AUTOMATIC DATA AUGMENTATION VIA INVARIANCE-CONSTRAINED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Underlying data structures, such as symmetries or invariances to transformations, are often exploited to improve the solution of learning tasks. However, embedding these properties in models or learning algorithms can be challenging and computationally intensive. Data augmentation, on the other hand, induces these symmetries during training by applying multiple transformations to the input data. Despite its ubiquity, its effectiveness depends on the choices of which transformations to apply, when to do so, and how often. In fact, there is both empirical and theoretical evidence that the indiscriminate use of data augmentation can introduce biases that outweigh its benefits. This work tackles these issues by automatically adapting the data augmentation while solving the learning task. To do so, it formulates data augmentation as an invariance-constrained learning problem and leverages Monte Carlo Markov Chain (MCMC) sampling to solve it. The result is a practical algorithm that not only does away with a priori searches for augmentation distributions, but also dynamically controls if and when data augmentation is applied. Our experiments illustrate the performance of this method, which achieves state-of-the-art results in automatic data augmentation benchmarks for CIFAR datasets. Furthermore, this approach can be used to gather insights on the actual symmetries underlying a learning task.

1 INTRODUCTION

Exploiting the underlying structure of data has always been a key principle in data analysis. Its use has been fundamental to the success of machine learning solutions, from the translational equivariance of convolutional neural networks (Fukushima and Miyake, 1982) to the invariant attention mechanism in Alphafold (Jumper et al., 2021). However, embedding invariances and symmetries in model architectures is hard in general and when possible, often incurs a high computational cost. This is the case, of rotation invariant neural network architectures that rely on group convolutions, which are feasible only for small, discrete transformation spaces or require coarse undersampling due to their high computational complexity (Cohen and Welling, 2016; Finzi et al., 2020).

A widely used alternative consists of modifying the data rather than the model. That is, to *augment* the dataset by applying transformations to samples in order to induce the desired symmetries or invariances during training. Data augmentation, as it is commonly known, is used to train virtually all state-of-the-art models in a variety of domains (Shorten and Khoshgoftaar, 2019). This empirical success is supported by theoretical results showing that, when the underlying data distribution is invariant to the applied transformations, data augmentation provides a better estimation of the statistical risk (Chen et al., 2019; Sannai et al., 2019; Lyle et al., 2020; Shao et al., 2022). On the other hand, applying the wrong transformations can introduce biases that may outweigh these benefits (Chen et al., 2019; Shao et al., 2022).

Choosing which transformations to apply, when to do so, and how often, is thus paramount to achieving good results. However, it requires knowledge about the underlying distribution of the data that is typically unavailable. Several approaches to learning an augmentation policy or distribution over a fixed set of transformations exist, such as reinforcement learning (Cubuk et al., 2018), genetic algorithms (Ho et al., 2019), density matching (Lim et al., 2019; Cubuk et al., 2020; Hataya et al., 2020), gradient matching (Zheng et al., 2022), bi-level optimization (Li et al., 2020b; Liu et al., 2021), jointly optimizing over transformations using regularised objectives (Benton et al., 2020) and

variational bayesian inference (Chatzipantazis et al., 2021). Optimization based methods often require computing gradients with respect to transformations (Chatzipantazis et al., 2021; Li et al., 2020b). Moreover, several methods resort to computationally intensive search phases, the optimization of auxiliary models, or additional data, while failing to outperform fixed user-defined augmentation distributions (Müller and Hutter, 2021).

In this work, we formulate data augmentation as an invariance-constrained learning problem. The use of invariance overcomes the need to search for a specific distribution over transformations. In addition, the constrained learning formulation mitigates the potential biases introduced by data augmentation whithout doing away with its potential benefits. More specifically, we rely on an *approximate* notion of invariance that is weighted by the probability of each data point. Hence, we require the output of our model to be stable only on the support of the underlying data distribution, and more so on common samples. By imposing this requirement as a constraint on the learning task and leveraging recent duality results, the amount of data augmentation can be automatically adjusted during training. We propose an algorithm that combines stochastic primal-dual methods and MCMC sampling to do away with the need for transformations to be differentiable. Our experiments show that it leads to state-of-the-art results in automatic data augmentation benchmarks in CIFAR datasets.

2 Data Augmentation in Supervised Learning

As in the standard supervised learning setting, let $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ denote a feature vector and $y \in \mathcal{Y} \subseteq \mathbb{R}$ its associated label or measurement. For classification tasks, we take $\mathcal{Y} \subseteq \mathbb{N}$. Let $\mathfrak{D}$ denote a probability distribution over the data pairs $(\mathbf{x}, y)$ and $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ be a non-negative, convex loss function, e.g., the cross entropy loss. Our goal is to learn a predictor $f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$ in some hypothesis class $\mathcal{H}_{\boldsymbol{\theta}} = \{f_{\boldsymbol{\theta}} \mid \boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^p\}$ that minimizes the expected loss, namely

$$\underset{\boldsymbol{\theta}\in\Theta}{\text{minimize}} \;\; R(f_{\boldsymbol{\theta}}) := \mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}[\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), y)]. \tag{SRM}$$

We consider the distribution $\mathfrak{D}$ to be unknown, except for the dataset $\{(\mathbf{x}_i, y_i), i = 1, \ldots, n\}$ of n i.i.d. samples from $\mathfrak{D}$. Therefore, we rely on the empirical approximation of the objective of (SRM), explicitly

$$\hat{R}(f_{\boldsymbol{\theta}}) := \frac{1}{N} \sum_{i=1}^{n} \ell(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i). \tag{1}$$

The aim of data augmentation is to improve the approximation $\hat{R}$ of the statistical risk R when dealing with a dataset that is not sufficiently representative of the data distribution. To do so, we consider transformations of the feature vector $g : \mathcal{X} \to \mathcal{X}$, taken from the (possibly infinite) transformation set $\mathcal{G}$. Common examples include rotations and translations in images. Data augmentation leverages these transformations to generate new data pairs $(g\mathbf{x}, y)$ by sampling transformations according to a probability distribution $\mathfrak{G}$ over $\mathcal{G}$, leading to the learning problem

$$\underset{\boldsymbol{\theta}\in\Theta}{\text{minimize}} \;\; \hat{R}_{\text{aug}}(f_{\boldsymbol{\theta}}) := \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{g\sim\mathfrak{G}} \left[\, \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}_i), y_i) \right]. \tag{2}$$

Note that the empirical risk approximation $\hat{R}$ in (1) can be interpreted as an approximation of the data distribution $\mathfrak{D}$ by a discrete distribution that places atoms on each data point. In that sense, $\hat{R}_{\text{aug}}$ in (2) can be thought of as the Vicinal Risk Minimization (Chapelle et al., 2000) counterpart of (1), in which the atoms on $\mathbf{x}_i$ are replaced by a local distribution over the transformed samples $g\mathbf{x}_i$, i.e.,

$$\hat{R}_{\text{aug}}(f_{\boldsymbol{\theta}}) = \frac{1}{N} \sum_{i=1}^{N} \int \ell\left(h(g\mathbf{x}_i), y_i\right) dP(g\mathbf{x}_i), \tag{3}$$

where the distribution P over $\mathcal{X}$ is induced by the distribution $\mathfrak{G}$ over $\mathcal{G}$. As it can be seen from (3) if $\mathfrak{G}$ is not chosen adequately, $\hat{R}_{aug}$ can be a poor estimate of R, introducing biases that outweigh the benefits of data augmentation (Chen et al., 2019; Shao et al., 2022).

3 DATA AUGMENTATION VIA INVARIANCE CONSTRAINTS

Finding a transformation distribution $\mathfrak{G}$ that leads to a good approximation of the underlying distribution of the data $\mathfrak{D}$ in (3) can be challenging. What is more, using a fixed $\mathfrak{G}$ as in (2) prevents us from controlling when and how much augmentation is used during training, running the risk of biasing the final solution. On the other hand, it is straightforward to specify a set of transformations $\mathcal{G}$ to which the solution should be *approximately* invariant (e.g., image rotations and translations).

In the next sections, we explain how a data augmentation distribution can be obtained from such an invariance requirement. We first show how *invariance* can be interpreted as an augmentation distribution (Section 3.1). We then incorporate this invariance in a constrained learning problem (Section 3.2).

3.1 FROM INVARIANCE TO AN AUGMENTATION DISTRIBUTION

The goal of this section is to replace the task of choosing an augmentation distribution by the task of choosing a set of transformations we wish to be (approximately) invariant to. To do so, we we will show how invariance can be used to implicitly determine the augmentation distribution.

However, rather than requiring the output of the model to be invariant, i.e., $f_{\boldsymbol{\theta}}(\mathbf{x}) = f_{\boldsymbol{\theta}}(g\mathbf{x})$ for all $g \in \mathcal{G}$, we will consider invariance in terms of quality of its predictions as evaluated by the loss function. Namely,

$$\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), y) = \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y), \ \forall g \in \mathcal{G}.$$

This notion of invariance explicitly incorporates the structure of the learning task by using the loss to identify which changes in the output of the model would lead to a significant change in prediction performance. More precisely, we want to limit the difference in our model's performance on a sample and its transformed versions, i.e., we wish to have $|\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), y)) - \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y))|$ small for all g in $\mathcal{G}$. Equivalently, we wish to control the magnitude of

$$\max_{g \in \mathcal{G}} |\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), y) - \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)|. \tag{4}$$

However, rather controlling (4) for all $x \in \mathcal{X}$, which may be overly conservative, we want to restrict our attention to the support of the data distribution. Furthermore, we wish to weight different inputs depending on their probability, in order to reduce the importance of unlikely or pathological cases. We therefore average (4) over the data distribution to obtain

$$R_{\text{inv}}(f_{\boldsymbol{\theta}}) := \mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}\left[\max_{g \in \mathcal{G}} |\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), y) - \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)|\right]. \tag{5}$$

To connect the invariant risk R_{inv} in (5) to the data augmentation formulation in (2), notice that it can be bounded, using triangle inequality and the monotonicity of the expectation, by

$$R_{\text{inv}}(f_{\boldsymbol{\theta}}) \leq \mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}\left[\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), y)\right] + \mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}\left[\max_{g \in \mathcal{G}} \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)\right]. \tag{6}$$

The first term is simply the statistical risk R that is usually minimized in order to tackle the learning task. The second term resembles the objective typically found in adversarial learning problems, e.g., (Madry et al., 2017). This term, it turns out, can be interpreted as an augmentation distribution.

As shown by Robey et al. (2021a), the maximisation of the loss over transformations can be written as the semi-infinite constrained optimization problem

$$\max_{g \in \mathcal{G}} \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y) = \sup_{\lambda \in \mathcal{L}_+^2} \int_{\mathcal{G}} \lambda(g)\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)dg. \tag{7}$$
$$\text{s. to} \int_{\mathcal{G}} \lambda(g)dg = 1$$

Notice that the solution of this optimization problem $\lambda^\star(g)$ is a non-negative, normalized function and can therefore be interpreted as a distribution over transformations that depends on the sample

point $(\mathbf{x}, y)$ as well as the model $f_{\boldsymbol{\theta}}$. This allows us to re-interpret the maximization over $\mathcal{G}$ as an expectation, i.e.,

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}\left[\max_{g\in\mathcal{G}} \ell(f_{\boldsymbol{\theta}}\left(g\mathbf{x}\right), y)\right] = \mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}\left[\mathbb{E}_{g\sim\lambda^\star}\left[\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)\right]\right]. \tag{8}$$

Observe that the right-hand side of (8) resembles the statistical form of the data augmentation objective in (2).

3.2 A Constrained Learning Approach

Returning to the invariant risk bound in (6), notice that it is composed of two parts, namely, the statistical risk R from (SRM) and what we have shown in (8) to be a intance/model-dependent data augmentation. However, in order to address the potential biases introduced by transformations, rather than modifying the objective as in standard data augmentation, we propose to combine these two terms in a constrained learning problem. Explicitly,

$$\begin{aligned} P^\star \;=\; \min_{\boldsymbol{\theta}\in\Theta} \quad & \mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}\left[\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), y)\right] \\ \text{s. to} \;\; & \mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}\left[\mathbb{E}_{g\sim\lambda^\star}\left[\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)\right]\right] \leq \epsilon\,. \end{aligned} \tag{CSRM}$$

Notice that this formulation tackles the two terms forming the invariant risk bound in (6), but instead of combining them directly, it incorporates the data augmentation term as a constraint in the typical statistical risk minimization problem (SRM). This formulation has the advantage that if a solution to the unconstrained problem is feasible, i.e., satisfies the invariance constraint in (CSRM), the presence of that constraint has no effect on the statistical problem. Yet, it can be beneficial when approximating the solution of (CSRM) empirically. We will explore this fact in the next section, where we tackle the practical challenges involved in solving (CSRM).

4 Algorithm Development

Solving (CSRM) presents two challenges. First, it is a constrained statistical learning problem, which involves the unknown data distribution $\mathfrak{D}$. We address this by resorting to an empirical dual problem as explained on Section 4.1. Second, it can be hard to sample from $\lambda^\star$. We address this by introducing a smooth approximation that leverages MCMC methods on Section 4.2.

4.1 Empirical Dual Constrained Learning

To tackle the invariance-constrained statistical risk minimization problem we leverage recent duality results in constrained learning theory (Chamon et al., 2021), that approximate the problem by its empirical dual

$$D^\star_{\text{emp}} = \max_{\gamma\geq 0}\min_{\boldsymbol{\theta}\in\Theta} \frac{1}{n}\sum_{i=1}^n \ell(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i) + \gamma\left(\frac{1}{n}\sum_{i=1}^n \mathbb{E}_{g\sim\lambda^\star}\left[\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)\right]) - \epsilon\right). \tag{D-CERM}$$

The advantage of (D-CERM) is that it is an unconstrained problem that, provided we have enough samples and the parametrization is rich enough, can approximate the constrained statistical problem (CSRM). Namely, the difference between the optimal value of the empirical dual $D^\star_{\text{emp}}$ and the statistical primal $P^\star$, i.e., the empirical duality gap is bounded (Chamon et al., 2021).

As in regular data augmentation, we will also approximate the expectation over $\lambda^\star$ by sampling transformations as discussed in Section 4.2. Then, the problem D-CERM becomes an unconstrained deterministic problem, which can be solved using the algorithm described in Section 4.3.

Note that finding a Lagrangian minimizer for a fixed value of the dual variable (γ) is equivalent to minimising the risk under a fixed mixture augmentation distribution[1], where the value of γ controls the probability of sampling the identity. We can also interpret this as optimising a penalised or regularised learning objective. However, solving the constrained problem, namely maximising over γ, has fundamental differences.

[1]Explicitly, with probability $\frac{1}{1+\gamma}$, the identity is sampled, and with probability $\frac{\gamma}{1+\gamma}$, $g\sim\lambda^*$.

First, constraints explicit the requirement they represent. While the degree of invariance imposed should depend only on the statistical problem at hand, the value of γ needed to achieve it will depend on the sample size, the parametrization and the learning algorithm. In contrast, constrained learning dynamically adjusts the amount of augmentation — dictated by γ — to a particular learning setup.

Second, the optimal dual variable can give information about the trade-off minimising the loss over training samples and satisfying the invariance constraint. In penalised approaches, on the contrary, this trade-off is fixed.

Lastly, the aforementioned informativeness and interpretability can facilitate hyper-parameter tuning. The insights gathered from optimal dual variables can be leveraged a posteriori, for instance, to manually choose appropriate transformations, relax the invariance constraint levels, or change the learning setup (e.g. increase the capacity of the model class).

4.2 Sampling towards invariance

Sampling the augmentation distribution $\lambda^\star$ can be difficult when $\mathcal{G}$ is not finite and $f_{\boldsymbol{\theta}}$ is a deep neural network. Even when the transformation space is low dimensional, as in the case of translations and rotations, the highly non-convex loss landscape of these models makes the maximization over $\mathcal{G}$ challenging (Engstrom et al., 2017). If the optimal distribution $\lambda^\star$ is not smooth, it is challenging to sample from it with sufficient accuracy (Homem-de Mello and Bayraksan, 2014). Consequently, obtaining an unbiased estimator of $\mathbb{E}_{g\sim\lambda^\star}\left[\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)\right]$ may not be possible. Therefore, we add an L_2 norm penalisation, which promotes smoothness, to leverage MCMC methods.

We then define the *c-smoothed distribution* $\lambda_c^\star$ as a solution to the regularised problem

$$\lambda_c^\star \subseteq \operatorname{argmax}_{\lambda\in\mathcal{L}_+^2} \int_{\mathcal{G}} \lambda(g)\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)dg + c\int_{\mathcal{G}} \lambda(g)^2 dg,$$
$$\text{s. to} \int_{\mathcal{G}} \lambda(g)dg = 1$$

The regularization term introduces an optimality gap with respect to worst case perturbations, i.e., $\mathbb{E}_{g\sim\lambda_c^\star}\left[\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)\right] \leq \max_{g\in\mathcal{G}} \ell\left(f_{\boldsymbol{\theta}}\left(g\mathbf{x}\right), y\right)$. However, for particular values of c the regularized problem has a closed form solution (Robey et al., 2021a) that allows us to sample from it easily. Namely, there exists a constant $c \geq 0$ such that $\lambda_c^\star(\mathbf{x}, y, g) = \frac{\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)}{c}$.

Since $\lambda_c^\star$ is a smooth probability distribution, we do not need to estimate the multiplicative factor c to sample from it by leveraging Monte Carlo Markov Chain methods (MCMC).

MCMC methods (Hastings, 1970) are based on constructing a Markov chain that has the target distribution as an equilibrium distribution. *Independent* Metropolis Hastings uses a state independent - usually fixed - proposal for each step. In our case, it only requires applying a transformation and computing a forward pass of the neural network to evaluate the loss. This enables the use of non-differentiable transformations, and has the advantage that the density at consecutive proposals can be evaluated in parallel allowing speedups in the sampling step. Although MH methods thus allow to sample the proposal distribution with low computational cost, they exhibit random walk behaviour, which leads to slow convergence in high dimensional settings (Dellaportas and Roberts, 2003; Holden et al., 2009). However, a uniform augmentation distribution has been shown to give good results (Müller and Hutter, 2021) and sampling λ_c^* accurately is not a concern given that we are interested in promoting *approximate* invariance.

We can then approximate $\mathbb{E}_{g\sim\lambda_c^\star}\left[\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}_i), y_i)\right]$ by sampling transformations according to the loss. Namely, we can obtain a set of m samples drawn from $\lambda_c^\star$ and approximate the expectation over the group by the sample mean

$$\mathbb{E}_{g\sim\lambda_c^\star}\left[\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}_i), y_i)\right] \approx \frac{1}{m}\sum_{j=1}^{m} \ell(f_{\boldsymbol{\theta}}(\mathbf{g_j}\mathbf{x_i}), y_i),$$

where $g_1, \ldots, g_m \overset{\text{i.i.d.}}{\sim} \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}_i), y_i)/c$ are m transformations sampled from the smoothed distribution $\lambda_c^\star(f_{\boldsymbol{\theta}}, \mathbf{x}_i, y_i)$.

In the next section, the implementation of independent-MH with a uniform proposal is described in Algorithm 2, together with the primal-dual augmentation algorithm.

4.3 PRIMAL-DUAL ALGORITHM

Since the cost of the inner minimization, i.e. training the model, can be high, we adopt an alternating update scheme (K. J. Arrow and Uzawa., 1958) for the primal and dual variables, as in (Chamon et al., 2021; Fioretto et al., 2020).

A bounded empirical duality gap *does not* guarantee that the primal variables obtained after running the alternating primal-dual algorithm 1 and solving the saddle point problem approximately are near optimal or approximately feasible. Although stronger *primal recovery* guarantees can be obtained by randomizing the learning algorithm (Chamon et al., 2021), it requires storing model parameters $\boldsymbol{\theta}$ at each iteration, and there is empirical evidence (Chamon et al., 2021; Robey et al., 2021a; Elenter et al., 2022; Shen et al., 2022; Cervino et al., 2022; Zhang et al., 2022) that good solutions can still be obtained without randomization.

Algorithm 2 describes transformation sampling. By keeping only one sample ($m = 1$) we recover the usual augmentation setting, that yields one augmentation per sample in the training batch. In our experiments we address this setting, since it has lower computational costs. However, simply keeping more samples from the chain ($m > 1$) allows to extend the method to the batch augmentation setting (Hoffer et al., 2020), which creates several augmented samples from the same instance in each batch.

Although several steps of the chain may be required to deviate enough from the proposal distribution, we show in the experimental section that sampling the constraint approximately by using few sampling steps suffices.

Algorithm 1 Primal-Dual Augmentation

$\lambda = 0,\ \boldsymbol{\theta} = \boldsymbol{\theta}_0.$
for Batch in $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ **do**
  for $(\mathbf{x}_i, y_i) \in$ Batch **do**
    $g_{i1}, \ldots, g_{im} \sim^{iid} \ell(f_{\boldsymbol{\theta}}(g\mathbf{x}_i), y_i)/c$ ▷ Sample transformations
  $s = \frac{1}{|\text{Batch}|} \sum_{(\mathbf{x}_i, y_i) \in \text{Batch}} \left[\frac{1}{m} \sum_{j=1}^m \ell(f_{\boldsymbol{\theta}}(g_{ij}\mathbf{x}_i), y_i)\right] - \epsilon$ ▷ Evaluate constraint slack
  $\ell = \frac{1}{|\text{Batch}|} \sum_{(\mathbf{x}_i, y_i) \in \text{Batch}} \ell(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i)$ ▷ Evaluate Loss
  $\hat{L} = \ell + \gamma s$ ▷ Compute Lagrangian
  $\boldsymbol{\theta} = \boldsymbol{\theta} - \eta_p \nabla_{\boldsymbol{\theta}} \hat{L}$ ▷ Primal update
  $\gamma = [\gamma + \eta_d s]_+$ ▷ Dual Update

Algorithm 2 Independent MH sampler

$g^{(0)} \sim \mathcal{U}(\mathcal{G})$ ▷ Sample initial State
$\ell^{(0)} = \ell\left(f_{\boldsymbol{\theta}}\left(g^{(0)}\mathbf{x}\right), y\right)$ ▷ Evaluate loss
for $t = 1, \ldots, n_{steps}$ **do**
  $g_{\text{prop}} \sim \mathcal{U}(\mathcal{G})$ ▷ Sample next proposal
  $\ell_{\text{prop}} = \ell\left(f_\theta\left(g_{\text{prop}}\mathbf{x}\right), y\right)$ ▷ Evaluate Loss
  $p = \min\left(1, \frac{\ell_{\text{prop}}}{\ell^{(t-1)}}\right)$ ▷ Acceptance Prob
  w.p. p: ▷ Accept/Reject
    $g^{(t)} = g_{\text{prop}}, \ell^{(t)} = \ell_{\text{prop}}$
  else:
    $g^{(t)} = g^{(t-1)}, \ell^{(t)} = \ell^{(t-1)}$

5 AUTOMATIC DATA AUGMENTATION EXPERIMENTS

This section showcases Algorithm 1 in common image classification benchmarks. We compare it to state-of-the-art data augmentation methods in terms of classification accuracy. Furthermore, we discuss other advantageous properties of our method through ablations. Namely, we focus on the ability to control the effect of data augmentation by modifying the constraint level, the informativeness of dual variables, and the benefits of adapting the augmentation distribution throughout

training. We follow the setup (including the transformation sets) used in recent automatic data augmentation literature (Müller and Hutter, 2021). A complete list of transformations together with other hyperparameters and training settings can be found on Appendix B.

Throughout these experiments, we fixed the number of steps of the MH sampler (Algorithm 2) to two, which has the added advantage of reducing the computational cost of evaluating proposals. The constraint level was determined by a grid search targeting cross-validation accuracy. As shown in Table 1, in both transformation sets considered, we find that our approach improves or closely matches existing approaches. The failure to achieve large improvements in accuracy over baselines, which has been attributed to a stagnation in data augmentation research (Müller and Hutter, 2021), can also reflect the limits of the benchmarking setup.

	Standard			Wide	
	TA	DeepAA	**OURS**	TA	**OURS**
CIFAR10					
Wide-ResNet-40-2	96.62	–	$\mathbf{96.76 \pm 0.10}$	96.32	$\mathbf{97.05 \pm 0.13}$
Wide-ResNet-28-10	97.46	97.56	$\mathbf{97.75 \pm 0.06}$	97.46	$\mathbf{97.85 \pm 0.12}$
CIFAR100					
Wide-ResNet-40-2	79.99	–	$\mathbf{80.83 \pm 0.22}$	79.86	$\mathbf{81.19 \pm 0.24}$
Wide-ResNet-28-10	83.54	**84.02**	83.53 ± 0.12	84.33	$\mathbf{84.89 \pm 0.08}$
SVHNcore					
Wide-ResNet-28-10	98.05	–	$\mathbf{98.15 \pm 0.07}$	**98.11**	98.01 ± 0.03

Table 1: Image Classification accuracy for WideResnet architectures (Zagoruyko and Komodakis, 2016) trained using different augmentation policies, defined on standard (Cubuk et al., 2018) and wider (Müller and Hutter, 2021) augmentation search spaces. We include previous state-of-the-art results reported by TA (Müller and Hutter, 2021) and DeepAA (Zheng et al., 2022), and 95% confidence intervals computed over five runs for our method. In CIFAR datasets our approach leads to an improvement in accuracy, whereas in SVHN (Netzer et al., 2011) it leads to a degradation in performance in the wide setup.

Regardless, our approach yields improvements in test accuracy over a baseline model without augmentation for a wide range of constraint levels (Figure 1). This illustrates the robustness of the solution to this hyperparameter. Observe also that as the constraint is relaxed (by increasing ϵ), the training error decreases while the generalization gap, i.e., the difference between train and test errors, increases. In other words, by loosening the invariance requirement the model can fit better to training samples at the cost of worse generalization. Eventually, only the generalization gap increases while the training error stagnates ($\epsilon > 2.1$ for CIFAR100 and $\epsilon > 0.8$ for CIFAR10). This transition occurs at the same point at which the final value of the dual variable γ from (D-CERM) essentially vanishes (darker color). This showcases the infromativeness of the dual variable. In the case of CIFAR10, even the training error begins to increase at that point, suggesting that the invariance requirement need not be at odds with accuracy.

Not only does Algorithm 1 tune the effect of data augmentation on the solution (by adapting γ in step 9), but also modifies the distribution over transformations during training (step 4). To showcase the benefits of this over the use of a fixed distribution, Figure 2 compares the results obtained using our approach (sampling according to $\lambda_c^\star$) and one where step 4 is replaced by a uniform sampling over transformations. For the same constraint levels, lower test errors are obtained by sampling transformations according to $\lambda_c^\star$, i.e., promoting invariance. Note also that for $\epsilon = 2.1$ in CIFAR100 and $\epsilon = 0.8$ for CIFAR10, the performance gap is quite large. Once again, this occurs at the point in which γ vanishes (darker color) for the uniform distribution, i.e., no data augmentation occurs by the end of training. At this point, however, there is still value in promoting invariance by sampling from $\lambda_c^\star$ as evidenced by the positive value of the dual variable (lighter color) in this approach.

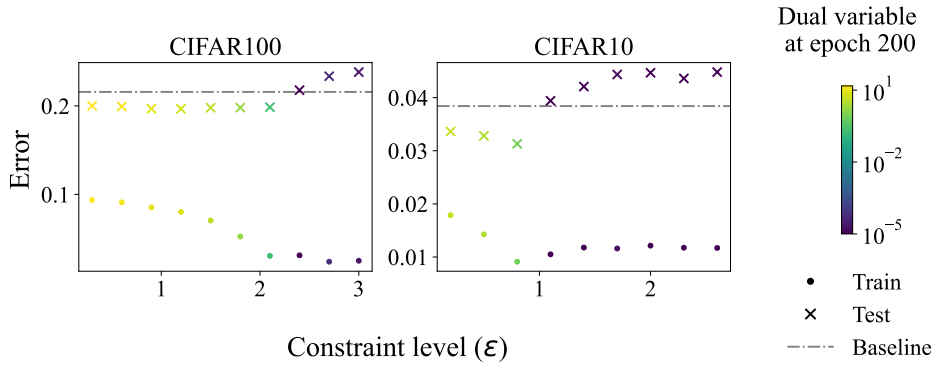

Figure 1: Constraint level ablation for WideResnet-40-2 in CIFAR datasets. We plot error rates computed over the train and test set and averaged over five runs, for different constraint levels. We include the test error of the unconstrained baseline (without augmentation) for comparison. The color of markers denotes the final value of the dual variable.

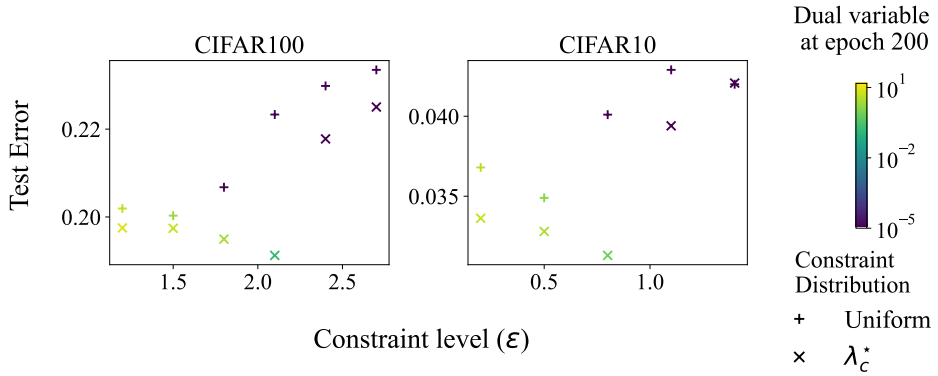

Figure 2: We compare our approach to a constraint on the uniform distribution, for WideResnet-40-2 in CIFAR datasets, at different constraint levels. We plot error rates computed over test set and averaged over five runs. Markers denote the augmentation distribution. The color of markers denotes the final value of the dual variable.

6 Related Work

Zhang et al. (2020) have shown that using adversarial transformations - which as already mentioned is related to promoting invariance - can give competitive results with respect to other automatic augmentation methods in image classification. However, Blaas et al. (2021) have since evidenced the importance of two factors: the implicit learning curricula and the suboptimality of the adversarial used by Zhang et al. (2020), which mitigate the biases introduced by worst-case transformations. Furthermore, Blaas et al. (2021) report that an explicit cyclic curricula in which augmentations are *mild* at first, then get harder as training progresses, and finally revert to milder augmentations at the end of training, performs better empirically. We note some interesting commonalities with our approach and experimental results. First, the dynamics of our primal-dual algorithm resemble the aforementioned heuristically defined curricula. Second, the suboptimality with respect to worst case perturbations can be related to the smoothed approximation used in our approach.

Lastly, Xu et al. (2021) have motivated a two-stage data augmentation policy, as learning subject to a constraint in the risk under a fixed augmentation distribution. Unlike our approach, this distribution still needs to be user defined.

7 CONCLUSION

In this paper, we have proposed a constrained learning approach for automatic data augmentation, which instead of using augmented samples as a modified learning objective, imposes an invariance-constraint. We have shown that this yields an augmentation distribution that adapts during training, and found that coarse sampling approximations based on MCMC methods can improve generalization in automatic data augmentation benchmarks. Furthermore, our experiments showed that the dual variable can give insights about the resulting augmentation distribution. We also found that strictly feasible solutions were obtained for a wide range of constraint levels, with notably different generalization gaps, and that in some cases tightening the constraint even led to a lower training error. Thus, analysing the interplay between the learning problem and the optimization algorithm is a promising future work direction.

REFERENCES

Benton, G., Finzi, M., Izmailov, P., and Wilson, A. G. (2020). Learning invariances in neural networks from training data. *Advances in Neural Information Processing Systems*, 33:17605–17616.

Bietti, A. and Mairal, J. (2019). Group invariance, stability to deformations, and complexity of deep convolutional representations. *The Journal of Machine Learning Research*, 20(1):876–924.

Blaas, A., Suau, X., Ramapuram, J., Apostoloff, N., and Zappella, L. (2021). Challenges of adversarial image augmentations. In *I (Still) Can't Believe It's Not Better! NeurIPS 2021 Workshop*.

Caruana, R., Lawrence, S., and Giles, C. (2000). Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In Leen, T., Dietterich, T., and Tresp, V., editors, *Advances in Neural Information Processing Systems*, volume 13. MIT Press.

Cataltepe, Z., Abu-Mostafa, Y. S., and Magdon-Ismail, M. (1999). No free lunch for early stopping. *Neural Computation*, 11(4):995–1009.

Cervino, J., Ruiz, L., and Ribeiro, A. (2022). Training stable graph neural networks through constrained learning. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4223–4227.

Chamon, L. F. O., Paternain, S., Calvo-Fullana, M., and Ribeiro, A. (2021). Constrained learning with non-convex losses. *CoRR*, abs/2103.05134.

Chapelle, O., Weston, J., Bottou, L., and Vapnik, V. (2000). Vicinal risk minimization. *Advances in neural information processing systems*, 13.

Chatzipantazis, E., Pertigkiozoglou, S., Dobriban, E., and Daniilidis, K. (2021). Learning augmentation distributions using transformed risk minimization. *arXiv preprint arXiv:2111.08190*.

Chen, S., Dobriban, E., and Lee, J. H. (2019). A group-theoretic framework for data augmentation.

Cheung, T.-H. and Yeung, D.-Y. (2022). Adaaug: Learning class- and instance-adaptive data augmentation policies. In *International Conference on Learning Representations*.

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

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

Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V. (2020). Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Dellaportas, P. and Roberts, G. O. (2003). An introduction to mcmc. In *Spatial statistics and computational methods*, pages 1–41. Springer.

DeVries, T. and Taylor, G. W. (2017). Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*.

Duvenaud, D., Maclaurin, D., and Adams, R. (2016). Early stopping as nonparametric variational inference. In Gretton, A. and Robert, C. C., editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 1070–1077, Cadiz, Spain. PMLR.

Elenter, J., Naderializadeh, N., and Ribeiro, A. (2022). A lagrangian duality approach to active learning. *ArXiv*, abs/2202.04108.

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., and Madry, A. (2017). Exploring the landscape of spatial robustness.

Esteves, C., Allen-Blanchette, C., Makadia, A., and Daniilidis, K. (2017). 3d object classification and retrieval with spherical cnns. *CoRR*, abs/1711.06721.

Finzi, M., Stanton, S., Izmailov, P., and Wilson, A. G. (2020). Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data.

Fioretto, F., Hentenryck, P. V., Mak, T. W., Tran, C. D., Baldo, F., and Lombardi, M. (2020). Lagrangian duality for constrained deep learning. In *ECML/PKDD*.

Fukushima, K. and Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, pages 267–285. Springer.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109.

Hataya, R., Zdenek, J., Yoshizoe, K., and Nakayama, H. (2020). Faster autoaugment: Learning augmentation strategies using backpropagation. In *European Conference on Computer Vision*, pages 1–16. Springer.

Ho, D., Liang, E., Chen, X., Stoica, I., and Abbeel, P. (2019). Population based augmentation: Efficient learning of augmentation policy schedules. In *International Conference on Machine Learning*, pages 2731–2741. PMLR.

Hoffer, E., Ben-Nun, T., Hubara, I., Giladi, N., Hoefler, T., and Soudry, D. (2020). Augment your batch: Improving generalization through instance repetition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8129–8138.

Holden, L., Hauge, R., and Holden, M. (2009). Adaptive independent Metropolis–Hastings. *The Annals of Applied Probability*, 19(1):395 – 413.

Homem-de Mello, T. and Bayraksan, G. (2014). Monte carlo sampling-based methods for stochastic optimization. *Surveys in Operations Research and Management Science*, 19(1):56–85.

Ji, Z., Li, J. D., and Telgarsky, M. (2021). Early-stopped neural networks are consistent. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*.

Jumper, J. M., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Zídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D. A., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., and Hassabis, D. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, 596:583 – 589.

K. J. Arrow, L. H. and Uzawa., H. (1958). *Studies in linear and non-linear programming*. Stanford University Press.

Kondor, R. and Trivedi, S. (2018). On the generalization of equivariance and convolution in neural networks to the action of compact groups.

Li, M., Soltanolkotabi, M., and Oymak, S. (2020a). Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *International conference on artificial intelligence and statistics*, pages 4313–4324. PMLR.

Li, Y., Hu, G., Wang, Y., Hospedales, T., Robertson, N. M., and Yang, Y. (2020b). Dada: differentiable automatic data augmentation. *arXiv preprint arXiv:2003.03780*.

Lim, S., Kim, I., Kim, T., Kim, C., and Kim, S. (2019). Fast autoaugment. *Advances in Neural Information Processing Systems*, 32.

LingChen, T. C., Khonsari, A., Lashkari, A., Nazari, M. R., Sambee, J. S., and Nascimento, M. A. (2020). Uniformaugment: A search-free probabilistic data augmentation approach. *ArXiv*, abs/2003.14348.

Liu, A., Huang, Z., Huang, Z., and Wang, N. (2021). Direct differentiable augmentation search. *CoRR*, abs/2104.04282.

Lyle, C., van der Wilk, M., Kwiatkowska, M., Gal, Y., and Bloem-Reddy, B. (2020). On the benefits of invariance in neural networks.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Mallat, S. (2016). Understanding deep convolutional networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150203.

Müller, S. G. and Hutter, F. (2021). Trivialaugment: Tuning-free yet state-of-the-art data augmentation. *CoRR*, abs/2103.10158.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning. In *Advances in Neural Information Processing Systems (NIPS)*.

Olah, C., Cammarata, N., Voss, C., Schubert, L., and Goh, G. (2020). Naturally occurring equivariance in neural networks. *Distill*. https://distill.pub/2020/circuits/equivariance.

Petrini, L., Favero, A., Geiger, M., and Wyart, M. (2021). Relative stability toward diffeomorphisms indicates performance in deep nets. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*.

Püschel, M. and Moura, J. M. F. (2006). Algebraic signal processing theory. *CoRR*, abs/cs/0612077.

Rice, L., Wong, E., and Kolter, Z. (2020). Overfitting in adversarially robust deep learning. In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8093–8104. PMLR.

Robey, A., Chamon, L. F. O., Pappas, G. J., Hassani, H., and Ribeiro, A. (2021a). Adversarial robustness with semi-infinite constrained learning. *ArXiv*, abs/2110.15767.

Robey, A., Pappas, G. J., and Hassani, H. (2021b). Model-based domain generalization. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*, volume 34, pages 20210–20229. Curran Associates, Inc.

Sannai, A., Imaizumi, M., and Kawano, M. (2019). Improved generalization bounds of group invariant / equivariant deep networks via quotient feature spaces.

Shao, H., Montasser, O., and Blum, A. (2022). A theory of pac learnability under transformation invariances.

Shen, Z., Cervino, J., Hassani, H., and Ribeiro, A. (2022). An agnostic approach to federated learning with class imbalance. In *International Conference on Learning Representations*.

Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48.

Weiler, M. and Cesa, G. (2019). General e(2)-equivariant steerable cnns. *CoRR*, abs/1911.08251.

Xu, Y., Noy, A., Lin, M., Qian, Q., Hao, L., and Jin, R. (2021). Wemix: How to better utilize data augmentation.

Zagoruyko, S. and Komodakis, N. (2016). Wide residual networks. *CoRR*, abs/1605.07146.

Zhang, H., Zhang, Y.-F., Liu, W., Weller, A., Schölkopf, B., and Xing, E. P. (2022). Towards principled disentanglement for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8024–8034.

Zhang, X., Wang, Q., Zhang, J., and Zhong, Z. (2020). Adversarial autoaugment. In *International Conference on Learning Representations*.

Zheng, Y., Zhang, Z., Yan, S., and Zhang, M. (2022). Deep autoaugment. In *International Conference on Learning Representations*.

Zhou, F., Li, J., Xie, C., Chen, F., Hong, L., Sun, R., and Li, Z. (2021). Metaaugment: Sample-aware data augmentation policy learning. In *AAAI*.

A ADDITIONAL RELATED WORK

A.1 CONSTRAINED LEARNING AND DATA AUGMENTATION

Xu et al. (2021) also formulate data augmentation as a constrained learning problem. They impose a constraint on the excess risk, i.e. the difference between the statistical risk and its optimal value, on augmented data. Thus the constraint level on the augmented risk is also determined by the data distribution, augmentations considered, and model class, and the existence of a strictly feasible point is guaranteed.

$$\min_{\boldsymbol{\theta}\in\Theta} R(f_{\boldsymbol{\theta}}) \quad \text{s.t.} \quad R_{aug}\left(f_{\boldsymbol{\theta}}\right) - \min_{\hat{\boldsymbol{\theta}}\in\Theta} R_{aug}(f_{\hat{\boldsymbol{\theta}}}) \leq \epsilon, \quad \epsilon > 0,$$

where R and R_{aug} are the statistical risk under the original and augmented distribution, exlipicity

$$R(f_{\boldsymbol{\theta}}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathfrak{D}}[\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), y)], \quad R_{\text{aug}}(f_{\boldsymbol{\theta}}) = \mathbb{E}_{\substack{(\mathbf{x},y)\sim\mathfrak{D},\\ g\sim\mathfrak{G}}}[\ell(f_{\boldsymbol{\theta}}(g\mathbf{x}), y)].$$

Unlike our formulation, this formulation assumes a fixed distribution of augmentations $\mathfrak{G}$ is given.

By formulating it as a constrained problem, they aim to avoid introducing a bias when the data distribution is not invariant to augmentations. Two types of biases induced by augmentation are explicitly addressed, covariate shift (i.e. label-preserving augmentations) and concept shift (i.e. label mixing augmentations).

Interestingly, they show that under some conditions on the risk, augmented risk and constraint level, by utilizing the augmented data to constrain the solution to a small region SGD can achieve lower error (Xu et al., 2021, Proposition 1).

Instead of resorting to constrained optimization algorithms, they propose a two stage algorithm that consists of first finding an approximate minimizer of the augmented risk and then using that solution as an initialisation to the (unconstrained) statistical risk minimization problem. The first stage obtains a feasible point, and then under some conditions the SGD iterates obtained when solving the second problem remain feasible (Xu et al., 2021, Theorem 1).

A.2 CONSTRAINED LEARNING AND DOMAIN GENERALIZATION

Domain Generalization (DG) involves training the model in different but related data distributions, and evaluating in an unseen *domain*. For example, a common benchmark consists of domains created by rotating images in the MNIST dataset by different angles. Constrained formulations have been proposed in this context (Robey et al., 2021b; Zhang et al., 2022), that enforce invariance under learnt *domain translation* transformations. In contrast, our approach addresses pre-defined transformations that are commonly used in data augmentation pipelines in order to improve generalization in the same domain used to train the model.

A.3 GROUP INVARIANCE

The relationship between convolutional structure and equivariance has been long known in algebraic signal processing theory (Püschel and Moura, 2006). Recently, necessity results have re-gained attention in the context of neural network architecture design (Kondor and Trivedi, 2018). Several Group Convolutional neural network architectures that generalize CNNs to different groups by leveraging group convolutions have been proposed (Cohen and Welling, 2016; Esteves et al., 2017; Finzi et al., 2020). In the case of images, it has been shown that that SE2 equivariant layers can be implemented efficiently using regular 2D convolutions (Weiler and Cesa, 2019). This approach allows to derive a parametrization for CNN filters under finite subgroups of SE2. Among other works (Bietti and Mairal, 2019; Sannai et al., 2019) give theoretical analyses of the benefits of group invariance in learning settings.

As already mentioned, achieving invariance through architecture design is both challenging and limited in the sense that it relies on transformations having a specific structure (e.g: a group). The goal of exact invariance over the whole input space is more strict than the approximate invariance notion that our work addresses. As argued by Mallat (2016), CNNs can learn *locally* invariant

features with respect to arbitrary transformation groups, which could explain their generalization properties. Furthermore, empirical studies evidence modern CNN architectures learn approximately equivariant features to transformations such as scaling and rotations (Olah et al., 2020), or diffeomorphisms (Petrini et al., 2021), even when trained without direct augmentation.

However, commonly used augmentations do not form a group. Our approach does not require this structure.

B EXPERIMENTAL SETUP DETAILS

B.1 TRANSFORMATIONS

As in recent automatic augmentation literature (Ho et al., 2019; Lim et al., 2019; Hataya et al., 2020; Zhang et al., 2020; Cubuk et al., 2020; LingChen et al., 2020; Zhang et al., 2020; Müller and Hutter, 2021; Zhou et al., 2021; Zheng et al., 2022; Cheung and Yeung, 2022), we focus on image classification datasets and employ a transformation search space comprising 14 operations, introduced by Cubuk et al. (2018). For those that have parameters, their magnitudes are discretized in thirty levels, which does not compromise performance and greatly reduces the search space (Cubuk et al., 2020). Most approaches compose transformations, i.e. applying more than one transformation to the same image. However, recently Müller and Hutter (2021) have shown that applying only one transformation at a time, defined over a wider magnitude space (noted Wide in Table 2) can outperform other approaches. We thus use the same transformation space as (Müller and Hutter, 2021).

Table 2 from Müller and Hutter (2021) lists the operations and their magnitude ranges. In our experiments we used the wide search space, the standard ranges from Cubuk et al. (2020) are included for comparison. We extend the codebase provided by Müller and Hutter (2021), which uses the Pillow[2] implementation of all transformations except for cutout, and refer to its documentation for further details about image operations.

Operation	**Magnitude**	
	Standard	Wide
Identity	–	–
ShearX	$[-0.3, 0.3]$	$[-0.99, 0.99]$
ShearY	$[-0.3, 0.3]$	$[-0.99, 0.99]$
TranslateX	$[-10, 10]$	$[-32, 32]$
TranslateY	$[-0.45, 0.45]$	$[-32, 32]$
Rotate	$[-30, 30]$	$[-135, 135]$
AutoContrast	–	–
Invert	–	–
Equalize	–	–
Solarize	$[0, 256]$	
Posterize	$[4, 8]$	$[2, 8]$
Contrast	$[0.1, 1.9]$	$[0.01, 2]$
Color	$[0.1, 1.9]$	$[0.01, 2]$
Brightness	$[0.1, 1.9]$	$[0.01, 2]$
Sharpness	$[0.1, 1.9]$	$[0.01, 2]$
Flips	–	–
Cutout	$16(60)$	$16(60)$
Crop	–	–

Table 2: Pillow image operations in the data augmentation search space and the range of magnitudes corresponding to the standard (Cubuk et al., 2020) and wide (Müller and Hutter, 2021) search spaces. Some operations do not use magnitude parameters.

B.2 TRAINING SETUP

In order to enable comparisons and reproducibility we use the same training pipeline as in previous works Müller and Hutter (2021) . We apply the vertical flip and the pad-and-crop augmentations and a 16 pixel cutout (DeVries and Taylor, 2017) after any augmentation method. We trained Wide-ResNet (Zagoruyko and Komodakis, 2016) models in the 40-2 and 28-10 configurations.

Except for epoch ablation experiments, we use SGD with Nesterov Momentum and a learning rate of 0.1, a batch size of 128, a 5e-4 weight decay, and cosine learning rate decay. In ablation experiments we also trained for 600 epochs and used a custom learning rate schedule. For the first 185 epochs

[2] https://github.com/python-pillow/Pillow

we followed the same cosine learning rate decay schedule, and then switching to a custom step learning rate scheduler detailed on Table 3. This schedule was implemented after observing that just scaling the cosine learning rate schedule to 600 epochs resulted in slow convergence, thus yielding solutions similar to training for 600 epochs, and failing reflect the effects of early stopping which this experiment addressed.

We used 2 MH steps for the smoothed adversarial unless stated otherwise, and a learning rate of 10e-3 for the dual ascent step.

No other hyperparameters were tuned or modified with respect to standard settings. The constraint level were set for Wide-ResNet-40-2 in the wide augmentation space by maximising three fold cross validation over the grid specified on Table 4. Then the constraint levels were adjusted for other architectures and search spaces so that the dual variables at the end of training were small but not zero (of the order of 10^{-1}, which empirically showed good results. The resulting constraint levels corresponding to the results in 1 are detailed in table 5.

Epochs	**LR Scheduler Step**
180-230	10
230-430	20
430-600	40

Table 3: Learning Rate Custom schedule used when training for 600 epochs. We use the standard scheduler for the first 180 epochs, and then update the LR only every n epochs, where n is the number indicated in the second column. This hand designed schedule outperforms using a cosine learning rate schedule for 600 epochs, but could improvements convergence speed and performance by exploring other LR-schedulers or tuning it.

Dataset	**Constraint Level Grid Range**
CIFAR10	[0.2, 2.3]
CIFAR100	[0.3, 2.7]
SVHNcore	[0.1, 1.5]

Table 4: Constraint level grid search space. For each dataset, we evaluated three fold cross validation accuracy for 8 evenly spaced constraint levels in the ranges given in the second column. The one with the highest cross validation score was then selected and used to train the model with the full dataset.

	Constraint Level	
	Standard	Wide
CIFAR10		
Wide-ResNet-40-2	0.6	0.8
Wide-ResNet-28-10	0.4	0.8
CIFAR100		
Wide-ResNet-40-2	0.9	0.8
Wide-ResNet-28-10	0.9	1.2
SVHNcore		
Wide-ResNet-28-10	0.1	0.2

Table 5: Constraint levels for different datasets and architectures, for the results presented in Table 1

C ADDITIONAL RESULTS

The following section contains additional ablations and discussions about our algorithm. As in previous sections, we use the wide Müller and Hutter (2021) augmentation space and CIFAR image classification benchmarks. Our main motivation is to analyse how the different hyperparameter choices and the learning algorithm affect the generalization and invariance of the obtained solutions. First, we show how the dual variables adapts to different constraint levels during training, and link its dynamics to heuristically defined learning curricula. We then evaluate the effect of training for more epochs and link our observations to the known properties of early stopping in unconstrained learning, in section C.2. In section C.3, we analyse how the sampling approximation affects regularisation, by performing an ablation on the number of MH steps. Lastly, we include the observed frequencies of sampled transformations for different setups, so as to obtain further insights in how the distribution adapts throughout training (Section C.4).

C.1 DUAL VARIABLE DYNAMICS

As already mentioned, the dual variables control the weight of augmented samples during training, thus balancing the trade-off between fitting the primal objective (i.e. loss over training samples) and satisfying the constraints (i.e. loss over transformed samples). In penalised approaches, on the contrary, this trade-off is fixed. In Figure 3 we show how the dual variable adapts to different constraint levels, for Wide-ResNet-40-2 in CIFAR datasets using the standard setup. Note that for stricter constraint levels, the algorithm has not converged when it reaches 200 epochs. We also observe that the dynamics of our primal-dual algorithm resembles the augmentation learning curricula proposed by Blaas et al. (2021).

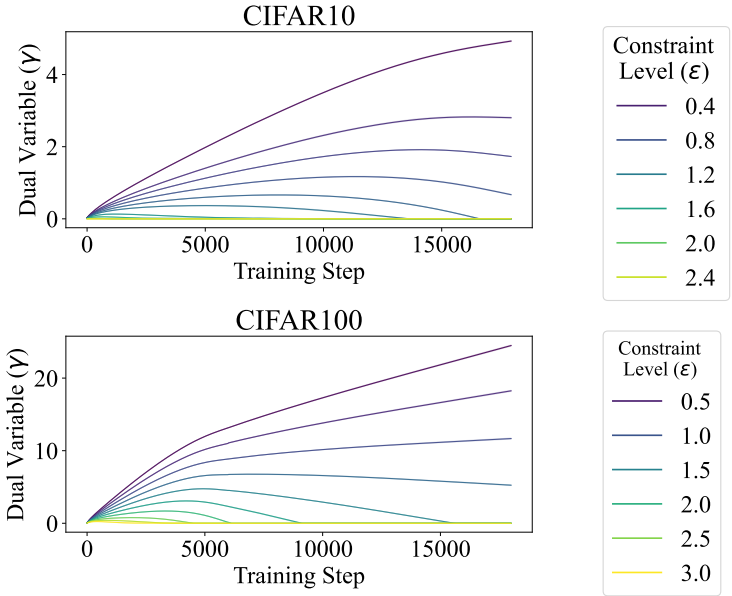

Figure 3: Evolution of dual variables during training for Wide-ResNet-40-2 in CIFAR datasets. For most levels, augmentation increases until the constraint becomes feasible and then decreases towards the end of training. For stricter levels, the algorithm does not converge in 200 epochs using the standard learning settings. However, the solutions obtained still show good properties.

C.2 EARLY STOPPING

Our approach has slower convergence than unconstrained approaches and stopping training arbitrarily - after a fixed number of epochs - can result in solutions that are unfeasible or sub-optimal. However, early stopping is a popular regulariser, particularly for neural networks trained through gradient descent. Several empirical (Caruana et al., 2000; Rice et al., 2020) and theoretical (Ji et al., 2021;

Li et al., 2020a; Duvenaud et al., 2016) results show its advantages in terms of generalisation and robustness to noisy training data. In general, the advantages of early stopping do not lie in the sub-optimality of the solution in terms of training error, but on nature of the regularization or prior imposed, which leads to non-uniform model selection among models with a given training error (Cataltepe et al., 1999).

To the best of our knowledge, there is no literature that explicitly addresses early stopping in empirical primal-dual learning. Whether the generalisation gap in constraint satisfaction can be reduced by early stopping regularisation, in the same manner early stopping regularisation can reduce the generalisation gap in unconstrained learning, thus remains unclear.

Figure 4 shows an ablation on the number of epochs. Non-zero dual variables and strict feasiblility show the algorithm has not yet converged at 200 epochs. At 600 epochs whereas constraint satisfaction shows little change, training loss decreases and the generalization gap increases. Thus, we observe early stopping has a larger impact on the primal objective than on the constraint. that although for stricter levels of the constraint the algorithm has not converged when training is stopped at 200 epochs, it can yield solutions that are still feasible and have a smaller generalization gap.

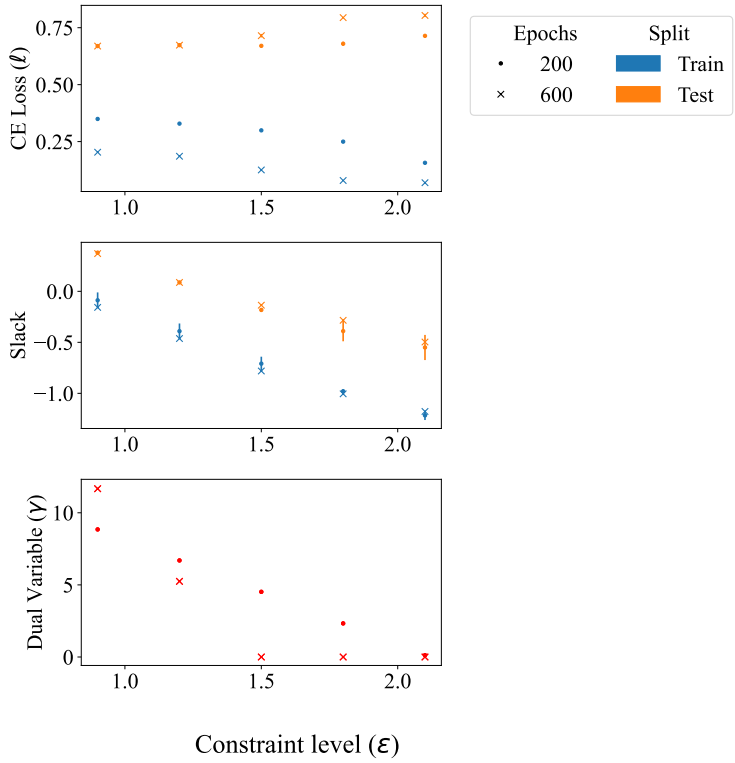

Figure 4: Training WideResnet-40-2 for 200 (standard) and 600 epochs in the CIFAR100 dataset, with different constraint levels.

C.3 Sampling Steps Ablation

As already mentioned in Section 4.3, we used the Metropolis-Hastings algorithm with independent uniform proposals. As shown in Figure 5, using more steps of the chain allows samples to deviate further from the uniform distribution, as measured by the decrease in entropy. For a fixed constraint level, augmentation distributions with higher entropy result in weaker regularization. As already mentioned, dual variables give information of the trade-off between fitting clean and augmented data. We observe the final value of the dual variable is highly correlated with the entropy of sampled transformations and final training loss. In 6 we show the evolution of dual variables for different sampling steps. Sampling distributions that are closer to worst case perturbations result in more stringent requirements, and thus dual variables adapt accordingly.

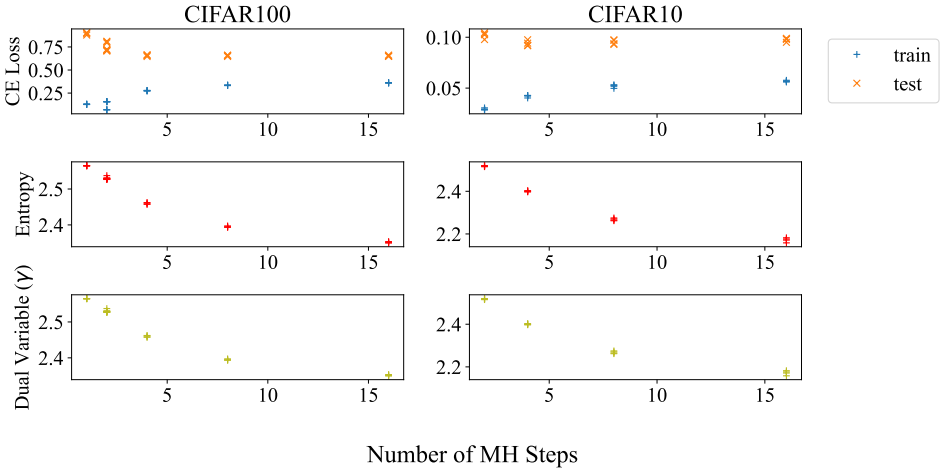

Figure 5: Number of Metropolis Hastings steps ablation for WideResnet-40-2 in CIFAR datasets. The constraint level is fixed for each dataset (0.8 in CIFAR10 and 2.1 for CIFAR100). We compute the entropy of the augmentations sampled at the last epoch of training. The gap between train and test loss and the final value of the dual variable, and the augmentation distribution entropy increase with the number of steps.

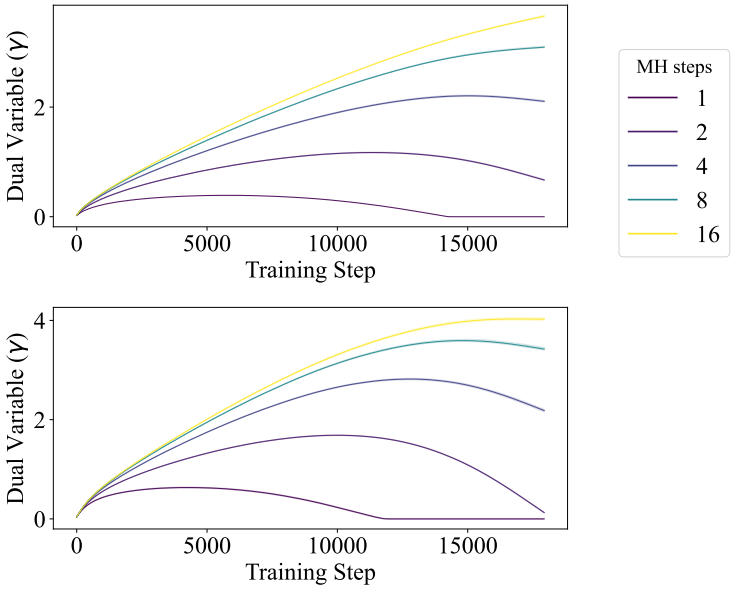

Figure 6: Evolution of dual variables during training for Wide-ResNet-40-2 in CIFAR datasets. The constraint level is fixed for each dataset (0.8 in CIFAR10 and 2.1 for CIFAR100). As the number of MH steps increases so does the growth of dual variables, which reflects harder to satisfy constraints.

C.4 SAMPLED TRANSFORMATIONS

We now include plots of the empirical transformation distributions. As shown in Figure 7, the frequency with which transformations are sampled varies depending on the dataset, which is a desirable property. Similar to Zhang et al. (2020), we observe a prevalence of geometric transformations, unlike Cubuk et al. (2018). In SVHN color transformations are less frequent than in CIFAR datasets, and the frequency of geometric transformations increases, which is interesting due to the perceptual importance of shape in the digit recognition task.

As already noted, using more steps allows the chain to deviate further from the uniform proposal distribution. We thus plot the frequency of sampled transformations accross training epochs on Figure 8. We observe that as training progresses, the observed frequencies also deviate further from uniformity. This suggests that, due to the dataset and architecture, some transformations may be harder to fit than others.

Similarly, we include histograms for the sampled transformation levels in Figure 9. Extreme levels are sampled more frequently, but the empirical distributions vary depending on the transformation. The deviation from uniformity and the differences between transformations are accentuated as sampling steps increase (Figure 10).

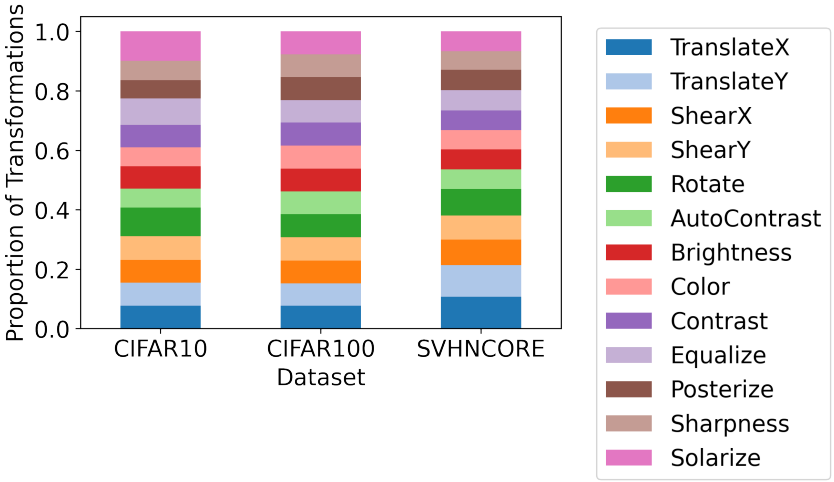

Figure 7: Frequency of sampled transformations for CIFAR and SVHN datasets using two MH steps, for Wide-Resnet-28-10.

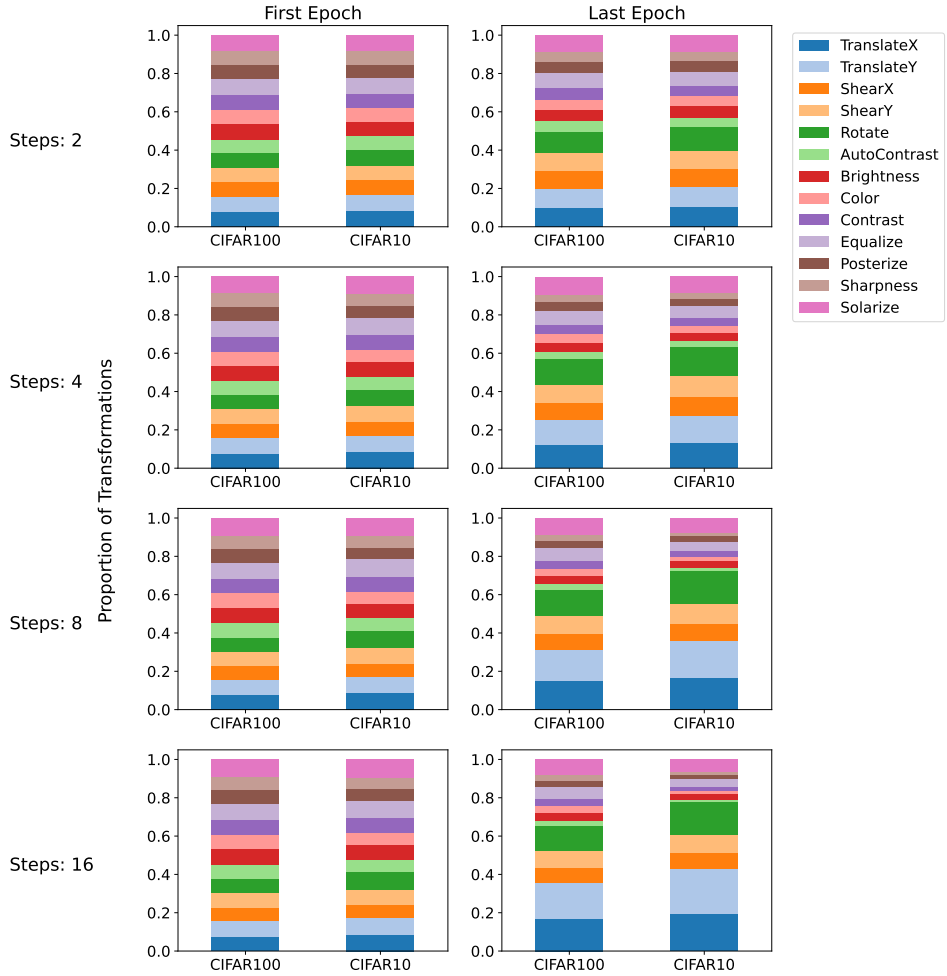

Figure 8: Frequency of sampled transformations for CIFAR datasets in the first and last epochs of training, for Wide-Resnet-40-2. As the number of steps increases, the the entropy of sampled transformations decreases, i.e., observed frequencies get further from uniformity.

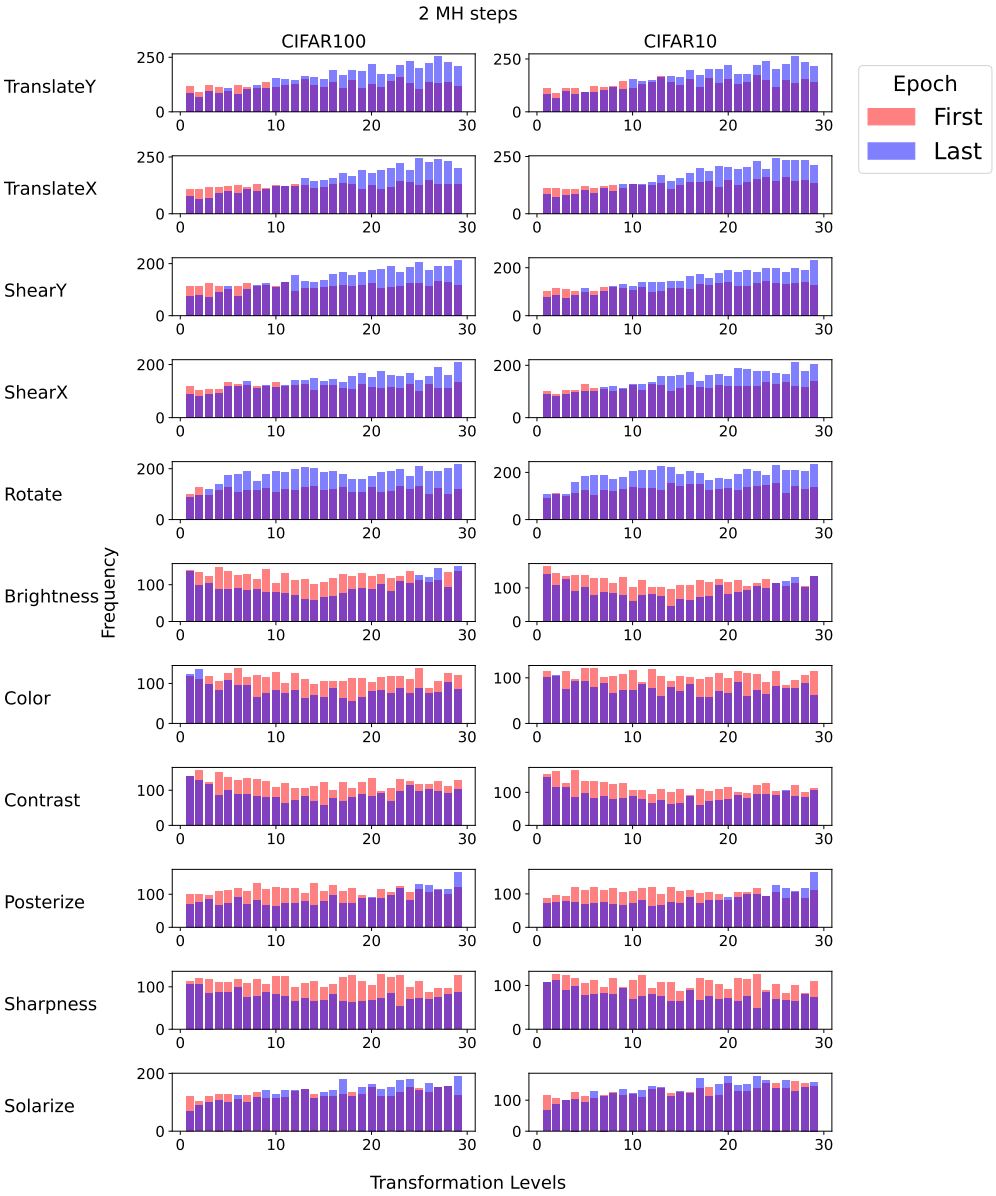

Figure 9: Frequency of sampled transformation levels across epochs, for different transformations, using two MH steps. Extreme levels are sampled more frequently, but some transformations deviate further from uniformity.

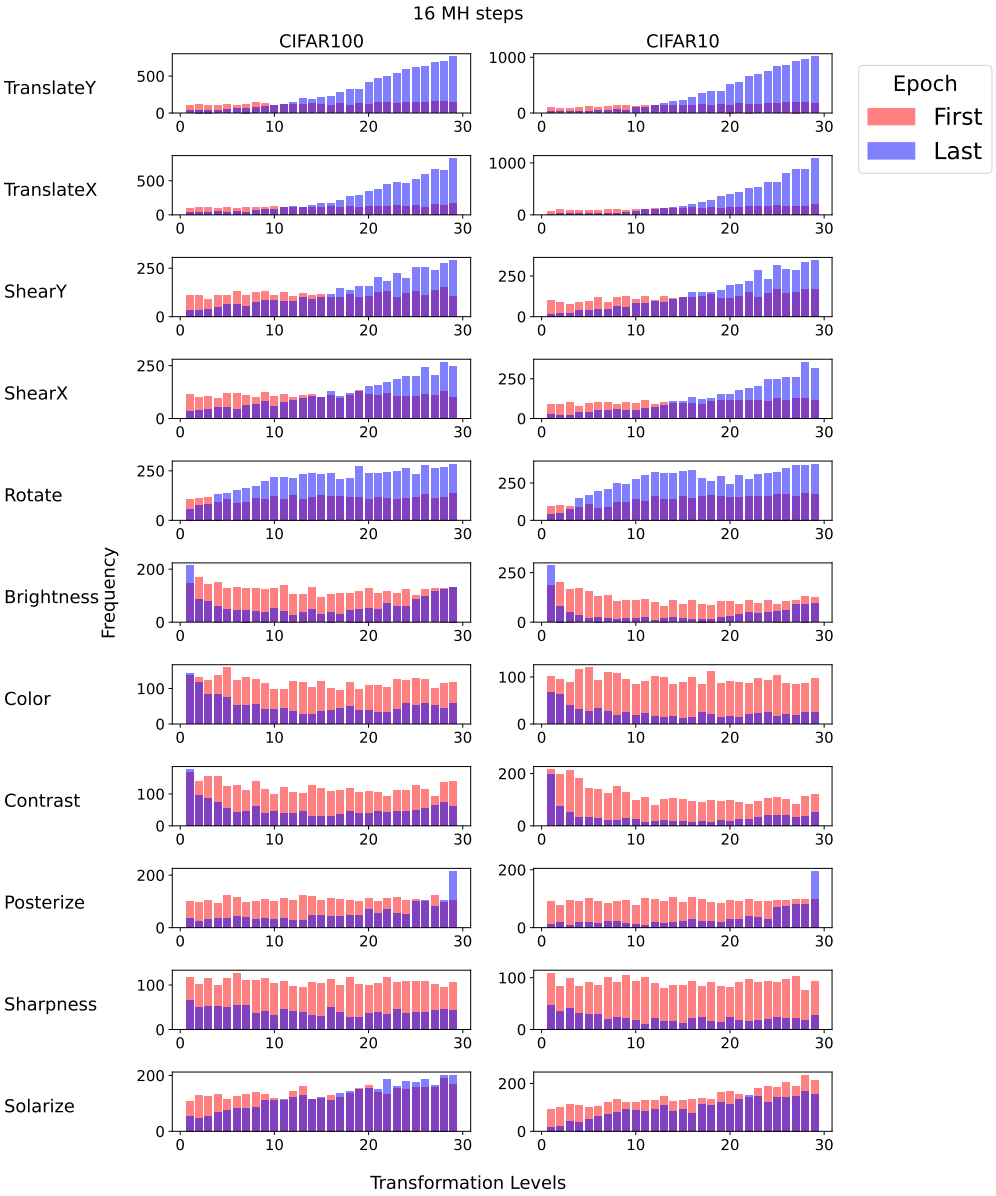

Figure 10: Frequency of sampled transformation levels for the first and last epochs, for different transformations, using sixteen MH steps. The frequencies concentrate in extreme values for some transformations.