What You See is What You Get: Distributional Generalization for Algorithm Design in Deep Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

We investigate and leverage a connection between Differential Privacy (DP) and the recently proposed notion of Distributional Generalization (DG). Applying this connection, we introduce new conceptual tools for designing deep-learning methods that bypass "pathologies" of standard stochastic gradient descent (SGD). First, we prove that differentially private methods satisfy a "What You See Is What You Get (WYSIWYG)" generalization guarantee: whatever a model does on its train data is almost exactly what it will do at test time. This guarantee is formally captured by distributional generalization. WYSIWYG enables principled algorithm design in deep learning by reducing *generalization* concerns to *optimization* ones: in order to mitigate unwanted behavior at test time, it is provably sufficient to mitigate this behavior on the train data. This is notably false for standard (non-DP) methods, hence this observation has applications even when privacy is not required. For example, importance sampling is known to fail for standard SGD, but we show that it has exactly the intended effect for DP-trained models. We use these insights to construct simple algorithms which match or outperform SOTA in several distributional robustness applications, and to significantly improve the privacy vs. disparate impact tradeoff of DP-SGD. Finally, we also improve on known theoretical bounds relating DP, stability, and distributional generalization.

1 Introduction

Much of machine learning (ML), both in theory and in practice, operates under two assumptions. First, we have i.i.d. samples from a distribution. Second, we care only about a single averaged scalar (e.g. error/loss/risk) over this distribution. 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.

However, the focus on scalar metrics such as average error 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 [Buolamwini and Gebru, 2018, Koenecke et al., 2020]. In general, ML models are applied to the heterogeneous and long-tailed data distributions of the real world [Van Horn and Perona, 2017]. 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 by their train sets. This presents an obstacle for algorithm design, because interventions which alter a network's properties on its train set do not always transfer to the test

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

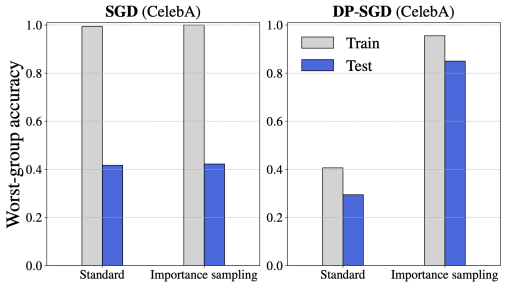


Figure 1: **Differential privacy ensures the desired behavior of importance sampling on test data.** The train and test accuracy of ResNets on CelebA, evaluated on the worst-performing ("male, blond") subgroup. *Left:* Standard SGD has a large generalization gap on this subgroup, and Importance Sampling (IS) has little effect. *Right:* DP-SGD provably has small generalization gap on all subgroups, and IS improves subgroup performance as intended. See Section 5 for details.

set. For example, consider the setting of *importance sampling*: suppose we know that a certain subgroup of inputs is underrepresented in the train dataset compared to the test distribution (breaking the "i.i.d. train/test" assumption). For underparameterized models, we can simply up-sample this underrepresented group to account for the distribution shift [see, e.g., Gretton et al., 2009]. This approach, however, is known to empirically fail for overparameterized models [Byrd and Lipton, 2019]. Because "what you see" (on the train dataset) 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 [2016]) fail for overparameterized deep networks, and require ad-hoc modifications [Sagawa et al., 2019].

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 train dataset 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 train dataset. It cannot be the case, for example, that a WYSIWYG model has small disparate impact on the train dataset, 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 train dataset. This means that algorithm designers can be concerned only with achieving "good behavior" on the train dataset, and the WYSIWYG property guarantees good behavior at test time. 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 Figure 1 for an illustration).

As WYSIWYG is a high-level conceptual property, we have to formalize it to use in practice. We do so using the recent notion of *Distributional Generalization* (DG), as introduced by Nakkiran and Bansal [2020], Kulynych et al. [2022]. If classical generalization ensures that the values of the model's loss on the training dataset and at test time are close on average [Shalev-Shwartz et al., 2010], distributional generalization ensures that values of any other bounded test function (not only loss) are close on training and test time. We say that a model that satisfies an appropriate level of distributional generalization exhibits the WYSIWYG property.

Achieving DG via Differential Privacy (DP). Our key observation is that distributional generalization (DG) is formally implied by *differential privacy* (DP) [Dwork et al., 2006, 2014]). The spirit of this observation is not novel: DP training is known to satisfy much stronger notions of generalization (e.g., *robust generalization*, see Section 6 for more details), and stability than standard SGD [Dwork et al., 2015a, Cummings et al., 2016, Bassily et al., 2016, Steinke and Zakynthinou, 2020]. 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 DG. This guarantees the WYSIWYG property for any method that is differentially-private, including DP-SGD on deep neural networks [Abadi et al., 2016].

We demonstrate how DG can be a useful design principle in two concrete settings. First, we show that we can mitigate disparate impacts of DP training [Bagdasaryan et al., 2019, Pujol et al., 2020] by leveraging importance sampling. Second, we study the setting of distributionally robust optimization. We show how ideas from DP can be used to construct heuristic optimizers, which do not formally satisfy DP, yet empirically exhibit high DG. Our heuristics lead to competitive results with SOTA algorithms in five datasets in the distributional robustness setting.

Our Contributions. We develop the theoretical connection between Differential Privacy (DP) and Distributional Generalization (DG), and we leverage our theory to improve empirical performance in privacy, fairness, and robustness applications. Theoretically (Sections 2 to 4):

1. We provide tighter bounds than previously known connecting DP and strong forms of generalization, and show that DP training methods satisfy DG, thus the WYSIWYG property.
2. We introduce DP-IS-SGD, an importance-sampling version of DP-SGD, and show it satisfies DP and DG.

Experimentally (Section 5):

1. We use our framework to shed light on *disparate impact* of DP [Bagdasaryan et al., 2019]: The disparity in accuracy across groups at test time are provably reflected by the accuracy disparity *on the train dataset*.
2. We use our DP-IS-SGD algorithm to largely mitigate the disparate impact of DP using importance sampling.
3. Based on our theoretical intuitions, we propose a DP-inspired heuristic: addition of gradient noise. We find this empirically achieves competitive and even improved results in several DRO settings.

Taken together, our results emphasize the central role of distributional generalization—and the WYSIWYG property it implies—in designing machine learning algorithms which avoid the "pathologies" of standard SGD. We also establish DP as a useful tool for achieving DG, and thus extend its applications further beyond privacy.

2 Theory of "What You See is What You Get" Generalization

We first review the notion of distributional generalization and demonstrate why it implies the WYSIWYG property. Second, we show that stability implies distributional generalization. Finally, we improve on the known stability guarantees of differential privacy. As a result, we extend the connections between differential privacy, stability, and generalization to *distributional* generalization, showing that stability and privacy imply the WYSIWYG property.

Notation. We consider a learning task with a set of examples $\mathbb{X}$ and labels $\mathbb{Y}$. We assume that a *source distribution* of labeled examples $z \triangleq (x, y) \sim \mathcal{D}$ is defined over $\mathbb{D} = \mathbb{X} \times \mathbb{Y}$. Given an i.i.d.-sampled *dataset* $S \sim \mathcal{D}^n$ of size n, we use a randomized *training algorithm* $\theta(S)$ that outputs a model's parameter vector θ from the set Θ. We denote by $f_\theta(x)$ the resulting prediction function.

2.1 Distributional Generalization and WYSIWYG

If on-average generalization [Shalev-Shwartz et al., 2010] guarantees closeness only of loss values on train and test data, distributional generalization (DG) also guarantees closeness of values of all test functions $\phi(z; \theta) \in [0, 1]$ beyond only loss:

Definition 2.1 (Based on Nakkiran and Bansal [2020])**.** An algorithm $\theta(S)$ satisfies δ-distributional generalization if for all $\phi : \mathbb{D} \times \Theta \to [0, 1]$,

$$\left| \mathop{\mathbb{E}}_{\substack{S \sim \mathcal{D}^n \\ z \sim S}} \phi\big(z; \theta(S)\big) - \mathop{\mathbb{E}}_{\substack{S \sim \mathcal{D}^n \\ z \sim \mathcal{D}}} \phi\big(z; \theta(S)\big) \right| \leq \delta. \tag{1}$$

By the variational characterization of the total-variation distance, Equation (1) is equivalent to the bound $d_{\mathsf{TV}}(P_1, P_0) \leq \delta$, where P_1 and P_0 are both distributions of $\big(z, \theta(S)\big)$ over the randomness of $S \sim \mathcal{D}^n$ and the training algorithm $\theta(\cdot)$, with the difference that $z \sim S$ in the case of P_1 (train distribution), and $z \sim \mathcal{D}$ in the case of P_0 (test distribution).

It might seem that DG only ensures average closeness of bounded tests on train and test data. This is not, however, the full picture. Consider generalization in terms of a broader class of functions:

Definition 2.2 (Kulynych et al. [2022])**.** An algorithm $\theta(S)$ satisfies (δ, π)-distributional generalization if for a given property function $\pi : \mathbb{D} \times \Theta \to \mathbb{R}^k$ it holds that $d_{\mathsf{TV}}(\pi\sharp P_1, \pi\sharp P_0) \leq \delta$, where $\pi\sharp P$ is the distribution of $\pi(T)$ for $T \sim P$.

Because TV distance is preserved under post-processing, we can see that δ-distributional generalization implies (δ, π)-distributional generalization for *all* property functions. Informally, δ-DG means that for *all* numeric property functions $\pi(z; \theta)$ of a model, the distributions of the property values are close on the train and test data, on average. This fact captures the high-level idea of the *"What You See is What You Get"* (WYSIWYG) guarantee. Some example property functions:

- *Subgroup loss:* $\pi(z; \theta) = \mathbb{1}\{z \in G\} \cdot \ell(z; \theta)$, for some subgroup $G \subset \mathbb{D}$.
- *Counterfactual fairness:* $\pi((x, y); \theta) = f_\theta(\mathrm{CF}(x)) - f_\theta(x)$, where $\mathrm{CF}(x)$ is a counterfactual version of x had it had a different value of a sensitive attribute [Kusner et al., 2017].
- *Robustness to corruptions:* $\pi(z; \theta) = \ell(A(z); \theta)$, where $A(x)$ is a possibly randomized transformation that distorts the example, e.g., by adding Gaussian noise.
- *Adversarial robustness:* $\pi(z; \theta) = \ell(A_\theta(z); \theta)$, where $A_\theta(z)$ is an adversarial example, e.g. generated using the PGD attack [Madry et al., 2018].

In the next sections, we show how a training algorithm can provably satisfy DG and therefore provide WYSIWYG guarantees for all properties, including the ones above.

2.2 Distributional Generalization from Stability and Differential Privacy

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 [Bousquet and Elisseeff, 2002, Shalev-Shwartz et al., 2010]. In turn, differential privacy implies strong forms of stability, thus ensuring generalization through the chain Privacy $\Rightarrow$ Stability $\Rightarrow$ Generalization [Raskhodnikova et al., 2008, Dwork et al., 2015b,a, Wang et al., 2016].

Let us formally define differential privacy:

Definition 2.3 (Differential Privacy [Dwork et al., 2006, 2014])**.** 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 [Dwork et al., 2015b]. 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 [Geng et al., 2019], and total-variation privacy [Barber and Duchi, 2014]. In the context of learning, it has been called total-variation (TV) stability [Bassily et al., 2016]. We take this last approach and refer to it as TV stability:

Definition 2.4 (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 $\Pr[\theta(S) \in K] \leq \Pr[\theta(S') \in K] + \delta$.

It is easy to see that (ϵ, δ)-DP immediately implies $(\exp(\epsilon) - 1 + \delta)$-TV stability.

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

Theorem 2.5. *Suppose that the training algorithm is δ-TV stable. Then, the algorithm satisfies δ-DG.*

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

As DP implies TV-stability, by Theorem 2.5 we have that DP also implies DG. Next, we show that DP algorithms enjoy a significantly stronger stability guarantee than the previously known one, which means that the DG guarantee that one obtains from DP is also stronger.

Proposition 2.6. *Suppose that the training algorithm is (ϵ, δ)-DP. Then, the algorithm satisfies δ'-TV stability with* $\delta' = {(\exp(\epsilon)-1+2\delta)}/{(\exp(\epsilon)+1)}$.

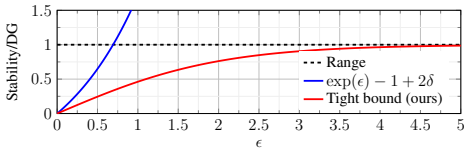


Figure 2: Bound on TV stability (therefore DG) from DP, assuming $\delta = 0$. x axis: ϵ level of DP. y axis: δ-level of TV stability/DG.

Figure 2 shows that the known bound of $\exp(\epsilon) + 1 - \delta$ quickly become vacuous, unlike the one in Proposition 2.6. In Appendix A we discuss the relationship of this result to other works in generalization literature, and show that our bound is tight in Appendix B.

Stronger Distributional Generalization Guarantees. Although DG implies generalization bounds for all properties in a black-box way, it is sometimes possible to obtain tighter bounds for specific properties. For example, if we are concerned with accuracy on subgroups, then 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 Appendix C.1 we show that TV stability in fact implies "subgroup DG", which guarantees that the accuracy on even arbitrarily small subgroups generalizes well in expectation. As another example, in Appendix C.2 we show that TV stability also ensures the generalization of calibration properties of the learning algorithm.

3 Example Applications

To demonstrate that DG is a useful property in algorithm design, in the remainder of this paper we use it to construct simple and high-performing algorithms for two example applications: mitigation of disparate impact of DP and ensuring group-distributional robustness.

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. [2019] show that using DP-SGD—a standard algorithm for satisfying DP [Abadi et al., 2016]—in place of regular SGD causes a significant accuracy drop on "darker skin" faces in models trained on the CelebA dataset of celebrity faces [Liu et al., 2015], but a less severe drop on "lighter skin" faces. Our goal is to mitigate such disparate impact. This issue—a quality-of-service harm [Madaio et al., 2020]—is but one of many possible harms due to ML systems. We do not aim to mitigate any other fairness-related issues, nor claim this is possible within our framework, or with technical solutions in general.

Formally, we assume the data distribution $\mathcal{D}$ is a mixture of m groups indexed by set $\mathcal{G} = \{1, \ldots, m\}$, such that $\mathcal{D} = \sum_{i=1}^{m} q_i \mathcal{D}_i$. The vector $(q_i, \ldots, q_m) \in [0, 1]^m$ represents the group probabilities, with $\sum_{i=1}^{m} q_i = 1$. For given parameters (ϵ, δ), we want to learn a model θ that simultaneously satisfies (ϵ, δ)-DP, has high overall accuracy, and incurs small *loss disparity*:

$$\max_{g,g' \in \mathcal{G}} \left| \mathop{\mathbb{E}}_{z \sim \mathcal{D}_g} [\ell(z; \theta)] - \mathop{\mathbb{E}}_{z \sim \mathcal{D}_{g'}} [\ell(z; \theta)] \right| . \tag{2}$$

Group-Distributional Robustness. Next, we consider a setting of *group-distributionally robust optimization* [e.g., Sagawa et al., 2019, Hu et al., 2018]. If in the standard learning approach we want to train a model that minimizes *average* loss, in this setting, we 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 [Sagawa et al., 2019].

Formally, we want to learn a model θ that minimizes the *worst-case group loss*:

$$\max_{g \in \mathcal{G}} \mathop{\mathbb{E}}_{z \sim \mathcal{D}_g} [\ell(z; \theta)] . \tag{3}$$

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, as described next.

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

Input: Training dataset S, model f_θ parameterized by θ, loss function ℓ; max gradient norm C, noise parameter σ, number of epochs T, sampling rate $\bar{p}$; group probabilities $\vec{q} = (q_1, \ldots, q_m)$

Output: θ_T

for $t = 1, \ldots, T$ **do**
  $\mathsf{Batch}_t \leftarrow \mathsf{Sample}_{p(\cdot)}(S)$, with $p(z) \triangleq \bar{p}/_{m \cdot q_{g(z)}}$
  $\Delta_t\theta \leftarrow 0$
  for $(x_i, y_i) \in \mathsf{Batch}_t$ **do**
    $\Delta_t\theta \leftarrow \Delta_t\theta + \mathsf{Clip}(\nabla_\theta \ell(f_\theta(x_i), y_i), C)$
  end for
  Sample $\xi_t \sim \sigma C \cdot \mathcal{N}(0, I_{d\times d})$
  $\Delta_t\theta \leftarrow \frac{1}{|\mathsf{Batch}_t|}\Delta_t\theta + \xi_t$
  $\theta_t \leftarrow \mathsf{Optimizer}(\theta_{t-1}, \Delta_t\theta)$
end for

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 $p(z) = \bar{p}$).

4 Algorithms which Distributionally Generalize

In this section, we construct algorithms for the applications in Section 3. 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*.

4.1 DP Training with Importance Sampling

Our first algorithm, DP-IS-SGD (Algorithm 1), is a version of DP-SGD [Abadi et al., 2016] 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 [Nelson and Reuben, 2019]. 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 [Chaudhuri and Mishra, 2006, Li et al., 2012]. For example, Li et al. [2012] 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 Appendix B 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) [Dong et al., 2019]. 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 $\theta(S)$ [Dong et al., 2019]. 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 4.1. *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 $\mu(p^*, \sigma, C, T)$.*

4.2 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 [Chaudhuri et al., 2011]. 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 [Hardt et al., 2016, Neelakantan et al., 2015], 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) [Gretton et al., 2009], and gDRO [Sagawa et al., 2019]. 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 D for more details.

5 Experiments

We empirically study the distributional generalization in real-world applications. The code for the experiments is available in the supplementary material.

Datasets. We consider five datasets, studied in the distributional robustness and fairness literature: CelebA [Liu et al., 2015], UTKFace [Zhang et al., 2017], iNaturalist2017 (iNat) [Horn et al., 2018], CivilComments [Borkan et al., 2019], and MultiNLI [Williams et al., 2018, Sagawa et al., 2019]. 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". We present more details on the datasets, their groups, and used model architectures in Appendix E.

5.1 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 the rest of the paper, we use the difference between train-time and test-time worst-group accuracy. This (1) follows the empirical approach by Nakkiran and Bansal [2020] which proposes to estimate the gap in Equation (1) 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 Appendix E.2.

In Figure 3, we train a model on CelebA using DP-SGD for different levels of privacy ϵ. We see that the gap between training and testing worst-group accuracy increases as the level of privacy ε is weakened, which is consistent with our theoretical bounds.

In Appendix E.3 we also explore how regularization methods which do not formally imply DG, can empirically improve DG.

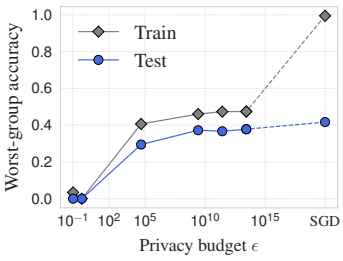
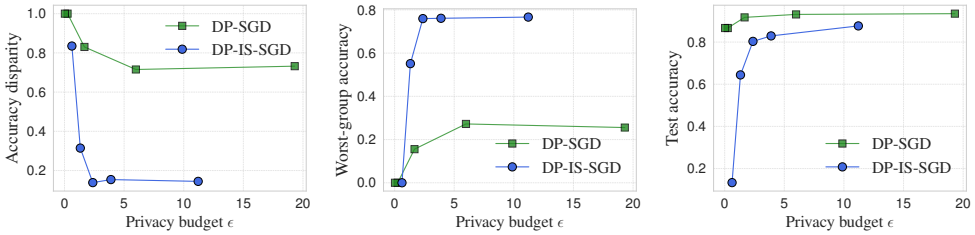


Figure 3: **Privacy induces DG.** Train/test worst-case group accuracies as a function of privacy parameter ϵ of DP-SGD on CelebA (x axis). Increasing privacy (left on x-axis) corresponds to a smaller generalization gap.

5.2 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.

Figure 4 shows the accuracy disparity, test accuracy, and worst-case group accuracy, computed as in Equation (2), 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



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

Figure 4: **Importance Sampling helps Disparate Impact of DP-SGD.** The accuracy disparity of the models trained with DP-SGD and DP-IS-SGD on CelebA. Adding importance sampling (IS) improves disparate impact at most privacy budgets in this setting. We set $\delta = 1/2n$, where n is the number of training examples. We use GDP accountant to compute the privacy budget ε.

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, as shown in Appendix E. 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 appear in Appendix E.

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

5.3 Group-DRO

We investigate whether our proposed versions of standard algorithms with Gaussian gradient noise (Section 4.2) can improve group-distributional robustness. We evaluate empirical DG using worst-group accuracy as a proxy for DG gap as in Section 5.1, following the evaluation criteria in prior work [Sagawa et al., 2019, Idrissi et al., 2021]. 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 three noisy-gradient variations: IS-SGD-n, IW-SGD-n, and gDRO-n, 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 Appendix E for details on the experimental setup.

Table 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.

6 Related Work

DP and Strong Generalization. DP is known to imply a stronger than standard notion of generalization, called *robust generalization* [Cummings et al., 2016, Bassily et al., 2016]. 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: DG formulated as in Equation (1) enables us to prove tight

Table 1: The worst-group accuracy of each algorithm. gDRO-ℓ_2-SOTA shows the average and standard error over five runs from Idrissi et al. [2021]. Baselines are in the top rows; our algorithms are in the bottom. For CelebA, we show mean and standard error over three random splits. Our noisy gradient algorithms produce competitive results compared to their counterparts with ℓ_2 regularization.

	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	$\mathbf{89.0} \pm 0.9$	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	$\mathbf{78.0} \pm 0.3$
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	$\mathbf{88.5} \pm 0.4$	**88.5**	70.9	69.9	69.7
gDRO-n	83.3 ± 0.5	87.5	56.4	71.3	**78.0**

theoretical bounds that are independent of sample size. A more comprehensive discussion of relations to robust generalization is in Appendix A. It can also be useful to consider intermediary definitions, varying in strength from DG to robust generalization. In Appendix C, we introduce such an notion ("strong DG") and show its connections to DP.

Other than robust generalization, our connections in Section 2 can also be derived from weaker generalization bounds that rely on information-theoretic measures [Xu and Raginsky, 2017, Russo and Zou, 2019, Steinke and Zakynthinou, 2020]. We detail this in Appendix A.

Disparate Impact of DP. Bagdasaryan et al. [2019], Pujol et al. [2020] have shown that ensuring DP in algorithmic systems can cause error disparity across population groups. Concurrently, Xu et al. [2021] have proposed DP-SGD-F, a variant of DP-SGD [Abadi et al., 2016] that dynamically adapts gradient-clipping bounds for different groups to reduce such disparate impact. We show that a simpler approach based on importance sampling is also an effective way to reduce disparities, and we provide a more general blueprint for designing algorithms with similar mitigations.

Group-Distributional Robustness. Group-distributional robustness aims to improve the worst-case group performance. Existing approaches include using worst-case group loss [Mohri et al., 2019, Sagawa et al., 2019, Zhang et al., 2020], balancing majority and minority groups by reweighting or subsampling [Byrd and Lipton, 2019, Sagawa et al., 2020, Idrissi et al., 2021], leveraging generative models [Goel et al., 2020], and applying various regularization techniques [Sagawa et al., 2019, Cao et al., 2019]. Although some work [Sagawa et al., 2019, Cao et al., 2019] discusses the importance of regularization in distributional robustness, they have not explored potential reasons for this importance (e.g. via the connection to distributional generalization). Another line of work studies how to improve group performance without group annotations [Duchi et al., 2021, Liu et al., 2021a, Creager et al., 2021], which is a different setting from ours as we assume the group annotations are known.

7 Conclusions

We argue that distributional generalization (DG) is a desirable property for learning algorithms, enabling more principled algorithm design in settings including deep learning. DG captures a high-level "What You See is What You Get" property, which ensures that *any* training-time behavior is reproduced at test time. We show that this property is possible to achieve, by proving it is implied by differential privacy. This enables us to leverage advances in differential privacy (such as DP-SGD for neural networks) to enforce DG in many applications.

We propose enforcing DG as a general design principle, and we use it to construct simple yet effective algorithms in two settings. In certain fairness settings, we largely mitigate the disparate impact of differential privacy by using importance sampling and enforcing DG in our new algorithm DP-IS-SGD. In certain distributional robustness settings, inspired by DP-SGD, we propose using a noisy gradient regularizer instead of the traditional ℓ_2 regularizer to enforce DG. Compared to SOTA algorithms, noisy gradient regularization can achieve competitive results across many standard benchmarks. We hope future work can explore extending this algorithmic design principle to other settings, such as adversarial robustness, counterfactual fairness, and calibration.

References

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.

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.

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

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.

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.

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

Yuheng Bu, Shaofeng Zou, and Venugopal V Veeravalli. Tightening mutual information-based bounds on generalization error. *IEEE Journal on Selected Areas in Information Theory*, 2020a.

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

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

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.

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.

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

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

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

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.

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.

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.

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

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

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

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

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*, 2015a.

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*, 2015b.

Vitaly Feldman and Tijana Zrnic. Individual privacy accounting via a Renyi filter. *Advances in Neural Information Processing Systems*, 2021.

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.

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.

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.

Mahdi Haghifam, Jeffrey Negrea, Ashish Khisti, Daniel M Roy, and Gintare Karolina Dziugaite. Sharpened generalization bounds based on conditional mutual information and an application to noisy, iterative algorithms. *Advances in Neural Information Processing Systems*, 33:9925–9935, 2020.

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.

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`.

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.

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.

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.

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.

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

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.

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.

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.

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

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.

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.

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.

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, 2021a.

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

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.

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.

Michael A Madaio, Luke Stark, Jennifer Wortman Vaughan, and Hanna Wallach. Co-designing checklists to understand organizational challenges and opportunities around fairness in ai. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

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.

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

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

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.

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.

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

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

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.

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.

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.

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

Daniel Russo and James Zou. How much does your data exploration overfit? controlling bias via information usage. *IEEE Transactions on Information Theory*, 2019.

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.

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.

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

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

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.

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

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

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.

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.

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

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.

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.

Aolin Xu and Maxim Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. *Advances in Neural Information Processing Systems*, 2017.

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.

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.

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.

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.

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.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] We address the limitations where appropriate, e.g., high-level limitations in Section 3, and the fact that some methods we use in the empirical evaluations deviate from theory in Sections 4 and 5.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] In Section 3 we caution against using WYSIWYG for fairness-washing through technical solutionism. Other than this, our potential negative societal impacts are the same as with any other academic work which aims to improve the performance of deep learning: we risk improving performance of deep learning in downstream applications that are harmful, e.g., privacy-invasive, or violating other civil liberties. It is beyond the scope of our work to discuss such broad issues.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix B.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We included the experiment code as supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix E.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] For experimental settings in which it was feasible for us to repeat experiments multiple times, we reported the standard errors (Table 1).
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix E.1.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Appendix E.1.
   (b) Did you mention the license of the assets? [Yes] See Appendix E.1.
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] See Appendix E.1.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See Appendix E.1.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Related Work Details

A.1 Differential Privacy and Robust Generalization

DP is known to imply a stronger notion of generalization, called *robust generalization*, which is a "tail bound" version of DG [Dwork et al., 2015a, Cummings et al., 2016, Bassily et al., 2016]. The original motivations for robust generalization are slightly different, but in our notation, a training procedure θ is said to satisfy (γ, η)-Robust Generalization if and only if for any test $\phi : \mathbb{D} \times \Theta \to [0, 1]$, we have

$$\Pr_{S\sim\mathcal{D}^n,\theta}\big(\big|\mathop{\mathbb{E}}_{z\sim S}\phi\big(z;\theta(S)\big) - \mathop{\mathbb{E}}_{z\sim\mathcal{D}}\phi\big(z;\theta(S)\big)\big| > \gamma\big) \leq \eta.$$

Comparing this to the definition of DG (Definition 2.1), it is immediate that any procedure that satisfies (γ, η)-Robust Generalization, also satisfies $(\gamma + \eta)$-DG. Any training method satisfying (ϵ, δ)-DP also satisfies $(\mathcal{O}(\epsilon), \mathcal{O}(\delta))$-robust generalization, as long as the sample size n is of size $\Omega(\log(1/\delta)/\epsilon^2)$ [Bassily et al., 2016, Theorem 7.2], therefore it satisfies $\mathcal{O}(\epsilon+\delta)$-DG by the previous implication. Thus, it is possible to recover the result that DP implies DG as a consequence of these previous works, although with looser bounds.

The difference between Distributional Generalization and Robust Generalization is that DG considers all quantities in expectation, while robust generalization considers tail bounds with respect to the train dataset. We focus on DG for two reasons: First, we believe DG is conceptually simpler, as it can be seen as simply the TV-distance between two natural distributions, and does not involve additional parameters. This simplicity is conceptually useful to the algorithm designer, but also enables us to prove simpler tight theoretical bounds that are independent of sample size. Second, it is often possible to lift results about DG to the stronger setting of robust generalization, with additional bookkeeping. Thus, we focus on DG in this paper, with the understanding that stronger guarantees can be obtained for these methods if desired.

A.2 Information-Theoretic Generalization Bounds

Other than robust generalization, it is also possible to obtain bounds on generalization of arbitrary test (loss) functions using information-theoretic measures [Xu and Raginsky, 2017, Russo and Zou, 2019, Steinke and Zakynthinou, 2020]. Recently, Steinke and Zakynthinou showed that one such information-theoretic measure — *conditional mutual information* (CMI) between training algorithm outputs and the training dataset — is bounded if the training algorithm is DP or TV stable. Thus, we could relate stability and DG as in Section 2.2 using CMI as an intermediate tool.

In particular, Steinke and Zakynthinou show that for bounded $\phi : \mathbb{D} \times \Theta \to [0, 1]$:

$$\Big|\mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n\\ z\sim S}}\phi\big(z;\theta(S)\big) - \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n\\ z\sim\mathcal{D}}}\phi\big(z;\theta(S)\big)\Big| \leq \sqrt{\frac{2}{n}\cdot \mathrm{CMI}_{\mathcal{D}}(\theta)}, \tag{4}$$

where $\mathrm{CMI}_{\mathcal{D}}(\theta)$ is the conditional mutual information of the training algorithm with respect to the data distribution. If the training algorithm satisfies $(\epsilon, 0)$-DP, they also show that $\mathrm{CMI}_{\mathcal{D}}(\theta) \leq \frac{n}{2}\epsilon^2$, where n is the dataset size. Plugging this into Equation (4), we can see that the generalization upper bound (right-hand side) is ϵ. This is significantly looser than the our bound in Section 2.2, as illustrated in Figure 5.

We can use our results to sharpen the CMI bounds of Steinke and Zakynthinou. Using the fact that δ-TV stability implies $\mathrm{CMI}_{\mathcal{D}}(\theta) \leq n\delta$, we can show a tighter CMI guarantee from DP:

Corollary A.1. *Suppose that the training algorithm $\theta : \mathbb{D}^n \to \Theta$ is (ϵ, δ)-DP. Then,*

$$\mathrm{CMI}_{\mathcal{D}}(\theta) \leq n \cdot \frac{\exp(\epsilon) - 1 + 2\delta}{\exp(\epsilon) + 1}.$$

This is immediate from Proposition 2.6. As we illustrate in Figure 5, however, this sharper bound on CMI from DP does not translate into a sharper bound on the generalization gap if we plug the result into Equation (4).

A recent line of work on information-theoretic bounds explored sharper bounds using individual-level measures [Bu et al., 2020a, Haghifam et al., 2020]. As a direction for future work, it could also be

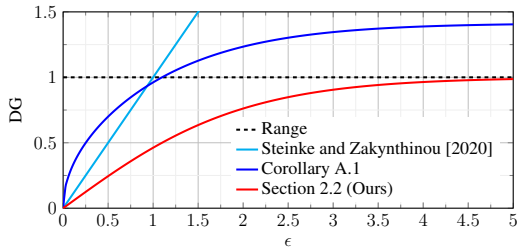


Figure 5: Bound on DG from DP, assuming $\delta = 0$. x axis: ϵ level of DP. y axis: δ-level of DG.

possible to obtain tighter bounds on DG using individual-level notions of stability [Feldman and Zrnic, 2021].

A.3 Tension between Differential Privacy and Algorithmic Fairness

Beyond empirical observations that training with DP results in disparate impact on performance across subgroups [Bagdasaryan et al., 2019, Pujol et al., 2020], Cummings et al. [2019] and, more recently, Sanyal et al. [2022] theoretically analyze the inherent tensions between DP and algorithmic fairness.

It might appear that the tensions contradict our results in which we use DP or similar noise-adding algorithms to reduce disparate impact (Section 4). However, Cummings et al. [2019] and Pujol et al. [2020] discuss the relationship between privacy and disparate performance (accuracy or false-positive/false-negative rates), whereas we discuss the relationship between privacy and generalization. Even if a DP model has to incur at least a certain error on small subgroups on average [Sanyal et al., 2022, Lemma 1], this error is guaranteed to be similar at train time and test time (from our theoretical results in Section 2.2).

In terms of empirical results, the lower bound on subgroup error in Lemma 1 from Sanyal et al. [2022] vanishes for subgroups of size greater than 100 even for small values of epsilon (e.g., 0.1). The subgroups and values of epsilon in our experiments in Section 5 are all larger than this, thus in our regime we can achieve meaningful subgroup performance using the DP-IS-SGD algorithm despite the fundamental trade-off.

B Proofs Omitted in the Main Body

B.1 TV-Stability implies Distributional Generalization

Proof of Theorem 2.5. 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:

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

The total variation between these two distributions is bounded:

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

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

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

which is the sought result. □

B.2 Tight Bound on TV-Stability from DP

To prove Proposition 2.6, we make use of the hypothesis-testing interpretation of DP [Wasserman and Zhou, 2010]. 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 B.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:

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

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

Proposition B.2 (Kairouz et al. [2015])**.** *Suppose that an algorithm $\theta(S)$ satisfies (ϵ,δ)-DP. Then, for any decision rule $\phi:\mathbb{D}\to[0,1]$:*

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

We can now prove Proposition 2.6:

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 Equation (10):

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

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

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

Combining this with Equation (11), we get:

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

□

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

Proposition B.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

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

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. □

B.3 Privacy Analysis of DP-IS-SGD

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

Lemma B.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 $p_i \leq p^*$. 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 B.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

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

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

Note that with probability $p_0 \leq p$ 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 $\Pr(\theta(\{z_0\} \cup \tilde{S}) \in K) \leq e^\epsilon \Pr(\theta(\tilde{S}) \in K) + \delta$ by the assumed DP guarantee of the algorithm $\theta(S)$. We can average over all possible subsets $\tilde{S}$ to get

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

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

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

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

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

Definition B.5 (f-Privacy Dong et al. [2019])**.** An algorithm $\theta(S)$ satisfies f-privacy if for any two neighbouring datasets S, S' the following holds:

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

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

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

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

Bu et al. [2020b] show that uniform Poisson subsampling (see Section 4.1) provides the following privacy amplification:

Proposition B.6 (Bu et al. [2020b])**.** *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 B.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 $p_i \leq p^*$. 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 B.6, and make use of the following lemma:

Lemma B.8 (Bu et al. [2020b])**.** *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 $\tau(P_i, Q_i) \geq f$ for all $i \in I$, then for any $p \in [0,1]$:*

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

Proof of Lemma B.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 B.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 4.1 is immediate from Lemma B.7 by the fact that GDP is a special case of f-privacy.

C Stronger Formalizations of Distributional Generalization

The main notion discussed in this paper is that of δ-Distributional Generalization. To recap, a learning algorithm $\theta(S)$ satisfies it if for any bounded test function ϕ we have

$$\Big| \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n,\theta\\ z\sim S}} \phi\big(z;\theta(S)\big) - \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n,\theta\\ z\sim S}} \phi\big(z;\theta(S)\big)\Big| \le \delta.$$

In Appendix A we have compared it to a stronger property called (γ,η)-Robust Generalization; namely learning algorithm $\theta(S)$ satisfies it if for any bounded ϕ we have:

$$\Pr_{S\sim\mathcal{D}^n,\theta}\Big(\Big| \mathop{\mathbb{E}}_{z\sim S} \phi\big(z;\theta(S)\big) - \mathop{\mathbb{E}}_{z\sim \mathcal{D}} \phi\big(z;\theta(S)\big)\Big| > \gamma\Big) \le \eta.$$

A natural middle ground between these two is the following notion.

Definition C.1. We say that a learning algorithm $\theta(S)$ satisfies *δ-strong Distributional Generalization* if and only if for all $\phi : \mathbb{D}\times\Theta \to [0,1]$:

$$\mathop{\mathbb{E}}_{S\sim\mathcal{D}^n,\theta}\Big| \mathop{\mathbb{E}}_{z\sim S} \phi\big(z;\theta(S)\big) - \mathop{\mathbb{E}}_{z\sim \mathcal{D}} \phi\big(z;\theta(S)\big)\Big| \le \delta.$$

If DG as defined in Section 2 is a (supremum over all test functions) version of expected generalization error [Shalev-Shwartz et al., 2010, Xu and Raginsky, 2017, Russo and Zou, 2019], strong DG is an analogous variant of *expected absolute error* [Steinke and Zakynthinou, 2020]. Clearly any algorithm satisfying (η,γ)-Robust Generalization satisfies also $(\eta+\gamma)$-strong DG, and any algorithm satisfying δ-strong-DG satisfies δ-DG — so in fact the δ-strong DG can be thought of as a strengthening of Distributional Generalization as well as Robust Generalization.

One of the motivations behind the definition of δ-strong DG is the following issue of the Distributional Generalization itself. As it turns out, it is possible to artificially construct a mechanism for binary classification which satisfies 0-DG, always outputs a classifier with test error $\frac{1}{2}+o(1)$ (almost completely useless), but with probability $1/2$ outputs a classifier with training error 0 (and with probability $1/2$ the training error is 1). That is problematic: it is natural to expect from a training procedure with strong generalization guarantee, that if we used it to train a model up to a small training error, we should have high confidence that the test error is also small — but this example shows that our confidence cannot be larger than $1/2$.

The δ-strong DG solves this problem: The following proposition is a formal statement of the desired property, and can be easily deduced from Markov inequality.

Proposition C.2. *If the learning procedure θ satisfies δ-strong-DG, then for any bounded property ϕ we have* $\Pr_{S,\theta}(|\,\mathbb{E}_{z\sim S}\,\phi(z,\theta(S)) - \mathbb{E}_{z\sim\mathcal{D}}\,\phi(z,\theta(S))| > \delta\lambda) < \lambda^{-1}$.

Equivalently, if a learning procedure satisfies δ-strong-DG it also satisfies $(\delta\lambda,\lambda^{-1})$-Robust Generalization for any $\lambda > 1$.

The δ-strong DG defined this way is also a more direct strengthening of the classical notions of generalization gap — which are usually defined as $\mathbb{E}_{S\sim\mathcal{D}^n,\theta}\,|\,\mathbb{E}_{z\sim S}\,\ell(z,\theta(S)) - \mathbb{E}_{z\sim\mathcal{D}}\,\ell(z,\theta(S))|$ where ℓ is the loss function.

We can follow the same proof strategy as in [Bassily et al., 2016, Theorem 7.2], simplifying it significantly, to show that TV stability (and in turn Differential Privacy) implies strong Distributional Generalization, as soon as the training set is large enough.

Lemma C.3. *If a learning algorithm $\theta(S)$ satisfies δ-TV stability, and the sample size $|S| \ge \frac{2}{\delta^2}$, then $\theta(S)$ satisfies 4δ-strong DG.*

Proof. Let us assume first that we have a δ-TV stable mechanism $\mathcal{M} : \mathcal{D}^n \to \mathcal{Q}$ where $\mathcal{Q}$ is a set of bounded queries $q : \mathcal{D} \to \mathbb{R}$ satisfying $|\sup_x(q(x)) - \inf_x(q(x))| \le 1$.

We wish to find another 2δ-TV stable mechanism $\tilde{M} : \mathcal{D}^n \to \mathcal{Q}$ such that if we use notation $q_S \triangleq M(S)$ and $\tilde{q}_S \triangleq \tilde{M}(S)$, we have

$$\mathop{\mathbb{E}}_{S,M} \left| \mathop{\mathbb{E}}_{z\sim\mathcal{D}} q_S(z) - \mathop{\mathbb{E}}_{z\sim S} q_S(z) \right| \le 2 \left(\mathop{\mathbb{E}}_{S,\tilde{M},z\sim S} \tilde{q}_S(z) - \mathop{\mathbb{E}}_{S,\tilde{M},z\sim\mathcal{D}} \tilde{q}_S(z) \right). \tag{14}$$

Indeed, consider the following mechanism $\tilde{M}(S)$: on a sample S, get $q_S \triangleq M(S)$ and evaluate $\tilde{\mu} \triangleq \mathbb{E}_{z\sim S}\, q_S(z) + \frac{1}{\delta n}\mathbf{U}$ where $\mathbf{U}$ is a uniform variable in $[-1/2, 1/2]$. Finally if $\tilde{\mu} \ge \mathbb{E}_{z\sim D}\, q_S(z)$ output q_S, otherwise output $-q_S + 2\,\mathbb{E}_{z\sim\mathcal{D}}\, q_S(z)$.

Since $d_{\mathsf{TV}}(\eta\mathbf{U}, \eta\mathbf{U} + \gamma) \le \frac{\gamma}{\eta}$, given q the mechanism $\mu_q(S) \triangleq \mathbb{E}_{z\sim S}\, q(z) + \frac{1}{\delta n}\mathbf{U}$ is δ-stable. Hence, by composition and post-processing this mechanism $\tilde{M}$ is 2δ-TV stable.

For $n \ge 2\delta^{-2}$ the inequality (14) follows from Lemma C.4. Indeed, let $\hat{Z} \triangleq \mathbb{E}_{z\sim S}\, q_S(z) - \mathbb{E}_{z\sim\mathcal{D}}\, q_S(z)$, $Z = |\hat{Z}|$, and $\sigma = \operatorname{sign}(\hat{Z} + \frac{1}{\delta n}\mathbf{U})$. Note that, expanding the chosen notation, we have $\mathbb{E}_{z\sim S}\, \tilde{q}_S(z) - \mathbb{E}_{z\sim\mathcal{D}}\, \tilde{q}_S(z) = \sigma Z$, and $\mathbb{E}[\sigma|Z] = \max(\frac{2Z}{\delta}, 1)$. The conclusion of the Lemma C.4 is $\mathbb{E}\, Z \le 2\,\mathbb{E}[Z\sigma]$, which is exactly (14).

Since the mechanism $\tilde{M}$ satisfies 2δ-TV stability, by Theorem 2.5 applied to the mechanism $\tilde{M}$, we can bound the right hand side of the inequality (14) by 4δ.

Finally, for an arbitrary learning algorithm θ satisfying δ-TV stability, and any bounded test function $\phi(z, \theta)$, we can describe mechanism $\tilde{M}$ which given a sample $S \in \mathcal{D}^n$ outputs a function $q_S(z) \triangleq \phi(z, \theta(S))$. By post-processing such a mechanism $\mathcal{M}$ also satisfies δ-TV stability, and the inequality (14) becomes

$$\mathop{\mathbb{E}}_{S,M} \left| \mathop{\mathbb{E}}_{z\sim S} \phi(z, \theta(S)) - \mathop{\mathbb{E}}_{z\sim\mathcal{D}} \phi(z, \theta(S)) \right| \le 4\delta.$$

Since ϕ was chosen in an arbitrary way, this proves that mechanism θ indeed satisfies the condition of 4δ-strong-DG. □

Lemma C.4. *Let Z be a non-negative random variable satisfying $\mathbb{E}\, Z \ge \delta$ and σ be a $\{\pm 1\}$ valued random variable with $\mathbb{E}[\sigma|Z] = \max(\frac{2Z}{\delta}, 1)$. Then $\mathbb{E}\, Z \le 2\,\mathbb{E}[Z\sigma]$.*

Proof. This is just a calculation.

$$\mathbb{E}\, Z\sigma = \mathbb{E}\, \mathbb{1}_{Z\ge\delta/2} Z + \mathbb{E}\, \mathbb{1}_{Z<\delta/2} \frac{2Z^2}{\delta} \ge \mathbb{E}\, \mathbb{1}_{Z\ge\delta/2} Z = \mathbb{E}\, Z - \mathbb{E}\, \mathbb{1}_{Z<\delta/2} Z \ge \mathbb{E}\, Z - \delta/2 \ge \frac{1}{2}\,\mathbb{E}\, Z.$$

□

C.1 Subgroup-level Distributional Generalization from TV Stability

TV-stability implies a more granular, subgroup-level notion of distributional generalization:

Definition C.5. Suppose that the data distribution $\mathcal{D}$ is a mixture of group-specific distributions $\mathcal{D}_g$, for $g \in \mathcal{G}$. We define $(\delta, \mathcal{G})$-subgroup-DG similarly to δ-DG as follows:

$$\forall \phi, g \in \mathcal{G} : \quad \left| \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n \\ z\sim S_g}} [\phi(z, \theta(S)) \mid |S_g| > 0] - \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n \\ z\sim\mathcal{D}_g}} [\phi(z, \theta(S))] \right| \le \delta,$$

where S_g denotes a subset of examples in the dataset S that belong to the group g.

Subgroup DG is a stronger notion of DG which says that the model's behavior on examples from each group in $\mathcal{G}$ distributionally generalizes in expectation, as long as the model encounters at least one representative of the group in training. In its definition, we explicitly prevent the case when the training dataset does not contain any group examples to avoid undefined behavior (what is the group accuracy on the training dataset if there are no group representatives in the dataset?).

We now show that TV-stability implies this granular notion of DG:

Proposition C.6. *δ-TV stability implies $(\delta, \mathcal{G})$-subgroup-DG for any group partitioning $\mathcal{G}$.*

Proof. Observe that the following distributions are equivalent:

$$\begin{aligned}\Pr_{\substack{S\sim\mathcal{D}^n\\ z\sim S_g}}[\phi\big(z;\theta(S)\big) \mid |S_g|>0] &\equiv \Pr_{\substack{S\sim\mathcal{D}^{n-1}\\ z\sim\mathcal{D}_g}}[\phi\big(z;\theta(S\cup\{z\})\big)],\\ \Pr_{\substack{S\sim\mathcal{D}^n\\ z\sim\mathcal{D}_g}}[\phi\big(z;\theta(S)\big)] &\equiv \Pr_{\substack{S\sim\mathcal{D}^{n-1}\\ z\sim\mathcal{D}\\ z'\sim\mathcal{D}_g}}[\phi\big(z';\theta(S\cup\{z\})\big)].\end{aligned} \tag{15}$$

The statement follows by applying each step from the proof of Theorem 2.5 to the equivalent distributions in Equation (15). Thus, the difference with the proof of Theorem 2.5 is that $z, z' \sim \mathcal{D}_g$, not $z, z' \sim \mathcal{D}$. □

C.2 Calibration Generalization via Strong Distributional Generalization

We that the strong Distributional Generalization (Definition C.1) implies generalization of calibration properties. For simplicity we consider binary classifiers, with $\mathbb{Y} = \{0, 1\}$ and $f_\theta : \mathbb{X} \to [0, 1]$. Let us define the notion of calibration we consider:

Definition C.7. The *calibration gap*, or Expected Calibration Error (ECE), of a classifier f_θ is defined as follows:

$$\begin{aligned}\textsc{cgap}(f_\theta) &\triangleq \mathbb{E}\left[|\mathbb{E}[y \mid f_\theta(x)] - f_\theta(x)|\right]\\ &= \int_0^1 |\mathbb{E}\left[y - p \mid f_\theta(x) = p\right]|\, dF(p), \qquad \text{(where } F(p) \text{ is the law of } f_\theta(x))\end{aligned}$$

where all expectations are taken w.r.t. $(x, y) \sim \mathcal{D}$.

To estimate this quantity from finite samples, we will discretize the interval $[0, 1]$. The *τ-binned calibration gap* of a classifier f_θ is:

$$\textsc{cgap}_\tau(f_\theta;\mathcal{D}) \triangleq \sum_{p\in\{0,\tau,2\tau,\ldots,1\}} \left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} [\mathbb{1}\{f_\theta(x) \in (p, p+\tau)\} \cdot (y - p)] \right|$$

So that $\textsc{cgap}_\tau \to \textsc{cgap}$ as the bin width $\tau \to 0$. The empirical version of this quantity, for train set S, is:

$$\textsc{cgap}_\tau(f_\theta;S) \triangleq \sum_{p\in\{0,\tau,2\tau,\ldots,1\}} \left| \mathop{\mathbb{E}}_{x,y\sim S} [\mathbb{1}\{f_\theta(x) \in (p, p+\tau)\} \cdot (y - p)]\,. \right|$$

Next, we show strong DG implies that the expected binned calibration gap is similar between train and test.

Theorem C.8. *If the training method θ satisfies δ-strong DG, then*

$$\left| \mathop{\mathbb{E}}_{\theta,S\sim\mathcal{D}^n} \textsc{cgap}_\tau(f_{\theta(S)};S) - \textsc{cgap}_\tau(f_{\theta(S)};\mathcal{D}) \right| \le \frac{\delta}{\tau}. \tag{16}$$

Proof. First, let us define the family of tests:

$$\phi_p(x, y; \theta) \triangleq \mathbb{1}\{f_\theta(x) \in (p, p+\tau)\} \cdot (y - p)$$

The assumed δ-strong DG implies that each of these tests ϕ_p have similar expectations between train and test distributions:

$$\forall p: \quad \left| \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n;(x,y)\sim S} [\phi_p(x, y; \theta(S))] - \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n;(x,y)\sim\mathcal{D}} [\phi_p(x, y; \theta(S))] \right| \le \delta$$

Thus, the difference in expected calibration gaps between train and test is:

$$
\begin{aligned}
&|\mathop{\mathbb{E}}_{S\sim\mathcal{D}^n}[\textsc{cgap}_\tau(f_\theta;S) - \textsc{cgap}_\tau(f_\theta;\mathcal{D})]| \\
&= \left| \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n} \sum_{p\in\{0,\tau,2\tau,\ldots,1\}} \left| \mathop{\mathbb{E}}_{x,y\sim S}[\phi_p(x,y;\theta)] \right| - \left| \mathop{\mathbb{E}}_{x,y\sim\mathcal{D}}[\phi_p(x,y;\theta)] \right| \right| \\
&\leq \sum_{p\in\{0,\tau,2\tau,\ldots,1\}} \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n} \left| \mathop{\mathbb{E}}_{x,y\sim S}[\phi_p(x,y;\theta) - \mathop{\mathbb{E}}_{x,y\sim\mathcal{D}}[\phi_p(x,y;\theta)] \right| \\
&\leq \frac{\delta}{\tau},
\end{aligned}
$$

where the last inequality follows from the definition of δ-strong DG applied to functions ϕ_p. □

D 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.

E Additional Experiment Details

E.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

All algorithms are implemented in `PyTorch`[1] [Paszke et al., 2019]. For DP-related utilities, we use `opacus`[2] [Yousefpour et al., 2021]. Other packages, including `numpy` [3] [Harris et al., 2020], `scipy` [4] [Virtanen et al., 2020], `tqdm` [5], and `pandas` [6] [pandas development team, 2020], are also used. For gDRO [Sagawa et al., 2019], we use the implementation from `wilds` [Koh et al., 2021]. 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. [2021]. 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. [2019]. Tables 2 to 6 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

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

Table 7: 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

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 [Sagawa et al., 2019], we use the ImageNet-1k pretrained ResNet50 [He et al., 2016] from `torchvision` for CelebA, UTKFace, and iNat, and use the pre-trained BERT-Base [Devlin et al., 2019] from `huggingface` [Wolf et al., 2019] for CivilComments and MNLI.

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 [Loshchilov and Hutter, 2019] optimizer. We use `opacus`'s [Yousefpour et al., 2021] implementation on 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.

E.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:

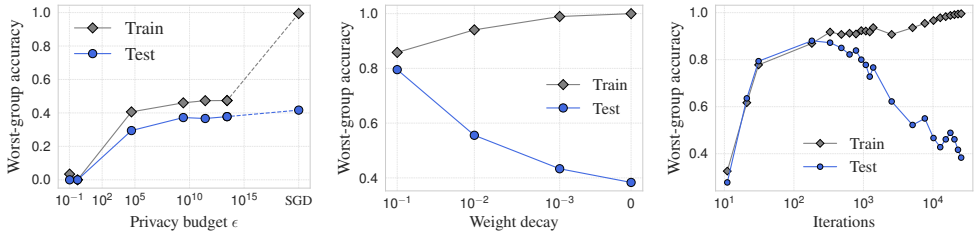
Definition E.1. The on-average generalization gap of the worst-case accuracy is defined as the following difference:

$$\text{WGGAP} \triangleq \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n}\left[\max_{g\in\mathcal{G}} \mathop{\mathbb{E}}_{z\sim S_g}[\ell(z,\theta(S))] \;\middle|\; |S_g|>0\right] - \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n}\left[\max_{g\in\mathcal{G}} \mathop{\mathbb{E}}_{z\sim\mathcal{D}_g}[\ell(z,\theta(S))]\right], \tag{17}$$

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 Equation (17)) 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:

$$\text{WGGAP} = \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n\\ z\sim S_{g^*}}}[\ell(z,\theta(S)) \mid |S_{g^*}|>0] - \mathop{\mathbb{E}}_{\substack{S\sim\mathcal{D}^n\\ z\sim\mathcal{D}_{g^*}}}[\ell(z,\theta(S))], \tag{18}$$



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

Figure 6: **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.

where $g^* \in \mathcal{G}$ is the worst-performing group. In Appendix C.1 we show that this simplified gap from Equation (18) is bounded by TV stability.

Therefore, in practice the generalization gap in Equation (17) offers a lower bound on the DG gap in Equation (1). Using it as a proxy for DG gap follows the spirit of the estimation approach by Nakkiran and Bansal [2020] 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* [Shokri et al., 2017], because DG and the accuracy of such attacks are equivalent, as showed by Kulynych et al. [2022]. 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 [Jayaraman and Evans, 2019, Liu et al., 2021b]. This body of evidence from the field of membership inference offers an alternative source of empirical support for our claims checked in Section 5.1.

E.3 Additional Details for Section 5.1

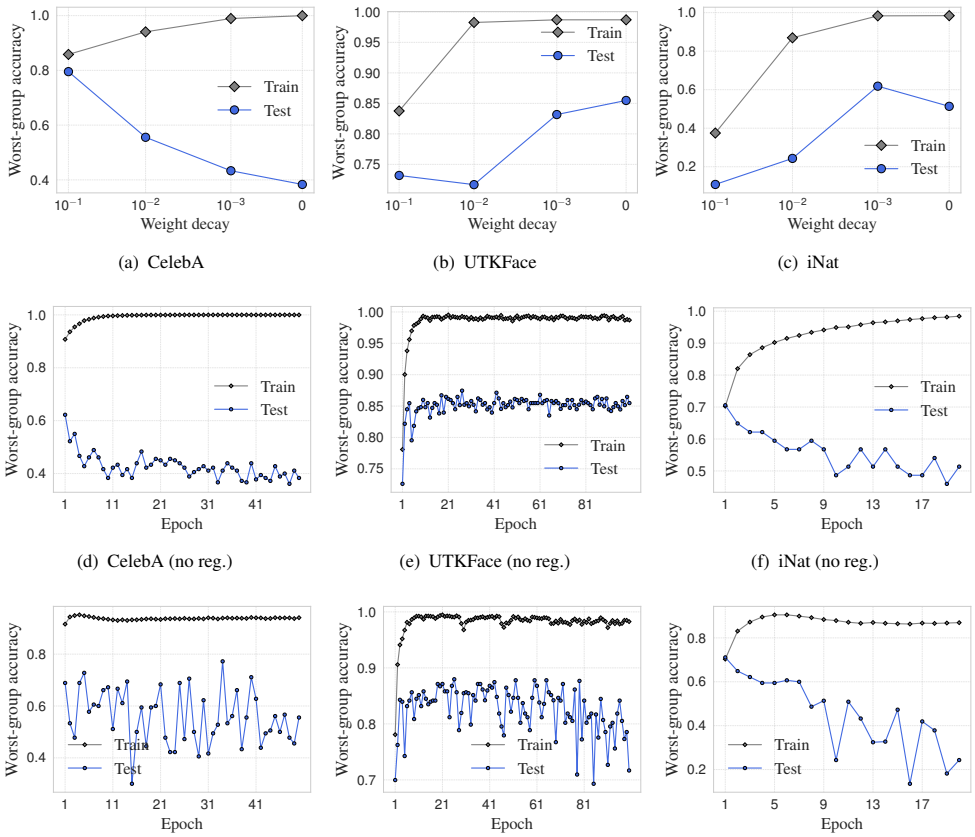
As mentioned in Section 4, many regularization methods can be used to improve different generalization gaps. For example, Sagawa et al. [2019] show that strong ℓ_2 regularization helps with improving group-distributional generalization, and Yang et al. [2020] show that dropout helps with adversarial-robustness 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 Figure 6, we train a neural network on CelebA using DP-SGD, and decrease the "regularization strength" in several different ways: by increasing privacy budget ϵ (Figure 6a), decreasing the ℓ_2 regularization (Figure 6b), or increasing the number of training iterations (Figure 6c).[7] We then measure the gap in worst-group accuracy on train vs. test (Appendix E.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 Figure 7, 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.

[7] Train time can be considered a regularizer, as its decrease induces stability (e.g. Hardt et al. [2016]).



(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 7: 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.

E.4 Additional Details for Section 5.2

Figure 8 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 possibly 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 $52\times$.

E.5 Additional Details for Section 5.3

We compare different algorithms, including SGD-ℓ_2 and IW-SGD-ℓ_2 as baselines, and two other algorithms, IS-SGD-ℓ_2 [Idrissi et al., 2021] and gDRO-ℓ_2 [Sagawa et al., 2019] 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,

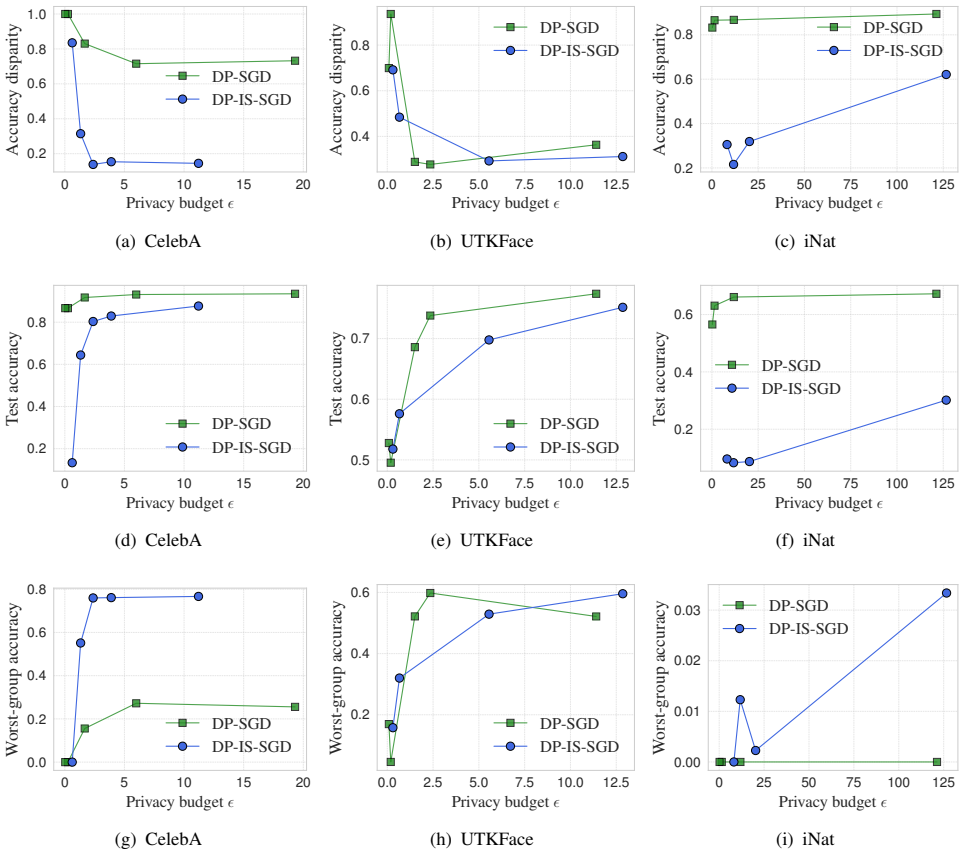


Figure 8: The disparity (lower the better) and test accuracies of the models trained with DP-SGD and IW-SGD on three datasets. If we care about privacy, DP-IS-SGD improves disparate impact at most privacy budgets. For CelebA, we train the model for 30 epochs. For UTKFace, we train for 100 epochs. For iNat, we train for 20 epochs. The GDP accountant is used to compute the privacy budget.

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

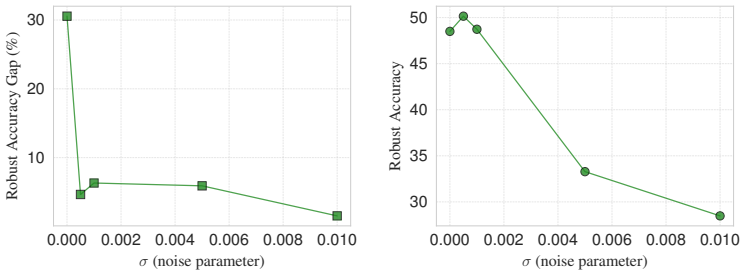
3. For IW-SGD-n, IS-SGD-n, and gDRO-n, we select the standard deviation of the random noise from 0.001, 0.01, 0.1, and 1.0 on CelebA, UTKFace, and iNat, and we select standard deviation of the random noise from 0.00001, 0.0001, and 0.001 on CivilComments and MNLI.

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[8], but this is not done in prior work.

E.6 Additional Experiments with Other Properties

As we detail in Section 2.1, a DG-inducing optimizer can ensure generalization of different types of properties. The focus of our applications in Section 3 is the subpopulation-performance property. As a proof of concept that our WYSIWYG framework is practical for other properties, we also

[8]For example, we do not tune the "group adjustments" parameter for gDRO, using the default from Koh et al. [2021] instead.



(a) (b)

Figure 9: **Noisy gradient reduces overfitting in adversarial training.** We show the generalization gap of adversarial accuracy (left), and adversarial accuracy on the test set (right) of adversarially trained models with different levels of gradient noise. The model trained without gradient noise exhibits "robust overfitting" (i.e., the gap between robust training accuracy and robust test accuracy is large) of about 30 p.p. Gradient noise reduces the generalization gap by more than 3× for all values of the noise parameter that we tried. This comes at a cost of decreased average adversarial accuracy as the noise gets larger.

apply it to ensure generalization of adversarial robustness, performance under test-time adversarial perturbations [see, e.g., Madry et al., 2018].

For images, a standard approach to modeling the adversary is assuming they attack the target model θ by feeding it *adversarial examples* which maximize the loss of the target model, yet are similar to some natural examples $(x, y) \sim \mathcal{D}$ in L_p distance:

$$\max_{\|\Delta x\|_p \leq \gamma} \ell(x + \Delta x; \theta).$$

A common way to train models that are adversarially robust in this sense is *adversarial training* [Madry et al., 2018]. In adversarial training, we want to train models that minimize the following objective, called *adversarial loss*:

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \max_{\|\Delta x\|_p \leq \gamma} \left[\ell(x + \Delta x; \theta)\right]. \tag{19}$$

Generalization of adversarial loss is an issue, as adversarially trained models are known to exhibit "robust overfitting": higher generalization gap of adversarial loss than that of the regular loss [Rice et al., 2020].

As in Section 4.2, inspired by our theoretical results, we apply the noisy-gradient regularizer to the standard adversarial training [Madry et al., 2018] which uses the projected gradient descent algorithm to solve the inner maximization in Equation (19). As with DRO, we expect this to induce stability, and ensure empirical distributional generalization, thus to reduce overfitting under adversarial perturbations.

Experimental Setup. We use the CIFAR-10 dataset [Krizhevsky et al., 2009], and ResNet-18 [He et al., 2016] 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.

Results. We summarize the results in Figure 9. The left plot displays the gap in robust accuracy, and the right plot displays the robust test accuracy. As shown in Figure 9 (a), without gradient noise, in standard adversarial training the gap between robust training accuracy and robust test accuracy is large at approximately 30 p.p., consistent with the prior observations of "robust overfitting" [Rice et al., 2020]. By injecting noise into the gradient, our proposed approach decreases the adversarial accuracy generalization gap by more than 3× to less than 10 p.p. More surprisingly, in our experiments, training with gradient noise achieves both a low adversarial accuracy gap and better robust accuracy compared to standard adversarial training, when using a low level of noise ($\sigma = 0.0005$). These experimental results further demonstrate how DG can be a useful design principle in practice.