On the Benefits of Invariance in Neural Networks

Clare Lyle [1] **Mark van der Wilk** [2] **Marta Kwiatkowska** [1] **Yarin Gal** [1] **Benjamin Bloem-Reddy** [3]

Abstract

Many real world data analysis problems exhibit invariant structure, and models that take advantage of this structure have shown impressive empirical performance, particularly in deep learning. While the literature contains a variety of methods to incorporate invariance into models, theoretical understanding is poor and there is no way to assess when one method should be preferred over another. In this work, we analyze the benefits and limitations of two widely used approaches in deep learning in the presence of invariance: data augmentation and feature averaging. We prove that training with data augmentation leads to better estimates of risk and gradients thereof, and we provide a PAC-Bayes generalization bound for models trained with data augmentation. We also show that compared to data augmentation, feature averaging reduces generalization error when used with convex losses, and tightens PAC-Bayes bounds. We provide empirical support of these theoretical results, including a demonstration of why generalization may not improve by training with data augmentation: the 'learned invariance' fails outside of the training distribution.

1. Introduction

Many real-world problems exhibit invariant structure. Tasks involving set-valued inputs such as point clouds are invariant to permutation. Image classification tasks are often rotation- and translation-invariant. Intuitively, models that capture the invariance of a problem should perform better than those that do not. This is supported by empirical results in a range of applications (Cohen & Welling, 2016; Fawzi et al., 2016; Salamon & Bello, 2017).

There are many ways of incorporating invariance into a model. One can build the invariance into the network as a convolution or weight-tying scheme, or average network predictions over transformations of the input (feature averaging), or simply train on a dataset augmented with these transformations (data augmentation). Each of these approaches has been demonstrated to perform well in various settings, but there remains a large divide between their impressive practical performance and solid theoretical understanding.

The lack of theory leaves open a number of questions. Firstly, if invariance is incorporated into a model or training algorithm, what are the theoretical guarantees on the performance of the trained model? Relatedly, as a matter of practice, how should a practitioner choose amongst the different approaches to incorporating invariance? Concretely, if an invariant model and a model trained with data augmentation both attain the same training error, which one should be preferred? Can one or the other be expected to converge faster? These questions are the key motivation for our work.

We focus the two most generically applicable methods, data augmentation and feature averaging. Our overall conclusion is that **feature averaging is better than data augmentation is better than doing nothing**; this holds even for stochastic (Monte Carlo) approximations of the averages involved in feature averaging and data augmentation. On the journey to the main conclusion, we uncover a number of intriguing properties and shed light on the mathematical structure driving the impressive practical performance of the methods.

1.1. Summary of Results

We consider the data-generating distribution $P_{\mathcal{D}}$ to be invariant to the action of a group $\mathcal{G}$: $P_{\mathcal{D}}(gX, Y) = P_{\mathcal{D}}(X, Y)$, for all $g \in \mathcal{G}$ (see Section 2 for details). Our main results relate baseline training of a generic neural network (or other predictive model) via empirical risk minimization (ERM) to performing either data augmentation or feature averaging. Table 1 summarizes the theoretical results.

Data augmentation (DA) (Section 3) improves on baseline training with ERM by minimizing an augmented risk, the risk averaged over the orbits induced by $\mathcal{G}$. This yields a lower-variance estimator of the model risk and its minima (Proposition 2). The variance reduction also applies to gradients of the risk, and therefore affects gradient-based learning. Our results to this end are essentially the same

[1]Department of Computer Science, University of Oxford, Oxford, United Kingdom [2]Department of Computing, Imperial College London, London, United Kingdom [3]Department of Statistics, University of British Columbia, Vancouver, Canada. Correspondence to: Clare Lyle <clare.lyle@univ.ox.ac.uk>.

Table 1. Summary of theoretical results.

	Baseline		**Data Augmentation**		**Feature Averaging**
Expected risk	$R_\ell(f)$	$=$ Proposition 2	$R_\ell(f)$	$\overset{\text{convex } \ell}{\geq}$ Proposition 5	$R_\ell(f^\circ)$
Empirical risk	$\widehat{R}_\ell(f, \mathcal{D}^n)$		$\widehat{R}^\circ_\ell(f, \mathcal{D}^n)$	$\overset{\text{convex } \ell}{\geq}$ Proposition 5	$\widehat{R}^\circ_\ell(f^\circ, \mathcal{D}^n) = \widehat{R}_\ell(f^\circ, \mathcal{D}^n)$
Variance of $\widehat{R}_\ell$	$\underset{\mathcal{D}^n \sim P_\mathcal{D}}{\text{Var}}\,[\widehat{R}_\ell(f, \mathcal{D}^n)]$	$\geq$ Proposition 2	$\underset{\mathcal{D}^n \sim P_\mathcal{D}}{\text{Var}}\,[\widehat{R}^\circ_\ell(f, \mathcal{D}^n)]$	$\overset{\text{convex } \ell}{\geq}$ Proposition 5	$\underset{\mathcal{D}^n \sim P_\mathcal{D}}{\text{Var}}\,[\widehat{R}_\ell(f^\circ, \mathcal{D}^n)]$
KL term in PAC-Bayes bound	$\text{KL}(Q \,\|\, P)$	$=$	$\text{KL}(Q \,\|\, P)$	$\geq$ Theorem 7	$\text{KL}(Q^\circ \,\|\, P^\circ)$
PAC-Bayes bound for 0-1 loss	B_0	$=$ Theorem 4	B_{DA}	$\geq$ Theorem 9	B_{FA}
Monte Carlo approx. ($k \geq 1$ samples)			PAC-Bayes bound holds Theorem 4		$\text{KL}(Q \,\|\, P) \geq \text{KL}(\widehat{Q}_{G^k} \,\|\, \widehat{P}_{G^k}) \geq \text{KL}(Q^\circ \,\|\, P^\circ)$

as some by Chen et al. (2019). In contrast to that work, we investigate PAC-Bayes bounds for generalization of DA. Traditional PAC-Bayes bounds based on i.i.d. data do not apply to DA because the augmented dataset violates the i.i.d. assumption. We show that the i.i.d. bounds also apply to DA and in particular to the augmented risk (Theorem 4), and that tighter bounds may be possible. However, training with DA is not guaranteed to produce an invariant (or even approximately invariant) function. We demonstrate empirically how this can fail; we also provide an example where minimizing the augmented risk yields an invariant function (Section 3.2).

Feature averaging (FA) (Section 4) yields a lower-entropy function class. In the case of convex loss, FA also obtains lower expected risk than DA and lower-variance estimates of risk and its gradient (Proposition 5). Furthermore, symmetrization compresses the model, and thus tightens PAC-Bayes bounds by reducing the KL term, a phenomenon that holds even for Monte Carlo approximations to FA (Theorem 7 and Proposition 8). As a byproduct, we prove a general lemma (Lemma 6) that connects the present paper to work on generalization and post-training compression (Zhou et al., 2019).

We illustrate our theoretical results with experiments in Section 6, and also investigate practical questions raised by the theory. We conclude (Section 7) by interpreting the theory and experiments as practical recommendations.

2. Background

"Invariance" has been used to describe a number of related but distinct phenomena in the machine learning and statistics literature. One perspective, which is shared by the present work, considers invariance of a neural network's output with respect to a group acting on its inputs (e.g., Cohen & Welling, 2016; Kondor & Trivedi, 2018; Bloem-Reddy & Teh).[1] Other work has used looser notions. For example, Zou et al. (2012) use "invariant" to mean "not changing very much". Related ideas are "local invariance" (Raj et al., 2017), "insensitivity" (van der Wilk et al., 2018), and "approximate invariance" (Chen et al., 2019, Sec. 6).

We focus on invariance under the action of a group $\mathcal{G}$. The action of $\mathcal{G}$ on a set $\mathcal{X}$ is a mapping $\alpha : \mathcal{G} \times \mathcal{X} \to \mathcal{X}$ which is compatible with the group operation. For convenience, we write $\alpha(g, x) = \alpha_x(g) = gx$, for $g \in \mathcal{G}$ and $x \in \mathcal{X}$. The **orbit** of any $x \in \mathcal{X}$ is the subset $\mathcal{G}_x$ of $\mathcal{X}$ that can be obtained by applying an element of $\mathcal{G}$ to x, $\mathcal{G}_x = \{gx : g \in \mathcal{G}\}$. For mathematical simplicity, we assume $\mathcal{G}$ to be compact, with (unique) normalized Haar measure denoted by λ.[2] We denote a random element of $\mathcal{G}$ by G. A mapping $f : \mathcal{X} \to \mathcal{Y}$ is **invariant** under $\mathcal{G}$ (or $\mathcal{G}$-invariant) if

$$f(gx) = f(x)\,, \quad g \in \mathcal{G},\ x \in \mathcal{X}\,. \tag{1}$$

Any function $f : \mathcal{X} \to \mathbb{R}$ can be **symmetrized** by averaging over $\mathcal{G}$. We denote this with a symmetrization operator $S_\mathcal{G}$, defined as

$$f^\circ(x) := S_\mathcal{G} f(x) = \mathbb{E}_{G \sim \lambda}[f(Gx)]\,, \quad x \in \mathcal{X}\,. \tag{2}$$

We consider a typical machine learning scenario, with a training data set $\mathcal{D}^n$ of n observations $(X_i, Y_i)_{i=1}^n \in (\mathcal{X}, \mathcal{Y})^n$ sampled i.i.d. from some (unknown) probability distribution $P_\mathcal{D}$. Furthermore, $P_\mathcal{D}$ *is known or assumed to be $\mathcal{G}$-invariant*,

$$P_\mathcal{D}(gX, Y) = P_\mathcal{D}(X, Y)\,, \quad g \in \mathcal{G}\,. \tag{3}$$

[1] These ideas (and the results in the present work) apply generally to functions, and therefore to a broader set of machine learning techniques; we focus on neural networks for continuity with the previous literature.

[2] λ is analogous to the uniform distribution on $\mathcal{G}$. Our results generalize—with some additional technicalities—to any group that acts property on $\mathcal{X}$ and has Haar measure.

For example, X may be an image of an animal, Y a label of the animal, and $\mathcal{G}$ the group of two-dimensional rotations.

The marginal distribution on X of any $\mathcal{G}$-invariant $P_{\mathcal{D}}$ has a disintegration into a distribution P_Φ over orbits of $\mathcal{X}$, each endowed with an **orbit representative** $\Phi \in \mathcal{X}$, and a conditional distribution $P_{X|\Phi} = \lambda \circ \alpha_\Phi^{-1}(\bullet)$ induced by applying a random $G \sim \lambda$ to Φ (see, e.g., Bloem-Reddy & Teh). That is, $(X, Y) \overset{\mathrm{d}}{=} (G\Phi, Y)$ and $P_{\mathcal{D}} = P_\Phi \times P_{X|\Phi} \times P_{Y|X}$. The specific relevance to this work is that expectations with respect to $P_{\mathcal{D}}$ can be iterated as

$$\begin{aligned}&\mathbb{E}_{(X,Y)\sim P_{\mathcal{D}}}[f(X,Y)] \\ &\quad = \mathbb{E}_{Y\sim P_{Y|X}}[\mathbb{E}_{\Phi\sim P_\Phi}[\mathbb{E}_{G\sim\lambda}[f(G\Phi,Y) \mid \Phi, Y] \mid Y]] \,.\end{aligned} \tag{4}$$

For a class of functions $F = \{f : \mathcal{X} \to \mathcal{Y}\}$, a probability distribution Q on F, and a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$. We denote various expected and empirical risks as follows:

$$\begin{aligned} R_\ell(f) &= \mathbb{E}_{(X,Y)\sim P_{\mathcal{D}}}[\ell(f(X), Y)] \\ R_\ell(Q) &= \mathbb{E}_{f\sim Q}[R_\ell(f)] \\ \widehat{R}_\ell(f, \mathcal{D}^n) &= \tfrac{1}{n}\textstyle\sum_{i=1}^n \ell(f(X_i), Y_i) \\ \widehat{R}_\ell(Q, \mathcal{D}^n) &= \mathbb{E}_{f\sim Q}[\widehat{R}_\ell(f, \mathcal{D}^n)] \end{aligned}$$

2.1. Modes of Invariance

Common sense indicates that when modeling $\mathcal{G}$-invariant $P_{\mathcal{D}}$, any good model will also be $\mathcal{G}$-invariant, at least to a good approximation. This has been achieved in practice through one of three approaches: trained invariance, encouraged during training via DA; network symmetrization, typically implemented as FA; and symmetric network design, obtained by composing a $\mathcal{G}$-invariant layer with a sequence of $\mathcal{G}$-equivariant layers.

Trained invariance is implemented as DA (Fawzi et al., 2016; Cubuk et al., 2018): (possibly random) elements G_{ij} of $\mathcal{G}$ are applied to each observation X_i of the training data, with the label Y_i left unchanged. The result is an augmented dataset $\mathcal{D}^n_{\mathcal{G}} = ((G_{ij}X_i, Y_i)_{j\le m})_{i\le n}$ used to minimize the augmented empirical risk

$$\begin{aligned}\widehat{R}^\circ_\ell(f, \mathcal{D}^n) &= \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{G\sim\lambda}[\ell(f(GX_i), Y_i)] \\ &\approx \frac{1}{nm}\sum_{i=1}^n\sum_{j=1}^m \ell(f(G_{ij}X_i), Y_i) \,.\end{aligned} \tag{5}$$

DA is now a standard method in practitioners' toolkit (Iyyer et al., 2014; Zhou & Troyanskaya, 2015; Salamon & Bello, 2017; Zhao et al., 2018), particularly due to its ease-of-implementation and flexibility: $\mathcal{G}$ may be a set of transformations that is not a group, which permits its use for encouraging exact or approximate invariance under an arbitrary set of transformations. Networks trained with DA have been observed to exhibit greater invariance to the desired transformations than those trained on the original dataset (Fawzi et al., 2016) despite the fact that invariance is not part of the built-in network architecture. Moreover, it can have positive effects on generalization even when the augmentation transformations are not present in the test set (Zhang et al., 2017).

Theoretical understanding of DA is still being developed. Recent theoretical work has established connections to FA and variance reduction methods. Specifically, Dao et al. (2019) showed that for a kernel linear classifier, minimizing the augmented risk is equivalent, to first order, to minimizing the feature averaged risk; and that a second-order approximation to the objective is equivalent to data-dependent variance regularization. Chen et al. (2019) showed that averaging over the set of transformations is a form of Rao–Blackwellization, and the resulting variance-reduction yields a number of desirable theoretical statistical properties.

Architectural invariance restricts the function class being learned to contain only invariant functions, typically through either FA or symmetric function composition. FA relies on computing an average over $\mathcal{G}$ at one or more layers, such that the overall network is invariant under $\mathcal{G}$ acting on the input. In practice, averaging is typically done at the penultimate or final layer, resulting in a $\mathcal{G}$-invariant network f°. A network f with D layers is written as the composition of $h_D \circ \cdots \circ h_1$, with the shorthand $h_d^{d'}$ referring to the composition of layers d through d'. The empirical risk of a network with FA at layer d evaluated on $\mathcal{D}^n$ is

$$\widehat{R}_\ell(f^\circ, \mathcal{D}^n) = \frac{1}{n}\sum_{i=1}^n \ell\big(h_d^D \circ \mathbb{E}_{G\sim\lambda}[h_1^{d-1}(GX_i)], Y_i\big) \,.$$

As with DA, FA can be applied to approximate and non-group invariance. The average over $\mathcal{G}$ might also be estimated by applying randomly sampled elements of $\mathcal{G}$, though when h_d^D is nonlinear the estimate of f° may be biased. Unlike DA, symmetrization guarantees that the output function f° will be invariant to $\mathcal{G}$ whenever the expectation over $\mathcal{G}$ can be computed exactly. The exact computation of this expectation, however, can be computationally expensive (linear in $|\mathcal{G}|$ when discrete) or even intractable (when $\mathcal{G}$ is infinite), in which case Monte Carlo estimates can be used.

The elegant, albeit less generically applicable approach of symmetric function composition uses properties of $\mathcal{G}$ to determine particular functional forms that are equivariant or invariant under $\mathcal{G}$. An invariant network f° is constructed by composing an invariant function (layer) h° with a sequence of equivariant functions (h^{e}_k): $f^\circ = h^\circ \circ h^{\mathrm{e}}_D \circ \cdots \circ h^{\mathrm{e}}_1$. A body of literature of varying degrees of generality has developed (Wood & Shawe-Taylor, 1996; Ravanbakhsh et al., 2017; Kondor & Trivedi, 2018; Bloem-Reddy & Teh; Cohen

et al., 2019). This includes convolutional networks. Empirical results indicate that this approach has advantages over trained invariance (e.g., Cohen & Welling, 2016). Theoretical results to this end are lacking, with the notable exception of the VC-dimension-based PAC bounds obtained by Shawe-Taylor (1991; 1995), which connect a tighter generalization bound to the reduction in parameters that results from symmetry constraints. We do not consider equivariant-invariant architectures further, and leave their theory as future work.

2.2. PAC-Bayes Generalizations Bounds

Understanding the generalization performance of deep learning models is a core research objective of modern machine learning. Many empirical results appear counterintuitive, and remain largely unexplained by theory. Networks with many more parameters than observations may generalize well, despite also having the capacity to memorize the training set (Zhang et al., 2017). Uniform generalization bounds often result in *vacuous* bounds, i.e., they are greater than the upper bound of the loss function (Dziugaite & Roy, 2017). However, PAC-Bayes bounds (McAllester, 1999) have been successfully applied to large deep network architectures to obtain nonvacuous generalization guarantees (Dziugaite & Roy, 2017; 2018; Zhou et al., 2019).

PAC-Bayes bounds characterize the risk of a randomized prediction rule; the randomizaton is interpreted as a Bayesian posterior distribution Q that can depend on $\mathcal{D}^n$. The typical bound on generalization error is expressed in terms of the empirical risk and the KL divergence between Q a fixed prior distribution P. The following is a standard bound due to Catoni (2007), which holds for general data generating distributions $P_{\mathcal{D}}$ and 0-1 loss.

Theorem 1 (Catoni (2007))**.** *Let $\mathcal{D}^n$ be sampled i.i.d. from $P_{\mathcal{D}}$, and let ℓ be 0-1 loss. For any prior P and any $\delta \in (0,1)$, with probability $1-\delta$ over samples $\mathcal{D}^n$, for all posteriors Q and for all $\beta > 0$,*

$$R_\ell(Q) \leq \frac{1 - e^{-\beta \widehat{R}_\ell(Q,\mathcal{D}^n) - \frac{1}{n}(\mathrm{KL}(Q \,||\, P) + \log \frac{1}{\delta})}}{1 - e^{-\beta}} \,. \quad (6)$$

For bounded loss functions, analogous bounds are in terms of the so-called KL generalization error (see, e.g., Dziugaite & Roy, 2017; 2018). We state all results only for variations of Catoni's bound (6), but versions for KL generalization error are straightforward to derive.

3. Data Augmentation Reduces Variance

In this section, we discuss the ways in which DA performs better than baseline ERM, and establish the validity of a PAC-Bayes bound for models trained with DA.

Recently, Chen et al. (2019) established that when $P_{\mathcal{D}}$ is $\mathcal{G}$-invariant, DA reduces the variance of ERM-based estimators by (approximately) averaging loss over the orbits of the observations, which can be seen as a form of Rao–Blackwellization. Specifically, for any integrable $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, symmetrizing is equivalent to taking the conditional expectation, conditioned on the orbit of X:

$$\mathbb{E}_{G\sim\lambda}[h(GX, Y)] = \mathbb{E}_{(X,Y)\sim P_{\mathcal{D}}}[h(X, Y) \mid \Phi(X)] \,.$$

The average of $\mathcal{G}$ appears in the augmented empirical risk (5), and reduces the variance of risk estimates. Specifically, the variance of the risk decomposes into within-orbit and across-orbit terms, and the within-orbit term vanishes for the augmented risk. The result follows directly from Chen et al. (2019, Lemma 4.1); we also give a proof in Appendix A that highlights the structure of the problem.

Proposition 2 (Chen et al. (2019))**.** *If $P_{\mathcal{D}}$ is $\mathcal{G}$-invariant and $\ell(f(\bullet), \bullet) \in L_2(P_{\mathcal{D}})$, then*

$$\mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}^n}\left[\widehat{R}_\ell^\circ(f, \mathcal{D}^n)\right] = \mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}^n}\left[\widehat{R}_\ell(f, \mathcal{D}^n)\right] \,, \text{ and}$$
$$\mathrm{Var}_{\mathcal{D}^n\sim P_{\mathcal{D}}^n}\left[\widehat{R}_\ell^\circ(f, \mathcal{D}^n)\right] \leq \mathrm{Var}_{\mathcal{D}^n\sim P_{\mathcal{D}}^n}\left[\widehat{R}_\ell(f, \mathcal{D}^n)\right] \,.$$

3.1. Practical Data Augmentation

Computing $\mathbb{E}_{G\sim\lambda}[\ell(f(GX), Y)]$ exactly may be infeasible: $\mathcal{G}$ may be discrete but large, or $\mathcal{G}$ may be continuous. In either case, practical DA relies on Monte Carlo estimates, typically within stochastic gradient descent (SGD). Specifically, with $G_{ij} \sim \lambda$, $\widehat{R}_\ell^\circ$ is approximated by

$$\widehat{\widehat{R}}_\ell^\circ(f, \mathcal{D}^n) := \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} \ell(f(G_{ij}X_i), Y_i) \,. \quad (7)$$

For "nice" loss functions—those for which we can interchange differentiation and $\mathbb{E}_{G\sim\lambda}$—the symmetrization reduces the variance of gradient estimates of augmented risk. Conversely, the variance of the Monte Carlo estimate of (7) may offset the reduction obtained from averaging. Furthermore, it has been argued that the noise in SGD implicitly regularizes the objective (Neyshabur, 2017); excessive variance reduction may have harmful effects. In short, the consequences of the interplay between the variance reduction of symmetrization and the variance increase of approximating that symmetrization, especially in the context of SGD for overparameterized models, are not clear. The details of those trade-offs are beyond the scope of this paper; we briefly investigate the effects empirically in Section 6.

3.2. Data Augmentation and Trained Invariance

While DA is sometimes referred to as an approach to train an invariant function, the learned function will not be invariant in general. The objective of training with DA is to minimize a symmetrized risk, not to find a symmetric function.

One setting in which minimizing the augmented risk will yield an invariant function is with a linear model $f_w(X) =$

$w^\top X$ and a convex loss, and with $\mathcal{G}$ a group whose action on $\mathcal{X}$ has a linear representation. To state the result, let V be a d-dimensional vector space over $\mathbb{R}$ with dual vector space V^*, and assume that $\mathcal{X}$ spans V. Furthermore, let $\mathcal{G}$ admit a linear representation, $\rho : \mathcal{G} \to GL(V)$, with corresponding dual $\rho_g^* = \rho_{g^{-1}}^\top$.

Proposition 3. *Suppose that $\mathcal{G}$ has a linear representation, as described above, let $f_w(X) = w^\top X$ and ℓ be strictly convex. Then the (global) minimizer $\hat{w}$ satisfies $\rho_g^* \hat{w} = \hat{w}$ for λ-almost all $g \in \mathcal{G}$. In particular, $f_{\hat{w}}$ is $\mathcal{G}$-invariant.*

Under suitable step-size conditions (e.g., the Robbins–Munro conditions) SGD will converge to an invariant set of weights. Thus, to learn a predictor that exhibits the desired invariance on the entire dataset, it is sufficient to train with SGD on augmented data with a convex loss.

Non-convex objectives with non-linear models do not yield similar results. As most settings for which we would use deep learning are both non-convex and non-linear, this suggests that although DA may appear to promote invariance, it may fail to learn networks that are truly invariant.

The example depicted in Fig. 1 demonstrates such a failure: the learned function appears to capture the target invariance on the training data, but, having not learned the appropriate symmetry in weight space, fails to generalize to novel data and displays high variance over orbits in evaluation. We train fully connected neural networks using DA on one of two related datasets: MNIST and fashionMNIST (28×28 pixel black and white images of handwritten digits and clothing categories respectively), each augmented by rotations of multiples of 90 degrees. We then evaluate the variance of the outputs over orbits (rotations by 90, 180, and 270 degrees) in the test set. Finally, we evaluate the two networks on orbits in the complementary dataset.

We observe that the networks attain low variance over orbits of data drawn from the same distribution as the training data. The performance of the networks on out-of-distribution data is more interesting. The MNIST network has increasingly *higher* variance of its predictions on the rotations of fashionMNIST as it reduces its prediction variance over orbits of MNIST. We also note that the variance between random seeds in the variances over orbits was significantly higher on the out-of-distribution data. We omit data for FA because averaging over each of the four rotations of the input trivially yields a variance of zero over each orbit.

3.3. PAC-Bayes Generalization Bounds

The PAC-Bayes bound in Theorem 1 holds with binary classification loss. Exact DA violates the assumptions because with the same loss, $\mathbb{E}_{G\sim\lambda}[\ell(f(GX), Y)] \in [0, 1]$. Monte Carlo approximations of $\mathbb{E}_{G\sim\lambda}[\ell(f(GX), Y)]$ also violate the assumptions of Theorem 1 because the augmented data

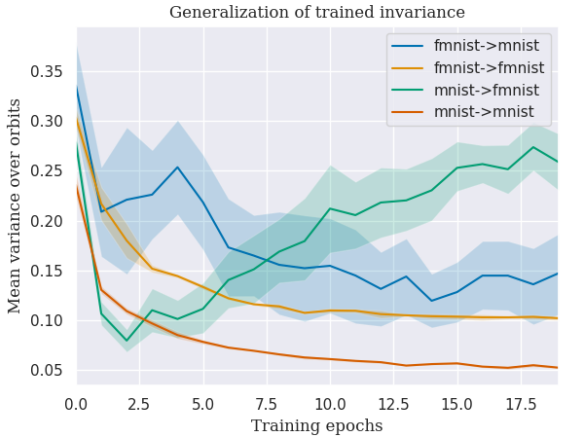

Figure 1. Variance of predictions w.r.t. rotations of input over the course of training. Labels indicate training → evaluation set.

set is not i.i.d. We address both issues with the following PAC-Bayes bound for DA.

Theorem 4. *Assume that $P_\mathcal{D}$ is $\mathcal{G}$-invariant. Then Theorem 1 holds with either of $\widehat{R}_\ell^\circ(Q, \mathcal{D}^n)$ as in (5) or $\widehat{R}_\ell^{\widehat{\circ}}(Q, \mathcal{D}^n)$ as in (7) substituted for $\widehat{R}_\ell(Q, \mathcal{D}^n)$.*

See Appendix A for the proof, which uses a general formula of Lever et al. (2013) and the invariance structure of $P_\mathcal{D}$. The bound (6) is looser than what is theoretically possible for DA. However, tighter bounds with an analytic form (see Appendix B.3) are computationally intractable.

4. Feature Averaging Can Do More

In this section we establish that FA should be preferred over DA in most situations. When the loss is convex, generalization error decreases both in expectation and per-dataset, and there is a further variance-reduction in risk estimates. More importantly, symmetrization compresses the model, resulting in a **symmetrization gap** in the PAC-Bayes bound.

For a group $\mathcal{G}$ that acts on $\mathcal{X}$, symmetrization of any function $f : \mathcal{X} \to \mathbb{R}$ can be performed by averaging over $\mathcal{G}$, as in (2). Fix a function class F, and let F° denote the class of $\mathcal{G}$-invariant functions obtained by symmetrizing the functions belonging to F. Clearly, $S_\mathcal{G}$ is surjective, but it may not be injective. The inverse image of f°, $S_\mathcal{G}^{-1} f^\circ$, yields the set of functions in F whose $\mathcal{G}$-symmetrization yields f°. Function symmetrization is naturally extended to probability measures on function classes: for any probability measure P on F, the induced probability measure on F° is the image of P under $S_\mathcal{G}$, $P^\circ = P \circ S_\mathcal{G}^{-1}$.

4.1. Further Variance Reduction with Convex Loss

With convex loss, Jensen's inequality can be applied to the augmented risk to compare DA and FA risk estimates. The proof of the following proposition is given in Appendix A.2.

Proposition 5. *Let $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$ be a loss function that is convex in its first argument. Then for any $f : \mathcal{X} \to \mathcal{Y}$,*

$$\widehat{R}_\ell(f^\circ, \mathcal{D}^n) = \widehat{R}^\circ_\ell(f^\circ, \mathcal{D}^n) \leq \widehat{R}^\circ_\ell(f, \mathcal{D}^n) \,,$$

and therefore analogous inequalities hold for $\widehat{R}_\ell(Q^\circ, \mathcal{D}^n)$, $R_\ell(f)$, and $R_\ell(Q)$. Furthermore, if $\ell(f(\bullet), \bullet) \in L_2(P_\mathcal{D})$ (i.e., has finite second moment),

$$\mathrm{Var}_{\mathcal{D}^n \sim P^n_\mathcal{D}}\left[\widehat{R}_\ell(f^\circ, \mathcal{D}^n)\right] \leq \mathrm{Var}_{\mathcal{D}^n \sim P^n_\mathcal{D}}\left[\widehat{R}^\circ_\ell(f, \mathcal{D}^n)\right] \,.$$

4.2. Reduction in KL via the Symmetrization Gap

In modern deep learning architectures, one typically has sufficiently large capacity to drive the empirical risk arbitrarily close to zero. Although the variance-reduction of the previous section can help during training, the dominant term in the generalization bound (6) is $\mathrm{KL}(Q \,||\, P)$. Indeed, much of the recent literature on obtaining nonvacuous PAC-Bayes bounds focuses on minimizing this term, subject to not overly inflating the empirical risk.

Consider the approach of Zhou et al. (2019): train a deep neural network, and use a compression algorithm to obtain a lossy compression of the trained network. Countering the potential for deterioration in the empirical risk, the KL term applied to the compressed network achieves a massive reduction in entropy; the compressed network is much less complex. Those basic concepts apply more generally, as formalized by the following lemma. Although fundamental, we have been unable to find a published proof (though it would be surprising if one does not exist). We give the proof in Appendix A.3.

Lemma 6. *Suppose that $(E_i, \mathcal{E}_i)$, $i = 1, 2$, are two measurable spaces, the second of which is standard, μ and ν are two probability measures on $(E_1, \mathcal{E}_1)$, and $\psi : (E_1, \mathcal{E}_1) \to (E_2, \mathcal{E}_2)$ is a measurable map. Then*

$$\mathrm{KL}(\mu \circ \psi^{-1} \,||\, \nu \circ \psi^{-1}) \leq \mathrm{KL}(\mu \,||\, \nu) \,. \tag{8}$$

*Furthermore, if $\mu \ll \nu$ with density m, then $\mu \circ \psi^{-1} \ll \nu \circ \psi^{-1}$ with density m_ψ, and the ψ-**gap** is*

$$\begin{aligned}\Delta_\psi(\mu \,||\, \nu) :&= \mathrm{KL}(\mu \,||\, \nu) - \mathrm{KL}(\mu \circ \psi^{-1} \,||\, \nu \circ \psi^{-1}) \\ &= \int_{E_1} \mu(dx) \log \frac{m(x)}{(m_\psi \circ \psi)(x)} \,. \end{aligned} \tag{9}$$

In particular, when ψ is non-injective, points of $(E_1, \mathcal{E}_1)$ become equivalent; $(E_2, \mathcal{E}_2)$ is a compressed version, and the probability measures μ and ν are similarly compressed.

The symmetrization gap. Applying Lemma 6 with $\psi = S_\mathcal{G}$ indicates that symmetrization can reduce the KL divergence term in the PAC-Bayes bound.

Theorem 7. *Let $\mathcal{X}$ be a compact metric space and $\mathcal{Y}$ a Polish space, $\mathcal{G}$ a group acting measurably on $\mathcal{X}$, and $F = C(\mathcal{X}, \mathcal{Y})$ the class of continuous functions $\mathcal{X} \to \mathcal{Y}$.[3] Let Q and P be probability measures on F such that $Q \ll P$ with density q, and $Q^\circ \ll P^\circ$ (density q°) their images under $S_\mathcal{G}$ on F°. Then*

$$\mathrm{KL}(Q^\circ \,||\, P^\circ) \leq \mathrm{KL}(Q \,||\, P) \,.$$

*Furthermore, the **symmetrization gap** is*

$$\Delta^\circ(Q \,||\, P) = \mathbb{E}_{f \sim Q}\left[\log \frac{q(f)}{q^\circ(S_\mathcal{G} f)}\right] \,. \tag{10}$$

Because Q° is the image of Q, the densities in (10) satisfy

$$\int_{S_\mathcal{G}^{-1} B} q(f) P(df) = \int_{S_\mathcal{G}^{-1} B} q^\circ(S_\mathcal{G} f) P(df) \,, \tag{11}$$

for all sets B in the σ-algebra on F°. Although this imposes a large number of constraints on q and $q^\circ \circ S_\mathcal{G}$, they may differ greatly across F. In particular, consider a $\mathcal{G}$-induced equivalence class $S_\mathcal{G}^{-1} S_\mathcal{G} f := \{f' \in F : S_\mathcal{G} f' = S_\mathcal{G} f\}$. In essence, the constraints (11) are integrals over one or more equivalence classes. $q^\circ \circ S_\mathcal{G}$ is constant on any equivalence class, while q may vary arbitrarily subject to (11). Inspection of (10) indicates that the symmetrization gap is zero if and only if q is constant on each $\mathcal{G}$-induced equivalence class of F. Conversely, the more q varies across each equivalence class, the higher the gap.

Symmetrization and compression via other means. The benefits of compression are not limited to symmetrization via averaging. *Any non-injective, $\mathcal{G}$-invariant map ψ will have a non-zero ψ-gap.* For example, each of $\sup_{g \in \mathcal{G}} f(gX)$, $\inf_{g \in \mathcal{G}} f(gX)$, and $\max\{0, f^\circ(X)\}$ satisfies the criteria.

4.3. Practical Feature Averaging

In practice, the expectation computed in FA may be computationally intractable. Instead, one may sample a set of k transformations with which to average the function output. While this will not output the exact expectation, it still takes advantage of a simplification of the function space via Lemma 6, by aggregating functions that have some probability of being mapped to the same approximately averaged function. To formalize the idea, let $g^k = \{g_1, g_2, \ldots, g_k\}$ be a set of elements of $\mathcal{G}$, and G^k a random realization sampled i.i.d. from λ. Let $S_{g^k} f(x) = k^{-1} \sum_{j \leq k} f(g_j x)$ denote the approximate symmetrization of f by g^k. Finally, let $Q^\circ_{g^k} = Q \circ S^{-1}_{g^k}$ denote the image of a distribution Q on F under S_{g^k}. The following result is a consequence of the fact that Lemma 6 is true for every g^k, and that for $g^{k+1} = g^k \cup \{g_{k+1}\}$, $S_{g^{k+1}} f(x) = f(g_{k+1} x) + \frac{k}{k+1} S_{g^k} f(x)$.

[3] The result can hold for other function classes F; the key requirement is that conditioning is properly defined in F and F°.

Proposition 8. *Assume the conditions of Theorem 7. Let $G_s = G_1, G_2, \ldots$ be a sequence of elements sampled i.i.d. from λ. Then with probability one over G_s,*

$$\begin{aligned}\mathrm{KL}(Q \mid\mid P) &\geq \mathrm{KL}(Q^{\widehat{\circ}}_{G^1} \mid\mid P^{\widehat{\circ}}_{G^1}) \geq \cdots \\ &\geq \mathrm{KL}(Q^{\widehat{\circ}}_{G^k} \mid\mid P^{\widehat{\circ}}_{G^k}) \geq \cdots \\ &\geq \mathrm{KL}(Q^{\circ} \mid\mid P^{\circ}) \,.\end{aligned}$$

As with practical DA, the interplay between SGD and approximate symmetrization remains an open question. However, Proposition 8 makes it clear that at test time, FA—even approximate—is favored.

Computing the symmetrized KL. One drawback to the generic applicability of FA is the difficulty of computing $\mathrm{KL}(Q^\circ \mid\mid P^\circ)$ within current approaches to specifying Q and P on neural networks. Specifically, in the approach pioneered by Langford & Caruana (2002) and refined by Dziugaite & Roy (2017) is (roughly) as follows: P is a mean zero uncorrelated multivariate Gaussian distribution on the weights of the network; Q is an uncorrelated multivariate Gaussian distribution, with mean equal to the trained weights and variances optimized to minimize the PAC-Bayes bound. Given that the the network represents a non-linear function, $\mathrm{KL}(Q^\circ \mid\mid P^\circ)$ cannot be computed in closed form. Whether there is a feasible alternative method to specifying P and Q that would allow for computation of $\mathrm{KL}(Q^\circ \mid\mid P^\circ)$ remains an open question. We give an example of when it can be computed with a linear model in Section 5 with a linear model.

Despite this drawback, the symmetrization gap in the theoretical bounds appears to have real effects on generalization, as shown by the experiments in Section 6.

4.4. PAC-Bayes Bounds

As discussed in Section 3.2, DA symmetrizes the loss function, which does not guarantee that the learned function f^* will be $\mathcal{G}$-invariant. Moreover, the generalization error of the learned predictor f^* will be estimated on untransformed test data, precluding randomized prediction distributions Q based on f^* from concentrating on F°. That is, the PAC-Bayes bound for DA does not benefit directly from the symmetrization gap.

Conversely, FA takes advantage of the symmetrization gap. When the empirical risk $\widehat{R}_\ell(Q, \mathcal{D}^n)$ is close to zero, which will be the case for a trained neural network, the symmetrization gap is the primary contributor to reductions in the PAC-Bayes generalization error bound. When the bound is non-vacuous, *the symmetrization gap is a measurement of the benefit of invariance.*

We formalize these statements in an ordering of the PAC-Bayes generalization upper bounds. Let B_0 be the upper bound on the right-hand side of (6), with B_{DA} and B_{FA} corresponding to the upper bounds for DA (using the augmented empirical risk $\widehat{R}^\circ_\ell(Q, \mathcal{D}^n)$) and FA (using $\mathrm{KL}(Q^\circ \mid\mid P^\circ)$), respectively. Finally, let $B_{\mathrm{DA}\,\#}$ denote the computationally intractable bound for DA given in Appendix B.3.

Theorem 9. *Assume the conditions of Theorem 1, and also that $P_{\mathcal{D}}$ is $\mathcal{G}$-invariant. Then $B_{\mathrm{FA}} \leq B_{\mathrm{DA}\,\#} \leq B_{\mathrm{DA}} = B_0$.*

Of course, without corresponding lower bounds, this does not imply a strict ordering on generalization error. However, the upper bounds are informative, they should carry some information about relative performance. We demonstrate this empirically in Section 6.

5. Example: Permutation-Invariant Linear Regression

The following example is a simple toy model, but it adheres to what may be done in practice. Specifically, consider linear regression $f_w(X) = w^\top X$, $w \in \mathbb{R}^k$, with the PAC-Bayes procedure of Dziugaite & Roy (2017): estimate $\hat{w}$ to optimize some loss function; define Q as a k-dimensional normal distribution with mean $\hat{w}$, covariance $S = s^2 I_k$, and P likewise with mean μ, covariance $\Sigma = \sigma^2 I_k$. Then

$$\mathrm{KL}(Q \mid\mid P) = \frac{k}{2}\left(\frac{s^2}{\sigma^2} - 1 + \ln\frac{\sigma^2}{s^2}\right) + \frac{||\mu - w||_2^2}{2\sigma^2} \,.$$

Alternatively, consider the same model averaged over all permutations of the inputs. Then for any w in the original model, there is the constant vector $w^\circ 1_k = \frac{1}{d!}\sum_{\pi \in \mathcal{S}_d} \pi w = k^{-1} 1_k 1_k^\top w$. The image of the prior therefore is equivalent to a 1-dimensional normal distribution with mean $\mu^\circ = k^{-1} 1_k 1_k^\top \mu$ and variance $k^{-1}\sigma^2$, and similarly for the image of the posterior. Therefore,

$$\mathrm{KL}(Q^\circ \mid\mid P^\circ) = \frac{1}{2}\left(\frac{s^2}{\sigma^2} - 1 + \ln\frac{\sigma^2}{s^2}\right) + \frac{k(\mu^\circ - w^\circ)^2}{2\sigma^2} \,.$$

In practice, the KL (and various measures of model complexity) is dominated by the terms involving $||w||_2^2$. Focusing on the difference in those terms, by the Cauchy–Schwarz inequality the symmetrization gap is

$$\Delta^\circ(Q \mid\mid P) \approx \frac{1}{2\sigma^2}\sum_{j=1}^{k}\left((\mu_j - w_j)^2 - (\mu^\circ - w^\circ)^2\right) \geq 0 \,.$$

We give a further example based on Boolean functions in Appendix B.1.

6. Experiments

We provide two examples to illustrate the theoretical results from sections Sections 3 and 4.

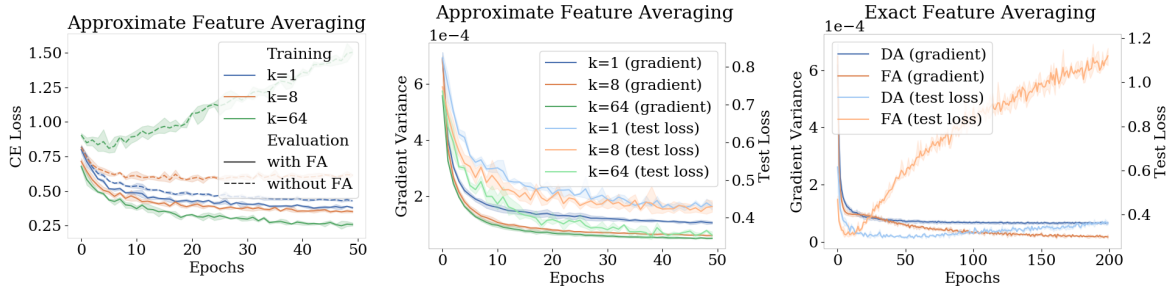

Figure 2. Measurements over the course of training a convolutional neural network using different data augmentation and feature averaging approaches. Left: models are trained with approximate feature averaging using k sampled rotations in the range $\{1, \dots, 360\}$, and then evaluated with and without that averaging scheme. Middle: per-epoch gradient variance and test loss in the same setting. Right: same dataset and architecture as before, but now augmentation set is composted or rotation by 90 degrees, so feature averaging is exact..

6.1. Training Behavior of DA and FA

In Section 3, we showed that feature averaging reduces variance in both function outputs and gradient steps when compared to data augmentation. We provide a demonstration of how this reduction in variance may play out in practice to ground the previous theoretical analysis and to give the reader a sense of the complexity of analyzing the interplay between feature averaging on gradient descent dynamics. For our evaluation, we train a series of convolutional neural networks on an augmentation of the FashionMNIST dataset. The class of an article of clothing is invariant to rotations: put simply, there is no way of rotating a shoe such that it can be mistaken for a t-shirt. We therefore consider two different augmentations of the dataset by rotations to construct invariant training distributions.

In the first, we augment the dataset by the 4-element group $\mathcal{G}$ of 90 degree rotations so that the data-generating distribution $\mathcal{P}$ is invariant to the action of $\mathcal{G}$, and train a simple convolutional neural network (CNN) once with feature averaging, and once without feature averaging. In this setting, the average over $\mathcal{G}$ can be computed exactly. Our findings agree with the results of Section 4: exact FA leads to a reduction in gradient variance, and also to lower training loss. However, the model trained with FA demonstrates overfitting, suggesting that the reduction in variance obtained by exact FA may not always be desirable during training.

We next consider an additional augmentation of FashionMNIST via the group $\mathcal{G}$ of rotations in the set $\{1°, \dots, 360°\}$. In this setting, we perform approximate feature averaging with k samples, where $k << |\mathcal{G}|$. We observe that the model trained with FA becomes increasingly dependent on feature averaging to obtain a low loss: the loss of each individual function computed by the network increases during training, and it is only when averaging over orbits that the network attains the lowest loss. In other words, the trajectory of the models trained with approximate feature averaging converge to regions of parameter space that don't correspond to functions that attain low loss when evaluated without feature averaging, and so may be quite different from the parameters learned by data augmentation.

Table 2. Generalization performance for a permutation-invariant point cloud classification task (see text for details).

Network	Train Error	Test Error	KL Divergence	PAC-Bayes Bound
Fully connected	0.002	0.65	24957	1.75
Partial-Pointnet	0.172	0.248	1992	0.67
Pointnet	0.24	0.245	944	0.533

6.2. Generalization in Neural Networks

We next provide a demonstration of the effect of invariance on PAC-Bayesian bounds for neural networks. We use the ModelNet10 dataset, which consists of LiDAR point cloud data for 10 classes of household objects. This dataset exhibits permutation invariance: the LiDAR reading is stored as a sequence of points defined by $\{x, y, z\}$ coordinates, and the order in which the points are listed is irrelevant to the class. We consider three different architectures: a PointNet-like architecture (Qi et al., 2017), which is invariant to permutations; a partitioned version of the PointNet architecture which is invariant to subgroups of the permutation group (details in the Appendix); and a fully connected model where the invariant pooling operation in the PointNet is replaced by a fully-connected layer. The invariance in the network is implemented via a max-pooling layer instead of an averaging layer and so is not a direct application of feature averaging; however, the results of Eq. (8) would apply, were we able to compute the PAC-Bayesian bound for the model exactly.

We compute the PAC-Bayes bounds following the procedure in Dziugaite & Roy (2017): we convert a deterministic network to a stochastic network by adding Gaussian noise to the weights, and then train this stochastic model using a differentiable surrogate loss that bounds the true PAC-Bayes bound. After this training procedure converges, we then compute the true PAC-Bayes bound. We attain an ordering consistent with the observations presented in the previous

section: the invariant architecture attains the lowest bound, followed by the partially invariant architecture, and finally followed by the fully connected network. We provide a decomposition of the distinct terms in the bound in Table 2.

7. Practical Implications and Conclusions

We refer back to Table 1 for a summary of our theoretical results. A few practical guidelines emerge.

Train with approximate data augmentation or feature averaging. The reduction in variance of risk estimates and their gradients obtained by averaging over $\mathcal{G}$ appears to be beneficial to training, though too much variance-reduction seems undesirable. Based on the experiments in Section 6, we advocate for training with approximate FA or DA.

Use feature averaging at test/deployment time. With a convex loss function, the generalization error $R_\ell(f)$ of a feature-averaged model is no worse, and possibly better, than that of its non-averaged counterpart, even when DA was used for training. Even with non-convex loss, a randomized prediction rule Q has looser generalization bounds than its $\mathcal{G}$-averaged counterpart Q°. Because of this, even a model trained with DA should generalize better when its outputs are averaged over $\mathcal{G}$ at test time. The experiments in Section 6 demonstrate this empirically.

Acknowledgements

MK has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant agreement No. 834115).

References

Bloem-Reddy, B. and Teh, Y. W. Probabilistic symmetry and invariant neural networks. *Journal of Machine Learning Research*. To appear.

Catoni, O. *PAC-Bayesian supervised classification: the thermodynamics of statistical learning*. Institute of Mathematical Statistics, 2007.

Chen, S., Dobriban, E., and Lee, J. H. Invariance reduces variance: Understanding data augmentation in deep learning and beyond. *arXiv preprint arXiv:1907.10905*, 2019.

Cohen, T. S. and Welling, M. Group equivariant convolutional networks. In Balcan, M. F. and Weinberger, K. Q. (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 2990–2999. PMLR, 2016.

Cohen, T. S., Geiger, M., and Weiler, M. A general theory of equivariant CNNs on homogeneous spaces. In *Advances in Neural Information Processing Systems 32*, pp. 9142–9153. 2019.

Cubuk, E. D., Zoph, B., Mané, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation policies from data. *ArXiv*, abs/1805.09501, 2018.

Dao, T., Gu, A., Ratner, A. J., Smith, V., De Sa, C., and Ré, C. A kernel theory of modern data augmentation. *Proceedings of the 36th International Conference on Machine Learning, PMLR 97*, 2019.

Dziugaite, G. K. and Roy, D. M. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In *UAI*, 2017.

Dziugaite, G. K. and Roy, D. M. Data-dependent PAC-Bayes priors via differential privacy. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *NeurIPS 31*, pp. 8430–8441. 2018.

Fawzi, A., Samulowitz, H., Turaga, D., and Frossard, P. Adaptive data augmentation for image classification. In *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3688–3692. Ieee, 2016.

Iyyer, M., Boyd-Graber, J., Claudino, L., Socher, R., and Daumé III, H. A neural network for factoid question answering over paragraphs. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 633–644, October 2014.

Kondor, R. and Trivedi, S. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In *Proc. ICML 35*, volume 80 of *PMLR*, pp. 2747–2755, 2018.

Langford, J. and Caruana, R. (Not) bounding the true error. In Dietterich, T. G., Becker, S., and Ghahramani, Z. (eds.), *Advances in Neural Information Processing Systems 14*, pp. 809–816. 2002.

Lever, G., Laviolette, F., and Shawe-Taylor, J. Tighter PAC-Bayes bounds through distribution-dependent priors. *Theoretical Computer Science*, 473:4–28, 2013.

McAllester, D. A. Some PAC-Bayesian theorems. *Machine Learning*, 37(3):355—363, 1999.

Neyshabur, B. *Implicit Regularization in Deep Learning*. PhD thesis, TTI-Chicago, 2017.

Qi, C. R., Su, H., Mo, K., and Guibas, L. J. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, pp. 652–660, 2017.

Raj, A., Kumar, A., Mroueh, Y., Fletcher, T., and Schoelkopf, B. Local Group Invariant Representations via Orbit Embeddings. In Singh, A. and Zhu, J. (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1225–1235. PMLR, 2017.

Ravanbakhsh, S., Schneider, J., and Poczos, B. Equivariance through parameter-sharing. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 2892–2901. JMLR. org, 2017.

Salamon, J. and Bello, J. P. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24(3): 279–283, 2017.

Shawe-Taylor, J. Threshold network learning in the presence of equivalences. In Moody, J. E., Hanson, S. J., and Lippmann, R. P. (eds.), *Advances in Neural Information Processing Systems 4*, pp. 879–886. Morgan-Kaufmann, 1991.

Shawe-Taylor, J. Sample sizes for threshold networks with equivalences. *Information and Computation*, 118(1):65 – 72, 1995.

van der Wilk, M., Bauer, M., John, S., and Hensman, J. Learning invariances using the marginal likelihood. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 9938–9948. 2018.

Wood, J. and Shawe-Taylor, J. Representation theory and invariant neural networks. *Discrete Applied Mathematics*, 69(1):33–60, 1996.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017.

Zhao, R., Wang, D., Yan, R., Mao, K., Shen, F., and Wang, J. Machine health monitoring using local feature-based gated recurrent unit networks. *IEEE Transactions on Industrial Electronics*, 65(2):1539–1548, Feb 2018.

Zhou, J. and Troyanskaya, O. G. Predicting effects of noncoding variants with deep learning-based sequence model. *Nature Methods*, 12(10):931, 2015.

Zhou, W., Veitch, V., Austern, M., Adams, R. P., and Orbanz, P. Non-vacuous generalization bounds at the imagenet scale: a PAC-Bayesian compression approach. In *International Conference on Learning Representations*, 2019.

Zou, W., Zhu, S., Yu, K., and Ng, A. Y. Deep learning of invariant features via simulated fixations in video. In *Advances in Neural Information Processing Systems*, pp. 3203–3211, 2012.

A. Proofs

Proof of Proposition 3. Let $w \in V^*$, and suppose that w is not invariant under the action of $\mathcal{G}$. Let $w^\circ = \mathbb{E}_{G\sim\lambda}[\rho_G^* w]$, which is $\mathcal{G}$-invariant by construction. Because $\mathcal{X}$ spans V, $w - w^\circ \neq 0$ implies that $w \neq w^\circ$.

Consider the minimizer

$$\hat{w} = \operatorname*{arg\,min}_{w\in V^*} \widehat{R}_\ell^\circ(f_w, \mathcal{D}^n) = \operatorname*{arg\,min}_{w\in V^*} \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{G\sim\lambda}[\ell(w^\top \rho_G X_i, Y_i)] \;,$$

which is unique because ℓ is strictly convex by assumption. Assume that $\hat{w}$ is not $\mathcal{G}$-invariant. Applying Jensen's inequality, we have

$$\begin{aligned}
\widehat{R}_\ell^\circ(f_{\hat{w}}, \mathcal{D}^n) &= \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{G\sim\lambda}[\ell(\hat{w}^\top \rho_G X_i, Y_i)] \\
&> \frac{1}{n}\sum_{i=1}^n \ell(\mathbb{E}_{G\sim\lambda}[\hat{w}^\top \rho_G X_i, Y_i)] \\
&= \frac{1}{n}\sum_{i=1}^n \ell(\mathbb{E}_{G\sim\lambda}[(\rho_{G^{-1}}^* \hat{w})]^\top X_i, Y_i)] \\
&= \frac{1}{n}\sum_{i=1}^n \ell(\hat{w}^\circ X_i, Y_i) = \widehat{R}_\ell^\circ(f_{\hat{w}^\circ}, \mathcal{D}^n) \;,
\end{aligned}$$

which cannot be the case because $\hat{w}$ minimizes $\widehat{R}_\ell^\circ$. Therefore, $\hat{w}$ must be $\mathcal{G}$-invariant. □

A.1. Proof of Theorem 4

The proof of our PAC-Bayes bound for data augmentation makes use of the following result due to Lever et al. (2013).

Theorem 10 (Lever et al. (2013), Theorem 1)**.** *For any functions $A(f)$, $B(f)$ over F, either of which may be a statistic of the training data $\mathcal{D}^n$, any distribution P over F, any $\delta \in (0,1]$, any $t > 0$, and a convex function $\mathscr{D} : \mathbb{R}\times\mathbb{R} \to \mathbb{R}$, with probability $P_{\mathcal{D}}^n$ at least $1-\delta$, for all distributions Q on F,*

$$\mathscr{D}\big(\mathbb{E}_{f\sim Q}[A(f)], \mathbb{E}_{f\sim Q}[B(f)]\big) \leq \frac{1}{t}\left(\mathrm{KL}(Q \,||\, P) + \log\frac{\mathcal{L}_P}{\delta}\right) , \tag{12}$$

where $\mathcal{L}_P := \mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}, f\sim P}[e^{t\mathscr{D}(A(f),B(f))}]$ is the Laplace transform of $\mathscr{D}(A(f), B(f))$.

As Lever et al. (2013) discuss, many PAC-Bayes bounds in the literature can be obtained as special cases of Theorem 10, including Catoni's bound in Theorem 1. In that case, which applies to 0-1 loss, $t = n$, $A(f) = \widehat{R}_\ell(f, \mathcal{D}^n)$, $B(f) = R_\ell(f)$, and

$$\mathscr{D}_C(q,p) := -\log(1 - p(1-e^{-C})) - Cq \;, \quad q, p \in (0,1),\ C > 0 \tag{13}$$

$$= -\log \mathbb{E}_{Z\sim\mathrm{Bern}(p)}[e^{-CZ}] - Cq \;. \tag{14}$$

Basic calculations show that with these quantities, $\mathcal{L}_P = 1$.

Recall that

$$\widehat{R}_\ell(f, \mathcal{D}^n) := \frac{1}{n}\sum_{i=1}^n \ell(f(X_i), Y_i) \tag{15}$$

$$\widehat{R}_\ell^\circ(f, \mathcal{D}^n) := \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{G\sim\lambda}[\ell(f(GX_i), Y_i)] \tag{16}$$

$$\widehat{\widehat{R}}_\ell^\circ(f, \mathcal{D}^n) := \frac{1}{nm}\sum_{i=1}^n\sum_{j=1}^m \ell(f(G_{ij}X_i), Y_i) \;. \tag{17}$$

Let (G_{ij}) denote the collection of $m \cdot n$ random augmentation transformations sampled i.i.d. from λ.

Lemma 11. *Let ℓ be binary loss, P any distribution on F, and assume that $P_{\mathcal{D}}$ is $\mathcal{G}$-invariant. Then*

$$\mathbb{E}_{f\sim P}\left[\mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}}\left[e^{n\mathscr{D}_C(\widehat{R}^\circ_\ell(f,\mathcal{D}^n),R_\ell(f))}\right]\right] \leq \mathbb{E}_{f\sim P}\left[\mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}}[e^{n\mathscr{D}_C(\widehat{R}_\ell(f,\mathcal{D}^n),R_\ell(f))}]\right] = 1 \tag{18}$$

and

$$\mathbb{E}_{f\sim P}\left[\mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}}\left[e^{n\mathscr{D}_C(\widehat{R}^\circ_\ell(f,\mathcal{D}^n),R_\ell(f))}\right]\right] \leq \mathbb{E}_{f\sim P}\left[\mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}}[e^{n\mathscr{D}_C(\widehat{R}_\ell(f,\mathcal{D}^n),R_\ell(f))}]\right] = 1\,. \tag{19}$$

Proof. Since the observations (X_i, Y_i) are i.i.d., the expectation over $\mathcal{D}^n$ on the left-hand side of (18) requires evaluating $\mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}}\left[e^{-C\mathbb{E}_{G\sim\lambda}[\ell(f(X_i),Y_i))]}\right]$. Using the convexity of e^{-x}, Jensen's inequality and Fubini's theorem yield

$$\begin{aligned}\mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}\left[e^{-C\mathbb{E}_{G\sim\lambda}[\ell(f(GX_i),Y_i))]}\right] &\leq \mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}\left[\mathbb{E}_{G\sim\lambda}\left[e^{-C\ell(f(GX_i),Y_i))}\right]\right] \\ &= \mathbb{E}_{G\sim\lambda}\left[\mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}\left[e^{-C\ell(f(GX_i),Y_i))}\right]\right]\,.\end{aligned} \tag{20}$$

Now, $\mathcal{G}$-invariance of $P_{\mathcal{D}}$ implies that $\mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}[h(gX_i,Y_i)] = \mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}[h(X_i,Y_i)]$ for all measurable functions $h : \mathcal{X}\times\mathcal{Y}\to\mathbb{R}_+$ and all $g\in\mathcal{G}$, which extends to independent random G by Fubini's theorem. Therefore,

$$\mathbb{E}_{G\sim\lambda}\left[\mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}\left[e^{-C\ell(f(GX_i),Y_i))}\right]\right] = \mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}\left[e^{-C\ell(f(X_i),Y_i))}\right] = \mathbb{E}_{Z\sim\text{Bern}(R_\ell(f))}[e^{-CZ}]\,,$$

which implies (18).

For the second inequality (19), observe that by Jensen's inequality,

$$\begin{aligned}\mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}}\left[e^{-nC\widehat{R}^\circ_\ell(f,\mathcal{D}^n)}\right] &= \prod_{i=1}^n \mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}\left[\mathbb{E}_{(G_{ij})_{j=1}^m\sim\lambda}\left[\exp\left(-\frac{C}{m}\sum_{j=1}^m \ell(f(G_{ij}X_i),Y_i)\right)\right]\right] \\ &\leq \prod_{i=1}^n \mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}\left[\mathbb{E}_{(G_{ij})_{j=1}^m\sim\lambda}\left[\frac{1}{m}\sum_{j=1}^m e^{-C\ell(f(G_{ij}X_i),Y_i)}\right]\right] \\ &= \prod_{i=1}^n \mathbb{E}_{(X_i,Y_i)\sim P_{\mathcal{D}}}\left[\mathbb{E}_{G\sim\lambda}\left[e^{-C\ell(f(GX_i),Y_i)}\right]\right]\end{aligned}$$

Using the $\mathcal{G}$-invariance of $P_{\mathcal{D}}$ once again, we have

$$\mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}}\left[e^{-nC\widehat{R}^\circ_\ell(f,\mathcal{D}^n)}\right] \leq \mathbb{E}_{\mathcal{D}^n\sim P_{\mathcal{D}}}\left[e^{-nC\widehat{R}_\ell(f,\mathcal{D}^n)}\right] = \left(\mathbb{E}_{Z\sim\text{Bern}(R_\ell(f))}[e^{-CZ}]\right)^n\,,$$

which implies (19). □

Proof of Theorem 4. Theorem 4 follows from Theorem 10 and Lemma 11. In particular, observe that the expectation of any of the risks (15)–(17) over $\mathcal{D}^n$ and $f\sim Q$ is $R_\ell(Q)$. Therefore, using any of those risks as $A(f)$ in Theorem 10 with $B(f) = R_\ell(f)$ will result in valid a PAC-Bayes bound; the only quantity that changes between the three situations is $\mathcal{L}_P$ in (12). Lemma 11 establishes that $\mathcal{L}_P$ when $A(f)$ is either of $\widehat{R}^\circ_\ell(f,\mathcal{D}^n)$ or $\widehat{R}^{\widehat{\circ}}_\ell(f,\mathcal{D}^n)$ is upper-bounded by $\mathcal{L}_P$ when $A(f) = \widehat{R}_\ell(f,\mathcal{D}^n)$, which is equal to 1.

The particular bound (6) follows from algebraic manipulations of (12). □

A.2. Proof of Proposition 5

Proof of Proposition 5. Let $\mathcal{G}$ be a group with some probability measure λ, and F a class of functions $f : \mathcal{X}\to\mathbb{R}$. Let $\ell : \mathbb{R}\times\mathbb{R}\to\mathbb{R}_+$ be a loss function such that $\ell(f(\bullet),\bullet)\in L_2(P_{\mathcal{D}})$ for every $f\in F$. Then the augmented risk of any function $f\in F$ is

$$\widehat{R}^\circ_\ell(f,\mathcal{D}^n) = \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{G\sim\lambda}[\ell(f(GX_i),Y_i)]\,.$$

If ℓ is convex in the first argument, then by Jensen's inequality,

$$\mathbb{E}_{G\sim\lambda}[\ell(f(GX_i),Y_i)] \geq \ell(\mathbb{E}_{G\sim\lambda}[f(GX_i)],Y_i)\,,\quad i = 1,2,\ldots,n\,. \tag{21}$$

On the other hand, the $\mathcal{G}$-symmetrization of $f(X)$ is $f^\circ(X) = \mathbb{E}_{G\sim\lambda}[f(GX)]$, with augmented risk

$$\begin{aligned}\widehat{R}^\circ_\ell(f^\circ, \mathcal{D}^n) &= \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{G\sim\lambda}[\ell(\mathbb{E}_{G\sim\lambda}[f(GX_i)], Y_i)] \\ &= \frac{1}{n}\sum_{i=1}^n \ell(\mathbb{E}_{G\sim\lambda}[f(GX_i)], Y_i) \\ &= \widehat{R}_\ell(f^\circ, \mathcal{D}^n)\ .\end{aligned}$$

Combined with (21), the reduction in empirical augmented risk follows. The reduction in $\widehat{R}^\circ_\ell(Q, \mathcal{D}^n)$ follows trivially.

The variance-reduction is established by extending the argument in the proof of Proposition 2. Specifically, by the conditional Jensen's inequality,

$$\mathrm{Var}_{\mathcal{D}^n\sim P_{\mathcal{D}}^n}\left[\widehat{R}^\circ_\ell(f, \mathcal{D}^n)\right] = \mathrm{Var}[\mathbb{E}[\widehat{R}^\circ_\ell(f, \mathcal{D}^n) \mid \Phi^n]] \geq \mathrm{Var}[\mathbb{E}[\widehat{R}_\ell(f^\circ, \mathcal{D}^n) \mid \Phi^n]] = \mathrm{Var}_{\mathcal{D}^n\sim P_{\mathcal{D}}^n}\left[\widehat{R}_\ell(f^\circ, \mathcal{D}^n)\right]\ .$$

□

A.3. Proof of Lemma 6 and Theorem 7

The proof of Lemma 6 relies on the chain rule of relative entropy. Let two probability measures, $\tilde{\mu} \ll \tilde{\nu}$ defined on the product space $(E_1 \times E_2, \mathcal{E}_1 \otimes \mathcal{E}_2)$, have marginal measures $\tilde{\mu}_1 \ll \tilde{\nu}_1$ on $(E_1, \mathcal{E}_1)$ (respectively, $\tilde{\mu}_2 \ll \tilde{\nu}_2$ on $(E_2, \mathcal{E}_2)$) and regular conditional probability measures $\tilde{\mu}_{2|1} \ll \tilde{\nu}_{2|1}$ (resp. $\tilde{\mu}_{1|2} \ll \tilde{\nu}_{1|2}$). Recall the chain rule of relative entropy is

$$\mathrm{KL}(\tilde{\mu} \,||\, \tilde{\nu}) = \mathrm{KL}(\tilde{\mu}_1 \,||\, \tilde{\nu}_1) + \mathbb{E}_{\tilde{\mu}}\left[\log \frac{d\tilde{\mu}_{2|1}}{d\tilde{\nu}_{2|1}}\right] = \mathrm{KL}(\tilde{\mu}_2 \,||\, \tilde{\nu}_2) + \mathbb{E}_{\tilde{\mu}}\left[\log \frac{d\tilde{\mu}_{1|2}}{d\tilde{\nu}_{1|2}}\right]\ . \tag{22}$$

Observe that each of the terms in the equalities is non-negative.

Proof of Lemma 6. Given probability measures on $(E_1, \mathcal{E}_1)$ $\mu \ll \nu$ (with density m such that $\mu = m \cdot \nu$) and a measurable map $\psi : (E_1, \mathcal{E}_1) \to (E_2, \mathcal{E}_2)$, construct the probability measure $\tilde{\mu}$ on $(E_1 \times E_2, \mathcal{E}_1 \otimes \mathcal{E}_2)$ as

$$\tilde{\mu}(A \times B) = \mu(A \cap \psi^{-1}B) = \int_A \mu(dx_1) \int_B \delta_{\psi(x_1)}(dx_2)\ , \quad A \in \mathcal{E}_1,\ B \in \mathcal{E}_2\ ,$$

and likewise for $\tilde{\nu}$. Then in the notation of (22), $\tilde{\mu}_1 = \mu \ll \nu = \tilde{\nu}_1$, and $\tilde{\mu}_{2|1} = \delta_{\psi(x_1)} = \tilde{\nu}_{2|1}$. Therefore,

$$\mathrm{KL}(\tilde{\mu} \,||\, \tilde{\nu}) = \mathrm{KL}(\tilde{\mu}_1 \,||\, \tilde{\nu}_1) = \mathrm{KL}(\mu \,||\, \nu)\ . \tag{23}$$

Alternatively, $\tilde{\mu}_2 = \mu \circ \psi^{-1}$, $\tilde{\nu}_2 = \nu \circ \psi^{-1}$, and it is straightforward to show that

$$\mathbb{E}_{\tilde{\mu}}\left[\log \frac{d\tilde{\mu}_{1|2}}{d\tilde{\nu}_{1|2}}\right] = \mathbb{E}_{\tilde{\mu}}\left[\log \frac{d\tilde{\mu}_1}{d\tilde{\nu}_1}\right] - \mathbb{E}_{\tilde{\mu}}\left[\log \frac{d\tilde{\mu}_2}{d\tilde{\nu}_2}\right] = \mathbb{E}_{\mu}\left[\log \frac{m}{m \circ \psi}\right] = \Delta_\psi(\mu \,||\, \nu) \geq 0.\ . \tag{24}$$

Therefore,

$$\mathrm{KL}(\tilde{\mu} \,||\, \tilde{\nu}) = \mathrm{KL}(\mu \,||\, \nu) = \mathrm{KL}(\mu \circ \psi^{-1} \,||\, \nu \circ \psi^{-1}) + \Delta_\psi(\mu \,||\, \nu)\ . \tag{25}$$

□

Proof of Theorem 7. For $\mathcal{X}$ a compact metric space and $\mathcal{Y}$ a Polish space, the space $F = C(\mathcal{X}, \mathcal{Y})$ of continuous functions $f : \mathcal{X} \to \mathcal{Y}$ is a Polish space, and therefore it (along with its Borel σ-algebra $\mathcal{B}(C(\mathcal{X}, \mathcal{Y}))$) is a standard Borel space. For a group $\mathcal{G}$ acting measurably on $\mathcal{X}$, the symmetrization operator $S_\mathcal{G} : F \to F^\circ$ is measurable, and the product space $(F \times F^\circ, \mathcal{B}(F) \otimes \mathcal{B}(F^\circ))$ is a standard Borel space. Thus, the conditions of Lemma 6 are satisfied and the result follows. □

B. Examples, Counterexamples, Tighter Bounds

B.1. Permutation-invariant Boolean Function

As an illustrative example, we consider the task of learning a permutation-invariant Boolean function. We consider the following toy learning algorithm. For a training set $\mathcal{D}^n$, each observation of which is a pair $(X_i, Y_i) \in \{0,1\}^k \times \{0,1\}$, the algorithm outputs a sample from $Q^n_{\mathcal{D}}$, the uniform distribution over all k-ary Boolean functions which agree with $\mathcal{D}^n$. If the full function space under consideration is the set of k-ary Boolean functions $F = \{f : \{0,1\}^k \to \{0,1\}\}$, then $|F| = 2^{2^k}$. Moreover, the number of Boolean functions consistent with a training data set containing $|\mathcal{D}^n|$ unique binary vectors is $2^{2^k - |\mathcal{D}^n|}$. Thus, letting P denote the uniform distribution over k-ary Boolean functions,

$$\mathrm{KL}(Q \mid\mid P) = \log_2 \frac{2^{2^k}}{2^{2^k - |\mathcal{D}^n|}} = |\mathcal{D}^n| \leq n \ ,$$

where for convenience we have used $\log_2$ inside the KL.

In contrast, we can consider the same learning algorithm applied to the class of permutation-invariant Boolean functions.[4] The permutation-invariant Boolean functions are those that are constant on all input vectors containing the same number of 1-valued entries, and therefore is equivalent to the set of functions $F_{\text{inv}} = \{f : \{0, \ldots, k\} \to \{0,1\}\}$, with $|F_{\text{inv}}| = 2^{k+1}$. The restriction of the uniform prior P to F_{inv} remains uniform, $P_{\text{inv}}(f_{\text{inv}}) = 2^{-(k+1)}$. The restriction of Q depends on $|\mathcal{D}^n|_{\text{inv}}$, the number of $j \in \{0, \cdots, k\}$ such that at least observation in $\mathcal{D}^n$ has exactly j 1-valued entries: $Q_{\text{inv}}(f_{\text{inv}}) = 2^{-(k+1-|\mathcal{D}^n|_{\text{inv}})}$. Thus,

$$\mathrm{KL}(Q_{\text{inv}} \mid\mid P_{\text{inv}}) = |\mathcal{D}^n|_{\text{inv}} \leq |\mathcal{D}^n| \ .$$

In this simple case, if the observations are consistent with the assumptions, i.e., the output is constant across input vectors with the same number of 1-valued entries, then the invariant model obtains a KL gap of $|\mathcal{D}^n| - |\mathcal{D}^n|_{\text{inv}}$.

B.2. Counterexamples

Feature averaging and non-convex losses. We consider the binary classification setting with the zero-one loss and some function class f bounded in $[0,1]$ – that is $\ell(x,y) = \mathbb{1}[|f(x) - y| > 1]$. Suppose that there exists some invariance $\mathcal{G}$ in the data such that $y(x) = y(gx)$ for all x, g. Then consider a function which, for some small ϵ, outputs $f(x) = \frac{1}{2} + y\epsilon$ on a $1 - 2\epsilon$ fraction of each equivalence class of the inputs, and $1 - y$ on 2ϵ of the inputs in each equivalence class. Then $\mathbb{E}[f(gx)] = (1 - 2\epsilon)(\frac{1}{2} + y\epsilon) + 2\epsilon(1 - y)$. When $y = 0$, this expectation is $\frac{1}{2} + \epsilon$, and when $y = 1$ it is $\frac{1}{2}[1 - \epsilon - 2\epsilon^2] < \frac{1}{2}$, so the feature-averaged model would have risk 1 whereas the original model had risk 0.

Non-uniform data-generating distributions. When the data-generating distribution is not uniform over the set $\mathcal{T}$, then performing data augmentation with $\mathcal{T}$ will not necessarily lead to a more accurate estimate of the model's empirical risk. For example, consider the task of learning a function g satisfying $g(x) = g(-x)$, bounded in magnitude by some constant A. Suppose, however, that positive numbers are much more likely under the data generating distribution, with $p(\mathbb{R}^+) = 1 - \epsilon$ for small ϵ. Then the function $f(x) = \mathbb{1}[x > 0]g(x)$ will satisfy $\mathbb{E}[\|f(X_S) - g(X_s)\|] \neq \mathbb{E}[\|f(X_{S^{\text{aug}}}) - g(X_{S^{\text{aug}}})\|]$. So the augmented risk is no longer an unbiased estimator of the empirical risk. Further, in this particular case its variance is also higher, as it will be equal to $\frac{1}{2}\mathrm{Var}(g(x))$, in contrast to $\epsilon\mathrm{Var}(g(x))$.

B.3. Tighter PAC-Bayes Bounds for Data Augmentation

Although Theorem 4 establishes that the i.i.d. PAC-Bayes bound (6) is valid for exact DA, the proof of Theorem 4 indicates that a tighter bound is possible. In particular, recall that when $P_{\mathcal{D}}$ is $\mathcal{G}$-invariant (Bloem-Reddy & Teh; Chen et al., 2019),

$$\mathbb{E}_{G\sim\lambda}[\ell(f(GX), Y)] = \mathbb{E}_{(X,Y)\sim P_{\mathcal{D}}}[\ell(f(X), Y) \mid \Phi] := \ell^{\circ}_f(\Phi) \ .$$

$\ell^{\circ}_f(\Phi)$ is a random variable, the average loss on the random orbit with representative Φ, whose distribution is induced by $P_{\mathcal{D}}$. Therefore, we can write $\mathcal{L}_P$ in (12) as

$$\mathcal{L}_P = \mathbb{E}_{f\sim P}\left[\left(\frac{\mathbb{E}_{\Phi\sim P_{\mathcal{D}}}\left[e^{-C\ell^{\circ}_f(\Phi)}\right]}{\mathbb{E}_{Z\sim\mathrm{Bern}(R_\ell(f))}\left[e^{-CZ}\right]}\right)^n\right] \leq 1 \ .$$

[4] Note that the class of permutation-invariant Boolean functions is a strict subset of the Boolean functions, because symmetrization via averaging produces a function with image $[0,1]$ (and thus the Boolean functions are not closed under averaging).

In general, this cannot be computed in closed form. However, it might be possible to estimate using the data (with appropriate modifications to the resulting bound) and samples $f \sim P$.

C. Computation Details for PAC-Bayes Bounds

PAC-Bayes bounds for neural networks are computed via the following procedure: a deterministic neural network is trained to minimize the cross-entropy loss on the dataset. After it has reached a suitable training accuracy, we use these parameters as the initialization for the means and variances of the stochastic neural network weights used for the PAC-Bayes bounds. We directly optimize a surrogate of the PAC-Bayes bound (using the cross-entropy loss instead of the zero-one accuracy and using the reparameterization trick to get the derivatives of the variance parameters). The exact computation of the PAC-Bayes bound uses the union bound and discretization of the PAC-Bayes prior as described in (Dziugaite & Roy, 2017). Reported values are at optimization convergence.

C.1. Experiment parameters and computation details

The experiment code is provided with the paper submission, but we describe here at a high level the different models used in our empirical evaluations.

FashionMNIST CNN: the convolutional network used for FashionMNIST consists of two convolutional layers (with batch norm and max pooling) followed by a single fully connected layer.

LiDAR Permutation-Invariant Network: we use a scaled-down version of the PointNet architecture (Qi et al., 2017). We include two layers of 1D convolutions followed by a max-pooling layer that selects the maximum over input points for each channel. This layer is followed by two fully-connected layers leading into the final output.

Partially-Invariant Network: we alter the previous architecture slightly so that it is only invariant to *subgroups* of the permutation group on its inputs. Specifically, we partition the input into 8 disjoint subsets, and apply the previous model's permutation-invariant embedding layers to each partition. The result is a feature representation that is invariant to permutations within each partition of the input, but not between partitions. This representation is then fed through the same architecture. We note that we keep the number of convolutional filters per layer constant, which results in a larger feature embedding by a factor of 8 that is fed into the first fully connected layer. As a result, this model has significantly more parameters than the fully permutation-invariant model.

Fully Connected Network: the max-pooling operator of the previous two architectures is omitted. This network has many more parameters than either of the first two models, and is not invariant to any subgroup of the permutation group.

D. Additional Empirical Evaluations

In addition to the results shown in Fig. 2, we include further plots to characterize training the FA as opposed to DA, and provide some insights here.

1. Feature averaging at evaluation uniformly improves the loss function compared to sampling a single input.
2. Feature averaging at evaluation doesn't appear to significantly harm accuracy (a non-convex loss function), but doesn't see the same improvement as for the cross-entropy loss.
3. Models trained with feature averaging tend to achieve lower training loss, but in the exact feature averaging setting this improved training loss is accompanied by increased overfitting.
4. Models trained with feature averaging perform worse over time when evaluated with a single sample. This gap increases as the model is trained.

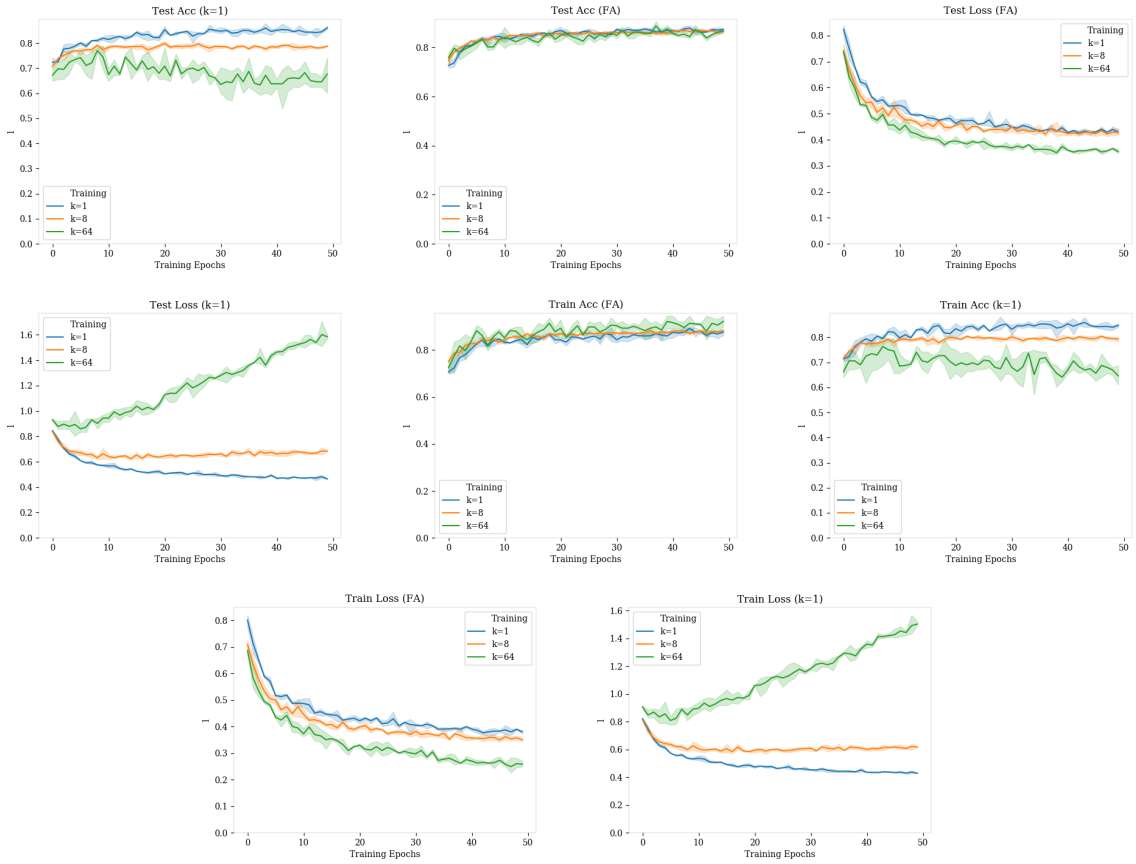

Figure 3. Measures of performance for networks trained with approximate feature averaging. Number of samples used in approximate FA during training range from $k = 1$ to $k = 64$. FA indicates that the model was evaluated using the same number of samples that it was trained on, while k=1 indicates that a single sample is drawn at evaluation time.

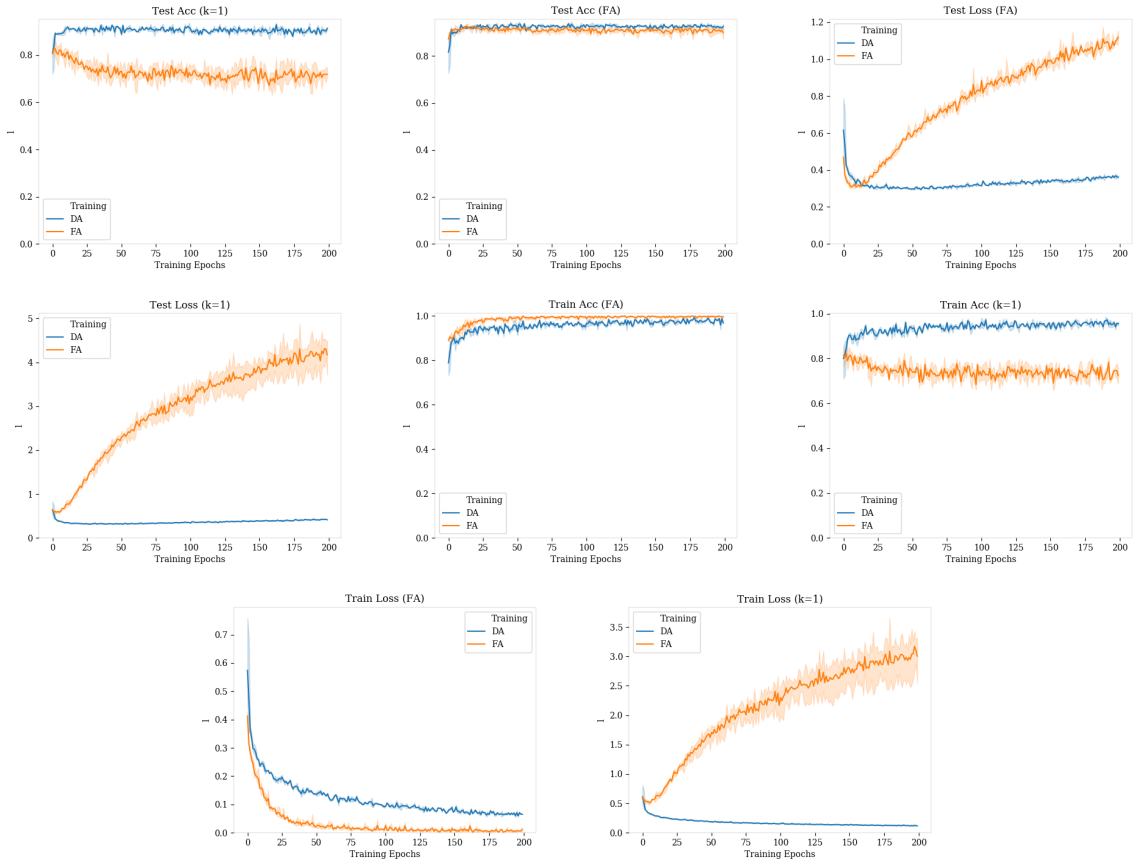

Figure 4. Different measures of performance of networks under different training regimes. Evaluation format (with a single sample or with averaging) is included in title, and training method (trained with data augmentation or feature averaging) is distinguished within each plot by colour.