MAP: Towards Balanced Generalization of IID and OOD through Model-Agnostic Adapters

Min Zhang[1], Junkun Yuan[1], Yue He[2], Wenbin Li[3], Zhengyu Chen[1], Kun Kuang[1]*
[1]Zhejiang University [2]Tsinghua University [3]Nanjing University
heyuethu@mail.tsinghua.edu.cn, liwenbin@nju.edu.cn,
{zhangmin.milab, yuanjk, chenzhengyu, kunkuang}@zju.edu.cn

Abstract

*Deep learning has achieved tremendous success in recent years, but most of these successes are built on an independent and identically distributed (IID) assumption. This somewhat hinders the application of deep learning to the more challenging out-of-distribution (OOD) scenarios. Although many OOD methods have been proposed to address this problem and have obtained good performance on testing data that is of major shifts with training distributions, interestingly, we experimentally find that these methods achieve excellent OOD performance by making a great sacrifice of the IID performance. We call this finding the IID-OOD dilemma. Clearly, in real-world applications, distribution shifts between training and testing data are often uncertain, where shifts could be minor, and even close to the IID scenario, and thus it is truly important to design a deep model with the balanced generalization ability between IID and OOD. To this end, in this paper, we investigate an intriguing problem of balancing IID and OOD generalizations and propose a novel **M**odel **A**gnostic ada**P**ters (MAP) method, which is more reliable and effective for distribution-shift-agnostic real-world data. Our key technical contribution is to use auxiliary adapter layers to incorporate the inductive bias of IID into OOD methods. To achieve this goal, we apply a bilevel optimization to explicitly model and optimize the coupling relationship between the OOD model and auxiliary adapter layers. We also theoretically give a first-order approximation to save computational time. Experimental results on six datasets successfully demonstrate that MAP can greatly improve the performance of IID while achieving good OOD performance.*

1. Introduction

Deep learning has achieved unprecedented success in various applications of computer vision, *e.g.*, image classification [15, 17, 18], but most of these successes are

*Corresponding author.

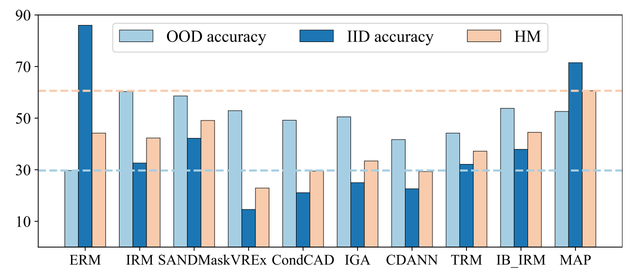

Figure 1. Comparison of OOD and IID accuracy of OOD, IID and our proposed MAP methods on ColoredMNIST. HM is the harmonic mean of IID and OOD accuracy. All OOD methods achieve high OOD performance with the sacrifice of IID accuracy compared with ERM (*i.e.*, an IID method). MAP achieves balanced generalization by incorporating IID inductive bias into OOD generalization learning. More results are shown in Table 1.

based on an independent and identically distributed (IID) assumption, *i.e.*, training and testing data are drawn from the same distribution [41, 16]. However, out-of-distribution (OOD) shifts between training and testing data are usually inevitable in the real world due to the widespread existence of unobserved confounders or data bias [46, 8]. Under such circumstances, deep models trained by empirical risk minimization (ERM) [51] with the IID assumption usually suffer from poor performance on OOD data. Therefore, it is important to improve the OOD generalization of deep models.

Recently, many OOD methods have been proposed to learn representations or predictors that are invariant to different distributions (or named environments) by introducing various regularizers [3, 4, 44, 1, 26, 2, 34, 64, 63, 50]. Although these methods achieve good OOD performance on testing data that is of major distribution shifts with training data, we experimentally found that they would significantly damage the performance on IID (with nearly no shift discrepancy) or minor shift data. We implement some representative OOD methods in both OOD and IID scenarios on ColoredMNIST and show results in Figure 1. We have an interesting observation that these methods have **significant** OOD accuracy but **lower** IID performance compared with the IID method (*e.g.*, ERM) with **higher** IID performance

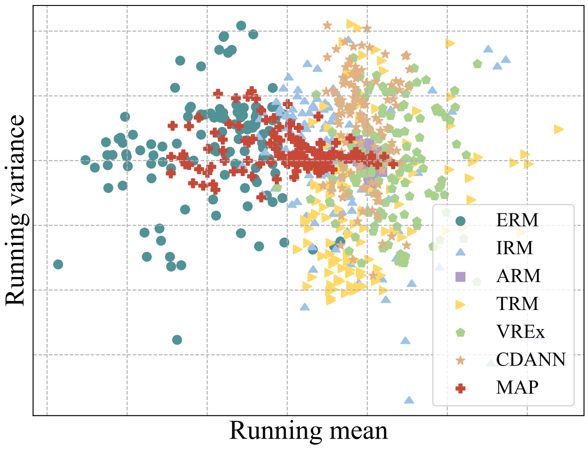

Figure 2. Comparison of running means and variances of BatchNorm layer on ColoredMNIST. IID and OOD methods learn different inductive biases and perform well in either IID or OOD scenarios. The proposed MAP method achieves balanced generalization performance by capturing both kinds of inductive bias.

but **lower** OOD accuracy. A possible reason for causing this phenomenon is that many OOD methods extract invariant features while possibly losing some information that contributes to IID generalization. Next, we explore this phenomenon from the perspective of inductive bias learned by different IID and OOD methods inspired by [36].

In Figure 2, following [53], we visualize channel-wise BatchNorm (BN) statistics of the 4-th layer as the inductive bias (other layers have similar observations). The inductive bias, *i.e.*, running means and variances, of IID and OOD methods are significantly different, *i.e.*, IID (OOD) methods are routed to BN_{IID} (BN_{OOD}). As pointed out by [2, 32], ERM extracts **easy-to-learn** variant features (*e.g.*, color on ColoredMNIST) in training distributions and generalizes well to testing data with the same distribution. In comparison, these OOD methods adopt regularizers to encourage the model to extract **hard-to-learn** invariant features (*e.g.*, digit on ColoredMNIST) and improve the performance of testing data that differ significantly from training distributions. Moreover, the regularizer guides the OOD model in different optimization directions compared with the IID method, which causes good OOD accuracy with low IID results. The finding is named as the IID-OOD dilemma. Both IID and OOD methods can only perform well in a specific scenario (IID or OOD), which limits their real-world applications with uncertain distribution shifts. Therefore, this observation motivates us to ask: *is it possible to design a model with a balanced performance between IID and OOD generalizations in the IID-OOD dilemma*?

In this paper, we take a step forward to propose a novel **M**odel **A**gnostic ada**P**ters (MAP) method that achieves balanced generalization performance in both IID and OOD evaluations. Specifically, we insert auxiliary adapter layers (AALs) in the OOD model to learn variant features with the inductive bias of the IID scenario, while keeping the ability to extract invariant features with the inductive bias of the OOD scenario. Training processes of the OOD model and AALs are viewed as two kinds of tasks: the OOD model learns OOD knowledge and AALs extract IID information. To achieve this, we formulate the learning into a bilevel optimization (BLO) problem. In the inner level, we optimize the OOD model with AALs by using an OOD loss. In the outer level, we utilize the IID criterion evaluated on the validation set based on the optimized OOD model in the inner level as the outer objective to guide the training of AALs. We alternatively perform the inner level and outer level and finally obtain a set of optimal parameters for the adapter and OOD model. To save computational time and memory, we theoretically give a first-order approximation of BLO. Note that AALs are model-agnostic and can be plugged into an arbitrary OOD method. Experiments evaluate the effectiveness of MAP, which improves the trade-off ability of OOD methods (see the HM metric in Figure 1) by capturing the inductive bias of both IID and OOD scenarios in Figure 2. Our main contributions are summarized as follows:

- We investigate a problem called the IID-OOD dilemma, *i.e.*, most OOD (or IID) methods achieve good OOD (or IID) performance with a sacrifice of IID (or OOD) accuracy, which is beyond the capability of these methods in real-world data with uncertain shifts.
- We propose a simple yet effective **M**odel **A**gnostic ada**P**ters (MAP) method to simultaneously learn inductive biases of both IID and OOD. To achieve this, a bilevel optimization (BLO) is used to train our MAP. Unlike the computationally intensive BLO solver, we theoretically give a first-order approximation.
- We conduct extensive experiments across six datasets, three model architectures, and sixteen baselines. We show that (1) MAP balances the performance of IID and OOD. (2) MAP is model-agnostic and can be plugged into any OOD method. (3) MAP is able to achieve reliable performance under various settings.

2. Related Work

Out-of-distribution generalization. To enable deep learning models to generalize to unknown data distributions, the task of out-of-distribution (OOD) generalization aims to train a generalizable model from one or multiple source domains and make it predict well on the previously unseen target domains. To get rid of the independent and identically distributed (IID) assumption employed by the conventional algorithms like empirical risk minimization (ERM) [51], a variety of OOD strategies have been proposed to overcome distribution shifts, including regularized training [60, 26, 38, 47, 23, 54, 57], meta-learning [27, 7, 59, 22, 62], data augmentation [55, 42], domain alignment [3, 28], causal learning [2, 33, 21], etc. Taking regularized training methods to extract invariant features as an example. IRM [4] learns invariant representations with a classifier optimal to domain changes as a regularization term. MTL [5] uses an extended input pattern with an estimated domain embedding and implements regularized learning over a reproducing kernel Hilbert space. ANDMask [37] and SAND-Mask [45] regularize the model training by updating parameters on the direction where gradient components have consistent signs across domains. Despite the good OOD performance these methods have achieved, we empirically found that compared with IID methods, existing OOD methods have a considerable sacrifice of IID to improve the OOD performance. However, real-world testing data is uncertain, which could even show similar statistical distributions to IID training data. In this paper, taking a step forward, we propose a novel method that achieves a balance between IID and OOD performance, which allows models to better handle uncertain and complex real-world data.

Bilevel optimization (BLO). BLO [49], also known as learning to learn or meta-learning, focuses on training a meta-learner that can learn how to train other models. The episodic training strategy from MAML [11], which simulates a set of tasks and makes the model learn to generalize its learning to different tasks, has been widely employed in OOD methods for overcoming distribution shifts [27, 31, 40, 61, 6, 48, 52, 39, 58, 14]. As a pioneer work, MLDG [27] makes the model learn how to generalize to unseen domains by simulating distribution shifts with virtual target data from source domains. ARM [60] optimizes the model for effective adaptation to shift by learning to adapt on training domains. Fish [47] augments the loss with an auxiliary term that maximizes the gradient inner product between domains to encourage the alignment between the domain-specific gradients. These methods aim to improve the OOD performance of the model by using bilevel optimization (BLO), while our goal is to optimize MAP so that it can balance the performance of IID and OOD data.

3. Preliminaries

In this section, we first formulate the problem definition. Then, we detail the optimization processes of IID and OOD.

Problem definition. Given a dataset $\mathcal{D} := \{(x_i, y_i)\}_{i=1}^n$ with n samples (x_i, y_i) is drawn from a joint space of $\mathcal{X}\times\mathcal{Y}$. In general, $\mathcal{D}$ contains the training data $\mathcal{D}_{tr}$ sampled from the training distribution $\mathcal{P}_{tr}(\mathcal{X}, \mathcal{Y})$ and the testing data $\mathcal{D}_{te}$ drawn from the testing distribution $\mathcal{P}_{te}(\mathcal{X}, \mathcal{Y})$. Supervised learning methods aim to predict labels y_i of x_i originate from a featurizer $f(\cdot;\theta)$ parameterized by θ and the classifier $g(\cdot;\phi)$ parameterized by ϕ, *i.e.*, $\theta : \mathcal{X} \to \mathcal{Z}$ and $\phi : \mathcal{Z} \to \mathcal{Y}$. $\mathcal{Z}$ represents the sample feature space.

IID learning. Deep learning usually assumes that training and testing data are both IID realizations from a common underlying distribution, *i.e.*, $\mathcal{P}_{tr}(\mathcal{X}, \mathcal{Y}) = \mathcal{P}_{te}(\mathcal{X}, \mathcal{Y})$. Based on such a hypothesis, empirical risk minimization (ERM) which minimizes the average loss on training samples could optimize the model for the testing distribution. Specifically, ERM minimizes the following objective:

$$\mathcal{L}_{ERM}(\mathcal{D}_{tr}, \theta, \phi) = \frac{1}{n}\sum_{i=1}^{n} \ell(g_\phi(f_\theta(x_i)), y_i), \tag{1}$$

where $\ell(\cdot,\cdot)$ is the loss function, *e.g.*, cross entropy for image classification. $z_i = f_\theta(x_i)$ is the feature representation and $\tilde{y}_i = g_\phi(z_i) = g_\phi(f_\theta(x_i))$ is the prediction.

OOD learning. There is an OOD problem when the testing distribution is unseen and different from the training distribution, *i.e.*, $\mathcal{P}_{tr}(\mathcal{X}, \mathcal{Y}) \neq \mathcal{P}_{te}(\mathcal{X}, \mathcal{Y})$. Specifically, following [4, 1], we have multiple environments (or distributions) $\mathcal{E} = \{e_1, e_2, \cdots, e_E\}$ in the sample space $\mathcal{X}\times\mathcal{Y}$ with the training distribution. The correlation between the non-sematic information (*i.e.*, variant features) and labels (*i.e.*, invariant features) is unstable among different environments. Existing OOD methods learn invariant representations or predictors by introducing the regularizer. The optimization process of OOD methods (IRM [4] and VERx [26] as an example due to its simple yet effective) is as below:

$$\begin{aligned}\mathcal{R}^{IRMv1}(\mathcal{D}_{tr}, \theta, \phi) = \sum_e \mathcal{L}_{ERM}(\mathcal{D}^e_{tr}, \theta, \phi) \\ + \lambda || \bigtriangledown_{\theta,\phi=1.0}\, \mathcal{L}_{ERM}(\mathcal{D}^e_{tr}, \theta, \phi)||_2^2,\end{aligned} \tag{2}$$

$$\begin{aligned}\mathcal{R}^{VREx}(\mathcal{D}_{tr}, \theta, \phi) = \sum_e \mathcal{L}_{ERM}(\mathcal{D}^e_{tr}, \theta, \phi) \\ + \lambda \mathcal{V}_e[\mathcal{L}_{ERM}(\mathcal{D}^e_{tr}, \theta, \phi)],\end{aligned} \tag{3}$$

where $\mathcal{V}_e[\mathcal{L}(\mathcal{D}^e, \phi)]$ is the variance of the loss across different environments. Equation (2) is the optimization objective of IRM using the fixed "dummy" classifier. Equation (3) is the VREx optimization objective. $\lambda \in [0, \infty)$ is a hyperparameter to balance between the ERM and regularizer loss.

4. Methodology

To address the interesting problem of the IID-OOD dilemma, in this section, we propose a model-agnostic adapters (MAP) method by introducing auxiliary adapter layers (AALs) in the OOD model. Specifically, as illustrated in Figure 3, we incorporate AALs into the OOD model to learn the inductive bias of IID and OOD data by using a bilevel alternating way. In the following subsections, we detail the bilevel optimization (BLO) process and the specific form of our proposed auxiliary adapter layers.

4.1. Bilevel Optimization

As shown in Figure 2, OOD and IID models learn different inductive biases from training data to improve the generalization in testing data. Under such circumstances, these models cannot achieve good performance under both IID and OOD. To improve the trade-off of IID and OOD, we design MAP to help OOD models learn IID knowledge by

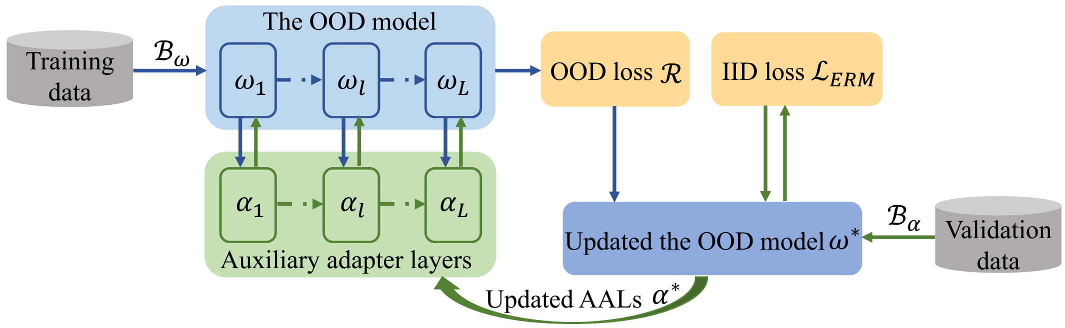

Figure 3. **Method overview.** (1) In the inner level (*i.e*, blue arrow $\rightarrow$), the OOD model is optimized to ω^* by using the batch $\mathcal{B}_\omega$ sampled from training data and the OOD loss $\mathcal{R}$, where each layer of the OOD model incorporates the proposed auxiliary adapter layers (AALs), *i.e.*, (ω_l, α_l) with $l \in [1, \cdots, L]$ (see Equation (5)). (2) In the outer level (*i.e.*, green arrow $\rightarrow$), the updated OOD model ω^* is used to optimize AALs with the batch $\mathcal{B}_\alpha$ sampled from validation data according to the feedback from the ERM loss $\mathcal{L}_{ERM}$ (see Equation (4)).

integrating AALs into OOD models. Our goal is to let OOD models extract invariant features (*e.g.*, object information) that help OOD generalization while also focusing on variant features (*e.g.*, background) that improve IID performance.

To this end, the learning paradigm of AALs and OOD models is viewed as two kinds of tasks: ❶ OOD models learn OOD inductive bias, and ❷ AALs recognize IID inductive bias. Moreover, we interpret the OOD-learning task (*i.e.*, ❶) and the AALs-learning task (*i.e.*, ❷) as two optimization levels, where the latter is formulated as an outer-level optimization problem, and it relies on the optimization of the inner-level OOD-learning task. To the best of our knowledge, the bilevel optimization (BLO) framework has not been considered for balancing IID and OOD in-depth and systematically. The model optimization is as the following BLO problem (❷ being nested inside ❶):

$$\underset{\alpha}{\text{minimize}} \quad \underbrace{\mathcal{L}_{ERM}(\mathcal{D}_{val}, \theta^*(\alpha), \phi^*(\alpha), \alpha)}_{\text{❶: Updating AALs } \alpha}, \tag{4}$$

$$s.t. \quad \underbrace{\{\theta^*(\alpha), \phi^*(\alpha)\} = \arg\min_{\theta,\phi} \overbrace{\mathcal{R}(\mathcal{D}_{tr}, \theta, \phi, \alpha)}^{\text{Equation (2) or (3)}}}_{\text{❷: Updating OOD models } \theta \text{ and } \phi}, \tag{5}$$

where $\mathcal{D}_{val}$ denotes validation data which is obtained by flipping training data $\mathcal{D}_{tr}$. α is the parameters of auxiliary adapter layers (AALs). $\{\theta^*, \phi^*\}$ is the inner-level solution obtained by minimizing the objective function $\mathcal{R}$ given α. By alternatively performing inner level and outer level, AALs gradually evolve to the state of being able to produce satisfactory OOD and IID performance with OOD training.

The BLO formulation has the following advantages: (1) BLO has the flexibility to use mismatched OOD and IID optimization objectives at outer and inner levels, respectively. Moreover, different objectives use different sample batches. To be specific, $\mathcal{B}_\omega$ with $\omega = \{\theta, \phi\}$ is sampled from training data $\mathcal{D}_{tr}$ to update inner-level OOD models while $\mathcal{B}_\beta$ is sampled from validation data $\mathcal{D}_{val}$ to optimize outer-level AALs. It can improve the generalization ability of the model [10, 20, 12]. (2) By alternatively performing inner and outer levels, AALs and OOD models can find optimal parameters to extract invariant features that help OOD learning and variant features that help IID generalization.

4.2. BLO with Gradient Approximation

The computation of implicit gradient (IG) is the key challenge of optimizing Equation (4). In this section, to solve the IG challenge, we propose an alternating approximation algorithm to save computational time and memory.

Updating ω in the inner level. In each outer iteration, instead of completely solving the inner level problem, we fix auxiliary adapter layers α and only consider gradient steps of the model parameters ω at the t-th iteration as follows:

$$\omega^{(t)} = \omega^{(t-1)} - \eta_\omega \bigtriangledown_\omega \mathcal{R}(\mathcal{B}_\omega, \omega^{(t-1)}, \alpha^{(t-1)}), \tag{6}$$

where $\bigtriangledown_\omega$ is partial derivatives of ω. η_ω is the learning rate for model parameters ω. $\mathcal{B}_\omega$ is batch sampled from $\mathcal{D}_{tr}$.

Updating α in the outer level. After obtaining the parameters $\omega^{(t)}$ (a reasonable approximation of $\omega^*(\alpha)$), we update α by calculating the outer level optimization objective as:

$$\alpha^{(t)} = \alpha^{(t-1)} - \eta_\alpha \bigtriangledown_\alpha \mathcal{L}_{ERM}(\mathcal{B}_\alpha, \omega^{(t)}, \alpha^{(t-1)}), \tag{7}$$

where $\bigtriangledown_\alpha$ means partial derivatives of α. η_α is the learning rate for adapter parameters α. $\mathcal{B}_\alpha$ is batch sampled from validation data $\mathcal{D}_{val}$ flipping by $\mathcal{D}_{tr}$. When we directly backpropagate the gradient, the IG problem occurs because $\omega^{(t)}$ nested inside $\alpha^{(t)}$. Therefore, we propose a method to approximate the gradient $\bigtriangledown_\alpha \mathcal{L}_{ERM}(\mathcal{B}_\alpha, \omega^{(t)}, \alpha^{(t-1)})$ of α^t (see supplementary for detailed derivations) as below:

$$\begin{aligned} &\underbrace{\bigtriangledown_\alpha \mathcal{L}_{ERM}(\omega^{(t)}, \alpha^{(t-1)})}_{\text{Gradient of AALs}} = \bigtriangledown_\omega \mathcal{L}_{ERM}(\omega^{(t)}, \alpha^{(t-1)}) \cdot \overbrace{\bigtriangledown_\alpha \omega^*(\alpha)}^{\text{IG}} \\ &= -\eta_\omega \frac{1}{\epsilon}(\bigtriangledown_\alpha \mathcal{R}(\omega^{(t-1)} + \epsilon v, \alpha^{(t-1)})) - \bigtriangledown_\alpha \mathcal{L}_{ERM}(\omega^{(t-1)}, \alpha^{(t-1)}), \end{aligned} \tag{8}$$

where $v = \bigtriangledown_\omega \mathcal{L}_{ERM}(\omega^{(t)}, \alpha^{(t-1)})$ with small $\epsilon > 0$. For ease of notation, we omit $\mathcal{B}_\omega$ and $\mathcal{B}_\alpha$ in loss $\mathcal{R}$ and

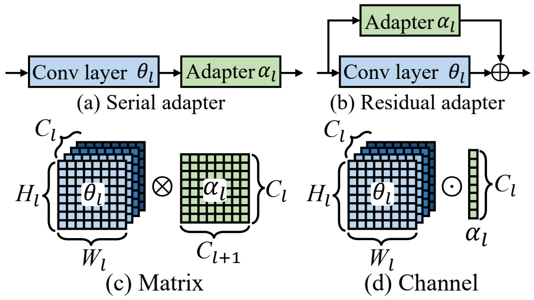

Figure 4. Two connection ways in (a) and (b) to insert our adapters into the OOD module. Two forms of adapters in (c) and (d).

$\mathcal{L}_{ERM}$, respectively. Equation (8) can be easily implemented by maintaining $\omega^{(t-1)}$ at last iteration to catch the OOD loss $\mathcal{R}(\omega^{(t-1)}, \alpha^{(t-1)})$ and compute the new loss $\mathcal{R}(\omega^{(t-1)}+\epsilon v, \alpha^{(t-1)})$. When η_ω is set to 0 in Equation (8), the second-order derivative will disappear, resulting in a first-order approximate. The complexity of the first order is the same as OOD methods, and the performance is as efficient as the second order (see results in Section 5.4).

4.3. Auxiliary Adapter Layers

In this subsection, we discuss that *how to connect the OOD featurizer with AALs and how to design the form of AALs*. For the first problem, we insert the adapter between the conv layer and BN, as shown in the right of Figure 5 in each OOD module (left of Figure 5). Concretely, The $l \in [1, \cdots, L]$-th layer of the featurizer f_θ is denoted as f_{θ_l} with the weights θ_l. And the adapter in each module is denoted as A_α parameterized by α. The information extracted by A_{α_l} can be incorporated into the output of the l-th layer as the input of $l+1$ layer. The formulation is as follows:

$$f_{\{\theta_l,\alpha_l\}}(z_l) = A_{\alpha_l}(f_{\theta_l}(z_l), z_l), \tag{9}$$

where $z_l \in \mathbb{R}^{W_l \times H_l \times C_l}$ is the feature tensor that is the input of the l-th module θ_l. W_l, H_l and C_l are the width, height and channel of the l-th convolutional layer, respectively. Motivated by [29], we consider two connection ways for incorporating adapter A_{α_l} into OOD featurizer f_{θ_l}: ❶ serial connection by subsequently applying it to the output:

$$f_{\{\theta_l,\alpha_l\}}(z_l) = A_{\alpha_l} \circ f_{\theta_l}(z_l), \tag{10}$$

which is illustrated in Figure 4 (a), and ❷ parallel connection by a residual addition and is illustrated in Figure 4 (b):

$$f_{\{\theta_l,\alpha_l\}}(z_l) = A_{\alpha_l} + f_{\theta_l}(z_l). \tag{11}$$

For the second problem, we consider two options for A_{α_l}: ❶ matrix multiplication with $\alpha_l \in \mathbb{R}^{C_l \times C_{l+1}}$ in Figure 4 (c), where C_l and C_{l+1} are the number of input and output channels, respectively. The formulation is as below:

$$A_{\alpha_l}(f_{\theta_l}(z_l)) = z_l \otimes \alpha_l, \tag{12}$$

Algorithm 1 Training process of MAP

Require: Training data D_{tr}, validation data D_{val} by flipping D_{tr}, inner- and outer-level learning rate η_ω and η_α

Randomly initialize all learnable parameters $\{\omega = (\theta, \phi), \alpha\}$
for Iteration $t = 0, 1, \cdots$ **do**
  Pick different random data batches $\mathcal{B}_\omega$ and $\mathcal{B}_\alpha$ for different levels of data $\mathcal{D}_{tr}$ and $\mathcal{D}_{val}$, respectively
  //Inner-level: update the OOD model ω using the OOD loss:
  $$\omega^{(t)} = \omega^{(t-1)} - \eta_\omega \bigtriangledown_\omega \mathcal{R}(\mathcal{B}_\omega, \omega^{(t-1)}, \alpha^{(t-1)})$$
  //Outer-level: update the adapter α using the ERM loss:
  $$\alpha^{(t)} = \alpha^{(t-1)} - \eta_\alpha \bigtriangledown_\alpha \mathcal{L}_{ERM}(\mathcal{B}_\alpha, \omega^{(t)}, \alpha^{(t-1)})$$
end for

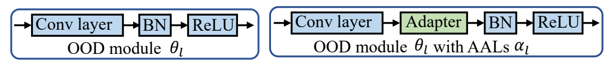

Figure 5. The difference between the OOD module with (right) and without (left) adapter. Conv layer, BN and RELU is the convolutional layer, BathchNorm and activate function, respectively.

where $\otimes$ denotes a convolutional operation with 1×1 kernels in our code. ❷ channel wise scaling in Figure 4 (d):

$$A_{\alpha_l}(f_{\theta_l}(z_l)) = z_l \odot \alpha_l, \tag{13}$$

where $\odot$ is a Hadamard product and $\alpha_l \in \mathbb{R}^{C_l}$. The whole optimization of MAP is illustrated in Algorithm 1.

5. Experiments

In this section, we evaluate the performance of the proposed MAP, aiming to answer the following questions: **Q1**: Could MAP balance the robustness of IID and OOD generalization compared with prior IID and OOD methods? (Section 5.2) **Q2**: Could MAP be the model-agnostic adapters? (Section 5.3) **Q3**: How effective is the proposed MAP in different settings (or ablation study)? (Section 5.4)[1].

5.1. Experimental Setup

Dataset. We use six OOD classification datasets, including three toy, *i.e.*, ColoredMNIST [4], ColoredCOCO [1], COCOPlaces [1] and three real, *i.e.*, NICO [19], CelebA [35], WILDSCamelyon [24] (more details in the supplementary).
Baseline methods. We compare our MAP to a large number of algorithms that span different learning strategies, including (1) IID learning: ERM [51] (2) OOD learning (fifteen methods): IRM [4], VREx [26], ARM [60], GroupDRO [44], MLDG [27], MMD [28], IGA [25], SANDMask [45], Fish [47], CDANN [30], TRM [54] IB_ERM [2], IB_IRM [2], CondCAD [42], CausIRL_CORAL [9] where ARM, MLDG and Fish also use the bilevel optimization (see more details in Section 2).
Backbone. We use the four-layer convolutional neural network (Conv4) for ColoredMNIST and ResNet18 pretrained by ImageNet [43] for three real datasets. Following [1], a

[1]The code is available at: https://github.com/remiMZ/MAP-ICCV23.

Methods	ColoredMNIST OOD	ColoredMNIST IID	ColoredMNIST **HM**	ColoredCOCO OOD	ColoredCOCO IID	ColoredCOCO **HM**	COCOPlaces OOD	COCOPlaces IID	COCOPlaces **HM**
ERM [51]	29.7 $\pm$ 0.2	**86.0 $\pm$ 0.2**	44.2	45.4 $\pm$ 0.9	77.0 $\pm$ 0.6	57.1	20.1 $\pm$ 0.7	**72.6 $\pm$ 0.5**	31.5
IRM [4]	**60.3 $\pm$ 2.8**	32.6 $\pm$ 7.0	42.3	49.2 $\pm$ 0.3	70.9 $\pm$ 1.7	58.1	**27.1 $\pm$ 0.9**	60.6 $\pm$ 2.6	37.5
VREx [26]	52.9 $\pm$ 1.2	14.6 $\pm$ 0.3	22.9	48.8 $\pm$ 0.7	73.6 $\pm$ 0.9	58.7	26.2 $\pm$ 0.7	64.5 $\pm$ 1.0	37.3
GroupDRO [44]	38.5 $\pm$ 1.5	51.5 $\pm$ 0.3	44.1	49.1 $\pm$ 0.6	74.8 $\pm$ 1.8	59.3	26.9 $\pm$ 0.6	64.8 $\pm$ 1.4	38.0
MLDG [27]	29.4 $\pm$ 0.6	50.3 $\pm$ 0.0	34.6	11.9 $\pm$ 0.8	20.5 $\pm$ 0.1	15.1	14.6 $\pm$ 0.5	16.7 $\pm$ 2.3	15.6
MMD [28]	50.6 $\pm$ 0.1	51.3 $\pm$ 0.6	51.0	50.4 $\pm$ 0.8	73.8 $\pm$ 1.0	59.9	26.3 $\pm$ 1.7	68.0 $\pm$ 1.5	37.9
IGA [25]	50.5 $\pm$ 0.1	25.0 $\pm$ 7.9	33.4	11.0 $\pm$ 0.6	17.5 $\pm$ 2.7	13.5	10.8 $\pm$ 0.3	11.6 $\pm$ 0.8	11.2
SANDMask [45]	58.6 $\pm$ 6.5	42.2 $\pm$ 7.2	49.1	49.2 $\pm$ 1.2	74.0 $\pm$ 0.7	59.1	25.9 $\pm$ 1.4	66.2 $\pm$ 0.3	37.2
Fish [47]	28.0 $\pm$ 1.5	46.4 $\pm$ 3.2	34.9	41.7 $\pm$ 0.5	71.7 $\pm$ 0.4	52.7	19.3 $\pm$ 2.1	55.9 $\pm$ 3.2	28.7
CDANN [30]	41.7 $\pm$ 3.5	22.6 $\pm$ 1.5	29.3	38.4 $\pm$ 1.5	70.1 $\pm$ 1.3	49.6	19.4 $\pm$ 1.0	58.2 $\pm$ 2.7	29.1
TRM [54]	44.2 $\pm$ 5.0	32.1 $\pm$ 9.5	37.2	47.5 $\pm$ 0.6	72.8 $\pm$ 0.2	57.5	24.8 $\pm$ 1.1	60.8 $\pm$ 0.6	35.2
IB_ERM [2]	50.2 $\pm$ 0.2	51.7 $\pm$ 1.7	50.9	45.4 $\pm$ 1.1	72.4 $\pm$ 2.5	55.8	20.2 $\pm$ 1.0	60.3 $\pm$ 0.6	30.2
CausIRL_CORAL [9]	28.7 $\pm$ 1.3	50.6 $\pm$ 0.2	36.6	**51.5 $\pm$ 1.1**	73.9 $\pm$ 0.9	60.7	26.1 $\pm$ 1.1	66.3 $\pm$ 1.3	37.5
CondCAD [42]	49.2 $\pm$ 0.5	21.1 $\pm$ 2.6	29.5	41.2 $\pm$ 0.7	67.2 $\pm$ 1.3	51.1	20.8 $\pm$ 0.3	60.6 $\pm$ 0.4	31.0
IB_IRM [2]	53.8 $\pm$ 1.8	37.9 $\pm$ 10.0	44.5	33.9 $\pm$ 0.6	67.3 $\pm$ 1.4	45.1	14.8 $\pm$ 2.3	53.5 $\pm$ 0.5	23.2
ARM [60]	28.1 $\pm$ 0.0	49.9 $\pm$ 0.1	36.0	33.0 $\pm$ 0.6	63.3 $\pm$ 0.6	43.4	25.1 $\pm$ 0.2	52.7 $\pm$ 0.4	34.0
MAP (ours)	52.6 $\pm$ 0.5	71.5 $\pm$ 0.7	**60.6**	50.9 $\pm$ 1.3	**78.1 $\pm$ 1.1**	**61.6**	26.9 $\pm$ 1.0	69.1 $\pm$ 0.8	**38.7**

Methods	NICO OOD	NICO IID	NICO **HM**	CelebA OOD	CelebA IID	CelebA **HM**	WILDSCamelyon OOD	WILDSCamelyon IID	WILDSCamelyon **HM**
ERM [51]	73.6 $\pm$ 1.9	**91.1 $\pm$ 1.0**	81.4	85.4 $\pm$ 1.1	**97.2 $\pm$ 0.1**	90.9	94.7 $\pm$ 0.1	**98.0 $\pm$ 0.0**	96.3
IRM [4]	75.8 $\pm$ 2.0	87.2 $\pm$ 0.9	81.1	83.9 $\pm$ 2.0	96.3 $\pm$ 0.3	86.7	95.4 $\pm$ 0.2	95.0 $\pm$ 0.1	95.2
VREx [26]	76.9 $\pm$ 0.7	88.5 $\pm$ 1.1	82.3	84.4 $\pm$ 0.7	95.9 $\pm$ 0.2	89.8	94.2 $\pm$ 0.1	97.0 $\pm$ 0.2	95.6
GroupDRO [44]	74.6 $\pm$ 2.4	85.4 $\pm$ 1.2	79.6	85.5 $\pm$ 0.7	96.6 $\pm$ 0.2	90.7	95.2 $\pm$ 0.1	95.6 $\pm$ 0.5	95.4
MLDG [27]	68.4 $\pm$ 2.7	52.4 $\pm$ 5.1	59.3	82.9 $\pm$ 2.3	93.1 $\pm$ 1.1	87.7	87.9 $\pm$ 1.0	86.6 $\pm$ 1.4	87.2
MMD [28]	**78.2 $\pm$ 1.2**	87.2 $\pm$ 0.8	**82.5**	86.2 $\pm$ 0.3	93.0 $\pm$ 1.7	89.5	94.7 $\pm$ 0.2	96.5 $\pm$ 0.1	95.6
IGA [25]	48.1 $\pm$ 1.3	53.9 $\pm$ 4.1	50.8	80.3 $\pm$ 2.0	92.2 $\pm$ 1.5	85.8	59.1 $\pm$ 1.3	88.0 $\pm$ 1.2	70.7
SANDMask [45]	72.8 $\pm$ 1.5	86.0 $\pm$ 2.2	78.9	84.8 $\pm$ 0.4	95.4 $\pm$ 0.2	90.2	**96.4 $\pm$ 0.1**	96.2 $\pm$ 0.3	96.3
Fish [47]	77.0 $\pm$ 1.2	88.0 $\pm$ 0.9	82.1	84.4 $\pm$ 0.8	96.9 $\pm$ 0.1	90.2	95.0 $\pm$ 0.1	96.5 $\pm$ 0.1	95.7
CDANN [30]	72.8 $\pm$ 1.8	84.4 $\pm$ 1.2	78.2	85.2 $\pm$ 1.0	96.9 $\pm$ 0.2	90.7	96.3 $\pm$ 0.1	97.2 $\pm$ 0.2	96.7
TRM [54]	73.0 $\pm$ 0.9	79.2 $\pm$ 5.0	76.0	83.9 $\pm$ 0.5	96.7 $\pm$ 0.2	89.8	96.3 $\pm$ 0.1	97.2 $\pm$ 0.2	96.7
IB_ERM [2]	77.7 $\pm$ 1.9	71.9 $\pm$ 12.2	74.7	84.7 $\pm$ 0.5	96.6 $\pm$ 0.2	90.2	96.1 $\pm$ 0.1	97.1 $\pm$ 0.1	96.6
CausIRL_CORAL [9]	75.7 $\pm$ 0.9	87.3 $\pm$ 0.6	81.1	84.9 $\pm$ 1.3	96.9 $\pm$ 0.2	90.5	95.4 $\pm$ 0.1	96.0 $\pm$ 0.2	95.7
CondCAD [42]	73.9 $\pm$ 1.4	88.1 $\pm$ 0.6	80.4	84.2 $\pm$ 0.6	96.2 $\pm$ 0.1	89.8	96.1 $\pm$ 0.1	96.8 $\pm$ 0.2	96.4
IB_IRM [2]	70.2 $\pm$ 2.2	86.6 $\pm$ 0.6	77.5	85.5 $\pm$ 0.7	94.1 $\pm$ 1.4	89.6	96.3 $\pm$ 0.1	97.3 $\pm$ 0.3	96.8
ARM [60]	76.4 $\pm$ 1.6	87.9 $\pm$ 1.4	81.7	86.9 $\pm$ 0.5	95.9 $\pm$ 0.2	91.2	93.5 $\pm$ 0.5	95.9 $\pm$ 0.5	94.7
MAP (ours)	76.8 $\pm$ 1.4	89.0 $\pm$ 0.6	**82.5**	**87.3 $\pm$ 0.5**	96.4 $\pm$ 0.1	**91.6**	95.3 $\pm$ 0.3	97.8 $\pm$ 0.2	**97.4**

Table 1. Experiments of three toy (top) and three real (bottom) datasets. Here, we show average accuracy (%) of IID and OOD. **HM** is the harmonic mean as a trade-off metric. We repeat experiments three times across 20 hyperparameter seeds by following DomainBed [13].

residual network trained from scratch is used for Colored-COCO and COCOPlaces and is called as ResNet8. For Conv4, AALs are placed behind the convolutional layer. For residual networks, we only use AALs in each block.

Model selection and implementation details. To evaluate the performance of IID and OOD, we split each environment for each dataset into two subsets of d_1 and d_2, where the number of samples of d_1 and d_2 is 9:1. The subset d_1 of training environments is used to train the model, and d_2 is used to evaluate IID accuracy. While the subset d_1 of testing environments is used to evaluate OOD accuracy, and d_2 is used to select the best model (or named oracle selection) by following the standard protocol of DomainBed [13, 56]. Then IID and OOD accuracy are calculated based on the selected model. Note that MAP uses the OOD loss of the VREx [26] method in the inner level.

5.2. Evaluating the Balance of IID and OOD Data

In Table 1, we report the overall performance of MAP and sixteen baselines under IID and OOD evaluations on six datasets. We further reported the harmonic mean ($HM = \frac{2x_1x_2}{x_1+x_2}$) of accuracy on IID and OOD data. Following [36], we use this metric to evaluate the trade-off between IID and OOD performance. According to Table 1, we have the following findings: (1) Compared with the IID method (*i.e.*, ERM), these fifteen OOD methods have good OOD perfor-

Methods	ColoredMNIST OOD	ColoredMNIST IID	ColoredMNIST **HM**	ColoredCOCO OOD	ColoredCOCO IID	ColoredCOCO **HM**	NICO OOD	NICO IID	NICO **HM**
IRM [4]	60.3 ± 2.8	32.6 ± 7.0	42.3	49.2 ± 0.3	70.9 ± 1.7	58.1	75.8 ± 2.0	87.2 ± 0.9	81.1
+ MAP	57.3 ± 3.2	55.3 ± 2.9	**56.3** +14.0	47.9 ± 1.1	75.6 ± 1.2	**58.6** +0.5	76.2 ± 1.1	88.7 ± 2.5	**82.0** +0.9
VREx [26]	52.9 ± 1.3	14.6 ± 0.3	22.9	48.8 ± 0.7	73.6 ± 0.9	58.7	76.9 ± 0.7	88.5 ± 1.1	82.3
+ MAP	52.6 ± 0.5	71.5 ± 1.2	**60.6** +37.7	50.9 ± 1.3	78.1 ± 1.1	**61.6** +2.9	77.6 ± 0.8	89.1 ± 0.6	**83.0** +0.7
ARM [60]	28.1 ± 0.0	49.9 ± 0.1	36.0	33.0 ± 0.6	63.3 ± 0.6	43.4	76.4 ± 1.6	87.9 ± 1.4	81.7
+ MAP	58.1 ± 4.5	69.4 ± 2.7	**63.2** +27.2	32.8 ± 0.5	66.5 ± 1.1	**44.0** +0.6	75.2 ± 1.5	89.6 ± 1.5	**81.8** +0.1
GroupDRO [44]	38.5 ± 1.5	51.5 ± 0.3	44.1	49.1 ± 0.6	74.8 ± 1.8	59.3	74.6 ± 2.4	85.4 ± 1.2	79.6
+ MAP	38.9 ± 1.5	64.3 ± 3.5	**48.5** +4.4	48.3 ± 0.7	76.8 ± 0.8	**59.3** +0.0	75.8 ± 1.4	87.5 ± 1.7	**81.2** +1.6
CDANN [30]	41.7 ± 3.5	22.6 ± 1.5	29.3	38.4 ± 1.5	70.1 ± 1.3	49.6	72.8 ± 1.8	84.4 ± 1.2	78.2
+ MAP	49.7 ± 0.4	44.8 ± 0.6	**47.1** +17.8	38.1 ± 0.7	74.9 ± 2.9	**50.5** +0.9	72.5 ± 1.0	85.6 ± 1.1	**78.5** +0.3
TRM [54]	44.2 ± 5.0	32.1 ± 9.5	37.2	45.0 ± 0.8	72.8 ± 0.2	55.6	73.0 ± 0.9	79.2 ± 5.0	76.0
+ MAP	50.3 ± 0.3	53.6 ± 1.3	**51.9** +14.7	44.8 ± 0.9	75.9 ± 0.8	**56.3** +0.7	75.1 ± 0.4	78.9 ± 2.7	**77.0** +1.0
IB_ERM [2]	50.2 ± 0.2	51.7 ± 1.7	50.9	45.4 ± 1.1	72.4 ± 2.5	55.8	77.7 ± 1.9	71.9 ± 12.2	74.7
+ MAP	52.5 ± 0.3	62.2 ± 1.9	**57.0** +6.1	46.9 ± 0.1	75.6 ± 2.1	**57.9** +2.1	75.6 ± 0.5	73.8 ± 1.2	**74.7** +0.0
IB_IRM [2]	53.8 ± 1.8	37.9 ± 10.0	44.5	33.9 ± 0.6	67.3 ± 1.4	45.1	70.2 ± 2.2	86.6 ± 0.6	77.5
+ MAP	56.3 ± 0.1	50.1 ± 7.5	**53.0** +8.5	33.4 ± 0.6	69.8 ± 1.1	**45.2** +0.1	77.8 ± 2.4	85.6 ± 0.7	**81.5** +4.0

Table 2. Average accuracy (%) of eight OOD methods with or without using our MAP on ColoredMNIST, ColoredCOCO and NICO.

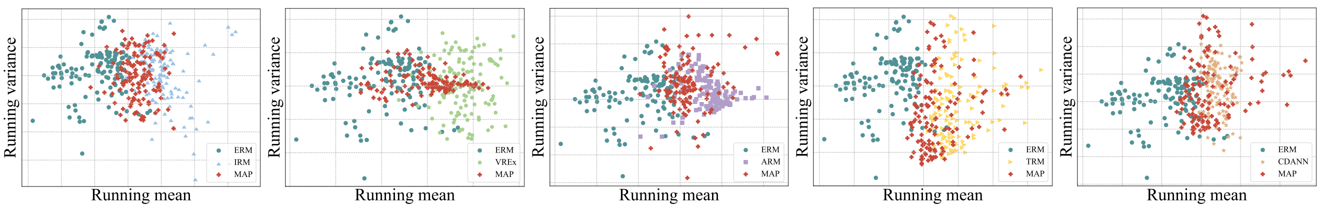

Figure 6. Running means and variances of channel-wise BathchNorm of five OOD methods with and without using the proposed MAP.

mance, but a significant drop in IID accuracy on all datasets, which demonstrates our motivation, *i.e.*, most OOD methods might lose some information that helps the IID learning in the OOD generalization. Furthermore, since the spurious correlation might be weaker on real datasets, the performance gap between IID and OOD methods becomes smaller. This phenomenon demonstrates that the IID model extracts easy-to-learn variant features to learn the inductive bias of training data, in comparison, the OOD model learns hard-to-learn invariant features to improve the performance of unseen testing distributions. (2) According to the HM metric, MAP achieves the balanced generalization ability on all datasets. This is because MAP has the advantage of simultaneously capturing the inductive bias between IID and OOD data, which demonstrates the effectiveness of our method. (3) Surprisingly, MAP even increases the OOD performance of OOD methods in some settings and the IID performance in ColoredCOCO outperforms ERM by 1.1%. Similar conclusions can be obtained based on Table 2. One possible reason is that the bilevel optimization can find suitable model parameters for IID and OOD generalizations (see more discussion in Sections 5.3).

5.3. Evaluating the Flexibility of MAP

We study how the proposed MAP improves the trade-off ability of many OOD methods. In Table 2, we show results by using eight OOD methods with or without MAP on three datasets. (1) We can observe that all OOD methods with MAP outperform those without MAP, especially in using VREx on ColoredMNIST brings a 37% improvement. It not only indicates that MAP is a model-agnostic framework, but also significantly improves the trade-off ability of these OOD methods. (2) With a deep look at the OOD evaluation, MAP shows its comparable performance. Surprisingly, MAP even increases the OOD performance in some OOD methods, especially on ColoredMNIST. We think that the iterative learning of the OOD model and AALs using bilevel optimization may help the OOD model to explore more knowledge that is helpful for OOD generalization.

In Figure 6, we visualize the inductive bias of five OOD methods with or without using MAP to observe the learned statistics information. The inductive bias learned by MAP can capture both IID and OOD scenarios. An interesting observation is that compared with CDANN, MAP not only uses adapters to capture IID generalization information but also explores more OOD generalization knowledge. A similar observation is made in the ARM method. This phenomenon shows that MAP has the ability to simultaneously improve the performance of IID and OOD evaluations.

5.4. Ablation Study

We conduct extensive ablation studies to evaluate the robustness of the proposed MAP under various settings.

Comparison of different structural designs of MAP. In Table 3, we analyze the impact of different connections (*i.e.*, serial or residual in Figure 4 (a) and (b)), different forms

Notes	Connection		Form		Init.		ColoredMNIST			NICO		
	serial	residual	matrix	channel	random	eye	OOD	IID	**HM**	OOD	IID	**HM**
VREx [26]	✗	✗	✗	✗	✗	✗	52.9 ± 1.2	14.6 ± 0.3	22.9	76.9 ± 0.7	88.5 ± 1.1	82.3
+ MAP	✓	✗	✓	✗	✓	✗	50.3 ± 1.2	52.6 ± 1.5	51.4	75.6 ± 2.1	88.4 ± 1.6	81.5
	✓	✗	✓	✗	✗	✓	49.7 ± 2.2	60.7 ± 1.3	54.7	76.5 ± 1.7	89.0 ± 1.5	82.3
	✓	✗	✗	✓	✓	✗	51.8 ± 1.7	40.3 ± 2.3	45.3	77.1 ± 0.1	88.2 ± 1.7	82.3
	✓	✗	✗	✓	✗	✓	50.0 ± 1.6	51.2 ± 1.3	50.6	77.4 ± 1.6	88.6 ± 0.5	82.6
	✗	✓	✓	✗	✓	✗	52.6 ± 0.9	71.5 ± 0.5	**60.6**	77.6 ± 0.8	89.1 ± 1.2	**83.0**
	✗	✓	✓	✗	✗	✓	43.1 ± 1.1	68.6 ± 2.9	53.0	75.9 ± 1.7	89.4 ± 0.9	82.1
	✗	✓	✗	✓	✓	✗	46.3 ± 2.7	68.8 ± 2.5	55.4	73.8 ± 1.1	88.9 ± 2.0	80.6
	✗	✓	✗	✓	✗	✓	47.5 ± 1.6	65.5 ± 2.7	55.1	75.1 ± 1.4	89.2 ± 0.7	81.5

Table 3. Experimental results using different forms of the adapter on ColoredMNIST and NICO. The Method in gray denotes the baseline model. The specific details of connection and form are shown in Figure 4. Init. represents the initialization form of our adapter parameters.

Methods	Major shifts → Minor shifts				
	0.1	0.3	0.5	0.7	0.9
ERM [51]	29.7	45.5 ↑	60.6 ↑	85.5 ↑	90.0 ↑
IRM [4]	60.3	53.8 ↓	46.2 ↓	41.7 ↓	33.5 ↓
VREx [26]	52.9	49.6 ↓	34.4 ↓	22.8 ↓	18.7 ↓
GroupDRO [44]	38.5	45.9 ↓	48.6 ↓	50.1 ↓	50.9 ↓
MLDG [27]	29.4	40.0 ↓	50.9 ↓	55.0 ↓	52.7 ↓
MMD [28]	50.6	50.3 ↓	56.0 ↑	53.5 ↓	49.8 ↓
IGA [25]	50.5	45.4 ↓	36.5 ↓	30.0 ↓	24.1 ↓
SANDMask [45]	58.6	53.2 ↓	50.7 ↓	46.5 ↓	42.6 ↓
CDANN [30]	41.7	35.5 ↓	29.4 ↓	27.6 ↓	23.1 ↓
TRM [54]	44.2	42.3 ↓	45.7 ↑	42.2 ↓	31.9 ↓
IB_IRM [2]	53.8	53.2 ↓	48.6 ↓	41.8 ↓	38.1 ↓
CondCAD [42]	49.2	47.1 ↓	36.1 ↓	31.7 ↓	20.9 ↓
MAP (ours)	52.6	54.4 ↑	62.9 ↑	71.1 ↑	80.5 ↑

Table 4. Various distribution shifts are constructed on ColoredMNIST to simulate real-world data. These ratios (*e.g.*, 0.1) represent the proportion between red and green samples in class 0 on testing data (more details in supplementary). ↑ (or ↓) is the increase (or decrease) in performance compared with the previous value.

(*i.e.*, matrix or channel in Figure 4 (c) and (d)) and different initializations (*i.e.*, random or eye). In all settings, a combination of residual, matrix and random has the best performance and this combination is also used in other experimental settings. Although different combinations bring different performances, in most cases, MAP is far superior to the baseline (close to **40%** improvement), especially in ColoredMNIST with strong spurious correlations. We will further study the best combination form in future work.

Could MAP perform well under different distribution shifts? In Table 4, we construct various distribution shifts, *i.e.*, from major shifts to minor shifts, to simulate uncertain real-world data. The performance of most OOD methods degrades as the shifts get smaller or closer to IID data, which demonstrates that these OOD methods extract invariant features while possibly losing some information that contributes to IID generalization. On the contrary, our MAP has good performance under different distribution shifts, which shows that MAP can learn the knowledge lost by OOD methods. More results are in the supplementary.

Is the bilevel optimization effective? We analyze the training strategy of first-order, second-order bilevel optimization (BLO) and without BLO (we optimize ω simultaneously with α on training data without validation and two levels).

Featurizer	without MAP	with MAP
Conv4	0.35M	0.39M+0.04M
ResNet8	4.67M	5.17M+0.5M
ResNet18	10.76M	11.82M+1.06M

Table 5. Comparison of model parameters with or without MAP.

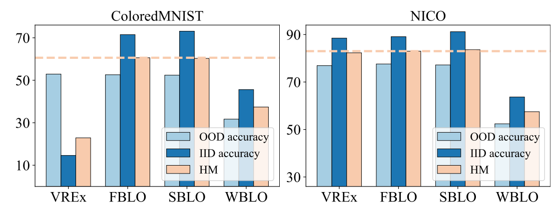

Figure 7. Experiments using different optimization algorithms. VREx, FBLO, SBLO and WBLO denote vanilla VREx, first-order BLO, second-order BLO and without BLO, respectively.

Figure 7 shows results on the ColoredMNIST and NICO datasets. We can observe that the performance of first-order and second-order BLO is similar and better than the vanilla VREx, but WBLO has poor performance in NICO. This demonstrates that the BLO can significantly improve generalization capability, and the first-order approximate is sufficient as the second order with a good performance.

Model parameters. In Table 5, we compared model parameters with or without MAP based on three featurizers. The form of adapters is residual and matrix, because this combination brings the largest parameters compared with other forms in Table 2. A few model parameters can bring significant performance, especially in Conv4, which shows the practical availability of the proposed MAP in applications.

6. Conclusion

In this paper, we investigate a problem of the IID-OOD dilemma and propose an effective **M**odel **A**gnostic ada**P**ters (MAP) method to achieve the balanced generalization performance between IID and OOD evaluations for uncertain real-world distribution shifts. Specifically, the proposed MAP method inserts auxiliary adapter layers (AALs) in the OOD model to simultaneously learn the inductive bias of IID and OOD data. To achieve this, a bilevel optimization (BLO) is used, which optimizes the OOD model in the inner level using an OOD loss and updates AALs in the outer level with an ERM loss based on the optimized OOD model in the inner level. Extensive experiments on six datasets demonstrate that our MAP is able to balance both IID and OOD performance compared with sixteen IID and OOD methods. In future work, we may extend our method to other tasks, *e.g.*, adversarial attacks, to improve the trade-off ability of clean accuracy and adversarial robustness.

Acknowledgement

This work was supported in part by National Key Research and Development Program of China (2022YFC3340900), National Natural Science Foundation of China (62006207, U20A20387), Young Elite Scientists Sponsorship Program by CAST (2021QNRC001), Zhejiang Province Natural Science Foundation (No. LQ21F020020), the StarryNight Science Fund of Zhejiang University Shanghai Institute for Advanced Study (SN-ZJU-SIAS-0010), and the Fundamental Research Funds for the Central Universities (226-2022-00142, 226-2022-00051).

References

[1] Faruk Ahmed, Yoshua Bengio, Harm van Seijen, and Aaron Courville. Systematic generalisation with group invariant predictions. In *International Conference on Learning Representations, ICLR*, 2020.

[2] Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Jean-Christophe Gagnon-Audet, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. Invariance principle meets information bottleneck for out-of-distribution generalization. *Advances in Neural Information Processing Systems, NeurIPS*, 34:3438–3450, 2021.

[3] Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand. Domain-adversarial neural networks. *arXiv preprint arXiv:1412.4446*, 2014.

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

[5] Gilles Blanchard, Aniket Anand Deshmukh, Ürun Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *The Journal of Machine Learning Research JMLR*, 22(1):46–100, 2021.

[6] Manh-Ha Bui, Toan Tran, Anh Tran, and Dinh Phung. Exploiting domain-specific features to enhance domain generalization. *Advances in Neural Information Processing Systems, NeurIPS*, 34:21189–21201, 2021.

[7] Zhengyu Chen and Donglin Wang. Multi-initialization meta-learning with domain adaptation. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1390–1394. IEEE, 2021.

[8] Zhengyu Chen, Teng Xiao, and Kun Kuang. Ba-gnn: On learning bias-aware graph neural network. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 3012–3024. IEEE, 2022.

[9] Mathieu Chevalley, Charlotte Bunne, Andreas Krause, and Stefan Bauer. Invariant causal mechanisms through distribution matching. *arXiv preprint arXiv:2206.11646*, 2022.

[10] Xiang Deng and Zhongfei Mark Zhang. Is the meta-learning idea able to improve the generalization of deep neural networks on the standard supervised learning? In *2020 25th International Conference on Pattern Recognition ICPR*, pages 150–157. IEEE, 2021.

[11] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning, ICML*, pages 1126–1135. PMLR, 2017.

[12] Stephen Gould, Basura Fernando, Anoop Cherian, Peter Anderson, Rodrigo Santa Cruz, and Edison Guo. On differentiating parameterized argmin and argmax problems with application to bi-level optimization. *arXiv preprint arXiv:1607.05447*, 2016.

[13] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020.

[14] Haitao He, Haoran Niu, Jianzhou Feng, Qian Wang, and Qikai Wei. A prototype network enhanced relation semantic representation for few-shot relation extraction. *Human-Centric Intelligent Systems*, 3(1):1–12, 2023.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision, ICCV*, pages 1026–1034, 2015.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence, TPAMI*, 37(9):1904–1916, 2015.

[17] 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, CVPR*, pages 770–778, 2016.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14, ECCV*, pages 630–645. Springer, 2016.

[19] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognition*, 110:107383, 2021.

[20] Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A two-timescale framework for bilevel optimization: Complexity analysis and application to actor-critic. *arXiv preprint arXiv:2007.05170*, 2020.

[21] Md Tariqul Islam, Khan Md Hasib, Md Mahbubur Rahman, Abdur Nur Tusher, Mohammad Shafiul Alam, and Md Rafiqul Islam. Convolutional auto-encoder and independent component analysis based automatic place recognition for moving robot in invariant season condition. *Human-Centric Intelligent Systems*, 3(1):13–24, 2023.

[22] Yinjie Jiang, Zhengyu Chen, Kun Kuang, Luotian Yuan, Xinhai Ye, Zhihua Wang, Fei Wu, and Ying Wei. The role

of deconfounding in meta-learning. In *International Conference on Machine Learning*, pages 10161–10176. PMLR, 2022.

[23] Daehee Kim, Youngjun Yoo, Seunghyun Park, Jinkyu Kim, and Jaekoo Lee. Selfreg: Self-supervised contrastive regularization for domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision ICCV*, pages 9619–9628, 2021.

[24] 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, ICML*, pages 5637–5664. PMLR, 2021.

[25] Masanori Koyama and Shoichiro Yamaguchi. When is invariance useful in an out-of-distribution generalization problem? *arXiv preprint arXiv:2008.01883*, 2020.

[26] 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, ICML*, pages 5815–5826. PMLR, 2021.

[27] 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, AAAI*, 2018.

[28] 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, CVPR*, pages 5400–5409, 2018.

[29] Wei-Hong Li, Xialei Liu, and Hakan Bilen. Cross-domain few-shot learning with task-specific adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 7161–7170, 2022.

[30] 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.

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

[32] Yong Lin, Hanze Dong, Hao Wang, and Tong Zhang. Bayesian invariant risk minimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16021–16030, 2022.

[33] Yong Lin, Shengyu Zhu, Lu Tan, and Peng Cui. Zin: When and how to learn invariance without environment partition? *Advances in Neural Information Processing Systems*, 35:24529–24542, 2022.

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

[35] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision, ICCV*, pages 3730–3738, 2015.

[36] Yulei Niu and Hanwang Zhang. Introspective distillation for robust question answering. *Advances in Neural Information Processing Systems, NeurIPS*, 34:16292–16304, 2021.

[37] Giambattista Parascandolo, Alexander Neitz, Antonio Orvieto, Luigi Gresele, and Bernhard Schölkopf. Learning explanations that are hard to vary. *arXiv preprint arXiv:2009.00329*, 2020.

[38] Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems NeurIPS*, 34:1256–1272, 2021.

[39] Fengchun Qiao and Xi Peng. Uncertainty-guided model generalization to unseen domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, CVPR*, pages 6790–6800, 2021.

[40] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 12556–12565, 2020.

[41] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems, NeurIPS*, 28, 2015.

[42] Yangjun Ruan, Yann Dubois, and Chris J Maddison. Optimal representations for covariate shift. *arXiv preprint arXiv:2201.00057*, 2021.

[43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision IJCV*, 115(3):211–252, 2015.

[44] Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning, ICML*, pages 8346–8356. PMLR, 2020.

[45] Soroosh Shahtalebi, Jean-Christophe Gagnon-Audet, Touraj Laleh, Mojtaba Faramarzi, Kartik Ahuja, and Irina Rish. Sand-mask: An enhanced gradient masking strategy for the discovery of invariances in domain generalization. *arXiv preprint arXiv:2106.02266*, 2021.

[46] 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.

[47] Yuge Shi, Jeffrey Seely, Philip HS Torr, N Siddharth, Awni Hannun, Nicolas Usunier, and Gabriel Synnaeve. Gradient matching for domain generalization. *arXiv preprint arXiv:2104.09937*, 2021.

[48] Yang Shu, Zhangjie Cao, Chenyu Wang, Jianmin Wang, and Mingsheng Long. Open domain generalization with domain-augmented meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 9624–9633, 2021.

[49] Sebastian Thrun and Lorien Pratt. *Learning to learn*. Springer Science & Business Media, 2012.

[50] Yunze Tong, Junkun Yuan, Min Zhang, Didi Zhu, Keli Zhang, Fei Wu, and Kun Kuang. Quantitatively measuring and contrastively exploring heterogeneity for domain generalization. *arXiv preprint arXiv:2305.15889*, 2023.

[51] Vladimir Vapnik and Vlamimir Vapnik. Statistical learning theory wiley. *New York*, 1(624):2, 1998.

[52] Riccardo Volpi, Diane Larlus, and Grégory Rogez. Continual adaptation of visual representations via domain randomization and meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 4443–4453, 2021.

[53] Haotao Wang, Aston Zhang, Shuai Zheng, Xingjian Shi, Mu Li, and Zhangyang Wang. Removing batch normalization boosts adversarial training. In *International Conference on Machine Learning, ICML*, pages 23433–23445. PMLR, 2022.

[54] Yilun Xu and Tommi Jaakkola. Learning representations that support robust transfer of predictors. *arXiv preprint arXiv:2110.09940*, 2021.

[55] Shen Yan, Huan Song, Nanxiang Li, Lincan Zou, and Liu Ren. Improve unsupervised domain adaptation with mixup training. *arXiv preprint arXiv:2001.00677*, 2020.

[56] Nanyang Ye, Kaican Li, Haoyue Bai, Runpeng Yu, Lanqing Hong, Fengwei Zhou, Zhenguo Li, and Jun Zhu. Ood-bench: Quantifying and understanding two dimensions of out-of-distribution generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 7947–7958, 2022.

[57] Guojun Zhang, Han Zhao, Yaoliang Yu, and Pascal Poupart. Quantifying and improving transferability in domain generalization. *Advances in Neural Information Processing Systems NeurIPS*, 34:10957–10970, 2021.

[58] Min Zhang, Siteng Huang, Wenbin Li, and Donglin Wang. Tree structure-aware few-shot image classification via hierarchical aggregation. In *European Conference on Computer Vision*, pages 453–470. Springer, 2022.

[59] Min Zhang, Siteng Huang, and Donglin Wang. Domain generalized few-shot image classification via meta regularization network. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3748–3752. IEEE, 2022.

[60] Marvin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: Learning to adapt to domain shift. *Advances in Neural Information Processing Systems NeurIPS*, 34:23664–23678, 2021.

[61] Min Zhang, Donglin Wang, and Sibo Gai. Knowledge distillation for model-agnostic meta-learning. In *ECAI 2020*, pages 1355–1362. IOS Press, 2020.

[62] Min Zhang, Zifeng Zhuang, Zhitao Wang, Donglin Wang, and Wenbin Li. Rotogbml: Towards out-of-distribution generalization for gradient-based meta-learning. *arXiv preprint arXiv:2303.06679*, 2023.

[63] Didi Zhu, Yinchuan Li, Yunfeng Shao, Jianye Hao, Fei Wu, Kun Kuang, Jun Xiao, and Chao Wu. Generalized universal domain adaptation with generative flow networks. *arXiv preprint arXiv:2305.04466*, 2023.

[64] Didi Zhu, Yincuan Li, Junkun Yuan, Zexi Li, Yunfeng Shao, Kun Kuang, and Chao Wu. Universal domain adaptation via compressive attention matching. *arXiv preprint arXiv:2304.11862*, 2023.