Rethinking the Evaluation Protocol of Domain Generalization

Han Yu, Xingxuan Zhang, Renzhe Xu, Jiashuo Liu, Yue He, Peng Cui*
Department of Computer Science, Tsinghua University
yuh21@mails.tsinghua.edu.cn, xingxuanzhang@hotmail.com, xrz199721@gmail.com
liujiashuo77@gmail.com, heyuethu@mail.tsinghua.edu.cn, cuip@tsinghua.edu.cn

Abstract

Domain generalization aims to solve the challenge of Out-of-Distribution (OOD) generalization by leveraging common knowledge learned from multiple training domains to generalize to unseen test domains. To accurately evaluate the OOD generalization ability, it is required that test data information is unavailable. However, the current domain generalization protocol may still have potential test data information leakage. This paper examines the risks of test data information leakage from two aspects of the current evaluation protocol: supervised pretraining on ImageNet and oracle model selection. We propose modifications to the current protocol that we should employ self-supervised pretraining or train from scratch instead of employing the current supervised pretraining, and we should use multiple test domains. These would result in a more precise evaluation of OOD generalization ability. We also rerun the algorithms with the modified protocol and introduce new leaderboards to encourage future research in domain generalization with a fairer comparison.

1. Introduction

The performance of traditional machine learning algorithms heavily depends on the assumption that the training and test data are independent and identically distributed (IID). However, in wild environments, the test distribution often differs significantly from the training distribution. This mismatch can lead to spurious correlations, ultimately causing the machine learning models to perform poorly and become unstable. Unfortunately, this limitation severely hinders the use of machine learning in high-risk domains like autonomous driving [27, 38], medical treatment [36], and law [4].

In recent years, there has been a growing interest among researchers in addressing the Out-of-Distribution (OOD) generalization problem, where the IID assumption does not stand [66, 83]. This issue is addressed by various branches of research, such as invariant learning [1, 13, 34, 49, 50], distributionally robust optimization [17, 51, 52, 67, 75], stable learning [14, 23, 35, 65, 78, 82], and domain generalization [42, 76, 87, 94, 95]. Of these, domain generalization assumes the heterogeneity of training data. Specifically, domain generalization attempts to learn common or causal knowledge from multiple training domains to develop a model capable of generalizing to unseen test data.

Despite a quantity of interesting and instructive works in domain generalization, previously they do not have a common standard training and evaluation protocol. Aware of this problem, DomainBed [19] proposes a framework for the hyperparameter search and model selection of domain generalization. It also sets a standard for experimental details like the model backbone, data split, data augmentation, etc. This unifies the protocol of domain generalization for subsequent works to follow. Nevertheless, conflicts remain between the current standard protocol and the accurate and reliable evaluation of OOD generalization ability. Since domain generalization is depicted as the ability to learn a model from diverse training domains that can generalize to **unseen/unknown** test data [76, 95], we should try to mitigate possible test data information leakage for a more precise evaluation of the OOD generalization ability.

In the current protocol, two key factors show the potential risk of test data information leakage. We make two recommendations for fairer and more accurate evaluation.

Recommendation 1 *Domain generalization algorithms should adopt self-supervised pretrained weights or random weights as initialization when evaluated and compared with each other.*

Most domain generalization algorithms take advantage of ImageNet supervised pretrained weights [15, 19] for better performance and faster convergence. Yet this introduces the information on both images and category labels in ImageNet, which may bear a resemblance to the test domain. Through comprehensive experiments, we demonstrate that more utilization of supervised pretrained weights and less utilization of training data can contribute to higher test do-

*Corresponding author

main performance under many common settings of domain generalization. This reveals that the ImageNet supervised pretrained weights may play a leading role in the test domain performance. Thus the accurate evaluation of OOD generalization is violated since the test domain performance does not really come from the generalization from training domains to test domains, for which most domain generalization algorithms are designed, but from the utilization of the supervised pretrained weights. We also demonstrate that when a test domain is quite similar to the ImageNet dataset, such a phenomenon becomes most evident while it does not occur if the test domain is rather different from ImageNet. This further confirms the test data information leakage.

To address such an issue, it is safest to train from scratch to purely evaluate domain generalization. However, on one hand, with the remarkable development and broad application of pretrained models these days [5, 60], it is too limited and not common practice to train from scratch in real applications without benefiting from pretraining. On the other hand, most of commonly used domain generalization datasets like PACS [39] and VLCS [30] are not large enough to support training from scratch. Thus we investigate different pretrained methods and model backbones towards a set of pretrained weights with which there is less test data information leakage and we can still conduct a relatively accurate evaluation of OOD generalization. Based on our experimental findings, we suggest that self-supervised pretrained weights are a good alternative.

Recommendation 2 *Domain generalization algorithms should be evaluated on multiple test domains.*

For each trained model, domain generalization algorithms are typically evaluated on a single test domain. Before DomainBed, the choice of hyperparameters is not well specified, and there is a chance that model selection is conducted with the help of test data, i.e., the oracle model selection [19]. This can introduce information leakage from the test data and undermine the validity of the evaluation. Even following the standard protocol of DomainBed, such a possibility still exists since the search space of hyperparameters could be pre-selected with the information of oracle data and fixed apriori for the DomainBed model selection pipeline. Moreover, the ultimate goal of domain generalization is to develop models that can generalize well to a wide range of unseen domains in real-world applications instead of tuning a set of hyperparameters for one single test domain. The current protocol allows models to select different hyperparameters for each test domain, which may not reflect the real-world scenario and could be inconsistent with the original purpose of domain generalization [90]. We suggest that we should evaluate algorithms on multiple test domains for each trained model since we empirically demonstrate that by doing so, the potential leakage from oracle model selection can be greatly mitigated.

New leaderboards Based on the aforementioned recommendations, we have conducted a re-evaluation of ten representative domain generalization algorithms following the revised protocol and presented three sets of new leaderboards. For ResNet50 [21] that is employed in the current protocol, we provide leaderboards with MoCo-v2 [10] pretraining across all commonly used datasets, and leaderboards with no pretraining on large-scale datasets like DomainNet [59] and NICO++ [90]. In addition, to support comparisons on more advanced network architectures like vision transformers, we also provide leaderboards for ViT-B/16 [16] with MoCo-v3 [12] pretraining. Combined with our previous analyses, the change in rankings of algorithms between the new leaderboard and the old one also implies that we are taking risks to improperly evaluate and rank existing methods with the current evaluation protocol. We believe the revised protocol and the leaderboards will stimulate future research in the field of domain generalization with more precise evaluation.

2. Rethinking the Evaluation Protocol

In this section, we will rethink the current evaluation protocol through comprehensive experimental analyses. First we review the definition of domain generalization.

Definition 1 (Domain Generalization) *Given M different training domains $\{S_1, S_2, ..., S_M\}$ where $S_j = \{x_i^{(j)}, y_i^{(j)}\}_{i=1}^{n_j}$ is sampled from $P_j(X, Y)$. The goal is to learn a function f that predicts well on the **unseen test domain** S_{te}. S_{te} should be different from the M training domains. With l denoting the loss function, the optimization target is:*

$$\min_f \mathbb{E}_{(x,y)\sim P_{te}}[l(f(x), y)] \tag{1}$$

Since the test domain is required to be unseen [76, 95], we should try to decrease the potential risk of leaking test data information to accurately evaluate the OOD generalization ability of algorithms.

Although DomainBed has established a standard protocol for researchers of domain generalization to follow, there are still some defects hindering the accurate and fair evaluation of OOD generalization ability. Specifically, Definition 1 does not explicitly address the use of pretrained weights, and the optimization target Equation 1 is defined for a single test domain P_{te}. We will discuss these issues in detail in the following sections.

We briefly introduce the domain generalization benchmark datasets we will use in our experiments.

Table 1. Results of linear-probing (LP) and fine-tuning (FT) with supervised pretrained ResNet-50 on commonly used domain generalization datasets.

PACS	P	A	C	S	Avg
LP	**97.7±0.1**	71.8±1.6	53.8±1.8	45.9±1.7	67.3±0.3
FT	97.4±0.1	86.1±0.9	80.4±1.4	77.1±2.5	85.3±0.6
VLCS	V	L	C	S	Avg
LP	**77.2±1.6**	58.1±0.6	**97.4±0.4**	71.4±1.1	76.0±0.6
FT	73.5±1.5	66.3±0.9	96.9±1.1	71.7±1.5	77.1±1.0
OfficeHome	A	C	P	R	Avg
LP	**64.0±0.4**	50.3±0.3	**77.7±0.5**	**79.7±0.2**	**67.9±0.2**
FT	61.1±0.6	51.1±0.3	73.9±0.5	75.7±0.7	65.5±0.2
TerraInc	L38	L43	L46	L100	Avg
LP	**43.4±5.4**	36.6±0.9	32.4±0.9	37.9±0.2	37.6±1.7
FT	43.2±2.4	56.1±0.2	38.4±5.7	54.8±5.9	48.1±3.1

- PACS [39]: consists of 4 domains to depict the distribution shift as a change of style: photo, art_painting, cartoon, sketch. It comprises 9,991 samples with 7 classes.
- VLCS [30]: is collected from 4 different datasets corresponding to four domains: Pascal VOC 2007, LabelMe, Caltech, SUN09. It contains 10,729 real photo examples with 5 common classes. It depicts the distribution shift with dataset bias.
- OfficeHome [74]: comprises 4 domains: art, clipart, product, real, with 65 classes and 15,588 examples. It also depicts the style transfer.
- Terra Incognita [3]: comprises 4 domains: L38, L43, L46, L100, with 10 classes and 24,788 examples. Its distribution shift is characterized by different locations when taking photos.
- DomainNet [59]: comprises 6 domains: clipart, infograph, painting, quickdraw, real, sketch, also characterized by style shift. It is a relatively large dataset with 586,575 samples and 345 classes in total.
- NICO++ [90]: comprises 6 publicly available domains: autumn, rock, dim, grass, outdoor, water. This part contains 88,866 examples with 60 classes. It is also a relatively large dataset that controls the distribution shift through the change of background contexts.

2.1. Pretraining

When the most widely used domain generalization benchmark dataset PACS is released [39], the authors proposed a baseline "Deep-All", an AlexNet [33] supervised pretrained on ImageNet and optimized using ERM [73]. To ensure comparability with prior research, subsequent studies have consistently followed this practice, even as the backbone architecture has evolved from AlexNet [39] to ResNet-18 [6, 25, 56, 57] and more recently to ResNet-50 [7, 19, 61]. It is worth noting that bringing in extra knowledge beyond

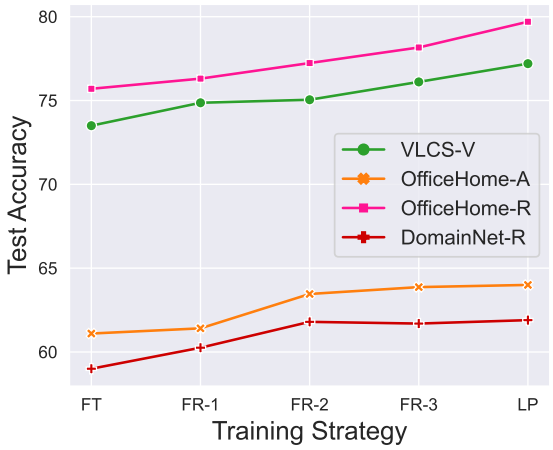

Figure 1. Test domain accuracy with the growing number of frozen layers when using ImageNet pretrained weights. Test domains include PASCAL from VLCS, Art and Real from OfficeHome, and Real from DomainNet.

training data may have a potential negative effect of test data information leakage, hurting the accurate evaluation of OOD generalization. For instance, VLCS [30], TerraInc [3] and NICO++ [90] comprise entirely real photos, which resemble the pretrained dataset ImageNet. Similarly, some domains in PACS [39], OfficeHome [74], and DomainNet [59] also include real photos that share similar characteristics with ImageNet. When using these data as test domains, there could be potential leakage of test data information from the pretrained weights.

Table 2. Relationship between dissimilarity of pretrained data and test domains, and the phenomenon of LP outperforming FT. The "Gap" is calculated as $\frac{\mathrm{Acc_{LP}}-\mathrm{Acc_{FT}}}{\mathrm{Acc_{FT}}}$.

PACS	P	A	C	S
Gap	0.004	-0.167	-0.331	-0.405
FID	103.75	128.07	148.52	209.68
OfficeHome	A	C	P	R
Gap	0.047	-0.017	0.051	0.054
FID	62.22	92.33	78.26	56.96

To confirm our concerns, we conduct experiments by comparing the effect of pretrained weights and training data on the test domain performance. We basically follow the protocol of DomainBed [19] with an ImageNet supervised pretrained ResNet-50. We compare two paradigms: LP and FT. LP stands for linear-probing where only the last layer is updated. In real applications where the cost of fine-tuning the whole network is too high to be afforded, LP is used to save computational resources and guarantee the generalization ability to some extent. FT stands for fine-tuning the

whole network. In general, LP relies more on the pretrained features and FT relies more on the fine-tuned training data. For both LP and FT, we employ the standard ERM to fine-tune models on the data of training domains. In traditional computer vision tasks, if the training data is sufficient for the network to be trained, FT is expected to yield superior results as compared to LP, as it is better equipped to leverage the training data [9, 11]. However, as evident from Tab. 1, LP performs comparably with FT under many settings, in some of which simple LP even just outperforms FT. For VLCS and OfficeHome, LP outperforms FT or performs comparably with LP across almost all settings. For PACS, when using real photos as the test domain, LP outperforms FT. For TerraInc, LP also outperforms FT in the domain L38. Such a phenomenon can also be observed in DomainNet and NICO++, which is left in Appendix B.1. As LP only updates the last layer, it strongly relies on the pretrained weights to generalize on the test domains, while FT relies more on the training domains and less on the pretrained weights than LP. The above results imply that the test domain performance under many settings is mostly attributed to the information from the pretrained weights rather than the information from the training data.

In order to further investigate the phenomenon that LP outperforms FT, we vary the number of frozen layers in ResNet-50, which we consider as a 4-layer structure. Specifically, we fine-tune the network by freezing the first 1, 2, and 3 layers, respectively. Fig. 1 shows that as the number of frozen layers increases, indicating higher utilization of pretrained weights and lower utilization of training domain data, the test domain accuracies also increase. Besides, we calculate Fréchet Inception Distance (FID) [24] between some domains and ImageNet in Tab. 2 and find that in most cases, a smaller FID is accompanied by a larger value of $\frac{\mathrm{Acc_{LP}}-\mathrm{Acc_{FT}}}{\mathrm{Acc_{FT}}}$, where $\mathrm{Acc_{LP}}$ and $\mathrm{Acc_{FT}}$ represent the corresponding test accuracy. This implies that the phenomenon of LP outperforming FT can really serve as evidence of test data information leakage, i.e. a higher similarity of pretrained data to test data.

These findings raise concerns about the fairness of comparing different algorithms with the existence of test data information leakage from supervised pretrained weights. If an algorithm can improve the utilization of the pretrained weights instead of the true OOD generalization ability from training domains to test domains (e.g. in some settings, simple LP already brings higher improvement than many domain generalization algorithms), it may generate comparable or even better test domain performance than the algorithms that are better at true OOD generalization. To solve this issue, the safest choice is to train from scratch when evaluating domain generalization algorithms. Nevertheless, as pretrained models have achieved rapid growth in recent years, pretraining is commonly used to improve

Table 3. Results of linear-probing (LP) and fine-tuning (FT) with different pretraining methods of different backbone architectures on OfficeHome.

OfficeHome			A	C	P	R	Avg
ResNet-18	Supervised	LP	**57.4±0.3**	46.0±0.2	**71.3±0.2**	**74.5±0.3**	**62.3±0.1**
		FT	50.8±0.5	46.5±1.1	67.4±0.4	69.2±0.4	58.5±0.3
ResNet-50	Supervised	LP	**64.0±0.4**	50.3±0.3	**77.7±0.5**	**79.7±0.2**	**67.9±0.2**
		FT	61.1±0.6	51.1±0.3	73.9±0.5	75.7±0.7	65.5±0.2
	MoCo	LP	27.5±0.6	17.3±0.1	32.8±0.3	40.1±0.6	29.4±0.2
		FT	45.3±0.7	36.9±1.1	62.4±0.8	64.2±0.1	52.2±0.6
	MoCo-v2	LP	41.5±0.4	25.3±0.2	49.9±0.4	56.7±0.2	43.4±0.1
		FT	49.6±2.7	45.2±2.0	65.8±1.3	68.6±0.2	57.3±0.3
	SimCLR	LP	9.7±0.2	7.9±0.0	12.6±0.3	17.0±0.3	11.8±0.1
		FT	24.6±0.6	26.3±1.5	44.3±1.0	41.7±0.6	34.2±0.4
	SimCLR-v2	LP	6.3±0.3	5.8±0.2	7.5±0.2	10.2±0.2	7.5±0.1
		FT	42.9±2.7	43.9±1.8	63.3±0.7	64.9±0.4	53.8±0.3
ViT-B/16	Supervised	LP	**72.7±0.4**	57.0±0.2	**82.7±0.1**	**83.8±0.1**	**74.1±0.1**
		FT	71.1±1.2	59.1±0.6	80.6±0.9	83.3±0.4	73.5±0.5
	MoCo-v3	LP	63.1±0.3	38.8±0.2	69.1±0.3	73.0±0.3	61.0±0.2
		FT	64.7±0.6	54.1±0.5	74.8±0.7	78.3±0.7	68.0±0.4

performance without incurring additional time or financial costs. It is somewhat limited to evaluate algorithms by training from scratch which may lead to a large gap between evaluation and real-world model deployment.

Therefore, we explore alternatives of pretraining that mitigate the risk of leakage, from perspectives of backbone architectures and pretraining methods. For changing backbone architectures, we conduct experiments with supervised pretrained ResNet-18 and ViT-B/16 [16]. For changing pretraining methods, we try several self-supervised pretraining including MoCo [22], MoCo-v2 [10], SimCLR [8], SimCLR-v2 [9] for ResNet-50, and MoCo-v3 [12] for ViT-B/16. From results in Tab. 3 that take OfficeHome as an example, we find that after changing from ResNet-50 to ResNet-18 (decreasing model capacity) or ViT-B/16 (increasing model capacity), the phenomenon of LP outperforming FT still exists. However, by changing from supervised pretraining to self-supervised pretraining, the phenomenon of LP outperforming FT disappears, for both ResNet-50 and ViT-B/16. Similar observations are made for other datasets, which we leave in Appendix B.2. These results demonstrate that changing model architectures does not help, but changing to self-supervised pretraining greatly helps in mitigating potential risks of test data information leakage. Since self-supervised pretraining only utilizes images compared with supervised pretraining utilizing both images and category labels, these results adhere to our intuition that self-supervised pretraining may bear less leakage.

Overall, we suggest that we should use self-supervised pretrained weights or train from scratch to mitigate the potential risk of test information leakage for fairer evaluation.

2.2. Oracle Model Selection

Model selection has emerged as a crucial problem for OOD generalization. In the context of IID generalization, training

Table 4. Results of increasing the number of test domains for mitigating test information leakage from oracle model selection for DomainNet and NICO++.

DomainNet		quickdraw	real	sketch	Avg	leakage	
IID		11.09	60.75	47.98	39.94	0.00	/
Oracle	K=1	11.42	62.57	48.30	40.76	0.82	-0%
	K=2	10.95	62.54	48.03	40.51	0.57	-31%
	K=3	10.81	62.54	48.07	40.47	0.53	-35%

DomainNet		clipart	infograph	painting	Avg	leakage	
IID		58.23	19.20	47.49	41.64	0.00	/
Oracle	K=1	58.23	19.23	47.67	41.71	0.07	-0%
	K=2	58.19	19.22	47.67	41.69	0.05	-24%
	K=3	58.15	19.23	47.67	41.68	0.04	-38%

NICO++		grass	outdoor	water	Avg	leakage	
IID		80.10	74.69	68.06	74.28	0.00	/
Oracle	K=1	81.19	75.58	69.85	75.54	1.26	-0%
	K=2	80.54	74.98	69.85	75.12	0.84	-33%
	K=3	80.47	75.58	68.97	75.01	0.72	-42%

NICO++		autumn	rock	dim	Avg	leakage	
IID		78.94	79.26	71.94	76.71	0.00	/
Oracle	K=1	79.44	79.58	72.58	77.20	0.49	-0%
	K=2	78.92	79.33	72.58	76.94	0.23	-53%
	K=3	79.44	79.58	71.42	76.81	0.10	-79%

data, validation data, and test data are typically drawn from the same distribution, so the generalization performances on validation data and test data are generally consistent. However, in the case of domain generalization, test distribution differs from training distribution. Since test data should be unknown, naturally the validation data should also come from the same distribution as training data does, thus the consistency between validation data performance and test data performance cannot be guaranteed. When evaluating on public benchmarks, despite not being reasonable, it is possible to exploit test data for hyperparameter tuning and model selection, referred to as "oracle" model selection. This serves as another form of test data information leakage. DomainBed [19] has proved that such an oracle model selection method outperforms the standard model selection strategy that utilizes a validation set sampled from the same distribution as training data. While DomainBed has integrated the validation process into the framework to reduce the possibility of using test data, there are still some degrees of freedom. For example, the hyperparameter search space can be customized and narrowed in order to reduce the computational cost of the evaluation process (In the current standard protocol of DomainBed, each setting randomly generates 20 hyperparameter sets for 3 random seeds each, leading to the requirement of training 60 models).

To further address this issue, we observe that the current protocol of domain generalization, as defined by Equation 1, only considers a single test domain for each trained model. Though the test domain varies across different settings, it is fixed for a single setting. For example, when evaluating on DomainNet with domains *real* and *clipart*, we do not directly train a single model for the two settings with each domain treated as the test domain. Instead, we train two separate models using different training data comprising domains other than the test domain. The two hyperparameter search and model selection processes are also performed independently for each model. As there is only one test domain, the room for increasing the test performance through oracle model selection is relatively large. Intuitively, using oracle model selection to "fit" the distribution of multiple test domains is usually harder than "fit" the distribution of a single test domain. Thus we consider introducing multiple test domains to alleviate the test data information leakage from the oracle model selection method.

We conduct experiments to confirm the above intuition. We adopt DomainNet and NICO++ due to their relatively larger number of domains. We split the domains of each dataset into two groups. For DomainNet, we split them into (quickdraw, real, sketch) and (clipart, infograph, painting). For NICO++, we split them into (grass, outdoor, water) and (autumn, rock, dim). For each dataset, we train on each group respectively and test on the other group. Validation data is randomly split from training data. For IID model selection, we choose the test accuracy corresponding to the hyperparameters with the highest accuracy on validation data. For oracle model selection, we directly choose the highest test accuracy across all hyperparameter sets. We vary the number of test domains K and compute the test accuracy of each test domain through the average of its accuracy across every combination of K test domains that includes this test domain. For example, for the domain *autumn* belonging to the group (autumn, rock, dim), for $K = 2$, we calculate the average of domain autumn's accuracy when using (autumn, rock) and (autumn, dim) for oracle model selection respectively. We quantify the difference in test accuracy between IID model selection and oracle model selection as the possible "leakage". Tab. 4 shows that increasing the number of test domains does help mitigate the possible leakage from oracle model selection. For DomainNet, when $K = 2$, the leakage can already be reduced by about 30 percent. For NICO++, the leakage can be reduced by about half when increasing K. If there are datasets that can support the evaluation of more test domains, such leakage can be further reduced. Overall, we recommend using multiple test domains to alleviate the possible information leakage from oracle model selection.

3. New Leaderboards

The above analyses suggest that to reduce the risk of test data information leakage and ensure accurate evaluation of

OOD generalization, it is advisable to use self-supervised pretrained weights or train from scratch, and to increase the number of test domains. We have accordingly introduced these two modifications to the DomainBed protocol and present new leaderboards that are compared with the old one which follows the current DomainBed protocol.

3.1. Experimental settings

Protocol modifications For self-supervised pretraining investigated in Tab. 3, we choose MoCo-v2 [10] pretrained ResNet-50 for the new evaluation protocol since it outperforms other self-supervised pretrained weights when using ResNet-50 as backbones. To support comparisons on more advanced architectures like ViT, we employ MoCo-v3 [12] pretrained ViT-B/16. Since there are only 4 domains in datasets other than DomainNet and NICO++, we still employ leave-one-domain-out strategy. For DomainNet and NICO++ with 6 domains each, we divide them into 3 groups: (clipart, infograph), (painting, quickdraw), (real, sketch) for DomainNet, and (autumn, rock), (dim, grass), (outdoor, water) for NICO++. We employ the leave-one-group-out strategy so that each time we have 2 domains for testing. Due to observations that there are relatively large random fluctuations in the results of TerraInc both in our experiments and in DomainBed, we do not adopt it in our new leaderboards. Besides, we present leaderboards without pretraining for DomainNet and NICO++ in Appendix C.4, since only they are large enough for ResNet and ViT to be sufficiently trained on from scratch.

Algorithms We test 10 algorithms following the modified protocol: ERM [73], SWAD [7, 29], RSC [25], GroupDRO [63], Fishr [61], CORAL [68, 69], MMD [42], SagNet [57], IRM [1], Mixup [80, 86].

ERM directly optimizes sample averaged loss, typically used in traditional machine learning tasks with IID assumption. Among these algorithms, Fishr, CORAL, MMD, SagNet, and IRM aim to achieve some form of invariance across domains. Mixup aims to enhance the diversity of the training data. Some of the algorithms were originally developed for other areas, such as CORAL and inter-domain Mixup for domain adaptation, IRM for invariant learning, GroupDRO for subpopulation shift, and SWAD adapted from the optimizer seeking flat minima.

Other details For PACS, VLCS, and OfficeHome, we set the number of iterations as 5,000 following DomainBed [19]. For DomainNet, we set it as 15,000 following Cha et al. [7] since the training loss has not converged yet at the iteration of 5,000. For NICO++, we set it as 10,000. To reduce the computational cost of the current DomainBed protocol, for each setting, namely each combination of an algorithm and a pair of training and test domains, we randomly search the hyperparameters over a predefined distribution with 10 trials (instead of 20 trials in DomainBed). We use the selected best hyperparameters to run 2 more times with different random seeds (instead of conducting 3 independent hyperparameter searches for each random seed). The total cost is training 12 models (instead of 60 models). As for the predefined hyperparameter search space, we change the search space of MMD gamma from log uniform distribution of $10^{\text{uniform}(-1,1)}$ to that of $10^{\text{uniform}(-2,0)}$ otherwise we observe that training loss will not decrease. Besides, in all our experiments, we use the Adam optimizer [32] with a Cosine Annealing Scheduler. For details of training from scratch, we put them in Appendix C.4. For other details, such as data augmentation and data split, we directly follow DomainBed.

3.2. Results

Tab. 5 presents the old leaderboard based on supervised pretrained ResNet-50 and the two new leaderboards. In the old leaderboard, the results of SWAD are from Cha et al. [7], the results of Fishr are from Rame et al. [61], and the others are from the standard leaderboard maintained in the official code repository of DomainBed [19]. The last column ΔR represents the change of ranking for the algorithm, where we mark the largest changes with bold type. The detailed leaderboards for each dataset are in Appendix C.

Performance rankings of some algorithms show great variations after applying the modified protocol. For instance, RSC ranks 2nd in both new leaderboards with self-supervised pretraining, implying its effectiveness, but it fails to outperform ERM in the old leaderboard with supervised pretraining. For a recent SOTA algorithm Fishr proposed based on the current DomainBed protocol, it achieves a high test accuracy in the old leaderboard but fails to outperform ERM in the new ones. A similar conclusion stands for SagNet too. Coupled with our analyses in Sec. 2.1, this raises concerns about the preciseness and fairness of the current evaluation protocol.

Rankings in self-supervised pretraining leaderboards are more consistent with leaderboards of training from scratch than supervised pretraining leaderboards. We calculate spearman rank correlation for algorithm performance in different leaderboards. The rank correlation between supervised pretraining and training from scratch is 0.261 while MoCo-v2 and MoCo-v3's rank correlations with training from scratch are 0.576 and 0.600 respectively. Besides, rank correlation between MoCo-v2 and MoCo-v3 is 0.794. This implies that evaluation based on self-supervised pretraining is more effective than the currently used supervised pretraining since it serves as a better surrogate of training from scratch. This also confirms that self-

Table 5. Leaderboards comparison between the current DomainBed protocol and the modified evaluation protocol of adopting self-supervised pretraining and using multiple test domains.

Old leaderboard: Supervised pretrained ResNet-50								
Algorithm	PACS	VLCS	OfficeHome	NICO++	DomainNet	Average	Ranking	ΔR
ERM	85.5±0.2	77.5±0.4	66.5±0.3	77.5±0.1	40.9±0.1	69.6	6	-
SWAD	88.1±0.1	79.1±0.1	70.6±0.2	80.2±0.1	46.5±0.1	72.9	1	-
RSC	85.2±0.9	77.1±0.5	65.5±0.9	78.1±0.2	38.9±0.5	69.0	7	-
GroupDRO	84.4±0.8	76.7±0.6	66.0±0.7	77.6±0.4	33.3±0.2	67.6	8	-
Fishr	85.5±0.4	77.8±0.1	67.8±0.1	78.4±0.1	41.7±0.0	70.2	3	-
CORAL	86.2±0.3	78.8±0.6	68.7±0.3	79.3±0.1	41.5±0.1	70.9	2	-
MMD	84.6±0.5	77.5±0.9	66.3±0.1	77.3±0.2	23.4±9.5	65.8	10	-
SagNet	86.3±0.2	77.8±0.5	68.1±0.1	78.6±0.2	40.3±0.1	70.2	4	-
IRM	83.5±0.8	78.5±0.5	64.3±2.2	77.5±0.3	33.9±2.8	67.5	9	-
Mixup	84.6±0.6	77.4±0.6	68.1±0.3	78.9±0.0	39.2±0.1	69.6	5	-

New leaderboard: MoCo-v2 pretrained ResNet-50								
Algorithm	PACS	VLCS	OfficeHome	NICO++	DomainNet	Average	Ranking	ΔR
ERM	84.1±0.3	76.9±0.8	57.3±0.3	71.1±0.1	39.3±0.3	65.7	6	-
SWAD	86.2±0.6	77.7±0.7	62.6±0.1	75.6±0.1	42.9±0.1	69.0	1	-
RSC	85.9±1.9	78.5±0.3	60.3±0.3	74.2±0.5	40.5±0.4	67.9	2	**+5**
GroupDRO	83.0±0.5	76.5±0.8	57.4±0.8	71.2±0.2	36.3±0.4	64.9	10	-2
Fishr	82.8±1.0	75.4±1.6	58.7±0.2	71.0±0.1	39.3±0.3	65.4	8	**-5**
CORAL	84.0±1.4	77.5±0.4	62.6±0.3	73.7±0.2	41.2±0.1	67.8	3	-1
MMD	84.3±1.2	76.7±1.3	57.6±0.8	71.6±0.3	39.3±0.3	65.9	5	**+5**
SagNet	82.5±1.1	75.5±0.8	59.2±1.0	70.2±0.3	38.6±0.2	65.2	9	**-5**
IRM	83.5±0.8	75.3±0.5	57.9±0.3	71.5±0.3	39.1±0.1	65.5	7	+2
Mixup	82.5±1.6	76.1±0.7	60.5±0.2	73.3±0.2	39.2±0.1	66.3	4	+1

New leaderboard: MoCo-v3 pretrained ViT-B/16								
Algorithm	PACS	VLCS	OfficeHome	NICO++	DomainNet	Average	Ranking	ΔR
ERM	85.8±0.3	78.4±0.6	68.0±0.4	79.6±0.2	47.4±0.2	71.8	5	+1
SWAD	88.1±0.3	79.0±0.1	71.4±0.4	80.8±0.0	49.6±0.1	73.8	1	-
RSC	86.8±0.5	78.2±1.1	67.9±0.1	79.7±0.1	47.3±0.1	72.0	2	**+5**
GroupDRO	85.6±0.8	78.2±0.9	68.0±0.2	79.7±0.2	44.9±0.1	71.3	8	-
Fishr	86.6±1.3	78.0±0.4	67.7±0.3	79.6±0.2	47.2±0.1	71.8	6	-3
CORAL	86.7±0.9	78.1±0.6	67.8±0.5	79.5±0.1	47.5±0.1	71.9	3	-1
MMD	86.4±0.9	64.4±4.3	67.2±0.2	69.7±1.0	47.3±0.1	67.0	10	-
SagNet	85.6±1.0	78.0±0.5	66.9±0.2	79.2±0.2	46.5±0.1	71.2	9	**-5**
IRM	84.7±0.5	78.1±0.3	68.2±0.5	79.7±0.1	47.3±0.1	71.6	7	+2
Mixup	83.4±0.8	78.2±0.8	70.0±0.1	79.8±0.2	48.0±0.1	71.9	4	+1

supervised pretraining helps in alleviating the potential test data information leakage.

SWAD consistently ranks 1st in every leaderboard. In leaderboards of supervised pretraining, self-supervised pretraining, and training from scratch, SWAD always ranks 1st and consistently improves upon ERM. This strongly demonstrates its effectiveness and universality in OOD generalization. Such a result, along with the high rank of RSC in the new leaderboards, indicates that intrinsic general-

ization properties and mechanisms like flatness or dropout could be less sensitive to test data information leakage and closer to essence of OOD generalization.

In conclusion, we present the new leaderboards using the modified protocol to mitigate the possible test data information leakage, so that we can promote a fairer and more accurate evaluation and comparison between the domain generalization algorithms for their OOD generalization ability.

4. Discussion

4.1. Position of pretraining for OOD generalization

With the rapid development of pretrained models, including large language models like ChatGPT and LLaMA [72], and large multi-modal models like GPT-4V [58] and LLaVA [48], nowadays typically we tend to take advantage of pretraining in real applications. Despite their remarkable performance, current studies show that they are far from perfect and still suffer from performance degradation under distribution shifts, even for GPT-3.5 [81, 85] and GPT-4V [93]. Besides, it is hard to collect sufficient diverse data to train or fine-tune large models due to high expenses and privacy issues in some areas like medical care, where distribution shifts prevail. Thus the problem of OOD generalization still holds great significance in the era of large pretrained models.

However, with the existence of pretrained models, proper evaluation of OOD generalization becomes a natural problem [83]. Since we focus on the ability of models generalizing from training data to test data, strong pretrained weights naturally bring about possible test information leakage as we have analyzed in Sec. 2.1. Kumar et al. [37] also provide theoretical analyses that with good pretrained weights and a strong distribution shift, LP will outperform FT in that FT can distort the pretrained features, which are already good enough to generalize to test data. Our work serves as an initial effort to provide a better evaluation protocol for OOD generalization under pretraining.

Recently, there have been works showing that pretraining on larger datasets with larger architecture backbones greatly improves test performance in OOD tasks [31, 77, 84], and some directly design algorithms for better utilization of pretrained weights to improve test domain performance [47, 62]. We believe this is an interesting and meaningful research direction of valuable practical usage. If we directly focus on improving test performance, we should employ pretrained weights that are as strong and powerful as possible. However, for a fairer and more accurate evaluation of OOD generalization from training data to test data during the fine-tuning stage, we should seek pretrained weights that exhibit less test data information leakage.

4.2. OOD model selection

Model selection is another important topic for OOD generalization. DomainBed [19] has pointed out that oracle model selection (test-domain validation) leaks test data information in domain generalization. A similar problem also exists in subpopulation shift and is even more severe. Idrissi et al. [28] demonstrate that model selection based on the average accuracy on an IID validation set can lead to significant performance degradation compared with using worst group accuracy. The latter strategy utilizes the group label information, which should be considered as oracle model selection at least for methods claiming no need for group information. In this paper, we adhere to DomainBed's IID validation (training-domain validation) to prevent test information leakage for the guarantee of a fair and accurate comparison, but it does not mean that IID validation is the only right way for model selection. Considering the natural inconsistency of performance on IID validation data and test data [70], maybe IID validation is not the best approach to guiding model selection. Improving the model selection strategy remains a fundamental research problem for OOD generalization in the future.

4.3. Datasets for domain generalization

It has been a long time since domain generalization algorithms are primarily evaluated on relatively small datasets, e.g. Colored MNIST [1], Rotated MNIST [18], PACS [39], VLCS [30], OfficeHome [74], and TerraInc [3]. Based on our analyses in this paper, our recommendation for expanding the number of test domains requires the establishment of larger datasets, and it is the same if we want to add more datasets to the leaderboard of training from scratch, which only DomainNet [59] and NICO++ [90] can support currently. It is much more challenging to construct a large, high-quality, and balanced dataset for domain generalization than to construct one for IID generalization, as it involves creating many domains with common classes. Recently, Lynch et al. [54] released a dataset Spawrious for better evaluation of OOD generalization, whose data is completely generated and collected from stable diffusion models. This provides a new direction for constructing domain generalization datasets with much lower cost.

5. Acknowledgements

This work was supported in part by National Natural Science Foundation of China (No. 62141607), China National Postdoctoral Program for Innovative Talents (BX20230195). Peng Cui is the corresponding author. All opinions in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

References

[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. 1, 6, 8, 13

[2] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. *Advances in neural information processing systems*, 31, 2018. 13

[3] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pages 456–473, 2018. 3, 8, 13

[4] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, 50(1):3–44, 2021. 1

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 2

[6] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019. 3

[7] 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: 22405–22418, 2021. 3, 6, 13

[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 4

[9] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020. 4

[10] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2, 4, 6

[11] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021. 4

[12] Xinlei Chen*, Saining Xie*, and Kaiming He. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021. 2, 4, 6

[13] Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *International Conference on Machine Learning*, pages 2189–2200. PMLR, 2021. 1

[14] Peng Cui and Susan Athey. Stable learning establishes some common ground between causal inference and machine learning. *Nature Machine Intelligence*, 4(2):110–115, 2022. 1

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[16] 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. 2, 4

[17] John C Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021. 1

[18] 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*, pages 2551–2559, 2015. 8, 13

[19] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2020. 1, 2, 3, 5, 6, 8, 13

[20] Sivan Harary, Eli Schwartz, Assaf Arbelle, Peter Staar, Shady Abu-Hussein, Elad Amrani, Roei Herzig, Amit Alfassy, Raja Giryes, Hilde Kuehne, et al. Unsupervised domain generalization by learning a bridge across domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5280–5290, 2022. 13

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[22] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 4

[23] Yue He, Peng Cui, Jianxin Ma, Hao Zou, Xiaowei Wang, Hongxia Yang, and Philip S Yu. Learning stable graphs from multiple environments with selection bias. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2194–2202, 2020. 1

[24] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 4

[25] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *European Conference on Computer Vision*, pages 124–140. Springer, 2020. 3, 6, 13

[26] Zeyi Huang, Haohan Wang, Dong Huang, Yong Jae Lee, and Eric P Xing. The two dimensions of worst-case training and their integrated effect for out-of-domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9631–9641, 2022. 13

[27] Brody Huval, Tao Wang, Sameep Tandon, Jeff Kiske, Will Song, Joel Pazhayampallil, Mykhaylo Andriluka, Pranav Ra-

jpurkar, Toki Migimatsu, Royce Cheng-Yue, et al. An empirical evaluation of deep learning on highway driving. *arXiv preprint arXiv:1504.01716*, 2015. 1

[28] Badr Youbi Idrissi, Martin Arjovsky, Mohammad Pezeshki, and David Lopez-Paz. Simple data balancing achieves competitive worst-group-accuracy. In *Conference on Causal Learning and Reasoning*, pages 336–351. PMLR, 2022. 8

[29] P Izmailov, AG Wilson, D Podoprikhin, D Vetrov, and T Garipov. Averaging weights leads to wider optima and better generalization. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, pages 876–885, 2018. 6

[30] Aditya Khosla, Tinghui Zhou, Tomasz Malisiewicz, Alexei A Efros, and Antonio Torralba. Undoing the damage of dataset bias. In *European Conference on Computer Vision*, pages 158–171. Springer, 2012. 2, 3, 8, 13

[31] Donghyun Kim, Kaihong Wang, Stan Sclaroff, and Kate Saenko. A broad study of pre-training for domain generalization and adaptation. In *European Conference on Computer Vision*, pages 621–638. Springer, 2022. 8

[32] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[33] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 3

[34] 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*, pages 5815–5826. PMLR, 2021. 1

[35] Kun Kuang, Ruoxuan Xiong, Peng Cui, Susan Athey, and Bo Li. Stable prediction with model misspecification and agnostic distribution shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4485–4492, 2020. 1

[36] Matjaž Kukar. Transductive reliability estimation for medical diagnosis. *Artificial Intelligence in Medicine*, 29(1-2): 81–106, 2003. 1

[37] 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*, 2021. 8

[38] Jesse Levinson, Jake Askeland, Jan Becker, Jennifer Dolson, David Held, Soeren Kammel, J Zico Kolter, Dirk Langer, Oliver Pink, Vaughan Pratt, et al. Towards fully autonomous driving: Systems and algorithms. In *2011 IEEE intelligent vehicles symposium (IV)*, pages 163–168. IEEE, 2011. 1

[39] 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*, pages 5542–5550, 2017. 2, 3, 8, 13

[40] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 13

[41] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M Hospedales. Episodic training for domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1446–1455, 2019. 13

[42] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5400–5409, 2018. 1, 6, 13

[43] Wen Li, Zheng Xu, Dong Xu, Dengxin Dai, and Luc Van Gool. Domain generalization and adaptation using low rank exemplar svms. *IEEE transactions on pattern analysis and machine intelligence*, 40(5):1114–1127, 2017.

[44] Ya Li, Mingming Gong, Xinmei Tian, Tongliang Liu, and Dacheng Tao. Domain generalization via conditional invariant representations. In *Proceedings of the AAAI conference on artificial intelligence*, 2018.

[45] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 624–639, 2018. 13

[46] Yiying Li, Yongxin Yang, Wei Zhou, and Timothy Hospedales. Feature-critic networks for heterogeneous domain generalization. In *International Conference on Machine Learning*, pages 3915–3924. PMLR, 2019. 13

[47] Ziyue Li, Kan Ren, Xinyang Jiang, Yifei Shen, Haipeng Zhang, and Dongsheng Li. Simple: Specialized model-sample matching for domain generalization. In *The Eleventh International Conference on Learning Representations*, 2022. 8

[48] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 8

[49] Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Heterogeneous risk minimization. In *International Conference on Machine Learning*, pages 6804–6814. PMLR, 2021. 1, 13

[50] Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Integrated latent heterogeneity and invariance learning in kernel space. *Advances in Neural Information Processing Systems*, 34:21720–21731, 2021. 1

[51] Jiashuo Liu, Zheyan Shen, Peng Cui, Linjun Zhou, Kun Kuang, Bo Li, and Yishi Lin. Stable adversarial learning under distributional shifts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8662–8670, 2021. 1

[52] Jiashuo Liu, Jiayun Wu, Bo Li, and Peng Cui. Distributionally robust optimization with data geometry. *Advances in neural information processing systems*, 35:33689–33701, 2022. 1

[53] Jiashuo Liu, Jiayun Wu, Renjie Pi, Renzhe Xu, Xingxuan Zhang, Bo Li, and Peng Cui. Measure the predictive heterogeneity. In *The Eleventh International Conference on Learning Representations*, 2022. 13

[54] Aengus Lynch, Gbètondji JS Dovonon, Jean Kaddour, and Ricardo Silva. Spawrious: A benchmark for fine control of spurious correlation biases. *arXiv preprint arXiv:2303.05470*, 2023. 8

[55] Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. In *International conference on machine learning*, pages 7313–7324. PMLR, 2021. 13

[56] Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11749–11756, 2020. 3

[57] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8690–8699, 2021. 3, 6

[58] OpenAI. Gpt-4 technical report, 2023. 8

[59] 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*, pages 1406–1415, 2019. 2, 3, 8, 13

[60] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2

[61] Alexandre Rame, Corentin Dancette, and Matthieu Cord. Fishr: Invariant gradient variances for out-of-distribution generalization. In *International Conference on Machine Learning*, pages 18347–18377. PMLR, 2022. 3, 6, 13

[62] Alexandre Rame, Kartik Ahuja, Jianyu Zhang, Matthieu Cord, Léon Bottou, and David Lopez-Paz. Model ratatouille: Recycling diverse models for out-of-distribution generalization. 2023. 8

[63] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2019. 6

[64] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In *International Conference on Learning Representations*, 2018. 13

[65] Zheyan Shen, Peng Cui, Tong Zhang, and Kun Kunag. Stable learning via sample reweighting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5692–5699, 2020. 1

[66] 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. 1, 13

[67] Aman Sinha, Hongseok Namkoong, and John Duchi. Certifying some distributional robustness with principled adversarial training. In *International Conference on Learning Representations*, 2018. 1

[68] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016. 6

[69] Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016. 6

[70] Damien Teney, LIN Yong, Seong Joon Oh, and Ehsan Abbasnejad. Id and ood performance are sometimes inversely correlated on real-world datasets. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 8

[71] Yunze Tong, Junkun Yuan, Min Zhang, Didi Zhu, Keli Zhang, Fei Wu, and Kun Kuang. Quantitatively measuring and contrastively exploring heterogeneity for domain generalization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2189–2200, 2023. 13

[72] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 8

[73] Vladimir Vapnik. Principles of risk minimization for learning theory. *Advances in neural information processing systems*, 4, 1991. 3, 6

[74] 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*, pages 5018–5027, 2017. 3, 8, 13

[75] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *Advances in neural information processing systems*, 31, 2018. 1, 13

[76] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip Yu. Generalizing to unseen domains: A survey on domain generalization. *IEEE Transactions on Knowledge and Data Engineering*, 2022. 1, 2, 13

[77] Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre-Alvise Rebuffi, Ira Ktena, Krishnamurthy Dj Dvijotham, and Ali Taylan Cemgil. A fine-grained analysis on distribution shift. In *International Conference on Learning Representations*, 2021. 8

[78] Renzhe Xu, Xingxuan Zhang, Zheyan Shen, Tong Zhang, and Peng Cui. A theoretical analysis on independence-driven importance weighting for covariate-shift generalization. In *International Conference on Machine Learning*, pages 24803–24829. PMLR, 2022. 1

[79] Zheng Xu, Wen Li, Li Niu, and Dong Xu. Exploiting low-rank structure from latent domains for domain generalization. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part III 13*, pages 628–643. Springer, 2014. 13

[80] Shen Yan, Huan Song, Nanxiang Li, Lincan Zou, and Liu Ren. Improve unsupervised domain adaptation with mixup training. *stat*, 1050:3, 2020. 6, 13

[81] Linyi Yang, Shuibai Zhang, Libo Qin, Yafu Li, Yidong Wang, Hanmeng Liu, Jindong Wang, Xing Xie, and Yue Zhang. Glue-x: Evaluating natural language understanding

models from an out-of-distribution generalization perspective. *arXiv preprint arXiv:2211.08073*, 2022. 8

[82] Han Yu, Peng Cui, Yue He, Zheyan Shen, Yong Lin, Renzhe Xu, and Xingxuan Zhang. Stable learning via sparse variable independence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10998–11006, 2023. 1

[83] Han Yu, Jiashuo Liu, Xingxuan Zhang, Jiayun Wu, and Peng Cui. A survey on evaluation of out-of-distribution generalization. *arXiv preprint arXiv:2403.01874*, 2024. 1, 8, 13

[84] Yaodong Yu, Heinrich Jiang, Dara Bahri, Hossein Mobahi, Seungyeon Kim, Ankit Singh Rawat, Andreas Veit, and Yi Ma. An empirical study of pre-trained vision models on out-of-distribution generalization. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021. 8

[85] Lifan Yuan, Yangyi Chen, Ganqu Cui, Hongcheng Gao, FangYuan Zou, Xingyi Cheng, Heng Ji, Zhiyuan Liu, and Maosong Sun. Revisiting out-of-distribution robustness in nlp: Benchmarks, analysis, and llms evaluations. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. 8

[86] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 6

[87] Xingxuan Zhang, Peng Cui, Renzhe Xu, Linjun Zhou, Yue He, and Zheyan Shen. Deep stable learning for out-of-distribution generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5372–5382, 2021. 1, 13

[88] Xingxuan Zhang, Yue He, Tan Wang, Jiaxin Qi, Han Yu, Zimu Wang, Jie Peng, Renzhe Xu, Zheyan Shen, Yulei Niu, et al. Nico challenge: Out-of-distribution generalization for image recognition challenges. In *European Conference on Computer Vision*, pages 433–450. Springer, 2022. 13

[89] Xingxuan Zhang, Linjun Zhou, Renzhe Xu, Peng Cui, Zheyan Shen, and Haoxin Liu. Towards unsupervised domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4910–4920, 2022. 13

[90] Xingxuan Zhang, Yue He, Renzhe Xu, Han Yu, Zheyan Shen, and Peng Cui. Nico++: Towards better benchmarking for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16036–16047, 2023. 2, 3, 8, 13

[91] Xingxuan Zhang, Renzhe Xu, Han Yu, Yancheng Dong, Pengfei Tian, and Peng Cui. Flatness-aware minimization for domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5189–5202, 2023. 13

[92] Xingxuan Zhang, Renzhe Xu, Han Yu, Hao Zou, and Peng Cui. Gradient norm aware minimization seeks first-order flatness and improves generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20247–20257, 2023. 13

[93] Xingxuan Zhang, Jiansheng Li, Wenjing Chu, Junjia Hai, Renzhe Xu, Yuqing Yang, Shikai Guan, Jiazheng Xu, and Peng Cui. On the out-of-distribution generalization of multimodal large language models. *arXiv preprint arXiv:2402.06599*, 2024. 8

[94] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *International Conference on Learning Representations*, 2020. 1

[95] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 1, 2, 13