DOMAIN GENERALIZATION IN REGRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

In the context of classification, *domain generalization* (DG) aims to predict the labels of unseen target-domain data only using labeled source-domain data, where the source and target domains usually share *the same label set*. However, in the context of regression, DG is not well studied in the literature, and the main reason is that the ranges of response variable in both domains are often *different*, even disjoint under some extreme conditions. In this paper, we study a new setting: *domain generalization in regression* (DGR), and propose a weighted meta-learning strategy to get optimal meta-initialization across disjoint domains to help address the DGR problem. Different from classification, the labels (responding values) in regression naturally have ordinal relatedness. The core problem in meta learning for regression is that the hard meta-tasks with less ordinal relatedness are less sampled from training domains. To pay attentions to the hard meta-tasks, we adopt the feature discrepancy in meta-space to calculate the discrepancy between any two domains and treat the discrepancy as the importance of meta-tasks in the meta-learning framework. The extensive regression experiments on standard benchmark DomainBed demonstrate the superiority of the proposed method.

1 INTRODUCTION

Being capable of out-of-distribution data is an important measure to see if a machine learning system is reliable in the real world. There are various related explorations in the field of machine learning, e.g., domain generalization/adaptation (Zhou et al., 2022; Wang et al., 2022a; Zhao et al., 2020), few/zero-shot learning (Wang et al., 2020b) and out-of-distribution detection (Yang et al., 2021a). Among them, *domain generalization* (DG) receives increasing attentions by the fascinating setting: learning models on source domains and making predictions on unseen but related target domains. However, most of them focus on classifications, which limits their practical applications.

For example, we often predict the recovery/survival time of patients in clinic or estimate the ages/skeleton joints/gaze direction of humans (Jiang et al., 2021; Wang et al., 2022b). These real-world tasks can be grouped into regression problems. Like classification, the distribution (domain) shifts in regression also have multiple patterns: (1) Marginal distribution shift of input data, e.g., Chen et al. (2021) try to transfer knowledge from the male gaze images to the female gaze; (2) Marginal distribution shift of labels (responding values), e.g., Yang et al. (2021b) learn a model on imbalanced regression setting and generalize to balanced scenarios; (3) Joint distribution shift of inputs and labels, e.g., Teshima et al. (2020) assume that the shifts exist in both the input and label distributions, but the causal mechanism between the input and the labels are not changed across domains.

Among these distribution-shift patterns, the label's marginal shift in regression is very different compared to classification. In classification, the shift usually denotes different class probability densities among domains (Liu et al., 2021b). In regression, the label shift can also have similar form, e.g., imbalanced domain regression. Meanwhile, it can further have a particular form, i.e., *interval shift*. For example, when the responding interval of the source domain is $[0.4, 0.5]$, the shifted responding interval of the target domain can be $[0.5, 0.6]$. The shift often arises in practical regression settings that need moderate extrapolation and interpolation. In some cases, this regression interval between the target and the source labels may have no overlap. We call this regression setting as *domain generalization in regression* (DGR). Fig. 1 denotes the differences among the traditional domain generalization, the imbalanced domain regression and the DGR.

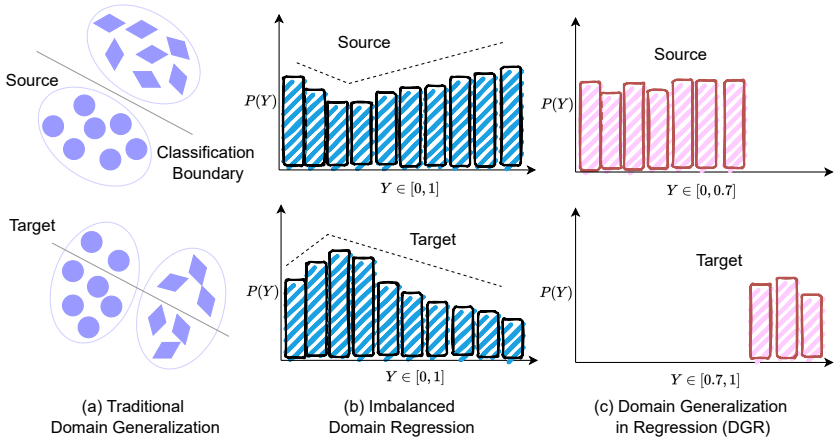


Figure 1: The illustration of three transfer settings. (a) In the traditional domain generalization, a source-trained classifier is directly applied to the target domain. (b) In the imbalanced domain regression, the responding values $Y \in [0, 1]$ have different probability densities among domains. (c) The DGR problem focuses on predicting unseen responding values in the target domain.

DGR can be viewed as a special case of the traditional setting of domain generalization, while the former emphasizes label shift. However, existing domain generalization methods can not be directly applied to the DGR problem. For example, feature alignment is the core idea of many domain generalization methods (Li et al., 2018b; 2020), but it is not necessary and even harmful in our DGR setting. Under the assumption that closer feature discrepancy means closer predictions, accurate feature alignment methods may let the model map all predictions exclusively into source interval or target interval, which can not reduce total regression risks. Furthermore, due to the existence of possible contradictory domains, simultaneously optimizing models on multiple source domains may suffer negative transfer (Wang et al., 2019).

Meta-learning algorithms, e.g., *model agnostic meta-learning* (MAML, Finn et al. (2017)) have been applied to traditional domain generalization (Li et al., 2018a; Dou et al., 2019; Du et al., 2021) and might be a direct solution to alleviate the above problem. In each meta task, these methods usually sample a support and a query classification task from two different domains and optimize the meta-model by a bi-level paradigm. Yet in fact, this paradigm is not enough for DGR. Because their task sampling strategy usually follows an implicit assumption, i.e., the meta-tasks have equal importance during training (Yao et al., 2021; Gao et al., 2022). We argue that the implicit assumption is not satisfied anymore in our regression setting.

Unlike classification, each pair of labels have ordinal relations in regression (Liu et al., 2021c). If we denote the regression margin as the label discrepancy between the support and query domain, *the meta-tasks that have a larger regression margin are less sampled compared to the meta-tasks with a smaller margin.* On the other side, the meta-task with a larger regression margin is usually harder to be optimized in meta-learning. *These key factors bring a sampling bias that harder meta-tasks are less sampled from training data.* As a result, the meta-model tends to put more attention to the easier meta-tasks, which means the limited exploration and interpolation of the meta-model. To alleviate this sampling bias, we use a simple but effective strategy, i.e., assigning more weights to harder meta-tasks. The weights can be simply computed with the feature discrepancy in meta-parameter space between the query and support examples of meta-task.

Finally, we build a standard generalization benchmark to evaluate the DGR problem. Except for toy causal experiments, we reformulate two real-world age estimation datasets. For example, the source domain data are 20 to 40 year-old face images of celebrities, the target domain data are 40 to 50 year-old face images corresponding to the celebrities. These images are sampled in different years and some works have a similar setting, e.g., the evolving or continuous domain dataset (Wang et al., 2020a; Liu et al., 2020). To summarize, our method can be named as *margin-aware meta regression* (MAMR), and the main contributions are as follows:

- We study a new domain generalization setting in regression scenario, which has practical significance and has not been well studied before.

- To implement better exploration and interpolation in DGR, we propose a margin-aware meta-learning framework to alleviate sampling bias and encourage the model to notice long-range ordinal relations.
- Although our solution achieves considerable improvements regarding baselines, our empirical analyses demonstrate that generalizing to unseen response values may be still very challenging.

2 Related Work

2.1 Domain Adaptation for Regression

Domain adaptation aims to migrate the knowledge from a source domain to a target domain, where there might be data distribution shift between the source and the target. Typical domain adaptation methods try to get confident decision boundaries for classification tasks based on clustering assumption (Liang et al., 2020). When dealing with cross-domain regression, this assumption is not satisfied, which makes it challenging for nowadays domain adaptation methods. Some pioneer works like (Cortes & Mohri, 2011) try to provide regression discrepancy in reproducing kernel Hilbert space. Most recent works try to address cross-domain regression in specific application scenarios, such as estimating object boxes in cross-domain/few-shot object detection (Zheng et al., 2020; Gao et al., 2022), regressing human skeleton key-points in cross-domain gesture estimation (Jiang et al., 2021) and calculating the gaze direction in cross-domain gaze tracing (Bao et al., 2022). Furthermore, Chen et al. (2021) propose a general cross-domain regression method via subspace alignment, which reduces domain gap by minimizing *representation subspace distance* (RSD) with the principal angles of representation matrices. Xia et al. (2022) propose an adversarial dual regressor to achieve direct alignment.

However, nearly all cross-domain regression methods inherently assume there only exists covariate shift in input examples, i.e., $p(x_s) \neq p(x_t)$, where $p(\cdot)$ is the probability density function and x_s, x_t denote the source and target examples. This means that they might be incapable of the label shift across domains. The label shift in cross-domain regression can arise as interval shift of responding values, e.g., the source interval $y_s \in [0.3, 0.5]$ while the target interval $y_t \in [0.6, 0.7]$. The responding values in the real world can be gasoline consumption data and vary significantly across developed and developing countries (Teshima et al., 2020). Chidlovskii et al. (2021) also consider the interval shift problem and tries to learn a ranking on the target domain, followed by mapping the ranking to responding values. This method assumes the availability of the responding interval on the target domain at the adaptation stage, which might be contradictory to the setting of unavailable labels. In contrast, we assume all target domain data are not available at the training stage, which is more practical and challenging in real-world scenarios.

2.2 Domain Generalization

In domain adaptation paradigm, both the labeled source data and the unlabeled target data are usually available at the training stage. Domain generalization introduces a more challenging setting where the model can only access the labeled source data at the training stage. We recommend the readers refer to the two related surveys (Zhou et al., 2022; Wang et al., 2022a) for more details. Nearly all the domain generalization methods focus on classification tasks, which heavily limits the application of this setting. When the interval shift happens in regression, a well-trained regressor from the source domain might collapse on the target domain.

Among existing domain generalization methods, the meta-learning paradigm might be potential for this interval shift problem. The spearhead work MLDG (Li et al., 2018a) introduces MAML (Finn et al., 2017) into domain generalization framework. Different from MLDG, we focus on the query task and do not simultaneously optimize the support and the query task. Dou et al. (2019) leverage class relationships and local sample clustering to capture the semantic features of different classes. These two operations are hard to be migrated to regression settings because the clustering assumption is usually not reasonable for regression. Moreover, in many regression tasks like age regression, the semantic features might be unimportant, e.g., distinguishing each face might be useless for age regression. Instead, the style features, like the texture of the faces might be important information for age regression. Another meta-learning method Meta-Norm (Du et al., 2021) stabilizes the batch

statistics in Batch Normalization Layers with MAML for domain generalization. Meta-Norm might be orthogonal with other meta-learning strategies and ours.

3 PRELIMINARIES AND NOTATIONS

In this section, we introduce the formal definition of the DGR problem. We denote the input space and the label space by $\mathcal{X}$ and $\mathcal{Y}$, where $\mathcal{Y}$ has a continuous range from 0 to 1 and can be further divided into two disjoint sub-spaces, e.g., $\mathcal{Y}_{source}$ and $\mathcal{Y}_{target}$, where $\mathcal{Y}_{source} \cap \mathcal{Y}_{target} = \emptyset$. $D_s = \{(\mathbf{x}, \mathbf{y}) \in \{\mathcal{X} \times \mathcal{Y}_{source}\}\}$ and $D_t = \{(\mathbf{x}, \mathbf{y}) \in \{\mathcal{X} \times \mathcal{Y}_{target}\}\}$ respectively denote the source and target domain data. The model can only utilize D_s at the training stage, and then predicts D_t without further adaptation. The above settings are very similar to the classification tasks of domain generalization. But the label space of different domains is disjoint in our regression setting. A prediction $\hat{y}$ by regression model R can be denoted with $\hat{y} = R(x) = G(F(x))$. We use $F : \mathcal{X} \to \mathcal{Z}$ to denote a feature encoder, where $\mathcal{Z}$ is feature space. After the encoder, we use a linear regressor with sigmoid activation to map the range of predictions into $[0, 1]$, i.e., $G : \mathcal{Z} \to \mathcal{Y}$.

4 MARGIN-AWARE META REGRESSION

Following the typical setting of domain generalization that domain labels are available. We split D_s into K source domains $\{D_1, D_2, \cdots, D_K\}$ and simulate the generalization setting between D_s and D_t. Note that the label space of any two source domains are also disjoint. In the following, we will provide empirical analyses of previous generalization strategies, e.g., domain alignment, meta-learning, and introduce our regression models. As we know, feature alignment is the core idea of many typical domain alignment solutions for domain adaptation (Ganin et al., 2016) as well as domain generalization (Li et al., 2018b). For domain generalization, the alignment is usually performed among multiple source domains to find domain-invariant semantic features. This alignment can be formalized using a general discrepancy measure, i.e., *integral probability metric* (IPM, Müller (1997)). Let X_1, X_2 denote two independent random variables from domain distributions $\mathbb{P}_i$ and $\mathbb{P}_j$. The domain discrepancy can be defined with:

$$\mathrm{IPM}(\mathbb{P}_i, \mathbb{P}_j) := \sup_{f \in \mathcal{H}} [\mathbb{E}[f(\mathbf{X}_1)] - \mathbb{E}[f(\mathbf{X}_2)]],$$

where $\mathbb{E}$ denotes the expectation, f denotes the transformation function in function space $\mathcal{H}$. Applying specific condition on $\mathcal{H}$, IPM can be transformed into many popular measures, such as *maximum mean discrepancy* (MMD, Liu et al. (2021a)) and *wasserstein distance* (WD, Shen et al. (2018)).

Incorporating the domain discrepancy between $\mathbb{P}_i$ and $\mathbb{P}_j$, the objective of the regressor can be formulated as:

$$\min_{\Theta} \sup_{\substack{(\mathbf{x}_1, \mathbf{y}_1) \in D_i, \\ (\mathbf{x}_2, \mathbf{y}_2) \in D_j}} \left[L_\Theta(\mathbf{x}_1, \mathbf{y}_1) + L_\Theta(\mathbf{x}_2, \mathbf{y}_2) + \widehat{\mathrm{IPM}}(\mathbf{x}_1, \mathbf{x}_2) \right],$$

where Θ is model parameter, $L_\Theta(\mathbf{x}, \mathbf{y}) = ||R_\Theta(\mathbf{x}) - \mathbf{y})||$ is the empirical risk and can be the squared loss, $\widehat{\mathrm{IPM}}$ is the estimator from two batch examples $\mathbf{x}_1$ and $\mathbf{x}_2$. For example, $\widehat{\mathrm{IPM}}$ can be the unbiased U-statistic estimator $\widehat{\mathrm{MMD}}_u^2(\mathbf{x}_1, \mathbf{x}_2)$ (Liu et al., 2021a). In general domain generalization for classification tasks, all terms in the above objective could be minimized. However, our regression setting is more like open domain generalization, which learns a model from the source domain and inferences in unseen target domains with novel classes (Shu et al., 2021). To regress unseen target values, one strategy is to simulate the scenario in the training stage. That means the labels in D_i and D_j have few or no overlaps. Therefore, when the domain discrepancy $\widehat{\mathrm{IPM}}$ is minimized, there might be only one term minimized between $L_\Theta(\mathbf{x}_1, \mathbf{y}_1)$ and $L_\Theta(\mathbf{x}_2, \mathbf{y}_2)$. This problem can be formally introduced with the following definition:

Regression Margin. Let (X_1, Y_1) and (X_2, Y_2) be the random variables of two domains D_i, D_j, the $[a, b]$ and $[c, d]$ be the regression interval of Y_1, Y_2. When $\widehat{\mathrm{IPM}}$ is reduced to 0, we have

$$\inf [\mathbb{E}[f(\mathbf{X}_1) - Y_1] - \mathbb{E}[f(\mathbf{X}_2) - Y_2]] = \inf [(\mathbb{E}[f(X_1) - \mathbb{E}[f(X_2)) + \mathbb{E}[Y_2 - Y_1]] = c - b.$$

The above analysis tells us that a large domain margin $(c-b)$ can lead to a divergent optimization when simultaneously minimizing the domain discrepancy and the empirical risks. To alleviate the above problem, one strategy is to bypass explicit the feature alignment. For example, in the meta-learning paradigm towards domain generalization, one can learn a meta-model by a bi-level optimization. If X_1, X_2 are random variables from source and target domains, in the inner optimization, the model learns on a support (source) domain, i.e., learning $f(X_1)$. In the outer optimization, the learned model tries to generalize to a query (target) domain, i.e, learning $f(X_2|X_1)$. This training strategy naturally avoids explicit feature alignment. Moreover, the bi-level optimization emphasizes the importance of query loss, which might alleviate the above regression margin because the inner function $f(X_1)$ and the outer function $f(X_2|X_1)$ can be viewed as different sampling in parameter space.

Existing meta-learning domain generalization methods are sub-optimal for the DGR problem. In the classification, each meta-task consisting of support tasks and query tasks is assumed to have the same sampling probability. However, the responding intervals of the support and query have ordinal relations in regression. When the regression margin is larger between the support and query tasks, the sampling probability is smaller in the regression problem. Intuitively thinking about the extreme case that when the regression margin is close to $1 = (c-b)$, the corresponding sampling probability of meta-tasks is close to 0. We named this phenomenon *sampling bias*:

Sampling Bias. Given limited training examples, the sampling probability of a larger regression margin meta-task is always smaller than the sampling probability of a small margin meta-task.

A larger regression margin between the support and query tasks usually means a harder meta-task for the meta-learning model. Therefore, without any specialized sampling strategy, the model is prone to be biased on the small margin tasks. To alleviate this problem, we want the large margin meta-task to have a larger weight in the meta-learning process. One direct strategy is to calculate the weight using the domain discrepancy, i.e., a larger regression margin means a larger meta-task weight. The learning objective can be redefined with:

$$\min_{\Theta} \sup_{\substack{(\mathbf{x}_q,\mathbf{y}_q)\in D_i,\\ (\mathbf{x}_s,\mathbf{y}_s)\in D_j}} L_{\Theta'}(\mathbf{x}_s,\mathbf{y}_s)\cdot d(\mathbf{x}_s,\mathbf{x}_q)$$
$$s.t.\ \Theta^{'} = \Theta - \beta\nabla_{\Theta}\left[L_{\Theta}(\mathbf{x}_s,\mathbf{y}_s\right],$$

where D_i, D_j respectively denote the query domain and the support domain, d is discrepancy functions like $\widehat{\mathrm{MMD}}_u^2(\cdot,\cdot)$ or simple Euclidean metric, and β is the inner loop learning rate on the support domain $\{\mathbf{x}_s, \mathbf{y}_s\}$. Compared with MAML, we usually need 1 or 2 optimization steps in the inner loop for the support domain. Unlike meta-learning for few-shot problem, more steps are not necessary for our setting. We want the learned meta-parameter to perform well without any fine turning at the test stage. Hence we want the adapted parameters $\Theta^{'}$ to approach the meta-parameters Θ. The smaller change between $\Theta^{'}$ and Θ brings another benefit for domain discrepancy $d(\cdot,\cdot)$, which can be calculated using the meta-parameter Θ. This is consistent with that we use Θ as the parameters of encoder F and regressor G at the test stage.

The graphic training process of one meta-task can be seen in Fig. 2. Different from existing meta-learning models, our MAMR model considers the domain discrepancy by discrepancy function $d()$, but the data node in $d(\mathbf{x}_s, \mathbf{x}_q)$ does not have gradients. As discussed, the reason is directly minimizing this domain discrepancy might harm the generalization ability of our MAMR model. Furthermore, with Euclidean distance $d()$, we describe the detailed method in Algorithm 1.

5 EXPERIMENTS

In this section, we will empirically explore what MAMR can learn and compare it to related works from the view of performance and methodology.

5.1 BASELINES

We use multiple domain generalization and the variants of domain adaptation methods as baselines, including: (1) risk minimization methods (**ERM** Vapnik. (1999), **IRM** Arjovsky et al. (2019)); (2)

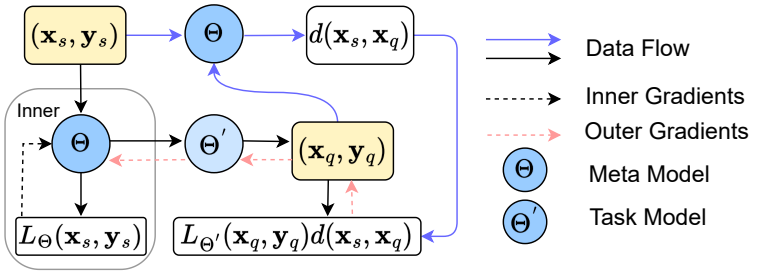


Figure 2: The graphic illustration of our model's training process. Note that the above two meta-models have identical parameters Θ, and the blue data flow does not have gradient backpropagation.

Algorithm 1 Training Algorithm of MAMR

Input: The source domains data D_s, the inner loop learning rate β, the out-loop learning rate α, the domain number K to split D_s.
Parameter: Model parameters Θ.
Output: The learned Θ.

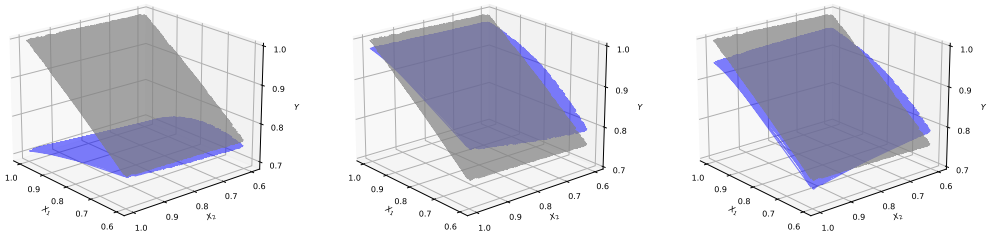
Split the source data D into sub-domains $\{D_0, D_1, \cdots D_K\}$.
while not convergence **do**
  Sample $T = K(K-1)/2$ domain pairs $\{(D_i, D_j)\}$ that $i \neq j$.
  for $index = 0 \rightarrow T$ **do**
    Sample a batch of support data $(\mathbf{x}_s, \mathbf{y}_s) \in D_j$ and query data $(\mathbf{x}_q, \mathbf{y}_q) \in D_i$;
    Compute task discrepancies: $d(\mathbf{x}_s, \mathbf{x}_q) = ||F(\mathbf{x}_s) - F(\mathbf{x}_q)||_2$;
    Get task-specific model parameters: $\Theta^{'} = \Theta - \beta \nabla_\Theta \left[L_\Theta(\mathbf{x}_s, \mathbf{y}_s\right]$;
    Compute the weighted regression error: $L_{\Theta'}(\mathbf{x}_s, \mathbf{y}_s) \cdot d(\mathbf{x}_s, \mathbf{x}_q)$;
    Update Θ: $\Theta = \Theta - \alpha \nabla_\Theta \left[L_{\Theta'}(\mathbf{x}_s, \mathbf{y}_s) \cdot d(\mathbf{x}_s, \mathbf{x}_q)\right]$;
  end for
end while

feature alignments and robust optimization (**MMD** Li et al. (2018b), **DANN** Ganin et al. (2016), **SD** Pezeshki et al. (2021), **Transfer** Zhang et al. (2021)); (3) subspace alignments (**RSD** Chen et al. (2021)); (4) self-supervised methods (**SelfReg** Kim et al. (2021), **CAD** Ruan et al. (2022)) (5) meta-learning (**MLDG** Li et al. (2018a)) and (6) disentangling method (**DDG** Zhang et al. (2022)). All the introductions of baselines can be seen in Appendix A.1. For fairness, we put all the baselines into a public evaluation benchmark DomainBed (Gulrajani & Lopez-Paz, 2021). We uniformly use ResNet12 as the backbone encoder F for all methods. ResNet12 is a popular ResNet (He et al., 2016) backbone in meta-learning. For regressor G, we use a single linear neural network followed by a sigmoid function.

5.2 Toy Example

To figure out what the MAMR model can learn in regression problems, we create a toy dataset in which the input examples and their responding values obey some causal mechanism. We assume the 1-dimensional random variables X_1 and X_2 follow a uniform distribution in [0,1], and the responding values Y are under the control of X_1 and X_2. The control mechanism can be complex as given in Appendix A.4. At training stage, regression models can only using $X_1 \in [0, 0.6]$ and $X_2 \in [0, 0.6]$. At test stage, we record the regression values when given $X_1 \in [0.6, 1]$ and $X_2 \in [0.6, 1]$.

The toy experiments sample 15000 and 10000 regression tasks at the training and test stage, respectively. We use a 4-layers fully connected neural network for ERM, RSD and our MAMR. Fig. 3 provides the test time explorations results of the three methods. On 10000 test tasks, the ground-truth responding values and the predicted values respectively form a gray region and a blue region. When given unseen values of X_1 and X_2, ERM fails to use the causal mechanism. The strong baseline method RSD captures a part of the causal mechanism. MAMR gets the best exploration performance by maximum causal discovery.



(a) ERM (MSE: 0.0219) (b) RSD (MSE:0.008) (c) Ours (MSE:0.0004)

Figure 3: The toy experiments illustrate the ground truth test landscape (gray color) and prediction regions (blue color). Each method's performance is reported with Mean Squared Error (MSE).

5.3 Cross-Domain Age Estimation Datasets

CACD[1]. Cross-Age Celebrity Dataset (CACD) contains 163,446 images from 2,000 celebrities collected from the Internet. The age of celebrities range from 16-62 and can be classified into 5 age intervals (domains), i.e., $[15-20), [20-30), [30-40), [40-50), [50-60]$. The images of each celebrity are sampled using different devices in different years. Therefore each domain has different facial characteristics.

AFAD[2]. The Asian Face Age Dataset (AFAD) originally is an age estimation dataset containing more than 160K face images and aging labels. We split the dataset into 5 age intervals (domains), i.e., $[15-20), [20-25), [25-30), [30-35), [35-40]$. Like CACD, each age interval has its own face characteristics and can be viewed as 5 related domains for regression.

For all datasets, we normalize the labels from 0 to 1 and leave out one domain at the training stage then make predictions on this domain at the test stage. To ensure a similar capacity among different age intervals, we make compensation for the small capacity interval by slightly relaxing the interval.

5.4 Training and Evaluation

Following the data configuration in the DomainBed[3] benchmark, we randomly split each domain into 90% and 10% subsets. The former is used in model training and the latter for model selection. We use two popular model selection methods in DG, i.e., test-domain validation and training domain validation. The former is also named the oracle method that the model is selected based on the 10% data of the test domain. The latter uses the 10% data of the training domain to select the best model. To find the proper hyper-parameters for each algorithm under limited computation resources, 5 times random hyper-parameter searches are conducted. Then we repeat 3 times with different seeds on each group of hyper-parameters. Appendix A.3 provides detailed settings of the hyper-parameters. Including toy experiments, all methods are implemented with Pytorch and can be executed on a NVIDIA RTX 3090 GPU.

5.5 Quantitative Comparisons

Comparison to **risk minimization** methods. ERM and IRM are typical risk minimization methods. From Tab. 1 and Tab. 2, the two methods get poor DGR performance on real-world datasets. We also find that ERM is better than IRM, which might imply that the gradient invariance in IRM is useless for our problem. Another result is that the naive ERM is surprisingly comparable with advanced methods, e.g., MMD, DANN and MTL. Previous works (Gulrajani & Lopez-Paz, 2021) also find a similar phenomenon in classification tasks.

[1] http://bcsiriuschen.github.io/CARC/

[2] https://afad-dataset.github.io/

[3] https://github.com/facebookresearch/DomainBed

Table 1: Regression results on CACD dataset with test-domain validation (Oracle). Each regression interval (domain) in all tables denotes the target interval with the others as source intervals. The minimum Mean Squared Errors with their standard variances are bolded.

Algorithm	[15-20)	[20-30)	[30-40)	[40-50)	[50-60]	Avg
ERM	$0.0247_{\pm 0.01}$	$0.0492_{\pm 0.01}$	$0.0801_{\pm 0.02}$	$0.1806_{\pm 0.02}$	$0.1760_{\pm 0.07}$	0.1021
IRM	$0.0247_{\pm 0.01}$	$0.0493_{\pm 0.01}$	$0.0987_{\pm 0.02}$	$0.1810_{\pm 0.02}$	$0.2366_{\pm 0.03}$	0.1181
MMD	$0.0204_{\pm 0.01}$	$0.0378_{\pm 0.02}$	$0.0896_{\pm 0.02}$	$0.1944_{\pm 0.01}$	$0.2703_{\pm 0.03}$	0.1225
DANN	$0.0326_{\pm 0.00}$	$0.0562_{\pm 0.01}$	$0.0924_{\pm 0.02}$	$0.1521_{\pm 0.04}$	$0.1593_{\pm 0.03}$	0.0985
MTL	$0.0329_{\pm 0.00}$	$0.0641_{\pm 0.00}$	$0.1199_{\pm 0.00}$	$0.2022_{\pm 0.00}$	$0.1426_{\pm 0.05}$	0.1123
SD	$0.0247_{\pm 0.01}$	$0.0493_{\pm 0.01}$	$0.0985_{\pm 0.02}$	$0.1809_{\pm 0.02}$	$0.2696_{\pm 0.03}$	0.1246
CAD	$0.0335_{\pm 0.00}$	$0.0641_{\pm 0.00}$	$0.2095_{\pm 0.07}$	$0.2022_{\pm 0.00}$	$0.2029_{\pm 0.00}$	0.1425
Transfer	$0.0330_{\pm 0.00}$	$0.0641_{\pm 0.00}$	$0.1199_{\pm 0.00}$	$0.2022_{\pm 0.00}$	$0.2366_{\pm 0.03}$	0.1312
MLDG	$0.0249_{\pm 0.01}$	$0.0089_{\pm 0.00}$	$0.0413_{\pm 0.03}$	$0.0092_{\pm 0.00}$	$0.1888_{\pm 0.06}$	0.0546
RSD	$0.0361_{\pm 0.00}$	$0.0099_{\pm 0.00}$	$0.0043_{\pm 0.00}$	$0.0132_{\pm 0.00}$	$0.0529_{\pm 0.00}$	0.0233
SelfReg	$0.0364_{\pm 0.00}$	$0.0114_{\pm 0.00}$	$0.0017_{\pm 0.00}$	$\mathbf{0.0077}_{\pm 0.00}$	$0.0427_{\pm 0.01}$	0.0200
DDG	$0.0324_{\pm 0.01}$	$0.0158_{\pm 0.00}$	$\mathbf{0.0015}_{\pm 0.00}$	$0.0101_{\pm 0.00}$	$0.0254_{\pm 0.01}$	0.0171
MAMR	$\mathbf{0.0099}_{\pm 0.01}$	$\mathbf{0.0028}_{\pm 0.00}$	$0.0019_{\pm 0.00}$	$0.0078_{\pm 0.00}$	$\mathbf{0.0200}_{\pm 0.01}$	**0.0085**

Table 2: Regression results on AFAD dataset with test-domain validation (Oracle).

Algorithm	[15-20)	[20-25)	[25-30)	[30-35)	[35-40]	Avg
ERM	$0.2247_{\pm 0.07}$	$0.2658_{\pm 0.02}$	$0.1334_{\pm 0.05}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.1602
IRM	$0.1413_{\pm 0.00}$	$0.2114_{\pm 0.00}$	$0.2356_{\pm 0.03}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.1530
MMD	$0.0728_{\pm 0.03}$	$0.0461_{\pm 0.03}$	$0.0678_{\pm 0.05}$	$0.0417_{\pm 0.03}$	$\mathbf{0.0332}_{\pm 0.01}$	0.0523
DANN	$0.1101_{\pm 0.02}$	$0.1575_{\pm 0.04}$	$0.1475_{\pm 0.03}$	$0.1052_{\pm 0.01}$	$0.0519_{\pm 0.01}$	0.1144
MTL	$0.2092_{\pm 0.06}$	$0.2114_{\pm 0.00}$	$0.1980_{\pm 0.00}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.1591
SD	$0.0174_{\pm 0.01}$	$0.0050_{\pm 0.00}$	$0.0031_{\pm 0.00}$	$0.0156_{\pm 0.00}$	$0.0540_{\pm 0.00}$	0.0190
CAD	$0.1413_{\pm 0.00}$	$0.2114_{\pm 0.00}$	$0.1990_{\pm 0.00}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.1457
Transfer	$0.1413_{\pm 0.00}$	$0.2114_{\pm 0.00}$	$0.1990_{\pm 0.00}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.1457
MLDG	$0.1558_{\pm 0.10}$	$0.0095_{\pm 0.00}$	$0.0023_{\pm 0.00}$	$\mathbf{0.0091}_{\pm 0.00}$	$0.0498_{\pm 0.00}$	0.0453
SelfReg	$0.0452_{\pm 0.00}$	$0.0139_{\pm 0.00}$	$0.0018_{\pm 0.00}$	$0.0108_{\pm 0.00}$	$0.0417_{\pm 0.00}$	0.0227
RSD	$0.0415_{\pm 0.00}$	$0.0179_{\pm 0.00}$	$0.0042_{\pm 0.00}$	$0.0101_{\pm 0.00}$	$0.0392_{\pm 0.00}$	0.0226
DDG	$0.0304_{\pm 0.01}$	$0.0118_{\pm 0.00}$	$\mathbf{0.0011}_{\pm 0.00}$	$0.0125_{\pm 0.00}$	$0.0495_{\pm 0.01}$	0.0211
MAMR	$\mathbf{0.0041}_{\pm 0.00}$	$\mathbf{0.0013}_{\pm 0.00}$	$0.0012_{\pm 0.00}$	$0.0189_{\pm 0.00}$	$0.0640_{\pm 0.00}$	**0.0179**

Comparison to the methods using **feature alignments and robust optimization**. As discussed in Sec. 4, directly using feature alignments, e.g., MMD, DANN, may perform poorly due to the regression margin. Furthermore, DANN and Transfer try to apply adversarial robustness to our problem. Our results in Tab. 1 and Tab. 2 also demonstrate they might bring opposite impact compared to ERM.

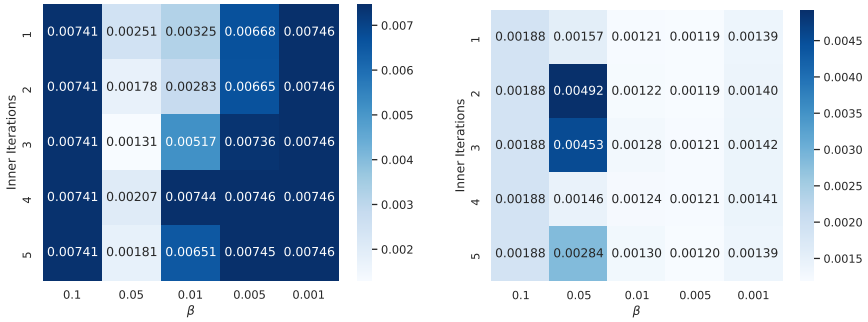
Comparison to **subspace alignments**, e.g., RSD. We find that RSD surpasses the feature alignment methods with a large margin. With principal angle alignment between sub-spaces, the sub-space alignments effectively slack the traditional feature alignments. This might imply that the domain adaptation method RSD can also generalize well to out-of-distribution data.

Comparison to **self-supervised methods**, e.g., SelfReg and CAD. The self-supervised methods, especially with contrastive learning, can be strong baselines for our problem. The reason might be that SelfReg uses strong data augmentation with Mixup in their models. We find the follow-up work CAD does not surpass SelfReg. The reason might be that the part of marginal distribution alignment in CAD harms the generalization ability like DANN.

Comparison to **meta-learning method**. MLDG simultaneously optimizes the support risks and query risks. While in DGR, the support and the query tasks usually change a lot, which makes the MLDG hard to be optimized. Our method does not simultaneously optimize the above two risks

Table 3: Ablation studies on CACD dataset with test-domain validation (Oracle).

Methods	**[15-20)**	**[20-30)**	**[30-40)**	**[40-50)**	**[50-60]**	**Avg**
MAMR-	$0.0159_{\pm 0.01}$	$0.0175_{\pm 0.00}$	$0.0021_{\pm 0.00}$	$0.0623_{\pm 0.03}$	$0.0397_{\pm 0.00}$	0.0275
MAMR-G	$0.0354_{\pm 0.01}$	$0.0440_{\pm 0.03}$	$0.0248_{\pm 0.02}$	$0.0431_{\pm 0.02}$	$0.0595_{\pm 0.05}$	0.0414
MAMR-P	$\mathbf{0.0099}_{\pm 0.01}$	$\mathbf{0.0028}_{\pm 0.00}$	$\mathbf{0.0019}_{\pm 0.00}$	$\mathbf{0.0078}_{\pm 0.00}$	$\mathbf{0.0200}_{\pm 0.01}$	**0.0085**



(a) [20,30) (b) [30,40)

Figure 4: The MSE heatmaps of regression tasks $[20, 30)$ and $[30, 40)$ in CACD by Oracle selection.

and is attentive to hard tasks. The experiments in Tab. 1 and Tab. 2 demonstrate that our method outperforms MLDG with a large margin.

Comparison to **disentangling**. DDG disentangles the latent representations into semantic features and variation features. Our experiments find that DDG may successfully capture the causal mechanism between the inputs and their responding values.

5.6 Detailed Analyses

Tab. 3 provides 3 ablation models. MAMR- is our method without the margin-aware weighting mechanism. MAMR-G computes a mean weight for query tasks using the MMD with Gaussian kernel. MAMR-P computes the pair-wised Euclidean distances among the support and query tasks and provides a weight for each query task. The results demonstrate the mean weight in MAMR-G may be invalid compared to pair-wised weights.

The key hyper-parameters of the MAMR model include the inner loop learning rate β, the outer loop learning rate α and the iteration steps of the inner loop. To reduce the searching of hyper-parameters, we set $\alpha = 0.1 * \beta$. We conduct a grid search for β and the iteration steps. Fig. 4 provides the MSE heatmaps on the CACD dataset using two generalization tasks. We find that more inner iteration steps do not have a significant influence on the generalization results. This phenomenon is consistent with our analysis in the method: different from 5 or 10 inner steps in meta-learning for few-shot learning, fast adaptation by multi-steps is not necessary for DGR.

6 Conclusion

We study a new problem setting named domain generalization in regression. A margin-aware meta-learning regression method is proposed to achieve long-range exploration and interpolation from the source domain. We build a regression benchmark to systematically investigate the existing domain generalization methods. Our empirical analyses demonstrate that domain generalization for regression still has a large exploration space. We hope more advanced methods in other fields can be introduced, such as imbalanced regression and open set domain generalization.

7 REPRODUCIBILITY STATEMENT

We make the following efforts to improve reproducibility:

- **Algorithms.** Algorithm 1 provides detailed implementation for our MAMR model.
- **Datasets.** The used public dataset can be downloaded from here[4] [5].
- **Evaluations.** we put our method and baselines into a popular benchmark DomainBed[6], to make fair comparisons.
- **Ablation studies.** Tab. 3 provides ablation results of MAMR.
- **Hyper-parameters settings.** Tab. 6 provides the used hyper-parameters in MAMR.
- **Hyper-parameters searching.** Fig. 4 provides the samples of hyper-parameters searching.
- **Codes.** The codes can be downloaded from anonymous link [7].

[4] http://bcsiriuschen.github.io/CARC/
[5] https://afad-dataset.github.io/
[6] https://github.com/facebookresearch/DomainBed
[7] https://anonymous.4open.science/r/MAMR-276E

REFERENCES

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2019. URL https://arxiv.org/abs/1907.02893.

Yiwei Bao, Yunfei Liu, Haofei Wang, and Feng Lu. Generalizing gaze estimation with rotation consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4207–4216, June 2022.

Gilles Blanchard, Aniket Anand Deshmukh, Ürun Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *J. Mach. Learn. Res.*, 22(1), jan 2021. ISSN 1532-4435.

Xinyang Chen, Sinan Wang, Jianmin Wang, and Mingsheng Long. Representation subspace distance for domain adaptation regression. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 1749–1759. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/chen21u.html.

Boris Chidlovskii, Assem Sadek, and Christian Wolf. Universal domain adaptation in ordinal regression, 2021. URL https://arxiv.org/abs/2106.11576.

Corinna Cortes and Mehryar Mohri. Domain adaptation in regression. In Jyrki Kivinen, Csaba Szepesvári, Esko Ukkonen, and Thomas Zeugmann (eds.), *Algorithmic Learning Theory*, pp. 308–323, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-24412-4.

Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/2974788b53f73e7950e8aa49f3a306db-Paper.pdf.

Yingjun Du, Xiantong Zhen, Ling Shao, and Cees G. M. Snoek. Metanorm: Learning to normalize few-shot batches across domains. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=9z_dNsC4B5t.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 1126–1135. JMLR.org, 2017.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. URL http://jmlr.org/papers/v17/15-239.html.

Ning Gao, Hanna Ziesche, Ngo Anh Vien, Michael Volpp, and Gerhard Neumann. What matters for meta-learning vision regression tasks? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14776–14786, June 2022.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=lQdXeXDoWtI.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Junguang Jiang, Yifei Ji, Ximei Wang, Yufeng Liu, Jianmin Wang, and Mingsheng Long. Regressive domain adaptation for unsupervised keypoint detection. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pp. 6780–6789. Computer Vision Foundation / IEEE, 2021. doi: 10.1109/CVPR46437.2021.00671. URL https://openaccess.thecvf.com/content/CVPR2021/html/Jiang_Regressive_Domain_Adaptation_for_Unsupervised_Keypoint_Detection_CVPR_2021_paper.html.

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*, pp. 9619–9628, 2021.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *AAAI Conference on Artificial Intelligence*, 2018a.

Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C. Kot. Domain generalization with adversarial feature learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5400–5409, 2018b. doi: 10.1109/CVPR.2018.00566.

Haoliang Li, Yufei Wang, Renjie Wan, Shiqi Wang, Tie-Qiang Li, and Alex Kot. Domain generalization for medical imaging classification with linear-dependency regularization. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 3118–3129. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/201d7288b4c18a679e48b31c72c30ded-Paper.pdf.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning (ICML)*, pp. 6028–6039, 2020.

Feng Liu, Wenkai Xu, Jie Lu, and Danica J. Sutherland. Meta two-sample testing: Learning kernels for testing with limited data. In *NeurIPS*, 2021a.

Hong Liu, Mingsheng Long, Jianmin Wang, and Yu Wang. Learning to adapt to evolving domains. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 22338–22348. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/fd69dbe29f156a7ef876a40a94f65599-Paper.pdf.

Xiaofeng Liu, Zhenhua Guo, Site Li, Fangxu Xing, Jane You, C.-C. Jay Kuo, Georges El Fakhri, and Jonghye Woo. Adversarial unsupervised domain adaptation with conditional and label shift: Infer, align and iterate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10367–10376, October 2021b.

Xiaofeng Liu, Site Li, Yubin Ge, Pengyi Ye, Jane You, and Jun Lu. Recursively conditional gaussian for ordinal unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 764–773, October 2021c.

Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997. ISSN 00018678.

Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=h8flNv9x8v-.

Yangjun Ruan, Yann Dubois, and Chris J. Maddison. Optimal representations for covariate shift. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=Rf58LPCwJj0.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. doi: 10.1609/aaai.v32i1.11784. URL https://ojs.aaai.org/index.php/AAAI/article/view/11784.

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*, pp. 9624–9633, 2021.

Takeshi Teshima, Issei Sato, and Masashi Sugiyama. Few-shot domain adaptation by causal mechanism transfer. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Vladimir Vapnik. *The nature of statistical learning theory*. Springer science business media, 1999.

Hao Wang, Hao He, and Dina Katabi. Continuously indexed domain adaptation. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9898–9907. PMLR, 13–18 Jul 2020a. URL https://proceedings.mlr.press/v119/wang20h.html.

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*, pp. 1–1, 2022a. doi: 10.1109/TKDE.2022.3178128.

Yaoming Wang, Yangzhou Jiang, Jin Li, Bingbing Ni, Wenrui Dai, Chenglin Li, Hongkai Xiong, and Teng Li. Contrastive regression for domain adaptation on gaze estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19376–19385, June 2022b.

Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.*, 53(3), jun 2020b. ISSN 0360-0300. doi: 10.1145/3386252. URL https://doi.org/10.1145/3386252.

Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11285–11294, 2019. doi: 10.1109/CVPR.2019.01155.

Haifeng Xia, Pu Wang, Toshiaki Koike-Akino, Ye Wang, Philip Orlik, and Zhengming Ding. Adversarial bi-regressor network for domain adaptive regression. In Lud De Raedt (ed.), *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pp. 3608–3614. International Joint Conferences on Artificial Intelligence Organization, 7 2022. doi: 10.24963/ijcai.2022/501. URL https://doi.org/10.24963/ijcai.2022/501. Main Track.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey, 2021a. URL https://arxiv.org/abs/2110.11334.

Yuzhe Yang, Kaiwen Zha, Ying-Cong Chen, Hao Wang, and Dina Katabi. Delving into deep imbalanced regression. In *International Conference on Machine Learning (ICML)*, 2021b.

Huaxiu Yao, Yu Wang, Ying Wei, Peilin Zhao, Mehrdad Mahdavi, Defu Lian, and Chelsea Finn. Meta-learning with an adaptive task scheduler. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 7497–7509. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/3dc4876f3f08201c7c76cb71fa1da439-Paper.pdf.

Guojun Zhang, Han Zhao, Yaoliang Yu, and Pascal Poupart. Quantifying and improving transferability in domain generalization. *Advances in Neural Information Processing Systems*, 2021.

Hanlin Zhang, Yi-Fan Zhang, Weiyang Liu, Adrian Weller, Bernhard Schölkopf, and Eric P Xing. Towards principled disentanglement for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8024–8034, 2022.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=r1Ddp1-Rb.

Sicheng Zhao, Xiangyu Yue, Shanghang Zhang, Bo Li, Han Zhao, Bichen Wu, Ravi Krishna, Joseph E. Gonzalez, Alberto L. Sangiovanni-Vincentelli, Sanjit A. Seshia, and Kurt Keutzer. A review of single-source deep unsupervised visual domain adaptation, 2020. URL https://arxiv.org/abs/2009.00155.

Yangtao Zheng, Di Huang, Songtao Liu, and Yunhong Wang. Cross-domain object detection through coarse-to-fine feature adaptation. In *2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–20, 2022. doi: 10.1109/TPAMI.2022.3195549.

A APPENDIX

A.1 INTRODUCTION OF BASELINES

The simple introduction of baselines are described as follows:

ERM (Vapnik., 1999). The Empirical Risk Minimization method is the most simple baseline that minimizes the regression loss on source domains and reports regression loss on unseen target domain.

IRM (Arjovsky et al., 2019). Invariant Risk Minimization estimates invariant correlations across multiple training domains. For implementation, it can apply the gradient correlations from two batches as a penalty.

MMD (Li et al., 2018b). The core of MMD is to align the distribution among different domains by the Maximum Mean Discrepancy measure. Li et al. (2018b) incorporate MMD into an adversarial auto-encoder to learn generalized feature representations.

MTL (Blanchard et al., 2021). Marginal Transfer Learning views DG as a kind of supervised learning problem by augmenting the original feature space with the marginal distribution of feature vectors.

MLDG (Li et al., 2018a). Meta-Learning for Domain Generalization (MLDG) is a pioneering work that applies MAML to domain generalization. MLDG optimizes meta-train and meta-test simultaneously in the outer loop. Original MAML only optimizes meta-test objective in the outer loop. The reason to optimize the meta-train objective is that we want the learned model to be capable of directly predicting on the target domain. Note that there are other meta-learning methods for DG, such as MetaNorm (Du et al., 2021) and MASF (Dou et al., 2019). But this baseline did not release codes, e.g., MetaNorm, or are specialized for classification tasks, e.g., MASF.

DANN (Ganin et al., 2016). Domain-Adversarial Neural Networks is originally proposed to address domain adaptation problems. Besides the introduced domain adversarial framework that aligns the domain distribution, DANN also proposes an elegant implementation with a gradient reversal layer.

SD (Pezeshki et al., 2021). Spectral Decoupling controls the learning dynamic of models and tries to reduce the learning speed for unrelated features for out-of-distribution generalization. In the training process, the model has two options to reduce the loss toward an example, i.e., to get more confident in a learned feature or to learn a new feature. SD tends to increase feature diversity by encouraging learning new features.

RSD (Chen et al., 2021). Representation Subspace Distance (RSD) tries to deal with general cross-domain regression via subspace alignment, which reduces domain gap by minimizing RSD via the principal angles of representation matrices.

SelfReg (Kim et al., 2021). SelfReg proposes a domain perturbation layer to make data augmentation methods like Mixup (Zhang et al., 2018) more useful in self-supervised contrastive regularization.

Transfer (Zhang et al., 2021). The method successfully finds more transferable features via representation learning using adversarial training.

DDG (Zhang et al., 2022). Disentanglement-constrained Domain Generalization (DDG) tries to disentangle the domain-agnostic semantic features and the domain-specific variation features to achieve out of distribution prediction. The data generation and augmentation technics are also utilized to disentangle the semantic and variation features.

CAD (Ruan et al., 2022). CAD also uses self-supervised learning like SelfReg but learns discriminative representations and aligns representation's marginal support among different domains.

A.2 MORE RESULTS ON REAL-WORD DATASETS

In this section, we provide additional experiments on CACD and AFAD datasets with training domain validation. Tab. 4 and Tab. 5 provide the detailed numerical result using MSE loss.

Table 4: Regression results on CACD dataset with training-domain validation.

Algorithm	[15-20)	[20-30)	[30-40)	[40-50)	[50-60]	Avg
ERM	$0.0247_{\pm 0.01}$	$0.0492_{\pm 0.01}$	$0.0801_{\pm 0.02}$	$0.1806_{\pm 0.02}$	$0.1847_{\pm 0.07}$	0.1038
IRM	$0.0247_{\pm 0.01}$	$0.0493_{\pm 0.01}$	$0.0987_{\pm 0.02}$	$0.1811_{\pm 0.02}$	$0.2560_{\pm 0.02}$	0.1220
MMD	$0.0204_{\pm 0.01}$	$0.0405_{\pm 0.02}$	$0.0896_{\pm 0.02}$	$0.1945_{\pm 0.01}$	$0.2819_{\pm 0.02}$	0.1254
DANN	$0.0326_{\pm 0.00}$	$0.0562_{\pm 0.01}$	$0.0924_{\pm 0.02}$	$0.1521_{\pm 0.04}$	$0.2311_{\pm 0.06}$	0.1129
MTL	$0.0329_{\pm 0.00}$	$0.0641_{\pm 0.00}$	$0.1199_{\pm 0.00}$	$0.2022_{\pm 0.00}$	$0.2112_{\pm 0.08}$	0.1261
SD	$0.0247_{\pm 0.01}$	$0.0493_{\pm 0.01}$	$0.0985_{\pm 0.02}$	$0.1809_{\pm 0.02}$	$0.2767_{\pm 0.02}$	0.1260
CAD	$\mathbf{0.0330}_{\pm 0.00}$	$0.0642_{\pm 0.00}$	$0.2095_{\pm 0.07}$	$0.2022_{\pm 0.00}$	$0.3036_{\pm 0.00}$	0.1625
Transfer	$\mathbf{0.0330}_{\pm 0.00}$	$0.0641_{\pm 0.00}$	$0.1199_{\pm 0.00}$	$0.2022_{\pm 0.00}$	$0.3040_{\pm 0.00}$	0.1447
MLDG	$0.0452_{\pm 0.00}$	$\mathbf{0.0143}_{\pm 0.00}$	$0.0421_{\pm 0.03}$	$0.0131_{\pm 0.00}$	$0.1916_{\pm 0.06}$	0.0613
RSD	$0.0464_{\pm 0.00}$	$0.0190_{\pm 0.00}$	$0.0045_{\pm 0.00}$	$0.0217_{\pm 0.00}$	$0.0650_{\pm 0.01}$	0.0313
DDG	$0.0490_{\pm 0.00}$	$0.0176_{\pm 0.00}$	$\mathbf{0.0016}_{\pm 0.00}$	$0.0153_{\pm 0.00}$	$0.0598_{\pm 0.00}$	0.0287
SelfReg	$0.0403_{\pm 0.00}$	$0.0151_{\pm 0.00}$	$0.0024_{\pm 0.00}$	$0.0128_{\pm 0.00}$	$0.0539_{\pm 0.00}$	0.0249
MAMR	$0.0331_{\pm 0.01}$	$\mathbf{0.0143}_{\pm 0.00}$	$0.0021_{\pm 0.00}$	$\mathbf{0.0078}_{\pm 0.00}$	$\mathbf{0.0371}_{\pm 0.01}$	**0.0189**

Table 5: Regression results on AFAD dataset with training-domain validation.

Algorithm	[15-20)	[20-25)	[25-30)	[30-35)	[35-40]	Avg
ERM	$0.3915_{\pm 0.00}$	$0.2932_{\pm 0.00}$	$0.1338_{\pm 0.05}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.1991
IRM	$0.3081_{\pm 0.07}$	$0.2662_{\pm 0.02}$	$0.2356_{\pm 0.03}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.1973
MMD	$0.2087_{\pm 0.08}$	$0.1108_{\pm 0.07}$	$0.0678_{\pm 0.05}$	$0.1193_{\pm 0.09}$	$0.2010_{\pm 0.11}$	0.1415
DANN	$0.2607_{\pm 0.08}$	$0.1658_{\pm 0.04}$	$0.1475_{\pm 0.03}$	$0.1170_{\pm 0.02}$	$0.1043_{\pm 0.05}$	0.1591
MTL	$0.3915_{\pm 0.00}$	$0.2936_{\pm 0.00}$	$0.1980_{\pm 0.00}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.2120
SD	$0.0324_{\pm 0.00}$	$0.0089_{\pm 0.00}$	$0.0034_{\pm 0.00}$	$0.0223_{\pm 0.00}$	$0.0738_{\pm 0.00}$	0.0281
CAD	$0.3915_{\pm 0.00}$	$0.2936_{\pm 0.00}$	$0.1990_{\pm 0.00}$	$0.1171_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.2123
Transfer	$0.3915_{\pm 0.00}$	$0.2936_{\pm 0.00}$	$0.1990_{\pm 0.00}$	$0.1168_{\pm 0.00}$	$0.0601_{\pm 0.00}$	0.2122
MLDG	$0.1614_{\pm 0.09}$	$0.0129_{\pm 0.00}$	$0.0036_{\pm 0.00}$	$0.0142_{\pm 0.00}$	$\mathbf{0.0553}_{\pm 0.00}$	0.0495
DDG	$0.0556_{\pm 0.00}$	$0.0166_{\pm 0.00}$	$\mathbf{0.0012}_{\pm 0.00}$	$0.0164_{\pm 0.00}$	$0.0610_{\pm 0.00}$	0.0302
SelfReg	$0.0474_{\pm 0.00}$	$0.0156_{\pm 0.00}$	$0.0028_{\pm 0.00}$	$\mathbf{0.0131}_{\pm 0.00}$	$0.0555_{\pm 0.00}$	0.0269
RSD	$0.0506_{\pm 0.00}$	$0.0194_{\pm 0.00}$	$0.0042_{\pm 0.00}$	$0.0171_{\pm 0.00}$	$0.0576_{\pm 0.00}$	0.0298
MAMR	$\mathbf{0.0281}_{\pm 0.00}$	$\mathbf{0.0068}_{\pm 0.00}$	$\mathbf{0.0012}_{\pm 0.00}$	$0.0190_{\pm 0.00}$	$0.0641_{\pm 0.00}$	**0.0238**

A.3 HYPER-PARAMETER SETTING

To help the readers reproduce the reported results, we provide more hyper-parameters in Tab. 6. The outer loop learning rate is used by our MAMR model, and the left hyper-parameters are shared by all methods.

Table 6: The hyper-parameter settings of our MAMR model and baselines.

Hyper-Parameters Setting	**Values**
Inner loop learning rate β	0.05
Outer loop learning rate α	$0.1 * \beta$
Inner loop iteration steps	1
Batch size of each support or query task	64
Holdout fraction for each domain	0.1
Trial seeds:	$3057, 3058, 3059$
Optimizer:	SGD
Optimizer weight decay:	$5e-4$
Data augmentation	RandomResizedCrop, RandomHorizontalFlip
Data normalization (mean)	mean=[0.485, 0.456, 0.406]
Data normalization (std)	std=[0.229, 0.224, 0.225]

A.4 Causal Mechanism in Toy Experiments

We provide the used causal mechanism in toy experiments. Fig. 5 demonstrate the mechanism to generate Y given two inputs X_1 and X_2. In this example, the dominant variable X_1 controls 5 generation factors, the auxiliary variable X_2 controls 3 generation factors. All the generation factors form a sum and the sum is normalized to the interval $[0, 1]$ for Y.

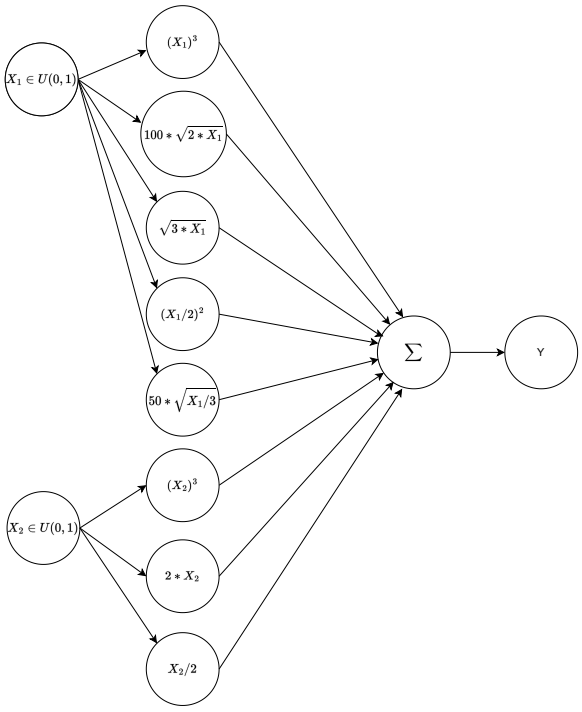


Figure 5: The example of the generation mechanism for toy experiments. Note that $\sum$ denotes the sum of all the coming elements, and the responding value Y is normalized to [0,1] after $\sum$.