 256 (for SGD), weight decay to 0.01, and

[2] Code and license can be found in `https://github.com/pytorch/pytorch`.
[3] Code and license can be found in `https://github.com/pytorch/opacus`.
[4] Code and license can be found in `https://github.com/numpy/numpy`
[5] Code and license can be found in `https://github.com/scipy/scipy`
[6] Code and license can be found in `https://github.com/tqdm/tqdm`
[7] Code and license can be found in `https://github.com/pandas-dev/pandas`

Table 8: The accuracy for each subgroup on CelebA. These results are acquired without any regularization or early stopping (trained on full 50 epochs).

		blond		not blond	
		female	male	female	male
SGD	train	1.00	0.99	1.00	1.00
	test	0.80	0.42	0.97	1.00
IW-SGD	train	0.98	0.99	0.98	0.99
	test	0.87	0.49	0.95	0.98
IS-SGD	train	1.00	1.00	1.00	1.00
	test	0.83	0.38	0.96	0.99
DP-SGD	train	0.80	0.41	0.96	0.99
	test	0.74	0.29	0.98	1.00
DP-IS-SGD	train	0.94	0.96	0.88	0.90
	test	0.92	0.85	0.91	0.92

number of epochs to 20. For the DP algorithms, we use gradient norm clipping to 0.5, and sampling rate of 0.005. For all training algorithms, we train five model times with different random seeds and we record the mean and standard error of the mean of our metrics. The noise parameter σ for DP-SGD-F and DP-SGD is set to 1.0, and we set the σ for DP-IS-SGD to 5.0 to achieve similar privacy budget $\epsilon \approx 0.7$. The additional noise parameter for DP-SGD-F σ_2 is set to 10σ as in Xu et al. [40].

Hyperparameters. We run 50 epochs for CelebA, 100 epochs for UTKFace, 20 epochs for iNat, and 5 epochs for CivilComments and MNLI. For image datasets (CelebA, UTKFace, and iNat), we use the SGD optimizer and for NLP datasets (CivilComments and MNLI), we use the AdamW [70] optimizer. We use `opacus`'s [62] implementation of DP-SGD and DP-AdamW to achieve DP guarantees.

We fix the batch size for none-DP algorithms to 64 for CelebA and UTKFace, 256 for iNat, 16 for CivilComments, and 32 for MNLI. For DP-SGD and DP-IS-SGD, we set the sample rate to 0.0001 for CelebA and iNat, 0.001 for UTKFace, and 0.00005 for CivilComments and MNLI.

G.2 Generalization of Worst-Case Group Accuracy as a Proxy for the DG Gap

Although generalization of worst-case group accuracy is not explicitly implied by DG, in our experiments it is practically equivalent to using the generalization gap of subgroup accuracy, which is bounded by TV stability. Let us first concretely define the generalization gap of the worst-case group accuracy:

Definition G.1. The on-average generalization gap of the worst-case accuracy is defined as the following difference:

$$\text{WGGAP} \triangleq \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n}\left[\max_{g\in\mathcal{G}} \mathop{\mathbb{E}}_{z\sim S_g}[\ell(z,\theta(S))] \,\middle|\, |S_g|>0\right] - \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n}\left[\max_{g\in\mathcal{G}} \mathop{\mathbb{E}}_{z\sim\mathcal{D}_g}[\ell(z,\theta(S))]\right], \quad (14)$$

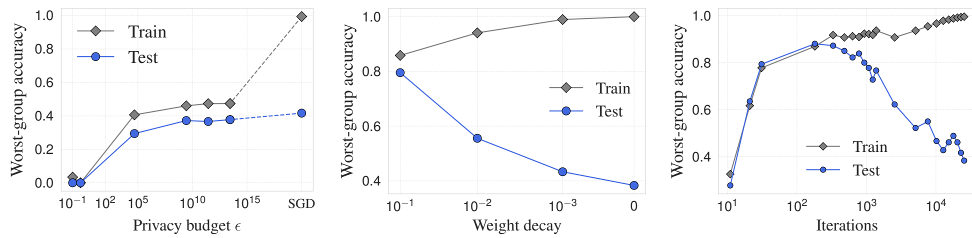
where we take $\ell((x,y),\theta) \triangleq \mathbb{1}[f_\theta(x)=y]$ to be the 0-1 loss. In this definition we explicitly restrict the datasets to include elements of each group $g \in \mathcal{G}$, which is a technicality needed in order to avoid undefined behavior.

In all our experimental results, the worst-performing groups (the maximizers in Eq. (14)) are always the same on the training and test data. As long as this holds—the worst-performing group is the same on the train and test data—the generalization gap above simplifies to:

$$\text{WGGAP} = \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n\\ z\sim S_{g^*}}}[\ell(z,\theta(S)) \mid |S_{g^*}|>0] - \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n\\ z\sim\mathcal{D}_{g^*}}}[\ell(z,\theta(S))], \quad (15)$$

where $g^* \in \mathcal{G}$ is the worst-performing group. In **??** we show that this simplified gap from Eq. (15) is bounded by TV stability.

Therefore, in practice the generalization gap in Eq. (14) offers a lower bound on the DG gap in **??**. Using it as a proxy for DG gap follows the spirit of the estimation approach by Nakkiran and Bansal



(a) Differential privacy as regularization (b) ℓ_2 regularization (c) Early-stopping as regularization

Figure 5: **Regularization induces DG.** The figure shows train/test worst-group accuracies as a function of regularization strength for SGD on CelebA, with different types of regularizers: differential privacy budget ϵ, weight decay, and train time. For DP-SGD, $\epsilon = \infty$ represents standard SGD. For all types of regularizers, increasing the strength (left on x-axis) corresponds to a smaller generalization gap in worst-group accuracy.

[8] which proposes to estimate the DG gap by taking the maximum of empirical generalization gaps for a finite set of relevant test functions (here, per-group accuracies).

Other Approaches to Estimate the DG Gap. The generalization gap of worst-case group accuracy can be loose as a proxy. Finding the worst-case test function is an object of study in the literature on *membership inference attacks* [71], because DG and the accuracy of such attacks are equivalent, as showed by Kulynych et al. [9]. We avoid using accuracy of a membership inference attack as a proxy for DG gap in this work as the fact that differential privacy and regularization impacts vulnerability to these attacks is known and well-documented [72, 73]. This body of evidence from the field of membership inference offers an alternative source of empirical support for our claims checked in App. B.1.

G.3 Additional Details for App. B.1

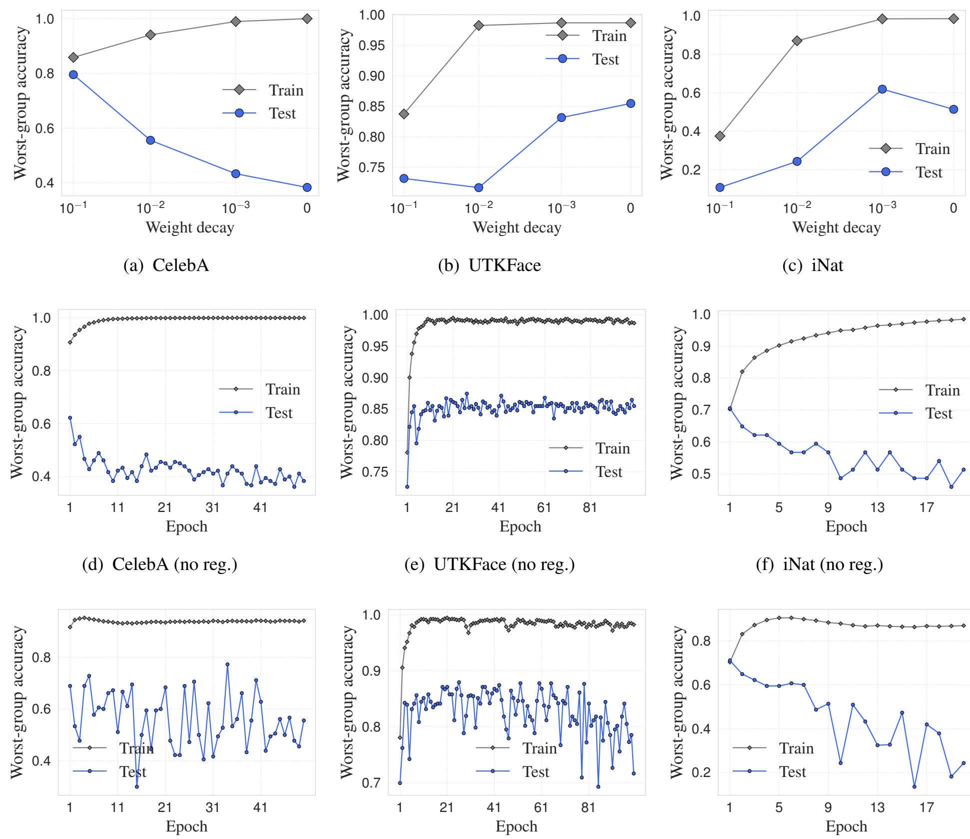
As mentioned in App. A, many regularization methods can be used to improve different generalization gaps. For example, Sagawa et al. [7] show that strong ℓ_2 regularization helps with improving group-distributional generalization, and Yang et al. [74] show that dropout helps with adversarial-robustness generalization. However, these works do not have theoretical justification.

Our framework suggests a unifying reason why strong regularization is helpful in distributional robustness: because it enforces DG. Following this theoretically-inspired intuition, other regularization methods beyond a combination of gradient noise and clipping (DP-SGD) can imply DG in practice. We verify this hypothesis empirically.

Privacy, ℓ_2 Regularization, and Early Stopping. In Fig. 5, we train a neural network on CelebA using DP-SGD, and decrease the "regularization strength" in several different ways: by increasing privacy budget ϵ (Fig. 5a), decreasing the ℓ_2 regularization (Fig. 5b), or increasing the number of training iterations (Fig. 5c).[8] We then measure the gap in worst-group accuracy on train vs. test (App. G.2). We observe that for all regularizers, the gap between training and testing worst-group accuracy increases as the regularization is weakened.

Investigating ℓ_2 Regularization in Depth. In Fig. 6, we show the training and testing worst-group accuracy with different strength of ℓ_2 regularization and on different epochs (w/ and w/o ℓ_2 regularization). We have three observations: (1) with properly tuned regularization parameter, the gap between training and testing worst-group accuracy can be narrowed, (2) the gap can start widening in very early stage of training, and (3) the testing worst-group accuracy can fluctuate largely, which highlights the importance of using validation set for early stopping in this task.

[8]Train time can be considered a regularizer, as its decrease induces stability (e.g. Hardt et al. [33]).



(a) CelebA (b) UTKFace (c) iNat

(d) CelebA (no reg.) (e) UTKFace (no reg.) (f) iNat (no reg.)

(g) CelebA (ℓ_2 reg.) (h) UTKFace (ℓ_2 reg.) (i) iNat (ℓ_2 reg.)

Figure 6: We show the training and testing worst-group accuracy with different strength of ℓ_2 regularization and on different epochs (w/ and w/o ℓ_2 regularization). The network is trained with IS-SGD on CelebA, UTKFace, and iNat. For (a), (b), and (c), we show the result of the last epoch. For (g), (h), and (i), we set weight decay to 0.01.

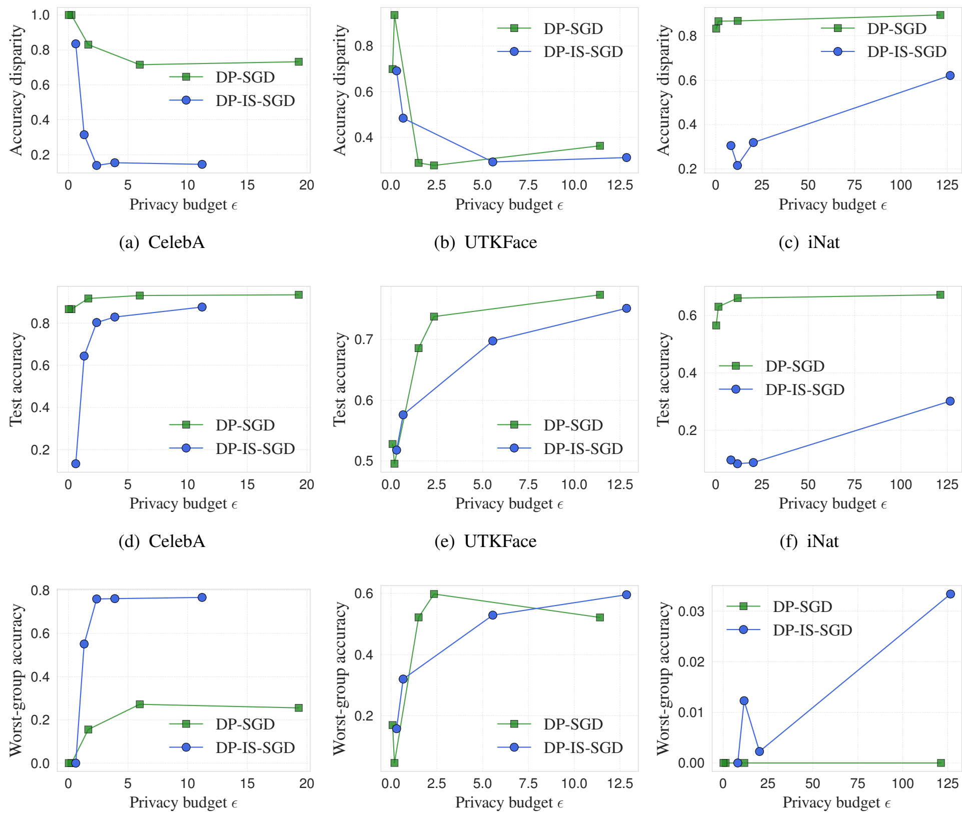
G.4 Additional Details for App. B.2

Fig. 7 shows the accuracy disparity, test accuracy, and worst-group accuracy for CelebA, UTKFace, and iNat on DP-SGD and DP-IS-SGD.

The reason that UTKFace has a similar disparity between DP-SGD and DP-IS-SGD is likely because UTKFace has a relatively small difference in the number of training examples between the largest group and the smallest group. In UTKFace, the majority group has around seven times more examples than in the minority group, whereas in CelebA, this difference is $52\times$.

Comparison with DP-SGD-F [40]. We did not manage to obtain good performance from DP-SGD-F on CelebA, UTKFace, and iNat, possibly because of the different domain—images—than tabular data considered by Xu et al. [40]. To proceed with the comparison, we evaluate the algorithms on the census data—ADULT dataset [39] (see Tab. 7 for dataset statistics)—that Xu et al. [40] used in their work. As subgroups, we consider four intersectional groups composed of all possible values of the "sex" attribute and prediction class (an income higher/lower than 50k).

We show the results in Tab. 9. For a comparable epsilon value (0.69 for DP-SGD-F, and 0.7 for our DP-IS-SGD), we see that our method has smaller accuracy disparity (Eq. 2) across the groups, although also lower overall accuracy.



(a) CelebA (b) UTKFace (c) iNat

(d) CelebA (e) UTKFace (f) iNat

(g) CelebA (h) UTKFace (i) iNat

Figure 7: The disparity (lower the better) and test accuracies of the models trained with DP-SGD and IW-SGD on three datasets. If we care about privacy, DP-IS-SGD improves disparate impact at most privacy budgets. For CelebA, we train the model for 30 epochs. For UTKFace, we train for 100 epochs. For iNat, we train for 20 epochs. The GDP accountant is used to compute the privacy budget.

Table 9: **DP-IS-SGD has lower disparity DP-SGD-F on ADULT and better accuracy at the same privacy level.** The table shows the privacy level, maximum accuracy disparity across groups, and overall accuracy for all algorithms.

Algorithm	ϵ	Accuracy disparity	Overall accuracy
SGD	-	0.660 ± 0.000	0.836 ± 0.000
DP-SGD	0.6573	0.852 ± 0.005	0.802 ± 0.001
DP-SGD-F	0.6964	0.657 ± 0.023	0.832 ± 0.001
DP-IS-SGD	0.7059	0.246 ± 0.034	0.766 ± 0.010

G.5 Additional Details for App. B.3

We compare different algorithms, including SGD-ℓ_2 and IW-SGD-ℓ_2 as baselines, and two other algorithms, IS-SGD-ℓ_2 [25] and gDRO-ℓ_2 [7] in terms of the group robustness. We set the learning rate as 0.001 for CelebA, UTKFace, and iNat, 0.00002 for MNLI, and 0.00001 for CivilComments. We use the validation set to select the hyperparameters:

1. For SGD-ℓ_2, IW-SGD-ℓ_2, IS-SGD-ℓ_2, and gDRO-ℓ_2, we select the weight decay from 0.0001, 0.01, 0.1, and 1.0.
2. For DP-IS-SGD, we fix the gradient clipping to 1.0 (except for iNat, where we set the value to 10.0 as 1.0 does not converge). We select the noise parameter from 1.0, 0.1, 0.01, 0.001 on CelebA and UTKFace, select the noise parameter from 0.0000001, 0.000001, 0.00001,

and 0.0001 on iNat and select the noise parameter from 0.01 and 0.001 on CivilComments and MNLI.

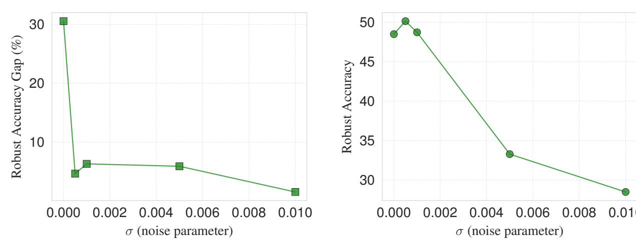
3. For IW-SGD-n, IS-SGD-n, and gDRO-n, we select the standard deviation of the random noise from 0.001, 0.01, 0.1, and 1.0 on CelebA, UTKFace, and iNat, and we select standard deviation of the random noise from 0.00001, 0.0001, and 0.001 on CivilComments and MNLI.

Statistical Concerns. Although our results appear to be comparable to or better than SOTA, we caution readers about the exact ordering of methods due to high estimation variance: these benchmarks have small validation and test sets (e.g., CelebA has 182 validation examples), and so hyperparameter tuning is subject to both overfitting and estimation error. For example, we observe validation accuracies which differ from their test accuracies by up to 5% in our experiments. We attempt to mitigate this using three random train/val/test splits on CelebA, and avoid large hyperparameter sweeps[9], but this is not done in prior work.

G.6 Additional Details for App. B.4

We use the CIFAR-10 dataset [27], and ResNet-18 [67] as the network architecture. We train the model to be robust against L_∞ perturbations of at most $\gamma = 8/255$ bound, which is a standard setup for adversarial training on this dataset. We vary σ (noise parameter) from 0.0 (regular adversarial training without gradient noise) to 0.01.

In this experiment, we measure robust accuracy and its respective generalization gap, thus setting $\ell((x, y), \theta) \triangleq \mathbb{1}[f_\theta(x) = y]$ to be the 0-1 loss.



(a) Gen. gap of robust accuracy (b) Robust accuracy.

Figure 8: **Noisy gradient reduces overfitting in adversarial training.** We show the generalization gap of robust accuracy (left), and test-time robust accuracy (right) of adversarially trained models with different levels of noise magnitude. The model trained without noise exhibits "robust overfitting" of about 30 p.p. Gradient noise reduces the generalization gap by more than $3\times$ for all values of the noise parameter at a cost of decreased robust accuracy as the noise gets larger.

[9] For example, we do not tune the "group adjustments" parameter for gDRO, using the default from Koh et al. [66] instead.