Hacking Task Confounder in Meta-Learning

Jingyao Wang[1,2] , **Yi Ren**[1] , **Zeen Song**[1,2] , **Jianqi Zhang**[1,2] ,
Changwen Zheng[1] and **Wenwen Qiang**[1,2]*

[1]Institute of Software Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences
{wangjingyao23, renyi, songzeen, zhangjianqi, changwen, qiangwenwen}@iscas.ac.cn

Abstract

Meta-learning enables rapid generalization to new tasks by learning knowledge from various tasks. It is intuitively assumed that as the training progresses, a model will acquire richer knowledge, leading to better generalization performance. However, our experiments reveal an unexpected result: there is negative knowledge transfer between tasks, affecting generalization performance. To explain this phenomenon, we conduct Structural Causal Models (SCMs) for causal analysis. Our investigation uncovers the presence of spurious correlations between task-specific causal factors and labels in meta-learning. Furthermore, the confounding factors differ across different batches. We refer to these confounding factors as "Task Confounders". Based on these findings, we propose a plug-and-play Meta-learning Causal Representation Learner (MetaCRL) to eliminate task confounders. It encodes decoupled generating factors from multiple tasks and utilizes an invariant-based bi-level optimization mechanism to ensure their causality for meta-learning. Extensive experiments on various benchmark datasets demonstrate that our work achieves state-of-the-art (SOTA) performance. The code is provided in https://github.com/WangJingyao07/MetaCRL.

1 Introduction

Meta-learning aims to develop models that can be rapidly transferred to previously unseen tasks. To achieve this, it first learns from diverse tasks to obtain models with high learning capacities. Then, it fine-tunes these models with little data from unseen tasks to obtain the desired ones. Recently, meta-learning has been widely applied in various fields, e.g., affective computing [Li *et al.*, 2023], image classification [Qiang *et al.*, 2023], and robotics [Schrum *et al.*, 2022].

During the training phase, each batch consists of a series of randomly sampled N-way K-shot tasks, where N denotes the number of classes per task and K denotes the number of samples per class. The samples in each task are divided into a support set and a query set. Then, meta-learning models are trained in a bi-level optimization manner [Wang *et al.*, 2021; Wang *et al.*, 2023]. In brief, at the first level, the desired model for each task is fine-tuned by training on the support set using the meta-learning model. At the second level, the meta-learning model is learned using the query sets from all training tasks and the corresponding expected models for each task. Therefore, a widely adopted hypothesis is that as training progresses, the meta-learning model will acquire richer knowledge that can be transferred well to downstream tasks, achieving better performance [Rivolli *et al.*, 2022].

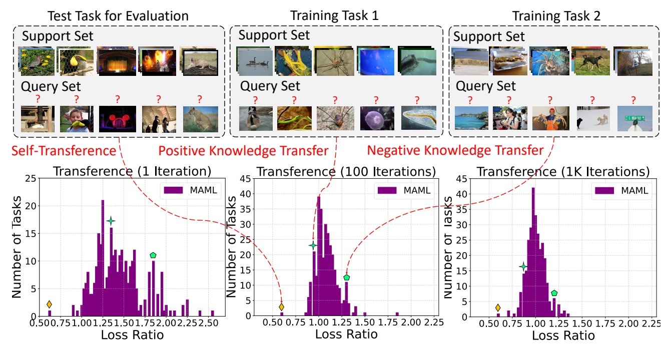
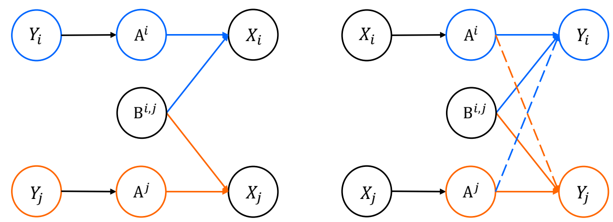


Figure 1: Knowledge transfer to a specific test task. For both positive knowledge transfer ($\mathcal{R}_{i,j} < 1$) and negative knowledge transfer ($\mathcal{R}_{i,j} > 1$), an exemplar task is shown. Here, we simply use the $\mathcal{R}_{i,j}$ threshold to classify the knowledge transfer as positive or negative. See Subsection 3.2 and Appendix F for more details.

However, our toy experiments reveal a conflicting phenomenon, i.e., the knowledge learned from the training tasks may be harmful to the unseen test tasks (See Subsection 3.2 for more details). Specifically, we first randomly sample 400 tasks from miniImageNet dataset [Vinyals *et al.*, 2016] and divide them into a training set and a test set. Then, we define a metric $\mathcal{R}_{i,j}$ to evaluate whether the meta-learning model trained on the training tasks can perform better on the test task, i.e., quantify the knowledge transfer performance from the training tasks to each test task. If $\mathcal{R}_{i,j} < 1$, the learned knowledge from the training task can help improve the model performance on the test task (positive knowledge transfer), while $\mathcal{R}_{i,j} > 1$ implies the learned knowledge is harmful to the test task (negative knowledge transfer). We

*Corresponding Author



(a) data generation mechanism (b) meta-learning process

Figure 2: Structural Causal Models (SCM) regarding two tasks τ_i and τ_j, where (X_i, Y_i) and (X_j, Y_j) are the samples and corresponding labels of these tasks. The solid line means the true causal correlation, and the dotted line means the spurious correlation. (a) is constructed based on the ground-truth causal mechanism, while (b) can be viewed as the inverse process of the generating mechanism.

use MAML [Finn *et al.*, 2017] as the baseline and record the score of $\mathcal{R}_{i,j}$ in the middle of training [Fifty *et al.*, 2020; Abdollahzadeh *et al.*, 2021]. Figure 1 shows the results. Ideally, all the knowledge transfer between tasks should be positive, i.e., $\mathcal{R}_{i,j} < 1$. The results show that there always exists negative knowledge transfer between tasks.

To explore the reasons behind this phenomenon, we propose using causal theory for analysis (See Subsection 3.3 for details). We begin by constructing Structural Causal Models (SCMs) for the training phase of ML, as shown in Figure 2. In the SCMs, A^i and A^j are the distinct causal factors of task τ_i and task τ_j, and $\mathrm{B}^{i,j}$ means the shared causal factors of these two tasks. Meanwhile, causal factors can be considered as different semantics of the data, e.g., color and shape, also considered as generating factors used for data generation [Zimmermann *et al.*, 2021]. Since meta-learning performs joint learning on all the training tasks, it acquires all the causal factors. Thus, the non-overlapping causal factors A^i of τ_i may cause spurious correlations with τ_j, and A^j holds the same with τ_i. These misleading correlations between training tasks will introduce bias into the learned knowledge and ultimately affect generalization, which is called "**task confounder**".

To address this issue, we propose a plug-and-play meta-learning causal representation learner (MetaCRL) to encode decoupled causal knowledge, thereby eliminating task confounders. It consists of two modules: the disentangling module and the causal module. The former aims to extract generating factors across all tasks and provide a subset of factors relevant to each task, while the latter is responsible for ensuring their causality. The modules achieve their objectives through a simple bi-level optimization mechanism with regularization terms. By incorporating MetaCRL into meta-learning, we dynamically eliminate task confounders during the meta-training process. Through extensive evaluations of multiple meta-learning benchmarks, we demonstrate that MetaCRL can significantly improve performance.

In summary, our contributions are as follows:

- We discover a counterintuitive phenomenon: there is negative knowledge transfer between tasks, resulting in reduced model generalization performance.
- We construct an SCM to analyze the phenomenon with causal theory, finding spurious correlations, named "Task Confounders", between non-shared causal factors of the meta-training tasks and the label space.
- We propose MetaCRL, a plug-and-play meta-learning causal representation learner to eliminate task confounders, thus improving generalization performance.
- Extensive experiments on various scenarios demonstrate the outstanding performance of our MetaCRL.

2 Related Work

Meta-learning aims to learn general knowledge from various training tasks, and then generalize to new tasks based on the acquired knowledge. Typical methods can be categorized into two types: optimization-based [Finn *et al.*, 2017; Nichol and Schulman, 2018; Guo *et al.*, 2024] and metric-based [Snell *et al.*, 2017; Sung *et al.*, 2018; Chen *et al.*, 2020] methods. They both rely on shared structures and bi-level learning mechanisms to learn general knowledge, resulting in remarkable performance on new tasks. However, meta-learning still faces the crisis of performance degradation. Various approaches have been proposed to address this issue, such as adding adaptive noise [Lee *et al.*, 2020], reducing inter-task disparities [Jamal and Qi, 2019], limiting the trainable parameters [Yin *et al.*, 2019; Oh *et al.*, 2020], and task augmentation [Yao *et al.*, 2021]. Despite alleviating performance degradation, they ignore the interaction between tasks, which is shown to be crucial in Section 3. In this study, we analyze the knowledge transfer effects between different training tasks with causal theory, and focus on the fundamental causes of performance degradation in meta-learning.

Causal learning aims to explore the causal relationships between variables in machine learning, modeling the target with a directed acyclic graph, also known as a causal model. It has been shown to aid models in unearthing underlying causal factors [Yang *et al.*, 2021; Zhang *et al.*, 2020; Nogueira *et al.*, 2022]. Current research attempts to combine causal knowledge with meta-learning methods to address domain challenges. Yue et al. [Yue *et al.*, 2020] removed performance limitations of pre-trained knowledge through backdoor regulation. Ton et al. [Ton *et al.*, 2021] utilized causal knowledge to distinguish causes and effects in a bivariate environment with limited data. Jiang et al. [Jiang *et al.*, 2022] used causal graphs to remove undesirable memory effects. While they all combine meta-learning and causal learning, their focus is on addressing problems that differ from ours.

3 Problem Formulation and Analysis

In this section, we first present the notation and problem definition of meta-learning. Next, we conduct experiments to evaluate the interaction between different tasks and illustrate the empirical evidence, i.e., the knowledge learned from the training tasks may be harmful to the unseen test tasks, reducing generalization performance. Finally, we construct SCMs to explore the reasons behind the empirical evidence.

3.1 Preliminaries

Given a task distribution $p(\mathcal{T})$, the meta-training dataset $\mathcal{D}_{tr}$ and the meta-test dataset $\mathcal{D}_{te}$ are all sampled from $p(\mathcal{T})$ without class-level overlap. During the training phase of ML, each batch contains N_{tr} tasks, denoted as $\{\tau_i\}_{i=1}^{N_{tr}} \in \mathcal{D}_{tr}$, and each task τ_i consists of a support set $\mathcal{D}_i^s = (X_i^s, Y_i^s) = \{(x_{i,j}^s, y_{i,j}^s)\}_{j=1}^{N_i^s}$ and a query set $\mathcal{D}_i^q = (X_i^q, Y_i^q) = \{(x_{i,j}^q, y_{i,j}^q)\}_{j=1}^{N_i^q}$, where $(x_{i,j}^{\cdot}, y_{i,j}^{\cdot})$ represents the sample and the corresponding label, and $N_i^{\cdot}$ denotes the number of the samples. The meta-learning model $f_\theta = h \circ g$ utilizes the feature encoder g and the classifier h to learn the above tasks.

The learning mechanism of meta-learning is regarded as a bi-level optimization process. At the first level, it fine-tune the desired model f_θ^i for task τ_i by training on the support set $\mathcal{D}_i^s$ using the meta-learning model f_θ, presented as:

$$\begin{aligned} & f_\theta^i \leftarrow f_\theta - \alpha \nabla_{f_\theta} \mathcal{L}(Y_i^s, X_i^s, f_\theta) \\ s.t. \quad & \mathcal{L}(Y_i^s, X_i^s, f_\theta) = \tfrac{1}{N_i^s} \textstyle\sum_{j=1}^{N_i^s} y_{i,j}^s \log f_\theta(x_{i,j}^s) \end{aligned} \tag{1}$$

where α is the learning rate. At the second level, the meta-learning model f_θ is learned using the query sets $\mathcal{D}^q$ from all training tasks and the expected models for each task:

$$\begin{aligned} & f_\theta \leftarrow f_\theta - \beta \nabla_{f_\theta} \tfrac{1}{N_{tr}} \textstyle\sum_{i=1}^{N_{tr}} \mathcal{L}(Y_i^q, X_i^q, f_\theta^i) \\ s.t. \quad & \mathcal{L}(Y_i^q, X_i^q, f_\theta^i) = \tfrac{1}{N_i^q} \textstyle\sum_{j=1}^{N_i^q} y_{i,j}^q \log f_\theta^i(x_{i,j}^q) \end{aligned} \tag{2}$$

where β is the learning rate. Note that f_θ^i is obtained by taking the derivative of f_θ, so f_θ^i can be regarded as a function of f_θ. Therefore, the update of f_θ mentioned in Eq.2 can be viewed as calculating the second derivative of f_θ.

3.2 Empirical Evidence

From above and [Wang *et al.*, 2021], meta-training on one batch can be viewed as a multi-task learning process. Meanwhile, a well-learned model should contain knowledge of all training tasks. Therefore, intuitively, one might assume that as training progresses, the meta-learning model will acquire richer knowledge (related to all tasks) and transfer better to downstream tasks, achieving great generalization. However, our toy experiments reveal that this is not always true.

Before introducing the toy experiments, we first present a method to quantify the influence of transferring knowledge learned from one task to the target task. For task τ_i, the model f_θ uses the support set $\mathcal{D}_i^s$ to obtain f_θ^i via Eq.1. Here, f_θ^i is considered to integrate the knowledge of task τ_i into f_θ. Then, for task τ_j, we first obtain the model $f_\theta^{j,1}$ by training f_θ^i on the support set $\mathcal{D}_j^s$, and then obtain the model $f_\theta^{j,2}$ by training f_θ on $\mathcal{D}_j^s$. Next, we calculate their losses on the query set $\mathcal{D}_j^q$, expressed as $\mathcal{L}(\mathcal{D}_j^q, f_\theta^{j,1})$ and $\mathcal{L}(\mathcal{D}_j^q, f_\theta^{j,2})$, respectively. Finally, we calculate the ratio between these two losses, denoted as $\mathcal{R}_{i,j}$, which quantifies the performance of knowledge transfer from task τ_i to task τ_j. Thus, we have:

$$\mathcal{R}_{i,j} = \frac{\mathcal{L}(\mathcal{D}_j^q, f_\theta^{j,1})}{\mathcal{L}(\mathcal{D}_j^q, f_\theta^{j,2})} \tag{3}$$

if $\mathcal{R}_{i,j} < 1$, it means that task τ_i has a positive knowledge transfer effect on task τ_j. On the other hand, if $\mathcal{R}_{i,j} > 1$, it indicates the negative knowledge transfer effect of τ_i on τ_j.

Next, we conduct experiments based on the quantitative method described above. We first randomly sample 400 tasks from miniImageNet dataset, which are divided into a training set of 300 tasks and a test set of 100 tasks. Then, we use MAML as the baseline to calculate the score of $\mathcal{R}_{i,j}$ from the training tasks to each test task in the middle of training.

Figure 1 shows the histograms of the knowledge transfer in the training phase of meta-learning along with exemplar tasks. From the results, we observe that as training proceeds, although the knowledge transfer effects become more and more positive, there always exists negative knowledge transfer between different tasks. It indicates that the training process of meta-learning cannot always obtain effective knowledge for unseen test tasks, and the aforementioned intuitive hypothesis is limited. Note that we also conduct experiments under various different settings, including using multiple meta-learning baselines, using different datasets, and training on multiple tasks simultaneously (the effect of multiple training tasks to a single test task), the impact of negative knowledge transfer always exists. More details and the full results are provided in Appendix F.

3.3 Causal Analysis and Motivation

To explore the reasons behind the above phenomenon, we propose using causal theory for analysis. We first construct a Structural Causal Model (SCM) based on the ground-truth causal mechanisms [Suter *et al.*, 2019; Hu *et al.*, 2022], as shown in Figure 2a. Specifically, this SCM contains two tasks τ_i and τ_j, where Y_i and Y_j denote the label variables for tasks τ_i and τ_j, X_i and X_j signify the corresponding generated samples for these two tasks, respectively. Meanwhile, A^i and A^j represent the distinct sets of causal factors specific to tasks τ_i and τ_j, while $\mathrm{B}^{i,j}$ encompasses shared causal factors. In this SCM, we assume that the samples X_i and X_j are generated by disentangled causal mechanisms using the causal factors, then $p(X_i|\mathrm{A}^i, \mathrm{B}^{i,j}) = \prod_k p(X_i|\mathrm{A}_k^i) \prod_t p(X_i|\mathrm{B}_t^{i,j})$, where A_k^i denotes the k-th factor of A^i, and $\mathrm{B}_t^{i,j}$ denotes the t-th factor of $\mathrm{B}^{i,j}$. Since A^i, A^j, and $\mathrm{B}^{i,j}$ represent high-level knowledge of the data, we could naturally define the task label variable Y_i for task i as the cause of the $\mathrm{B}^{i,j}$ and A^i. For the task τ_i, we call $\mathrm{B}^{i,j}$ and A^i as the causal feature variables that are causally related to Y_i, and we call A^j as the non-causal feature variables to task τ_i. Therefore, we have $p(X_i|\mathrm{A}^i, \mathrm{B}^{i,j}, \mathrm{A}^j) = p(X_i|\mathrm{A}^i, \mathrm{B}^{i,j})$.

Based on the proposed SCM, an ideal meta-learning predictor for each task should only utilize causal factors and be invariant to any intervention on non-causal factors. However, the joint learning of multiple tasks in meta-learning could give rise to the issue of using non-causal factors for unseen tasks, also known as spurious correlations, thereby making it challenging to achieve optimal predictions. To verify this claim, we consider the scenario of two binary classification tasks for simple but clear explanations. Let Y_i and Y_j be variables from $\{\pm 1\}$, we assume τ_i and τ_j have non-overlapping factors, i.e., $\mathrm{B}^{i,j} = \emptyset$, and the elements in A^i and A^j satisfy the constraint of Gaussian distribution. Then, we have:

Theorem 1. *If the correlation between Y_i and Y_j is not equal to 0.5, the optimal classifier has non-zero weights for non-causal factors for each task. If the correlation between Y_i and Y_j equals 0.5 with limited training data, the optimal classifier* *also has non-zero weights for non-causal factors in each task.*

As inferred from the aforementioned theorem, the learned model leverages the causal factors from other tasks to facilitate the learning of the target task. Taking the task τ_i as an example, the meta-learning model uses the causal factors A^j belonging to the task τ_j for learning Y_i. Therefore, there is a spurious correlation between A^j and Y_i, which can be represented as a spurious path $\mathrm{A}^j \rightarrow Y_i$. Similarly, we can obtain the spurious path $\mathrm{A}^i \rightarrow Y_j$ for task τ_j. These spurious correlations are called "task confounders", which are the reasons that lead to negative knowledge transfer in Subsection 3.2. The learning process can be viewed as the inverse process of the generating mechanism. Therefore, we can obtain the SCM with two spurious paths as illustrated in Figure 2b, which reflects the internal mechanism of task confounders in multi-task learning. The proof is provided in Appendix A.

4 Methodology

Based on the above analysis, we know that task confounders cause spurious correlations between causal factors and labels. An ideal meta-learning model should identify knowledge that is causally related to each task and learn from the identified multi-task knowledge. Therefore, we propose MetaCRL, a plug-and-play meta-learning causal representation learner that can encode decoupled causal factors for more efficient ML. It consists of two modules: (i) the disentangling module which aims to extract generating factors and eliminate task confounders; and (ii) the causal module which aims to ensure the causality of the obtained generating factors. In this section, we first introduce the disentangling module and the causal module in Subsections 4.1 and 4.2, respectively. Next, we provide the overall objective in Subsection 4.3. The pseudocode and pipeline of MetaCRL are shown in Appendix B.

4.1 Disentangling Module

In this module, we aim to obtain the whole generating factors related to all tasks and the task-specific generating factors related to each single task. Specifically, we first obtain the whole generating factors by learning a semantic matrix Ξ. Next, we use a grouping function f_{gr} to acquire subsets of generating factors relevant to every single task. Note that this module does not guarantee the causality of the obtained generating factors, which will be addressed in the causal module.

For a pre-trained encoder, different channels of the feature representations are related to different kinds of semantics [Islam *et al.*, 2020]. Thus, we propose to use the feature representation to learn the generating factors. During the training phase, we denote the N_{tr} training tasks as $\{\tau_i\}_{i=1}^{N_{tr}}$. Suppose that the number of generating factors is N_k, then, we propose obtaining these N_k factors through the learning of a matrix $\Xi \in \mathbb{R}^{N_z \times N_k}$. Here, N_z represents the dimension of the feature representation, i.e., the output dimension of the encoder g, and each column of Ξ represents a distinct factor. Based on Ξ, we can obtain a new representation of each sample, which can be called a generating representation, e.g., the generating representation for $x_{i,j}^s$ can be presented as $\Xi^{\mathrm{T}} g(x_{i,j}^s)$.

Generally, generating factors in geometric space can be conceptualized as coordinate basis vectors, where each generating factor corresponds to a specific basis vector [Jensen and Shen, 2004]. Moreover, different coordinate bases can undergo mutual transformations via a reversible matrix, implying their equivalence. Hence, learning a task-specific matrix, serving as a base matrix, allows us to approximate task-related generating factors. Therefore, for Ξ to be considered a generating factor matrix, we need to constrain the column vectors of Ξ to be orthogonal to each other. Then we have:

$$\mathcal{L}_{\mathrm{DM}}(\Xi) = \sum_{i=1}^{N_k-1} \sum_{j=i+1}^{N_k} \Xi_{:,i}^{\mathrm{T}} \Xi_{:,j} \tag{4}$$

where $\Xi_{:,i}$ represents the i-th column of Ξ. Minimizing $\mathcal{L}_{\mathrm{DM}}(\Xi)$ makes the different columns of Ξ orthogonal to each other, thus leading Ξ to be task-related generating factors.

Next, for all the N_{tr} training tasks, the generating factors should be divided into N_{tr} overlapping groups, and each group corresponds to a task. To obtain these groups, we propose a learnable grouping function f_{gr}, which is implemented using Multi-Layer Perceptrons (MLPs) to acquire task-specific generating factors. Take task τ_i as an example, we first calculate the average sample x_i for this task, i.e., $x_i = \frac{1}{N_i^s + N_i^q}(\sum_{j=1}^{N_i^s} x_{i,j}^s + \sum_{j=1}^{N_i^q} x_{i,j}^q)$. Then, we input x_i into the encoder g, Ξ, and f_{gr}, i.e., $f_{gr}(\Xi^{\mathrm{T}} g(x_i))$, yielding a vector with all elements greater than zero and matching the dimensionality of the generating representation. Then, each element is subject to the normalization operation, denoted as $\mathrm{Norm}(\cdot)$. As a result, the individual elements of the output vector, i.e., $\mathrm{Norm}(f_{gr})$, can be interpreted as the probabilities that each generating factor belongs to task τ_i.

Note that each task is associated with a subset of factors in Ξ and can vary significantly from task to task. Meanwhile, the above calculation process of Ξ and f_{gr} may lead to degenerate solutions, e.g., the subset of generating factors for each task is the same. To address this issue, we propose a regularization term that consists of a L_1 norm and an entropy term, constraining the output of f_{gr} to be sparse and diverse. By minimizing the L_1 norm, we make the output of f_{gr} sparse, ensuring obtain subsets of generating factors only relevant to each single task. By maximizing the entropy term, we make the output of f_{gr} diverse, preventing the acquisition of task-specific generating factors suffering degenerate solutions. The regularization term is:

$$\begin{aligned}\mathcal{L}_{\mathrm{DM}}(f_{gr}) = &\sum_{i=1}^{N_{tr}} \left\| f_{gr}(\Xi^{\mathrm{T}} g(x_i)) \right\|_1 \\ &-\mathrm{Entropy}\Big(\frac{\sum_j f_{gr}(\Xi^{\mathrm{T}} g(x_i))_j}{\sum_i \sum_j f_{gr}(\Xi^{\mathrm{T}} g(x_i))_j}\Big)\end{aligned} \tag{5}$$

where $f_{gr}(\Xi^{\mathrm{T}} g(x_i))_j$ represents the j-th element of the output of f_{gr}. Through Eq.5, we obtain accurate task-specific generating factors, thus eliminating task confounders.

By combining Eq.4 and Eq.5, we obtain the loss of the disentangling module which can be expressed as:

$$\mathcal{L}_{\mathrm{DM}}(f_{gr}, \Xi) = \lambda_1 \cdot \mathcal{L}_{\mathrm{DM}}(\Xi) + \lambda_2 \cdot \mathcal{L}_{\mathrm{DM}}(f_{gr}) \tag{6}$$

where λ_1 and λ_2 denote the loss weights of $\mathcal{L}_{\mathrm{DM}}(\Xi)$ and $\mathcal{L}_{\mathrm{DM}}(f_{gr})$, respectively. Through the above process with three constraints, i.e., correlation, sparsity, and diversity, we can accurately obtain all the generating factors and the task-specific generating factors without task confounders.

4.2 Causal Module

In this module, we aim to ensure the causality of the generating factors obtained in the disentangling module. Following [Koyama and Yamaguchi, 2020], a model invariant to different distributions can learn causal correlations. Meanwhile, based on Theorem 9 described in [Arjovsky *et al.*, 2019], by enforcing invariance over multiple training datasets that exhibit distribution shifts, the task-specific models could only use task-related causal factors and assign zero weights to those non-causal generating factors. Therefore, the causal module is designed to facilitate causal learning by using this invariance, thereby ensuring the causality of the generating factors obtained by Ξ and f_{gr}.

During the training phase of ML, the training data can be divided into multiple support sets and query sets. As they comprise different samples, they can be regarded as different data distributions with distributional shifts. Meanwhile, the learning process of meta-learning can be depicted as follows: First, for every f_θ, optimizing Eq.1 can achieve an optimal f_θ^i and $\mathcal{L}(Y_i^s, X_i^s, f_\theta^i)$ on the support set. Next, altering the value of f_θ impacts the optimal f_θ^i, we seek the optimal f_θ to obtain the optimal f_θ^i by optimizing $\frac{1}{N_{tr}}\sum_{i=1}^{N_{tr}}\mathcal{L}(Y_i^q, X_i^q, f_\theta^i)$ on the query sets (Eq.2). Thus, the bi-level optimization of Eq.1 and Eq.2 can be interpreted as achieving optimality across multiple datasets using the same f_θ, and the causal factors are invariant on the support and query sets of the same task.

Based on the above illustration, we propose to utilize a bi-level optimization mechanism to learn Ξ and f_{gr} which is similar to Eq.1 and Eq.2, thus ensuring causality. Specifically, for the first level, we learn $\Xi^{'}$ and $f_{gr}^{'}$ with the support sets through the following objectives:

$$
\begin{aligned}
&\begin{cases} \Xi^{'} \leftarrow \Xi - \alpha_1 \nabla_{\Xi}\tilde{\mathcal{L}} \\ f_{gr}^{'} \leftarrow f_{gr} - \alpha_2 \nabla_{f_{gr}}\tilde{\mathcal{L}} \end{cases} \\
s.t. \quad &\tilde{\mathcal{L}} = \frac{1}{N_{tr}}\sum_{i=1}^{N_{tr}}\mathcal{L}(Y_i^s, X_i^s, \Xi, f_{gr}) + \mathcal{L}_{\mathrm{DM}}(\Xi, f_{gr}) \\
&\mathcal{L}(Y_i^s, X_i^s, \Xi, f_{gr}) = \frac{1}{N_i^s}\sum_{j=1}^{N_i^s} y_{i,j}^s \log \mathrm{z}_{i,j}^s \\
&\mathrm{z}_{i,j}^s = h\{\mathrm{Norm}[f_{gr}(\Xi^{\mathrm{T}} g(x_i))] \odot [\Xi^{\mathrm{T}} g(x_{i,j}^s)]\}
\end{aligned} \tag{7}
$$

and for the second level, we learn Ξ and f_{gr} with the query sets through the following objectives:

$$
\begin{aligned}
&\begin{cases} \Xi \leftarrow \Xi - \alpha_3 \nabla_{\Xi}\tilde{\mathcal{L}}' \\ f_{gr} \leftarrow f_{gr} - \alpha_4 \nabla_{f_{gr}}\tilde{\mathcal{L}}' \end{cases} \\
s.t. \quad &\tilde{\mathcal{L}}' = \frac{1}{N_{tr}}\sum_{i=1}^{N_{tr}}\mathcal{L}(Y_i^q, X_i^q, \Xi^{'}, f_{gr}^{'}) + \mathcal{L}_{\mathrm{DM}}(\Xi^{'}, f_{gr}^{'}) \\
&\mathcal{L}(Y_i^q, X_i^q, \Xi^{'}, f_{gr}^{'}) = \frac{1}{N_i^q}\sum_{j=1}^{N_i^q} y_{i,j}^q \log \mathrm{z}_{i,j}^q \\
&\mathrm{z}_{i,j}^q = h\{\mathrm{Norm}[f_{gr}(\Xi^{'\mathrm{T}} g(x_i))] \odot [\Xi^{'\mathrm{T}} g(x_{i,j}^q)]\}
\end{aligned} \tag{8}
$$

where $\odot$ represents the element-wise multiplication operator between two vectors, i.e., the generating representation $\Xi^{\mathrm{T}} g(x_{i,j}^{\cdot})$ and the weight $\mathrm{Norm}[f_{gr}(\Xi^{\mathrm{T}} g(x_i))]$, while α_1, α_2, α_3 and α_4 are the learning rates. Note that both in Eq.7 and Eq.8, the loss $\mathcal{L}(Y_i^{\cdot}, X_i^{\cdot}, \Xi, f_{gr})$ is calculated using the generating representations with causal weights instead of feature representations, which restrict the features of the samples in τ_i to be associated only with task-specific causal factors.

In summary, the learning process of Ξ and f_{gr} can be regarded as enforcing invariance over the support sets and the query sets, and the bi-level optimization mechanism for Ξ and f_{gr} can ensure causality. Meanwhile, Ξ and f_{gr} are learned independently with the fixed meta-learning model f_θ in the middle training following modularity design, thus rendering the MetaCRL a plug-and-play learner.

4.3 Overall Objective

In this subsection, we embed the above causal representation learning process into a meta-learning framework for joint optimization. The training process with MetaCRL in each batch is divided into two steps. In the first step, with Ξ and f_{gr} held fixed, we optimize the meta-learning model $f_\theta = h \circ g$. Specifically, the objective of the inner loop becomes:

$$
\begin{aligned}
&f_\theta^i \leftarrow f_\theta - \alpha\nabla_{f_\theta}\tilde{\mathcal{L}}(Y_i^s, X_i^s, f_\theta) \\
s.t. \quad &\tilde{\mathcal{L}}(Y_i^s, X_i^s, f_\theta) = \frac{1}{N_i^s}\sum_{j=1}^{N_i^s} y_{i,j}^s \log \mathrm{z}_{i,j}^s
\end{aligned} \tag{9}
$$

where $\mathrm{z}_{i,j}^s$ is calculated the same as Eq.7. Subsequently, the objective of the outer loop mentioned in Eq.2 becomes:

$$
\begin{aligned}
&f_\theta \leftarrow f_\theta - \beta\nabla_{f_\theta}\frac{1}{N_{tr}}\sum_{i=1}^{N_{tr}}\tilde{\mathcal{L}}(Y_i^q, X_i^q, f_\theta^i) \\
s.t. \quad &\tilde{\mathcal{L}}(Y_i^q, X_i^q, f_\theta^i) = \frac{1}{N_i^q}\sum_{j=1}^{N_i^q} y_{i,j}^q \log \mathrm{z}_{i,j}^q
\end{aligned} \tag{10}
$$

where $\mathrm{z}_{i,j}^q$ is calculated as mentioned in Eq.8. Next, in the second step, with the meta-learning model f_θ held fixed, we optimize Ξ and f_{gr} as mentioned in Eq.7 and Eq.8.

By incorporating the causal invariant-based optimization mechanism and the additional regularization term, we can effectively eliminate task confounders that lead to model degradation and improve generalization capability.

5 Experiments

In this section, we first evaluate MetaCRL on various scenarios, including sinusoid regression, image classification, drug activity prediction, and pose prediction in Subsections 5.1-5.4, respectively. Next, we conduct ablation studies and visualization in Subsections 5.5 and 5.6. Considering that MetaCRL is a plug-and-play method, we assess its performance on several meta-learning models, e.g., MAML [Finn *et al.*, 2017], ANIL [Raghu *et al.*, 2019], MetaSGD [Li *et al.*, 2017], and T-NET [Lee and Choi, 2018], and multiple causal-based baselines, e.g., IFSL [Yue *et al.*, 2020], Meta-Trans [Bengio *et al.*, 2019], Meta-Aug [Rajendran *et al.*, 2020], and MR-MAML [Yin *et al.*, 2019], to demonstrate its compatibility. Considering that MetaCRL addresses the "Task Confounder" problem to enhance generalization, we also compare it with the plug-and-play generalization baselines that are most relevant to our method, i.e., MetaMix [Yao *et al.*, 2021] and Dropout-Bins [Jiang *et al.*, 2022]. We delay all the details of datasets, baselines, implementation details, and additional experimental results in Appendices C-F, respectively.

Model	5-shot	10-shot
IFSL	0.592 ± 0.141	0.178 ± 0.040
Meta-Trans	0.577 ± 0.123	0.140 ± 0.024
Meta-Aug	0.531 ± 0.118	0.103 ± 0.031
MR-MAML	0.581 ± 0.110	0.104 ± 0.029
MAML	0.593 ± 0.120	0.166 ± 0.061
MAML + MetaMix	0.476 ± 0.109	0.085 ± 0.024
MAML + Dropout-Bins	0.452 ± 0.081	0.062 ± 0.017
MAML + Ours	$\mathbf{0.440 \pm 0.079}$	$\mathbf{0.054 \pm 0.018}$
ANIL	0.541 ± 0.118	0.103 ± 0.032
ANIL + MetaMix	0.514 ± 0.106	0.083 ± 0.022
ANIL + Dropout-Bins	0.487 ± 0.110	0.088 ± 0.025
ANIL + Ours	$\mathbf{0.468 \pm 0.094}$	$\mathbf{0.081 \pm 0.019}$
MetaSGD	0.577 ± 0.126	0.152 ± 0.044
MetaSGD + MetaMix	0.468 ± 0.118	0.072 ± 0.023
MetaSGD + Dropout-Bins	0.435 ± 0.089	0.040 ± 0.011
MetaSGD + Ours	$\mathbf{0.408 \pm 0.071}$	$\mathbf{0.038 \pm 0.010}$
T-NET	0.564 ± 0.128	0.111 ± 0.042
T-NET + MetaMix	0.498 ± 0.113	0.094 ± 0.025
T-NET + Dropout-Bins	0.470 ± 0.091	0.077 ± 0.028
T-NET + Ours	$\mathbf{0.462 \pm 0.078}$	$\mathbf{0.071 \pm 0.019}$

Table 1: Performance (MSE) comparison on the sinusoid regression problem. "+ours" means integrating MetaCRL into the existing methods, and the best results are highlighted in **bold**.

Model	Omniglot	miniImagenet	TC
IFSL	88.51 ± 0.49	36.21 ± 1.62	\
Meta-Trans	87.39 ± 0.51	35.19 ± 1.58	\
Meta-Aug	89.77 ± 0.62	34.76 ± 1.52	\
MR-MAML	89.28 ± 0.59	35.01 ± 1.60	\
MAML	87.15 ± 0.61	33.16 ± 1.70	0.00
MAML + MetaMix	91.97 ± 0.51	38.97 ± 1.81	+0.42
MAML + Dropout-Bins	92.89 ± 0.46	39.66 ± 1.74	-0.14
MAML + Ours	$\mathbf{93.00 \pm 0.42}$	$\mathbf{41.55 \pm 1.76}$	**+4.12**
ANIL	89.17 ± 0.56	34.96 ± 1.71	0.00
ANIL + MetaMix	92.88 ± 0.51	37.82 ± 1.75	-0.10
ANIL + Dropout-Bins	92.82 ± 0.49	38.09 ± 1.76	+0.97
ANIL + Ours	$\mathbf{92.91 \pm 0.52}$	$\mathbf{38.55 \pm 1.81}$	**+3.56**
MetaSGD	87.81 ± 0.61	33.97 ± 0.92	0.00
MetaSGD + MetaMix	93.44 ± 0.45	40.28 ± 0.96	+0.05
MetaSGD + Dropout-Bins	93.93 ± 0.40	40.31 ± 0.96	+1.08
MetaSGD + Ours	$\mathbf{94.12 \pm 0.43}$	$\mathbf{41.22 \pm 0.93}$	**+6.19**
T-NET	87.66 ± 0.59	33.69 ± 1.72	0.00
T-NET + MetaMix	93.16 ± 0.48	39.18 ± 1.73	+0.28
T-NET + Dropout-Bins	93.54 ± 0.49	39.06 ± 1.72	+1.03
T-NET + Ours	$\mathbf{93.81 \pm 0.52}$	$\mathbf{40.08 \pm 1.74}$	**+4.65**

Table 2: Performance (accuracy $\pm$ 95% confidence interval) on (20-way 1-shot) Omniglot and (5-way 1-shot) miniImagenet. The "+" and "-" indicate the performance changes, and the "\" denotes that the result is not reported. See Appendix F for full results.

5.1 Sinusoid Regression

Firstly, we evaluate the performance of our MetaCRL on sinusoid regression. Following [Jiang *et al.*, 2022], we conduct 480 tasks and the data for each task is generated in the form of $A \sin w \cdot x + b + \epsilon$, where $A \in [0.1, 5.0]$, $w \in [0.5, 2.0]$, and $b \in [0, 2\pi]$. We add Gaussian observation noise with $\mu = 0$ and $\epsilon = 0.3$ to each data point sampled from the target task. In this experiment, we set λ_1 and λ_2 to 0.4 and 0.2. We use the Mean Squared Error (MSE) as the evaluation metric.

The results are shown in Table 1. Compared to the plug-and-play baselines, MetaCRL achieves improvements with an average MSE reduction of 0.034 and 0.013, respectively. MetaCRL also demonstrates significant improvements across all the meta-learning base models, with an MSE reduction of over 0.1. Compared to the causal-based baselines, adding MetaCRL to any meta-learning model can always achieve better performance. As expected, MetaCRL exhibits significant enhancements, showcasing its high compatibility.

5.2 Image Classification

Next, we conduct experiments on image classification, utilizing two benchmark datasets, i.e., miniImagenet and Omniglot. These two datasets contain 600 and 1623 tasks, respectively. We also introduce a specialized dataset called "TC", which comprises 50 groups of tasks (300 tasks in total) identified as being affected by task confounders, i.e., tasks with negative knowledge transfer as mentioned in Subsection 3.2. More details are provided in Appendix C. In this experiment, we set λ_1 and λ_2 to 0.5 and 0.35, respectively. The evaluation metric employed here is the average accuracy.

The results are shown in Table 2. MetaCRL consistently surpasses the SOTA baselines across all datasets, indicating that it can achieve better generalization improvements than the baselines do without the need for task-specific or general-label space augmentation that the baselines need. Notably, on the "TC" dataset, MetaCRL outperforms the baselines by a significant margin, which demonstrates a unique advantage of MetaCRL in handling task confounders. In summary, MetaCRL continues to exhibit remarkable performance and adeptly eliminates task confounders.

5.3 Drug Activity Prediction

We also evaluate MetaCRL on drug activity prediction. pQSAR [Martin *et al.*, 2019] is a dataset designed to forecast the activity of compounds on specific target proteins, encompassing a total of 4276 tasks. We adopt the same settings as [Yao *et al.*, 2021] and divide the tasks into four groups. In this experiment, λ_1 and λ_2 are both set to 0.3, and the evaluation metric is the squared Pearson correlation coefficient (R^2), reflecting the correlation between predictions and the actual values for each task. We record both the mean and median R^2 values, along with the count of R^2 values exceeding 0.3, which stands as a reliable indicator in pharmacology.

The results are shown in Table 3. MetaCRL attains performance levels akin to the SOTA baselines across all four groups of data. Notably, we achieve a noteworthy enhancement of 3 in the reliability index $R^2 > 0.3$. The achievement of this scenario underscores the effectiveness of our MetaCRL across disparate domains and the pervasive influence of task confounders. See Appendix F for full results.

5.4 Pose Prediction

Lastly, we undertake the fourth benchmark, focusing on pose prediction. This evaluation is constructed using the Pascal 3D dataset [Xiang *et al.*, 2014]. We randomly select 50 objects for meta-training and 15 additional objects for meta-testing. In this experiment, the values of λ_1 and λ_2 are set to 0.3 and 0.2, while the evaluation metric employed here is MSE.

The results are shown in Table 4. MetaCRL achieves the best performance. Notably, drawing insights from the findings presented in [Yao *et al.*, 2021], we posit that augment-

Model	Group 1			Group 2			Group 3			Group 4		
	Mean	**Med.**	**> 0.3**	**Mean**	**Med.**	**> 0.3**	**Mean**	**Med.**	**> 0.3**	**Mean**	**Med.**	**> 0.3**
MAML	0.371	0.315	52	0.321	0.254	43	0.318	0.239	44	0.348	0.281	47
MAML + Dropout-Bins	0.410	0.376	60	0.355	0.257	48	0.320	0.275	46	0.370	0.337	56
MAML + Ours	**0.413**	**0.378**	**61**	**0.360**	**0.261**	**50**	**0.334**	**0.282**	**51**	**0.375**	**0.341**	**59**
ANIL	0.355	0.296	50	0.318	**0.297**	**49**	0.304	0.247	46	0.338	0.301	50
ANIL + MetaMix	0.347	0.292	49	0.302	0.258	45	0.301	0.282	47	0.348	0.303	51
ANIL + Dropout-Bins	0.394	0.321	53	0.338	0.271	48	**0.312**	0.284	46	0.368	0.297	50
ANIL + Ours	**0.401**	**0.339**	**57**	**0.341**	0.277	**49**	**0.312**	**0.291**	**48**	**0.371**	**0.305**	**53**

Table 3: Performance comparison on drug activity prediction. "Mean", "Med.", and "> 0.3" are the mean, the median value of R^2, and the number of analyzes for $R^2 > 0.3$. The best results are highlighted in **bold**.

Model	10-shot	15-shot
MAML	3.113 ± 0.241	2.496 ± 0.182
MAML + MetaMix	2.429 ± 0.198	1.987 ± 0.151
MAML + Dropout-Bins	2.396 ± 0.209	1.961 ± 0.134
MAML + Ours	$\mathbf{2.355 \pm 0.200}$	$\mathbf{1.931 \pm 0.134}$
MetaSGD	2.811 ± 0.239	2.017 ± 0.182
MetaSGD + MetaMix	2.388 ± 0.204	1.952 ± 0.134
MetaSGD + Dropout-Bins	2.369 ± 0.217	1.927 ± 0.120
MetaSGD + Ours	$\mathbf{2.362 \pm 0.196}$	$\mathbf{1.920 \pm 0.191}$
T-NET	2.841 ± 0.177	2.712 ± 0.225
T-NET + MetaMix	2.562 ± 0.280	2.410 ± 0.192
T-NET + Dropout-Bins	2.487 ± 0.212	2.402 ± 0.178
T-NET + Ours	$\mathbf{2.481 \pm 0.274}$	$\mathbf{2.400 \pm 0.171}$

Table 4: Performance (MSE $\pm$ 95% confidence interval) comparison on pose prediction. More results are provided in Appendix F.

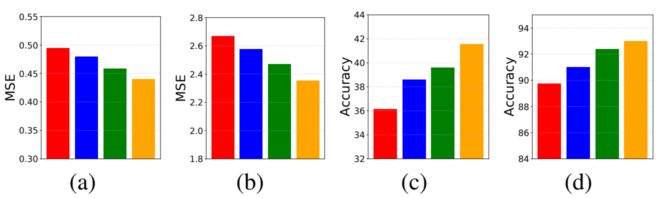


Figure 3: Ablation study, including (a) sinusoid regression, (b) pose prediction, (c) 5-way 1-shot miniImagenet, and (d) 20-way 1-shot Omniglot. The backbone is MAML. The red, blue, green, and orange bars represent the results of MetaCRL-$\mathcal{L}_{\mathrm{DM}}(f_{gr}, \Xi)$, MetaCRL-$\mathcal{L}_{\mathrm{DM}}(\Xi)$, MetaCRL-$\mathcal{L}_{\mathrm{DM}}(f_{gr})$, and MetaCRL.

ing the dataset could yield more effective results in this scenario, potentially outperforming the reliance solely on meta-regularization techniques. MetaCRL incorporates regularization terms instead of data augmentation and still manages to achieve enhanced performance, thereby affirming its efficacy.

5.5 Ablation Study

We conduct ablation studies to explore the impact of different regularization terms, that is $\mathcal{L}_{\mathrm{DM}}(\Xi)$, $\mathcal{L}_{\mathrm{DM}}(f_{gr})$, and their combination $\mathcal{L}_{\mathrm{DM}}(f_{gr}, \Xi)$ in Eq.6. We select both classification and regression scenarios, including four benchmark datasets. Figure 3 shows the results that $\mathcal{L}_{\mathrm{DM}}(\Xi)$ and $\mathcal{L}_{\mathrm{DM}}(f_{gr})$ promote the model in all datasets, and the improvement is the largest when combined. Moreover, despite eliminating the regularization terms, MetaCRL still significantly outperforms the base models, illustrating the effectiveness of the causal module. We also construct ablation studies targeting the accuracy of extracting task-specific causal factors and model efficiency (See Appendix F for details).

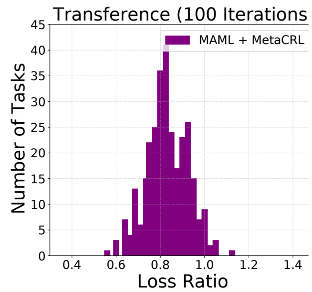


Figure 4: Knowledge transference after using MetaCRL.

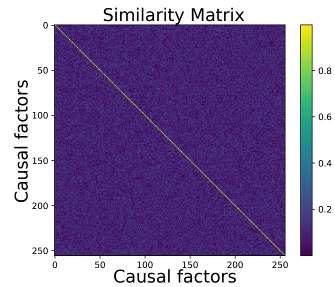


Figure 5: Visualization of the similarity between causal factors.

5.6 Visualization

To better evaluate the effect of MetaCRL, we visualize (i) knowledge transfer after using MetaCRL; and (ii) the similarity between causal factors. The former evaluates MetaCRL's efficacy in ensuring causality and avoiding negative knowledge transfer caused by task confounders, which use the same settings as in Subsection 3.2. The latter assesses the decoupling of causal factors using cosine similarity. Figures 4 and 5 show visualizations for these two aspects, respectively. Figure 4 shows that there are almost no training tasks that lead to negative knowledge transfer with fewer iterations than Figure 1, which indicates that MetaCRL effectively eliminates task confounders. Figure 5 shows that the similarity scores between different causal factors are very low, illustrating that the disentangling module successfully decouples causal factors. More details are provided in Appendix F.

6 Conclusion

In this paper, we discover a valuable problem called "Task Confounder", and propose a novel method called MetaCRL to address its unique challenges. We begin by analyzing a counterintuitive negative knowledge transfer phenomenon with SCM, revealing spurious correlations between causal factors of the training tasks and the label space, i.e., "Task Confounder". Then, we propose MetaCRL, which consists of two modules: (i) a disentangling module that acquires generating factors and eliminates task confounders; and (ii) a causal module that ensures causality of the obtained generating factors. It is a plug-and-play causal representation learner that can be applied to any meta-learning baseline. Extensive experiments demonstrate the effectiveness and robustness of MetaCRL. Our work uncovers a novel and significant issue in ML, providing valuable insights for future research.

Acknowledgements

The authors would like to thank the anonymous reviewers for their valuable comments. This work is supported in part by the Postdoctoral Fellowship Program of CPSF No. GZB20230790, the China Postdoctoral Science Foundation No. 2023M743639, and the Special Research Assistant Fund, Chinese Academy of Sciences No. E3YD590101. The Appendix is provided in https://arxiv.org/abs/2312.05771.

Contribution Statement

Jingyao Wang and Yi Ren made equal contributions. All the authors participated in designing research, performing research, analyzing data, and writing the paper.

References

[Abdollahzadeh *et al.*, 2021] Milad Abdollahzadeh, Touba Malekzadeh, and Ngai-Man Man Cheung. Revisit multimodal meta-learning through the lens of multi-task learning. *Advances in Neural Information Processing Systems*, 34:14632–14644, 2021.

[Arjovsky *et al.*, 2019] Martín Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *CoRR*, abs/1907.02893, 2019.

[Bengio *et al.*, 2019] Yoshua Bengio, Tristan Deleu, Nasim Rahaman, Rosemary Ke, Sébastien Lachapelle, Olexa Bilaniuk, Anirudh Goyal, and Christopher Pal. A meta-transfer objective for learning to disentangle causal mechanisms. *arXiv preprint arXiv:1901.10912*, 2019.

[Chen *et al.*, 2020] Jiaxin Chen, Li-Ming Zhan, Xiao-Ming Wu, and Fu-lai Chung. Variational metric scaling for metric-based meta-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 3478–3485, 2020.

[Fifty *et al.*, 2020] Christopher Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Measuring and harnessing transference in multi-task learning. *arXiv preprint arXiv:2010.15413*, 2020.

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

[Guo *et al.*, 2024] Huijie Guo, Ying Ba, Jie Hu, Lingyu Si, Wenwen Qiang, and Lei Shi. Self-supervised representation learning with meta comprehensive regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 1959–1967, 2024.

[Hu *et al.*, 2022] Ziniu Hu, Zhe Zhao, Xinyang Yi, Tiansheng Yao, Lichan Hong, Yizhou Sun, and Ed Chi. Improving multi-task generalization via regularizing spurious correlation. *Advances in Neural Information Processing Systems*, 35:11450–11466, 2022.

[Islam *et al.*, 2020] Md Amirul Islam, Sen Jia, and Neil DB Bruce. How much position information do convolutional neural networks encode? *arXiv preprint arXiv:2001.08248*, 2020.

[Jamal and Qi, 2019] Muhammad Abdullah Jamal and Guo-Jun Qi. Task agnostic meta-learning for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11719–11727, 2019.

[Jensen and Shen, 2004] Richard Jensen and Qiang Shen. Semantics-preserving dimensionality reduction: rough and fuzzy-rough-based approaches. *IEEE Transactions on knowledge and data engineering*, 16(12):1457–1471, 2004.

[Jiang *et al.*, 2022] 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.

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

[Lee and Choi, 2018] Yoonho Lee and Seungjin Choi. Gradient-based meta-learning with learned layerwise metric and subspace. In *International Conference on Machine Learning*, pages 2927–2936. PMLR, 2018.

[Lee *et al.*, 2020] Hae Beom Lee, Taewook Nam, Eunho Yang, and Sung Ju Hwang. Meta dropout: Learning to perturb latent features for generalization. 2020.

[Li *et al.*, 2017] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*, 2017.

[Li *et al.*, 2023] Ximan Li, Weihong Deng, Shan Li, and Yong Li. Compound expression recognition in-the-wild with au-assisted meta multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5734–5743, 2023.

[Martin *et al.*, 2019] Eric J Martin, Valery R Polyakov, Xiang-Wei Zhu, Li Tian, Prasenjit Mukherjee, and Xin Liu. All-assay-max2 pqsar: activity predictions as accurate as four-concentration ic50s for 8558 novartis assays. *Journal of chemical information and modeling*, 59(10):4450–4459, 2019.

[Nichol and Schulman, 2018] Alex Nichol and John Schulman. Reptile: a scalable metalearning algorithm. *arXiv preprint arXiv:1803.02999*, 2(3):4, 2018.

[Nogueira *et al.*, 2022] Ana Rita Nogueira, Andrea Pugnana, Salvatore Ruggieri, Dino Pedreschi, and João Gama. Methods and tools for causal discovery and causal inference. *Wiley interdisciplinary reviews: data mining and knowledge discovery*, 12(2):e1449, 2022.

[Oh *et al.*, 2020] Jaehoon Oh, Hyungjun Yoo, ChangHwan Kim, and Se-Young Yun. Boil: Towards representation change for few-shot learning. *arXiv preprint arXiv:2008.08882*, 2020.

[Qiang *et al.*, 2023] Wenwen Qiang, Jiangmeng Li, Bing Su, Jianlong Fu, Hui Xiong, and Ji-Rong Wen. Meta attention-generation network for cross-granularity few-shot learning. *International Journal of Computer Vision*, 131(5):1211–1233, 2023.

[Raghu *et al.*, 2019] Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of maml. *arXiv preprint arXiv:1909.09157*, 2019.

[Rajendran *et al.*, 2020] Janarthanan Rajendran, Alexander Irpan, and Eric Jang. Meta-learning requires meta-augmentation. *Advances in Neural Information Processing Systems*, 33:5705–5715, 2020.

[Rivolli *et al.*, 2022] Adriano Rivolli, Luís PF Garcia, Carlos Soares, Joaquin Vanschoren, and André CPLF de Carvalho. Meta-features for meta-learning. *Knowledge-Based Systems*, 240:108101, 2022.

[Schrum *et al.*, 2022] Mariah L Schrum, Erin Hedlund-Botti, Nina Moorman, and Matthew C Gombolay. Mind meld: Personalized meta-learning for robot-centric imitation learning. In *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 157–165. IEEE, 2022.

[Snell *et al.*, 2017] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

[Sung *et al.*, 2018] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1199–1208, 2018.

[Suter *et al.*, 2019] Raphael Suter, Djordje Miladinovic, Bernhard Schölkopf, and Stefan Bauer. Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness. In *International Conference on Machine Learning*, pages 6056–6065. PMLR, 2019.

[Ton *et al.*, 2021] Jean-François Ton, Dino Sejdinovic, and Kenji Fukumizu. Meta learning for causal direction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9897–9905, 2021.

[Vinyals *et al.*, 2016] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

[Wang *et al.*, 2021] Haoxiang Wang, Han Zhao, and Bo Li. Bridging multi-task learning and meta-learning: Towards efficient training and effective adaptation. In *International conference on machine learning*, pages 10991–11002. PMLR, 2021.

[Wang *et al.*, 2023] Jingyao Wang, Chuyuan Zhang, Ye Ding, and Yuxuan Yang. Awesome-meta+: Meta-learning research and learning platform. *arXiv preprint arXiv:2304.12921*, 2023.

[Xiang *et al.*, 2014] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE winter conference on applications of computer vision*, pages 75–82. IEEE, 2014.

[Yang *et al.*, 2021] Xu Yang, Hanwang Zhang, and Jianfei Cai. Deconfounded image captioning: A causal retrospect. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[Yao *et al.*, 2021] Huaxiu Yao, Long-Kai Huang, Linjun Zhang, Ying Wei, Li Tian, James Zou, Junzhou Huang, et al. Improving generalization in meta-learning via task augmentation. In *International conference on machine learning*, pages 11887–11897. PMLR, 2021.

[Yin *et al.*, 2019] Mingzhang Yin, George Tucker, Mingyuan Zhou, Sergey Levine, and Chelsea Finn. Meta-learning without memorization. *arXiv preprint arXiv:1912.03820*, 2019.

[Yue *et al.*, 2020] Zhongqi Yue, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Interventional few-shot learning. *Advances in neural information processing systems*, 33:2734–2746, 2020.

[Zhang *et al.*, 2020] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. *Advances in Neural Information Processing Systems*, 33:655–666, 2020.

[Zimmermann *et al.*, 2021] Roland S Zimmermann, Yash Sharma, Steffen Schneider, Matthias Bethge, and Wieland Brendel. Contrastive learning inverts the data generating process. In *International Conference on Machine Learning*, pages 12979–12990. PMLR, 2021.