Empirical Study on Optimizer Selection for Out-of-Distribution Generalization

Anonymous authors
Paper under double-blind review

Abstract

Modern deep learning systems do not generalize well when the test data distribution is slightly different to the training data distribution. While much promising work has been accomplished to address this fragility, a systematic study of the role of optimizers and their out-of-distribution generalization performance has not been undertaken. In this study, we examine the performance of popular first-order optimizers for different classes of distributional shift under empirical risk minimization and invariant risk minimization. We address this question for image and text classification using DomainBed, WILDS, and Backgrounds Challenge as testbeds for studying different types of shifts—namely correlation and diversity shift. We search over a wide range of hyperparameters and examine classification accuracy (in-distribution and out-of-distribution) for over 20,000 models. We arrive at the following findings, which we expect to be helpful for practitioners: i) adaptive optimizers (e.g., Adam) perform worse than non-adaptive optimizers (e.g., SGD, momentum SGD) on out-of-distribution performance. In particular, even though there is no significant difference in in-distribution performance, we show a measurable difference in out-of-distribution performance. ii) in-distribution performance and out-of-distribution performance exhibit three types of behavior depending on the dataset—linear returns, increasing returns, and diminishing returns. For example, in the training of natural language data using Adam, fine-tuning the performance of in-distribution performance does not significantly contribute to the out-of-distribution generalization performance.

1 Introduction

The choice of numerical optimization method can make a big difference when it comes to successfully training deep neural networks. In particular, the choice of optimizer influences training speed, stability, and generalization performance. Several studies have compared a variety of optimizers and investigated their influence on trainability and generalization (Wilson et al., 2017; Schneider et al., 2019; Choi et al., 2019). Some concluded that non-adaptive optimizers yield better generalization (Wilson et al., 2017; Balles & Hennig, 2018), while others countered that optimizer selection does not affect generalization performance (Schneider et al., 2019; Schmidt et al., 2020).

The conflicting nature of past reports can be explained by disparities in the methodology used for hyperparameter search. For Adam, in particular, hyperparameter ϵ controls the degree of adaptation. Low values, which are used by default, lead to a highly adaptive method. Unusually high values lead to less adaptation. In the limit of large ϵ, Adam turns into a non-adaptive momentum method. In other words, when arbitrary tuning is allowed, methods like a Adam can be thought of as a superset of gradient descent with momentum. The authors in Choi et al. (2019) take this approach, and also consider the less adaptive and non-adaptive regimes of Adam. Not surprisingly, they find that in this full generality Adam never performs worse than gradient descent with momentum in terms of in-distribution generalization, and in some cases, it can perform better.

Although these studies have made substantial progress to improve our understanding of optimizer characteristics, they are based on a common, foundational assumption in learning: training and test data are drawn from the same distribution. In applications, however, it is often the case that test data obey a distribution different from the one for training data. This *distributional shift* violates the typical assumption of independent and identically distributed (i.i.d.) data for learning (Nagarajan et al., 2021). Comparing the generalization performance of different optimizers under this distributional shift, known as out-of-distribution (OOD) generalization (Shen et al., 2021), is of great interest in theory and practice.

In our large suite of experiments, we focus on this OOD generalization question for Natural Language Processing (NLP) and image classification tasks. We train deep neural networks under the principle of Empirical Risk Minimization (ERM) or Invariant Risk Minimization (IRM) (Arjovsky et al., 2019) using a variety of optimizers. Because our objective is to investigate the impact of commonly used optimizers, we target five of the most popular optimizers that have been used and studied in recent years (Schmidt et al., 2021): stochastic gradient descent (SGD), Momentum SGD (Polyak, 1964), and Nesterov accelerated gradient (NAG, also known as Nesterov's momentum) (Nesterov, 2003) in the family of non-adaptive methods as well as RMSProp (Tieleman & Hinton, 2012) and Adam (Kingma & Ba, 2015) in the family of adaptive optimizers. We evaluate the OOD generalization performance of these optimizers on 10 different benchmarks: DomainBed (which includes seven image datasets) (Gulrajani & Lopez-Paz, 2021), the Backgrounds Challenge dataset (Xiao et al., 2021), , and CivilComments-WILDS (Koh et al., 2021).

As discussed above, methods like RMSProp and Adam can be tuned not to be adaptive, in which case they would subsume methods like gradient descent with momentum, and we would trivially get that the more general method can never underperform. Instead, we focus on the more nuanced question of adaptive vs non-adaptive methods: we tune RMSProp and Adam using a range of values for the hyperparameter ϵ, which is strictly wider than ranges used in practice but still keeps the optimizers in their adaptive regime. In this context, we conduct an exhaustive hyperparameter search to select configurations that give good in-distribution validation accuracy for each optimizer. We then test the selected models on the above OOD test sets. Importantly, we demonstrate that our experiments explore more hyperparameter configurations than many existing benchmarks, as betrayed by the fact that we find better-performing models on said benchmarks.

In summary, our contributions are as follows:

- We design and perform a comparison of the effect of optimizers on OOD generalization on a number of OOD benchmarks. To the best of our knowledge, we are the first to consider such a wide variety and scale of datasets. Also, we conduct an empirical study using a wide range of hyperparameter configurations, examining over 20,000 models, evaluating their performance, and measuring the performance gap when moving from the in-distribution test set to the OOD test set.
- We demonstrate on a large number of image classification and NLP tasks that different optimizer choices lead to differences in OOD generalization performance. Furthermore, we show that adaptive optimizers yield more in-distribution overfitting and degrade OOD performance more than non-adaptive optimizers.
- We show that the observed correlation behaviors between in-distribution performance and OOD performance can be categorized into typical patterns: linear return, diminishing return, and increasing return [1]. This categorization should help practitioners better understand and select optimizers.

The following evidence supports the claim that non-adaptive methods outperform adaptive methods in OOD settings: i) There is no significant difference in the in-distribution generalization performance between adaptive and non-adaptive optimizers, ii) Adaptive optimizers perform worse in 8 out of 10 datasets regarding best out-of-distribution generalization performance, iii) We match models trained by adaptive optimizers to models trained by non-adaptive optimizers that yield the same in-distribution performance. Using this matching scheme for our comparison, models trained by non-adaptive methods achieve better OOD performance on 9 out of 10 datasets. These results are based on a comprehensive hyperparameter search and validated through soundness checks in the Appendix G.4.

---

[1] These show how much performance can be expected in the out-of-distribution if we increase the in-distribution performance. These terms are explained in detail in Section 4.3.

Given the similarity of the results between ERM and IRM training principles, we have opted to focus our analysis on the former. Therefore, in the main section of this paper, we primarily report the results of ERM. Our observations that adaptive optimizers perform worse than non-adaptive methods in OOD performance align with theoretical results (Zou et al., 2022) previously reported in the literature, highlighting the drawbacks of adaptive optimizers such as Adam under the i.i.d. assumption and their tendency to fit noise in the data.

The paper is structured as follows. In Section 2, we discuss optimizer selection under the i.i.d. assumption and outline the problem of OOD generalization in the presence of distributional shifts, which can be encountered in real-world problems. In Section 3, we outline the optimizers and the OOD datasets that are the subject of this study, as well as our model selection method. In Section 4, we present the experimental protocol and the experimental results of 10 different datasets and 5 optimizers in each experimental setting.

2 Related Work

2.1 Optimizer Selection

Understanding the characteristics of the many optimizers proposed for deep neural network training (Schmidt et al., 2021) is of great importance to the machine learning research community. In terms of convergence, preconditioned optimizers, including Adam, are known to be superior to non-adaptive optimizers (Kingma & Ba, 2015; Liu et al., 2019; Amari, 1998).

While preconditioned optimization methods seem to be better than non-preconditioned ones in terms of convergence, Zhang et al. (2019) argued that there is a trade-off between generalization performance and convergence rate (Zhang et al., 2019). Wadia et al. (2020) also reported that preconditioned methods, especially second-order methods, do not provide great generalization performance either empirically or theoretically. With a simple theoretical and empirical analysis, Wilson et al. (2017) showed that adaptive optimizers are worse at generalization than simple SGD. Balles & Hennig (2018) also reported that Adam generalizes worse than Momentum SGD.

Contrary to these studies, Schneider et al. (2019) found that no single optimizer is the “best" in general. Similarly, Schmidt et al. (2020) claimed that optimizer performance varies from task to task. Choi et al. (2019) have come to a different conclusion from all the studies cited above. As described in Section 1, Choi et al. (2019) tuned the hyperparameter ϵ of adaptive methods that control the degree of adaptivity, which leads to non-adaptive methods. As a result, they found that well-tuned adaptive optimizers never underperform simple gradient methods.

These studies provided insights into how optimizer selection influences generalization. However, they focused on the classical supervised learning setting, in which the test distribution is assumed to be the same as the training distribution. Our research differs from theirs in that we investigate optimizer selection’s influence on OOD generalization.

2.2 Out-of-Distribution Generalization

Taming the distributional shift is a big challenge in machine learning research (Sugiyama & Kawanabe, 2012; Ben-David et al., 2010; Pan & Yang, 2009; Szegedy et al., 2014; Arjovsky et al., 2019). Geirhos et al. (2020) argued that many modern deep neural network models sometimes learn shortcut features instead of intended features and overfit to a specific dataset.

Some studies have focused on generalization with adversarial noise to evaluate the impact of optimizer selection on OOD generalization. For instance, the theoretical and empirical analysis by Khoury (2019) showed that SGD is more robust against adversarial noise than adaptive optimizers. Wang et al. (2019) argued that adversarial examples to some methods are not necessarily adversarial to others. Abdelzad et al. (2020) found that the best optimizers for OOD detection vary by experimental setting. Metz et al. (2020) reported that a learned optimizer somehow unexpectedly outperformed a human-designed optimizer in terms of the OOD generalization.

Algorithm 1 Generic adaptive optimization method setup.

```
Require: $\{\eta_t\}_{t=1}^T$: step size, $\{\phi_t, \psi_t\}_{t=1}^T$ function to calculate momentum and adaptive rate, $\boldsymbol{\theta}_0$: initial parameter, $\ell(\boldsymbol{\theta})$: objective function
for $t = 1$ to $T$ do
    $\boldsymbol{g}_t \leftarrow \tilde{\nabla}_{\boldsymbol{\theta}} f_t(\boldsymbol{\theta}_{t-1})$ (Calculate stochastic gradients w.r.t. objective at timestep t)
    $\boldsymbol{w}_t \leftarrow \phi_t(\boldsymbol{g}_1, ..., \boldsymbol{g}_t)$ (Calculate momentum)
    $\boldsymbol{l}_t \leftarrow \psi_t(\boldsymbol{g}_1, ..., \boldsymbol{g}_t)$ (Calculate adaptive learning rate)
    $\boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t-1} - \eta_t \boldsymbol{w}_t \boldsymbol{l}_t$ (Update parameters)
end for
```

These studies provide valuable insights on optimizer selection for the OOD problem. However, we emphasize that the previous work discussed above explores hyperparameters in a limited range. Khoury (2019) conducted the most exhaustive hyperparameter search but tuned only the learning rate for adaptive optimizers. As we briefly explain in Section 1, an exhaustive hyperparameter search is crucial for the empirical investigation of an optimizer's effect. Thus, we explored more hyperparameters than previous studies, including searching for over 20,000 models.

Here we emphasize that only shifts such as adversarial noise have been studied in previous studies of optimization selection for OOD generalization. Thus, we use a much more diverse set of real OOD datasets, including image classification and NLP tasks where the distributional shift is significant, covariate shift, correlation shift, subpopulation shift, and background shift, not only domain generalization. The set of datasets we explored is the most exhaustive for evaluating the optimizer's role in OOD generalization, as far as we know.

Finally, we mention to some relevant works Kumar et al. (2022a); Wortsman et al. (2022); Chen et al. (2023); Kumar et al. (2022b). Kumar et al. (2022a); Wortsman et al. (2022); Kumar et al. (2022b) have studied the intricate balance between IID and OOD performance when fine-tuning pre-trained models. Chen et al. (2023) theoretically showed that optimizing the ERM with the relaxed OOD penalty is not likely to have a good performance and proposed a better practical solution.

3 Preliminaries

3.1 Optimizers Subjected to Our Analysis

Similar to previous studies (Wilson et al., 2017; Schneider et al., 2019; Choi et al., 2019), we compare two types of optimizers. The first one is non-adaptive optimizers. The update equation at iteration t of model parameter $\boldsymbol{\theta}_t$ is as follows:

$$\boldsymbol{v}_t \leftarrow \gamma \boldsymbol{v}_{t-1} + \eta_t \tilde{\nabla}_{\boldsymbol{\theta}_{t-1}} \ell(\boldsymbol{\theta}_{t-1}), \quad \boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t-1} - \boldsymbol{v}_t \tag{1}$$

where η_t is the learning rate, $\ell(\boldsymbol{\theta})$ is the loss, and $\tilde{\nabla}_{\boldsymbol{\theta}_{t-1}}$ is the stochastic gradient, in the particular case of stochastic gradient descent, $\gamma = 0$. Optimizers with momentum terms such as Momentum SGD (Polyak, 1964), and Nesterov momentum (Nesterov, 2003) are also classified as non-adaptive optimizers, and γ is the parameter for controlling the momentum term. For Nesterov momentum, $\ell(\boldsymbol{\theta}_{t-1})$ should be replaced $\ell(\boldsymbol{\theta}_{t-1} - \gamma \boldsymbol{v}_{t-1})$.

The second type of optimizer is adaptive methods. Adam and RMSprop are adaptive optimizers and they can be written in the form of the generic adaptive optimization method. The generic adaptive optimization method can be written as in Algorithm 1. This is based on what (Liu et al., 2020) and (Reddi et al., 2018) propose.

Our selection of optimizers aligns with prior research on optimizer comparison (Choi et al., 2019) and recent studies on out-of-distribution (OOD) tasks, which have predominantly focused on Adam rather than alternatives such as AdamW. Therefore, we concentrate on the five most commonly employed optimizers in practice (Schmidt et al., 2021). Given their widespread usage, we chose to focus on studying Adam's performance under the adaptive regime, as this would provide more pertinent findings for present-day practices. Experimental results for AdamW (Loshchilov & Hutter, 2017) and SAM (Foret et al., 2020), although not for all data sets, are also discussed in Appendix I.3.

3.2 Out-of-Distribution Generalization Datasets

We use the following datasets to evaluate the optimizer influence on OOD generalization: DomainBed (Gulrajani & Lopez-Paz, 2021), Backgrounds Challenge (Xiao et al., 2021), (Koh et al., 2021), and CivilComments-WILDS (Koh et al., 2021). These datasets include both artificially created and real-world data, and the applications include image classification and NLP tasks. Although we describe the details of these datasets in Appendix C, we summarize them below.

Image Classification Datasets: DomainBed consists of a set of benchmark datasets for domain generalization, which includes PACS (Fang et al., 2013), VLCS (Li et al., 2017), Office-Home (Venkateswara et al., 2017), Terra Incognita (Beery et al., 2018) DomainNet (Peng et al., 2019), Rotated MNIST (Ghifary et al., 2015), and Colored MNIST (Arjovsky et al., 2019). These datasets contain a variety of distributional shifts. VLCS is a set of different image datasets, for example, images of *birds* from several datasets. Terra Incognita is a dataset consisting of images taken by cameras at different locations. The difference in the location of the camera corresponds to the difference in the domain. PACS, Office-Home, and DomainNet are image datasets whose style varies by domain. For example, an image of PACS in one domain is photography, while that in another domain is a sketch. Rotated MNIST is an artificially generated dataset of domains that have been given different rotation angles.

Colored MNIST is an *anomaly* in the DomainBed dataset. $P(Y|X)$ remains the same for all datasets (covariate shift) except in Colored MNIST. This dataset is designed to make models fail by exploiting spurious correlations in training environments. In particular, the dataset is such that the model can only exploit the source of spurious correlation (color) and achieve a very high training accuracy without relying on the true invariant source of correlation (shape). In contrast, none of the other datasets have such strong spuriousness. The strength of the spurious correlation flips in the test domain and thus it induces a negative correlation between the validation and the test accuracy.

The Backgrounds Challenge dataset measures a model's robustness against background shift (Xiao et al., 2021). A model is trained on an image and evaluated on the same image with a different background. If the model exploits the background features during training, it will be fooled during evaluation. Therefore, if the model strongly depends on the background, this dataset is a difficult OOD dataset, while if the model does not, this dataset is easy to model. To further strengthen our claim, we also performed experiments on CIFAR10-C and CIFAR10-P. The results are shown in Appendix I.1.

Natural Language Processing (NLP) Datasets: The CivilComments-WILDS dataset is cast as a subpopulation shift problem. The shift problem tackles a binary classification problem that predicts the toxicity of comments on articles, and domain vectors are assigned according to whether the comment refers to one of eight demographic identities. In the subpopulation shift, the test distribution is a subpopulation of the training distribution.

The dataset has the characteristics of a hybrid shift of a subpopulation shift and domain shift. This dataset is cast as the problem of estimating a rating of 1–5 from the rating comments of each user. In this kind of dataset, each user corresponds to a domain, and the goal is to produce a high performance for all user comments.

3.3 Model Selection Method and Evaluation Metrics

In the training phase of DomainBed datasets, we do not access the data in the test domain but split data from the training domains into a training dataset and validation dataset. We choose the model with the highest average performance (accuracy) on the validation data in the training domain. As a metric for evaluation, we evaluated the generalization performance in the test domain as the OOD accuracy.

The Backgrounds Challenge uses Imagenet-1k as the training data set and selects models based on their accuracy on the validation data in the training domain. After that, we measure the in-distribution performance with IN9L (Xiao et al., 2021), which aggregates the test data into nine classes. As for the OOD performance, we measure the classification performance on the data where the background image of IN9L is replaced with the background image of other images.

In CivilComments-WILDS, we divide the data into training, validation, and test datasets and maximize worst-group accuracy in the validation data (and by association, maximize the average accuracy over all domains). Then, we perform model selection and evaluate the OOD accuracy on the test data. We adopt the same hyperparameter selection strategy for datasets as that for CivilComments-WILDS. As a metric, we do not evaluate the worst-group performance but rather the 10th-percentile accuracy for the performance of each domain by following the standard federated learning literature (Caldas et al., 2018).

4 Experiments

4.1 Experimental Overview

Our study aims to elucidate the influence of optimizer selection under distributional shifts. To that end, we perform image classification and NLP tasks and evaluate the trained model accuracy on the benchmark datasets introduced in Section 3.2. By comparing the test accuracy for in-distribution samples with that for OOD samples, we can observe how the solution found by each optimizer is robust to the distributional shift. We investigate both ERM and IRMv1 (Arjovsky et al., 2019), a problem setting to solve IRM, to clarify the relationship between the problem formulation and optimizer selection in the OOD problem. For VREx (Krueger et al., 2021) small-scale experimental results are also provided in Appendix I.7.

We compare five optimizers for all datasets except for the Backgrounds Challenge dataset. For the Backgrounds Challenge dataset, we compare only Momentum SGD and Adam due to the computational cost.

We describe the configurations of hyperparameters and protocol for the experiments in further detail in Appendix E and Appendix D respectively. The remaining experimental settings of environment are explained in Appendix B.

Hyperparameter Tuning: The hyperparameters are tuned using Bayes optimization functionality of Weights&Biases[2] by evaluating in-distribution validation accuracy. Bayesian optimization sequentially explored the potential hyperparameter candidate points, and we evaluated all the trained models in the search process for comparison. As a confirmation of the soundness of our hyperparameter search, Appendix H.1 shows the histogram that the explored hyperparameters are drawn from the reasonably wide range. In addition, we investigated the relationship between the number of trials to explore hyperparameters and the best OOD performance and show results in Appendix G.2. The impact of initialization strategies during hyperparameter optimization is also confirmed in Appendix H.3. The data shown in the box plot (Figure 2, 3, and 4) are from the evaluation of several trained models.

The number of epochs (the steps budget) used is in line with previous studies (Gulrajani & Lopez-Paz, 2021; Koh et al., 2021; Choi et al., 2019) to ensure the soundness of our experimental design. Since we use the fixed epoch, it might seem to be unfair than when tuning the epoch as well for the optimizer that converges faster. Thus, we studied the effect of early stopping, which corresponds to the tuned epoch in Appendix G.4, and the result confirms that employing a fixed epoch does not impair the fairness of our comparison experiments. We discuss the details in Appendix G.4.

Boxplot: We believe that sharing the whole distribution of tuning outcomes is important because: it gives an idea of how sensitive methods are to tuning and how much tuning effort is required. We also share the raw data as scatter plots (Figure 15, 16, and 16) in Appendix F.4.

---

[2] `https://wandb.ai/site`

Table 1: DomainBed, Backgrounds Challenge, and CivilComments-WILDS: Comparison of the best OOD accuracy of ERM between five optimizers. We use oracle (see definition at 3.1 in Gulrajani & Lopez-Paz (2021)) as model selection method in this table. The model selection results in the training-domain validation set are shown in Table 12. Except for a small set of problems, momentum SGD outperforms Adam in all but ten cases. As a soundness check, we confirm that our Adam results outperform all existing benchmark results using Adam. Details are given in Appendix G.4.

Model	OOD Dataset	Non-Adaptive Optimizer			Adaptive Optimizer	
		SGD	Momentum	Netsterov	RMSProp	Adam
4-Layer CNN	ColoredMNIST	34.01%	34.23%	40.56%	**89.30%**	73.92%
	RotatedMNIST	90.00%	95.41%	94.06%	96.27%	**96.40%**
ResNet50	VLCS	99.43%	**99.43%**	99.29%	99.29%	99.29%
	PACS	88.67%	**89.55%**	89.25%	88.81%	89.30%
	OfficeHome	64.64%	**65.01%**	63.82%	62.91%	63.82%
	TerraIncognita	**63.21%**	62.41%	62.85%	62.31%	61.35%
	DomainNet	58.38%	61.91%	**62.24%**	55.74%	58.48%
	BackgroundChallenge	-	**80.09%**	-	-	77.90%
DistilBERT	Amazon-WILDS	52.00%	**54.66%**	**54.66%**	53.33%	52.00%
	CivilComment-WILDS	51.66%	57.69%	**60.07%**	45.39%	46.82%
ResNet-20	ColoredMNIST	-	**33.50%**	-	-	31.47%
ViT	PACS	-	**91.80%**	-	-	91.26%

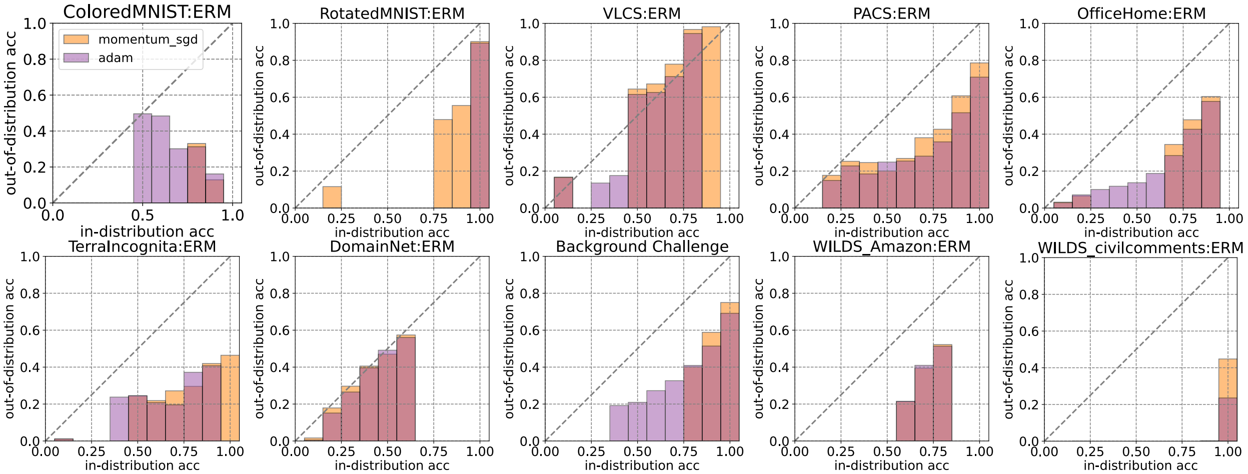

Figure 1. DomainBed, Backgrounds Challenge, and CivilComments-WILDS: Comparison of the in-distribution accuracy and the OOD accuracy of ERM between Momentum SGD and Adam. We also show the training results in the exhaustive hyperparameter search range as a **scatter plot** in Appendix F.3. In these ten plots, for each dataset, the in-distribution performance is separated by every ten bins 0.1. The average OOD performance when evaluating the checkpoints in that bin is shown on the vertical axis. We compare which optimizer shows better OOD performance for each model that achieves equivalent in-distribution performance. In most cases, momentum SGD outperforms Adam in OOD performance in the rightmost region of our interest (the region where high in-distribution performance is achieved).

4.2 Experimental Results

Figure 1 shows the relationship between the in-distribution and OOD accuracy in the ERM setting. The x-axis of the plot is the in-distribution accuracy, and the y-axis is the OOD accuracy. To clarify the trend, the in-distribution accuracy corresponding to the x-axis is divided into ten bins, and the average performance of the OOD accuracy in each bin is shown on the y-axis.

We conducted a comparison between Momentum SGD, the best non-adaptive optimizer, and Adam, the best adaptive optimizer. Our findings reveal that, in our field of study, where high in-distribution performance is achieved, Momentum SGD outperforms Adam on 9 out of 10 datasets (Figure 1). With respect to the best OOD performance, non-adaptive optimizers surpassed adaptive optimizer methods on 8 out of 10 datasets

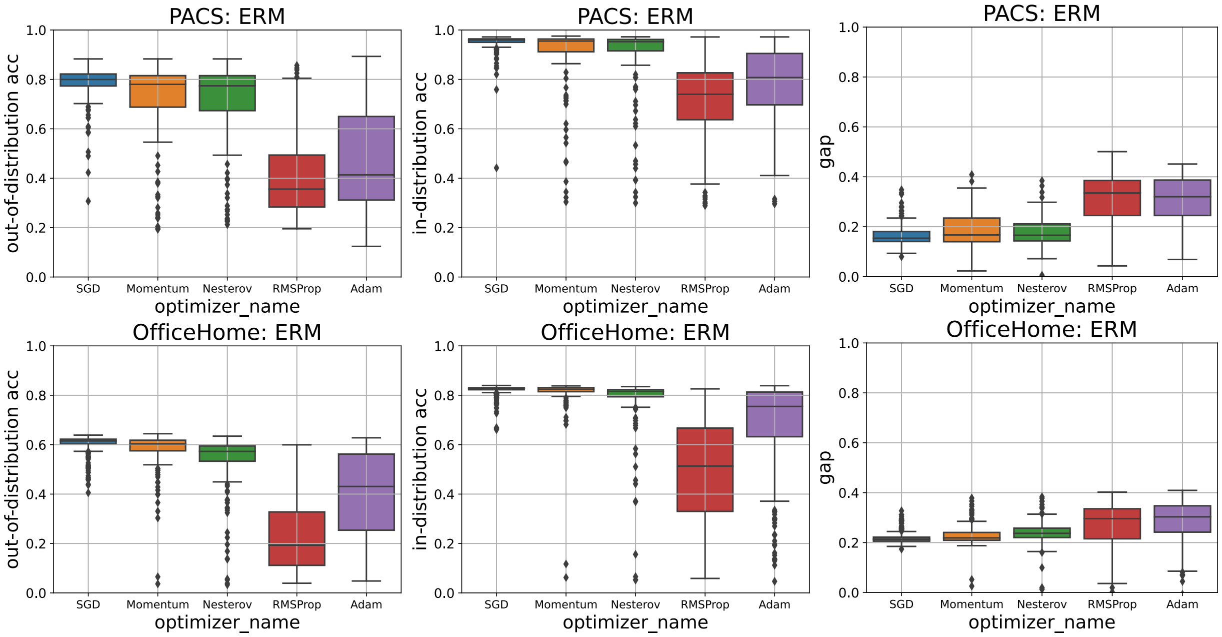

Figure 2. PACS and OfficeHome in DomainBed: Comparison of the in-distribution (validation) accuracy and the out-of-distribution (test) accuracy of ERM across five optimizers. Non-adaptive optimizers outperform the adaptive optimizers in terms of OOD generalization, and the gap between in-distribution performance and OOD performance is small. The details and results of other dataset experiments are described in Appendix F.2.

(Table 1). These results suggest that non-adaptive optimizers are more advantageous than adaptive optimizers in OOD, despite their similar performance in the IID environment. For a more detailed explanation of the experimental results for each dataset, please refer to the following section.

Experimental Results on DomainBed:

Figure 2 shows a box plot of the difference between the average in-distribution accuracy and average OOD accuracy for ERM. In the following discussion, we call this difference a *gap* for convenience. Due to the limitation of the paper length, only the plots of PACS and Office-Home are shown here. All results, including IRM results, are shown in Appendixes F.2.1 and **??**.

We found two distinct optimizer effect patterns, depending on the dataset: i) PACS, Office-Home, VLCS, Terra Incognita, Rotated MNIST, and DomainNet, and ii) Colored MNIST. Because these patterns appear in both the results of ERM and IRM, we focus on ERM for the explanation.

For PACS, Office Home, VLCS, TerraIncognita, RotatedMNIST, and DomainNet, the OOD accuracy is greater for non-adaptive optimizers. The gap between the mean OOD accuracy and the mean in-distribution accuracy was smaller for the non-adaptive optimizer except for TerraIncognita. This means the models trained with the non-adaptive optimizer are more robust on average. In TerraIncognita, the non-adaptive optimizer significantly outperforms the adaptive optimizer on the average in-distribution performance and OOD performance. We note, however, that except in this problem, the adaptive optimizer achieves a smaller gap between the two. As shown in Figure 1, when comparing models with the same in-distribution performance, the non-adaptive optimizer showed higher OOD accuracy than the adaptive optimizer. The results in Figures 1 and 2 confirm that the non-adaptive method achieves higher OOD accuracy than the adaptive method, both on average and for models with the same in-distribution performance.

Colored MNIST shows the opposite pattern of these results, where the adaptive optimizer is better than the non-adaptive optimizers. As outlined in Section 3.2, Colored MNIST differs from the other datasets. To comprehend why adaptive optimizers are more effective in OOD generalization for this dataset, we have

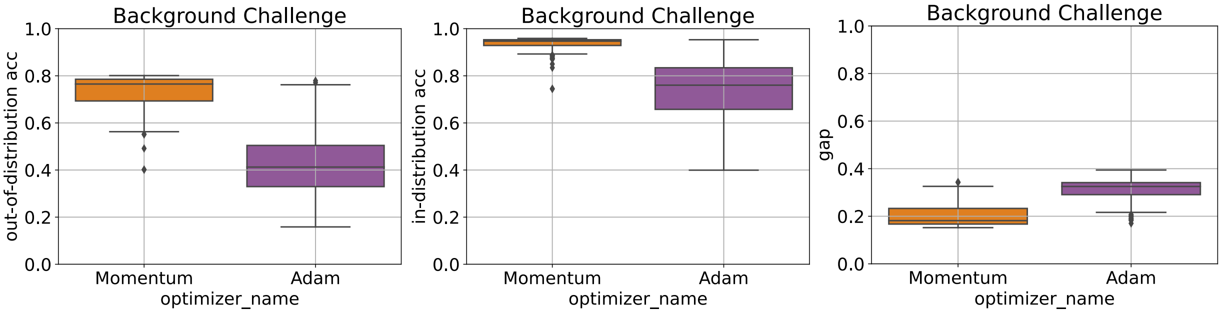

Figure 3. Backgrounds Challenge: Comparison of the in-distribution (validation) accuracy and the out-of-distribution (test) accuracy of ERM across two optimizers. In terms of OOD generalization, Momentum SGD outperforms Adam in both average and best OOD performance. The highest value of in-distribution accuracy remains the same, but Momentum SGD shows higher performance on average.

plotted the average training, validation, and test accuracy over time during training, as illustrated in Figure 33. Our explanation for this outcome can be found in Appendix G.3.

We note our considerations regarding the exceptional behavior of ColoredMNIST. ColoredMNIST is a binary classification task dataset with random labels, where spurious features are positively correlated with invariant features in in-distribution and negatively correlated with OOD. Non-adaptive optimizers learn the spurious features, achieve the oracle performance for in-distribution and perform worse than a random guess for OOD due to inverted correlations. Adaptive optimizers, in contrast, seem to overfit the training data, achieving 100% accuracy in the training set (more than oracle [3]) in this dataset, and failing to learn the spurious features. Due the aforementioned (synthetic) inverted correlations in the dataset, this overfitting behaviour in-distribution happens to favor adaptive optimizers. This behavior happens exceptionally on ColoredMNIST due to these synthetic flips in correlation. This exploitation unexpectedly enables Adam to avert being trapped in the training domain and produces better OOD generalization.

Experimental Results on Backgrounds Challenge:

Backgrounds Challenge requires training of ImageNet-1k on ResNet50 from scratch, and it takes 256 GPU hours to obtain a single trained model, so we only compared Momentum SGD with Adam. Because Momentum SGD achieved competitive performance among non-adaptive optimization methods in the DomainBed and WILDS experiments, we adopted Momentum SGD to represent non-adaptive optimization methods. In the same way, Adam outperformed RMSProp in almost all the benchmarks, so we adopted Adam as a representative of the adaptive optimizers.

Figure 3 compares the accuracy for ORIGINAL (in-distribution) and Mixed-Rand (out-of-distribution). The best in-distribution performance is the same for each optimizer, but concerning the best OOD performance, the non-adaptive optimizer outperformed the adaptive optimizer. As can be seen from Figure 1 (second row, middle column), particularly in the region of high in-distribution performance on the right, the OOD performance of Momentum SGD exceeds that of Adam.

Experimental Results on WILDS: A comparison of in-distribution and OOD averages for WILDS is shown in Figure 4. It can be clearly seen that the adaptive optimizer is fit too well to the in-distribution in the WILDS problem setting. For and CivilComment-WILDS, as in DomainBed and Backgrounds Challenge, the non-adaptive optimizer outperforms the adaptive optimizer in terms of OOD generalization. The gap between in-distribution accuracy and OOD accuracy is also tiny for non-adaptive optimizers. In particular, the CivilComments-WILDS experimental result is remarkable, as both non-adaptive and adaptive optimizers show similar high in-distribution performance, but in the OOD environment, adaptive methods significantly fail to make inferences.

---

[3] Since ColoredMNIST includes label flip as noise, even the best model that correctly learns the data rules will only perform 85%.

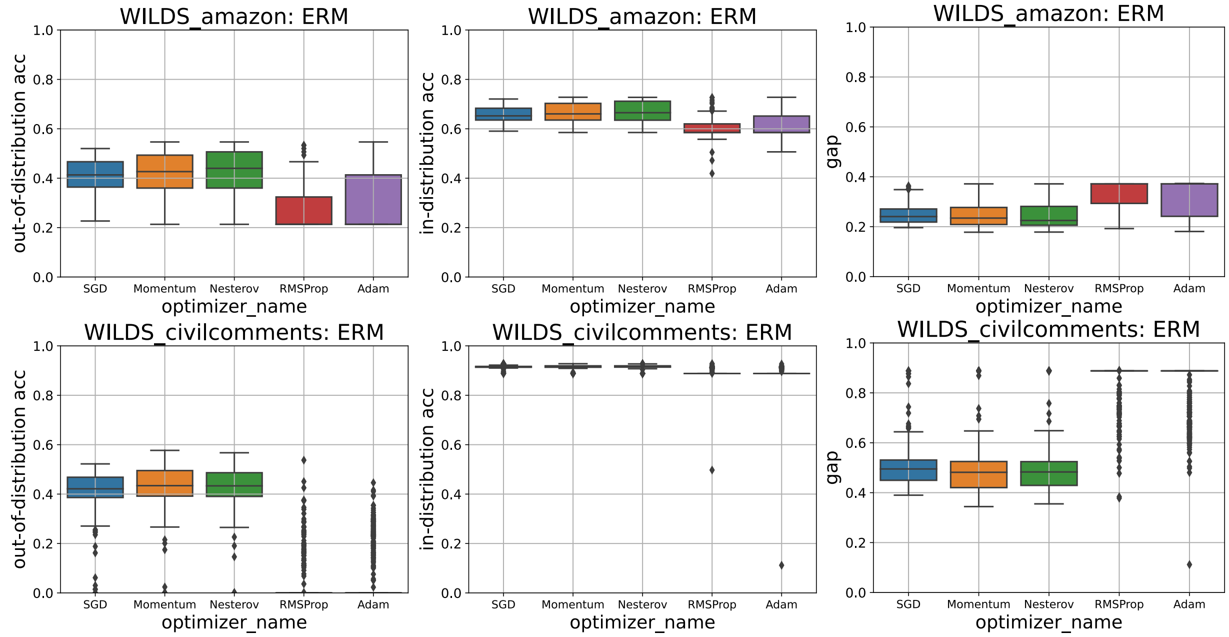

Figure 4. and CivilComments-WILDS: Comparison of the in-distribution (validation) accuracy and the out-of-distribution (test) accuracy of ERM across five optimizers. Both non-adaptive and adaptive optimizers have similar in-distribution accuracy, but the adaptive method significantly degrades the performance of the OOD generalization.

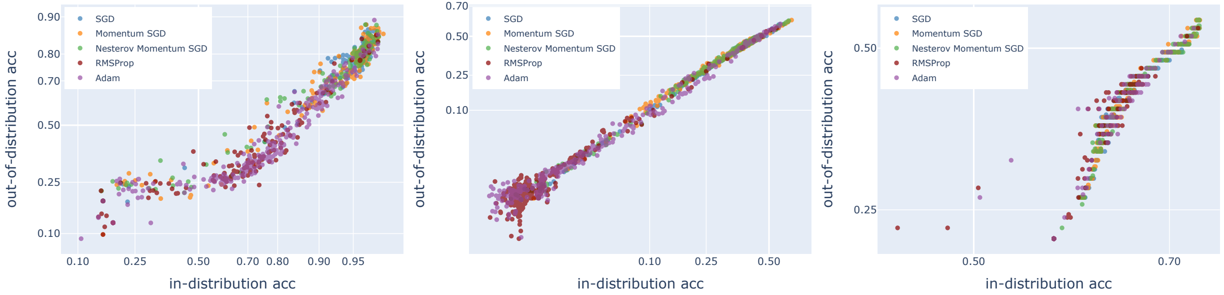

Figure 5. Three types of correlation behavior: increasing return (PACS), linear return (DomainNet), and diminishing return (). The legend circles on the right side of each figure show, in order, the SGD, Momentum SGD, Nesterov Momentum, RMProp, and Adam. The details and results of other dataset experiments are described in Appendixes 15, 16, and 16

4.3 Correlation Behaviors

Our results show that three typical types of behavior are observed in terms of the correlation between in-distribution performance and OOD performance for different datasets. Detailed results of the experiments on all datasets are shown in Appendix F.4. We follow Miller et al. (2021), who used a probit transform to show the relationship. The three types are increasing return, linear return, and diminishing return. These show how much performance in OOD can be expected if we increase the in-distribution performance.

The increasing return is an example, as shown in the leftmost part of Figure 5. The increasing returns in large regions of the in-distribution generalization significantly affect the OOD generalization, suggesting that the last tuning is significant for the OOD generalization, as seen in all domain generalization datasets except for DomainNet.

The linear return is as shown in the middle of Figure 5. The OOD accuracy increases linearly with the in-distribution accuracy. This is generally the same result for in-distribution validation and test.

Conversely, diminishing return behavior, illustrated in the rightmost part of Figure 5, indicates that substantial in-distribution improvement leads to a saturation of OOD generalization with only a marginal effect. This observation implies that the effort invested in fine-tuning might not always yield significant enhancements in OOD generalization. We have observed similar trends in settings with subpopulation shifts, such as CivilComments-WILDS.

In Appendix F.4, we present our experimental results without a probit transform, confirming that diminishing returns are not necessarily linear before probit transformation. This finding aligns with recent studies by Wenzel et al. (2022); Teney et al. (2022), stating that the accuracy of IID and OOD varies across datasets. Moreover, as Baek et al. (2022) suggests, the occurrence of linear return depends on the problem set. Our results support these claims, providing a comprehensive analysis of datasets and problem sets that Miller et al. (2021) did not address and uncovering trends they did not reveal.

For practitioners, our findings offer valuable insights into adjusting their expectations and strategies based on the observed behavior. For instance, if they work with a dataset similar to CivilComments, they can anticipate one of the identified types of behavior. Should their dataset exhibit saturating behavior, they can adjust their expectations accordingly; conversely, if their dataset demonstrates a regime where every slight improvement in in-distribution accuracy aids, they should pursue enhanced in-distribution optimization.

5 Discussion and Conclusion

We conduct an exhaustive empirical comparison of the generalization performance of various optimizers under different practical distributional shifts. Notably, ten state-of-the-art OOD datasets were used to study the environment of broad shifts in correlation and diversity shifts. The investigation elucidates how optimizer selection affects OOD generalization. As the main claim, the answer to our research objective is that the non-adaptive optimizer is superior to the adaptive optimizer in terms of OOD generalization.

The following evidence supports this: i) when comparing in-distribution accuracy is the same, OOD accuracy of non-adaptive optimizers is greater than that of adaptive optimizers (Figure 1); ii) on average and top performance, the OOD accuracy of the non-adaptive optimizer is higher (Figures 2, 3, and 4). All these points support our main claim that non-adaptive optimizers are superior in OOD generalization within our exhaustive experiments. We have tuned Adam to the range that it is used in practice and have updated Adam's scores on all OOD datasets, which use Adam optimizer as default for benchmarks. This supports the soundness of our experiments.

Overall, we can conclude that optimizer selection influences OOD generalization in the cases we are interested in. Future research should consider the algorithm or loss function and optimizers in the OOD problem. The results of IRM show a trend similar to ERM's, but a more detailed analysis is needed to consider the differences in loss landscapes. All these points support our main claim that non-adaptive optimizers are superior in OOD generalization within our exhaustive experiments.

Finally, we would like to mention the limitations of our work. One limitation is that we did not study recently proposed and less popular optimizers.

The choice of optimizers we study is in line with previous work (Choi et al., 2019), (Schmidt et al., 2021) on optimizer comparison and with most recent OOD work; those studies overwhelmingly focused on Adam rather than e.g., AdamW (Loshchilov & Hutter, 2017). Similarly, other less popular optimizers such as SWA (Izmailov et al., 2018), SWAD (Cha et al., 2021), have been omitted to allow for a more extensive study of the chosen methods.

Another limitation is that we employed a total of six models used in the DomainBed (ConvNet, ResNet20, ResNet50 and Vision Transformer (Dosovitskiy et al., 2020)), Backgrounds Challenge (ResNet50), and WILDS (DistilBERT (Sanh et al., 2019)) benchmarks.

References

Vahdat Abdelzad, Krzysztof Czarnecki, and Rick Salay. The effect of optimization methods on the robustness of out-of-distribution detection approaches. *arXiv preprint arXiv:2006.14584*, 2020.

Shunichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

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

Christina Baek, Yiding Jiang, Aditi Raghunathan, and Zico Kolter. Agreement-on-the-line: Predicting the performance of neural networks under distribution shift. *arXiv preprint arXiv:2206.13089*, 2022.

Lukas Balles and Philipp Hennig. Dissecting adam: The sign, magnitude and variance of stochastic gradients. In *International Conference on Machine Learning*, pp. 404–413. PMLR, 2018.

Peter Bandi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. *IEEE Transactions on Medical Imaging*, 2018.

Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 456–473, 2018.

Sara Beery, Elijah Cole, and Arvi Gjoka. The iwildcam 2020 competition dataset. *arXiv preprint arXiv:2004.10340*, 2020.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion Proceedings of The 2019 World Wide Web Conference*, 2019.

Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. *Advances in Neural Information Processing Systems*, 34, 2021.

Yongqiang Chen, Kaiwen Zhou, Yatao Bian, Binghui Xie, Bingzhe Wu, Yonggang Zhang, MA KAILI, Han Yang, Peilin Zhao, Bo Han, and James Cheng. Pareto invariant risk minimization: Towards mitigating the optimization dilemma in out-of-distribution generalization. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=esFxSb_0pSL.

Dami Choi, Christopher J Shallue, Zachary Nado, Jaehoon Lee, Chris J Maddison, and George E Dahl. On empirical comparisons of optimizers for deep learning. *arXiv preprint arXiv:1910.05446*, 2019.

Myung Jin Choi, Joseph J Lim, Antonio Torralba, and Alan S Willsky. Exploiting hierarchical context on a large database of object categories. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 129–136. IEEE, 2010.

Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Etienne David, Simon Madec, Pouria Sadeghi-Tehran, Helge Aasen, Bangyou Zheng, Shouyang Liu, Norbert Kirchgessner, Goro Ishikawa, Koichi Nagasawa, Minhajul A Badhon, Curtis Pozniak, Benoit de Solan, Andreas Hund, Scott C. Chapman, Frederic Baret, Ian Stavness, and Wei Guo. Global wheat head detection (gwhd) dataset: a large and diverse dataset of high-resolution rgb-labelled images to develop and benchmark wheat head detection methods. *Plant Phenomics*, 2020, 2020.

Etienne David, Mario Serouart, Daniel Smith, Simon Madec, Kaaviya Velumani, Shouyang Liu, Xu Wang, Francisco Pinto Espinosa, Shahameh Shafiee, Izzat S. A. Tahir, Hisashi Tsujimoto, Shuhei Nasuda, Bangyou Zheng, Norbert Kichgessner, Helge Aasen, Andreas Hund, Pouria Sadhegi-Tehran, Koichi Nagasawa, Goro Ishikawa, Sebastien Dandrifosse, Alexis Carlier, Benoit Mercatoris, Ken Kuroki, Haozhou Wang, Masanori Ishii, Minhajul A. Badhon, Curtis Pozniak, David Shaner LeBauer, Morten Lilimo, Jesse Poland, Scott Chapman, Benoit de Solan, Frederic Baret, Ian Stavness, and Wei Guo. Global wheat head dataset 2021: an update to improve the benchmarking wheat head localization with more diversity, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1657–1664, 2013.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pp. 178–178. IEEE, 2004.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11): 665–673, 2020.

Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *Proceedings of the IEEE international conference on computer vision*, pp. 2551–2559, 2015.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

Marc Khoury. Adaptive versus standard descent methods and robustness against adversarial examples. *arXiv preprint arXiv:1911.03784*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

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

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

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022a. URL `https://openreview.net/forum?id=UYneFzXSJWh`.

Ananya Kumar, Ruoqi Shen, Sébastien Bubeck, and Suriya Gunasekar. How to fine-tune vision models with sgd. *arXiv preprint arXiv:2211.09359*, 2022b.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 5542–5550, 2017.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. *arXiv e-prints*, pp. arXiv–1908, 2019.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. In *International Conference on Learning Representations*, 2020.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin Clement, Dawn Drain, Daxin Jiang, Duyu Tang, et al. Codexglue: A machine learning benchmark dataset for code understanding and generation. *arXiv preprint arXiv:2102.04664*, 2021.

Luke Metz, Niru Maheswaranathan, C Daniel Freeman, Ben Poole, and Jascha Sohl-Dickstein. Tasks, stability, architecture, and compute: Training more effective learned optimizers, and using them to train themselves. *arXiv preprint arXiv:2009.11243*, 2020.

John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International Conference on Machine Learning*, pp. 7721–7735. PMLR, 2021.

Norman Mu and Justin Gilmer. Mnist-c: A robustness benchmark for computer vision. *arXiv preprint arXiv:1906.02337*, 2019.

Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. In *International Conference on Learning Representations*, 2021.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1406–1415, 2019.

Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, 4(5):1–17, 1964.

Veselin Raychev, Pavol Bielik, and Martin Vechev. Probabilistic model for code with decision trees. *ACM SIGPLAN Notices*, 2016.

Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.

Bryan C Russell, Antonio Torralba, Kevin P Murphy, and William T Freeman. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*, 77(1-3):157–173, 2008.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Robin M Schmidt, Frank Schneider, and Philipp Hennig. Descending through a crowded valley–benchmarking deep learning optimizers. *arXiv preprint arXiv:2007.01547*, 2020.

Robin M Schmidt, Frank Schneider, and Philipp Hennig. Descending through a crowded valley-benchmarking deep learning optimizers. In *International Conference on Machine Learning*, pp. 9367–9376. PMLR, 2021.

Frank Schneider, Lukas Balles, and Philipp Hennig. DeepOBS: A deep learning optimizer benchmark suite. In *International Conference on Learning Representations*, 2019.

Zheyan Shen, Jiashuo Liu, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

Masashi Sugiyama and Motoaki Kawanabe. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press, 2012.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

J. Taylor, B. Earnshaw, B. Mabey, M. Victors, and J. Yosinski. Rxrx1: An image set for cellular morphological variation across many experimental batches. In *International Conference on Learning Representations (ICLR)*, 2019.

Damien Teney, Yong Lin, Seong Joon Oh, and Ehsan Abbasnejad. Id and ood performance are sometimes inversely correlated on real-world datasets, 2022. URL https://arxiv.org/abs/2209.00613.

Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5018–5027, 2017.

Neha S Wadia, Daniel Duckworth, Samuel S Schoenholz, Ethan Dyer, and Jascha Sohl-Dickstein. Whitening and second order optimization both destroy information about the dataset, and can make generalization impossible. *arXiv preprint arXiv:2008.07545*, 2020.

Yixiang Wang, Jiqiang Liu, Jelena Mišić, Vojislav B Mišić, Shaohua Lv, and Xiaolin Chang. Assessing optimizer impact on dnn model sensitivity to adversarial examples. *IEEE Access*, 7:152766–152776, 2019.

Florian Wenzel, Andrea Dittadi, Peter Vincent Gehler, Carl-Johann Simon-Gabriel, Max Horn, Dominik Zietlow, David Kernert, Chris Russell, Thomas Brox, Bernt Schiele, et al. Assaying out-of-distribution generalization in transfer learning. *arXiv preprint arXiv:2207.09239*, 2022.

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, 2017.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7959–7971, 2022.

Kai Yuanqing Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. In *International Conference on Learning Representations*, 2021.

Christopher Yeh, Anthony Perez, Anne Driscoll, George Azzari, Zhongyi Tang, David Lobell, Stefano Ermon, and Marshall Burke. Using publicly available satellite imagery and deep learning to understand economic well-being in africa. *Nature Communications*, 2020.

Guodong Zhang, Lala Li, Zachary Nado, James Martens, Sushant Sachdeva, George E Dahl, Christopher J Shallue, and Roger Grosse. Which algorithmic choices matter at which batch sizes? insights from a noisy quadratic model. *arXiv preprint arXiv:1907.04164*, 2019.

Difan Zou, Yuan Cao, Yuanzhi Li, and Quanquan Gu. Understanding the generalization of adam in learning neural networks with proper regularization, 2022. URL `https://openreview.net/forum?id=G7PfyLimZBp`.