Robust Fast Adaptation from Adversarially Explicit Task Distribution Generation

Qi (Cheems) Wang[†*]
Tsinghua University
Beijing, China

Yiqin Lv[*]
Tsinghua University
Beijing, China

Yixiu Mao[*]
Tsinghua University
Beijing, China

Yun Qu
Tsinghua University
Beijing, China

Yi Xu
Dalian University of Technology
Dalian, China

Xiangyang Ji[†]
Tsinghua University
Beijing, China

Abstract

Meta-learning is a practical learning paradigm to transfer skills across tasks from a few examples. Nevertheless, the existence of *task distribution shifts* tends to weaken meta-learners' generalization capability, particularly when the training task distribution is naively hand-crafted or based on simple priors that fail to cover critical scenarios sufficiently. Here, we consider explicitly generative modeling task distributions placed over task identifiers and propose robustifying fast adaptation from adversarial training. Our approach, which can be interpreted as a model of a Stackelberg game, not only uncovers the task structure during problem-solving from an explicit generative model but also theoretically increases the adaptation robustness in worst cases. This work has practical implications, particularly in dealing with task distribution shifts in meta-learning, and contributes to theoretical insights in the field. Our method demonstrates its robustness in the presence of task subpopulation shifts and improved performance over SOTA baselines in extensive experiments. The code is available at the project site (https://sites.google.com/view/ar-metalearn).

CCS Concepts

• **Computing methodologies** → **Machine learning**.

Keywords

Meta Learning, Generative Models, Game Theory

ACM Reference Format:
Qi (Cheems) Wang[†], Yiqin Lv, Yixiu Mao, Yun Qu, Yi Xu, and Xiangyang Ji. 2025. Robust Fast Adaptation from Adversarially Explicit Task Distribution Generation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1 (KDD '25), August 3–7, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3690624.3709337

[*]These authors contributed equally to this research.
[†]Correspondence: cheemswang@mail.tsinghua.edu.cn; xyji@tsinghua.edu.cn

KDD '25, Toronto, ON, Canada

ACM ISBN 979-8-4007-1245-6/25/08
https://doi.org/10.1145/3690624.3709337

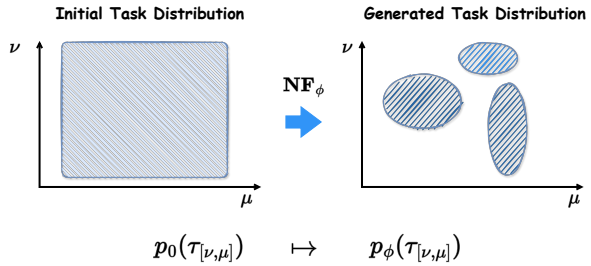

Figure 1: Diagram of Generating Task Distribution as the Adversary in Meta-Learning. Here, the initial task distribution $p_0(\tau)$ is a uniform distribution governed by two task identifiers $[\nu, \mu]$. Then, it is transformed into an explicit distribution $p_{\phi}(\tau)$ with the help of normalizing flows $\texttt{NF}_{\phi}$.

1 Introduction

Deep learning has made remarkable progress in the past decade, ranging from academics to industry [34]. However, training deep learning models is generally time-consuming, and the previously trained model on one task might perform poorly in deployment when faced with unseen scenarios [37].

Fortunately, meta-learning, or learning to learn, offers a scheme to generalize learned knowledge to unseen scenarios [11, 15, 24, 25]. The strategy is to leverage past experience, extract meta knowledge as the prior, and utilize a few shot examples to transfer skills across tasks. This way, we can avoid learning from scratch and quickly adapt the model to unseen but similar tasks, catering to practical demands, such as fast autonomous driving in diverse scenarios. Due to these desirable properties, such a learning paradigm is playing an increasingly crucial role in building foundation models [3, 29, 44, 70].

Literature Challenges: Despite the promising adaptation performance in meta-learning, several concerns remain. Among them, the automatically task distribution design is under-explored and challenging in the field, which closely relates to the model's generalization evaluation [10, 84].

Overall, task identifiers configure the task, such as the topic type in the corpus for large language models [5, 70], the amplitude and phase in sinusoid functions, or the degree of freedom in robotic manipulators [2, 13]. Most existing studies adopt simple prior, such as uniform distributions over task identifiers [15, 17, 56], or hand-crafted distributions, which heavily rely on domain-specific knowledge difficult to acquire.

Some scenarios even pose more realistic demands for task distributions. In testing an autopilot system, an ideal task distribution

deserves more attention on traffic accidents or even generates some while covering typical cases [53, 62]. Similar circumstances also occur during domain randomization for embodied robots [43]. These imply that the shift between commonly used task distributions, such as uniform, and the expected testing distributions raises robustness issues and probably causes catastrophic failures when adapting to risk-sensitive scenarios [42].

Proposed Solutions: Rather than exploring fast adaptation strategies, we turn to *explicitly create task distribution shifts at a certain level and characterize robust fast adaptation with a Stackelberg game* [49]. To this end, we utilize normalizing flows to parameterize the distribution adversary in Figure 1 for task distribution generation and the meta learner for fast adaptation in the presence of distribution shifts.

Importantly, we constitute the solution concept, adopt the alternative gradient descent ascent to approximately compute the equilibrium [32], and conduct theoretical analysis. The optimization process can be translated as *fast adaptation robustification through adversarially explicit task distribution generation.*

Outline & Primary Contributions: The remainder starts with related work in Section 2. We define the notation and recap fundamentals in Section 3. Then, we present the game-theoretical framework to handle constrained task distribution shifts and robustify fast adaptation in Section 4. The quantitative analysis is conducted in Section 5, followed by conclusions and limitations. In primary, our contributions are:

- This work translates the robust fast adaptation under distribution shifts into a Stackelberg game [67]. To reveal task structures during problem-solving, we explicitly generate the task distribution with normalizing flows over task identifiers and optimize the meta-learner in an adversarial way.
- In theoretical analysis and tractable optimization, we constitute the solution concept *w.r.t.* fast adaptation, approximately solve the game using alternating stochastic gradient descent, and perform convergence and generalization analysis under certain conditions.

Extensive experimental results show that our approach can reveal adaptation-related structures in the task space and achieve robustness improvement in task subpopulation shifts.

2 Literature Review

The past few years have developed a large body of work on skill transfer across tasks or domain generalization in different ways [25, 78–80]. This section overviews the field regarding meta-learning and adaptation robustness.

Meta Learning. Meta learning is a learning paradigm that considers a distribution over tasks. The key is to pursue strategies for leveraging past experiences and distilling extracted knowledge into unseen tasks with a few shots of examples [8, 25, 45]. Currently, there are various families of meta-learning methods. The optimization-based ones, like model agnostic meta-learning (MAML) [15] and its extensions [12, 22, 52, 68], aim at finding a good meta-initialization of model parameters for adapting to all tasks via gradient descent. The deep metrics methods optimize the task representation in a metric space and are superior in few-shot image classification tasks [1, 26, 38, 60, 81]. Typical context-based methods, e.g., neural processes (NPs) and variants [17, 18, 21, 30, 54, 59, 69, 72–74], constitute the deep latent variable model as the stochastic process to accomplish tasks. Besides, memory-augmented networks [58], hyper-networks [23], and so forth are designed for meta-learning purposes.

Robustness in Meta Learning. In most previous work, the task distribution is fixed in the training set-up. In order to robustify the fast adaptation performance, a couple of learning strategies or principles emerge. Increasing the robustness to worst cases is a commonly seen consideration in adaptation, and these scenarios include input noise, parameter perturbation, and task distributions [7, 33, 39, 41, 48, 64, 85]. To alleviate the effects of adversarial examples in few-shot image classification, Goldblum et al. [19] meta-train the model in an adversarial way. To handle the distribution mismatch between training and testing tasks, Zhang et al. [86] adopt the adaptive risk minimization principle to enable fast adaptation. Wang et al. [71] propose to optimize the expected tail risk in meta-learning and witness the increase of robustness in proportional worst cases. Ours is a variant of a distributionally robust framework [75], and we seek equilibrium for fast adaptation.

Task Distribution Studies in Meta Learning. Task distributions are directly related to the generalization capability of meta-learning models, attracting increasing attention recently. Aiming to alleviate task overfitting, Murty et al. [46], Ni et al. [47], Rajendran et al. [51], Yao et al. [82] enrich the task space with augmentation techniques. Task relatedness can improve generalization across tasks, Fifty et al. [14] devise an efficient strategy to group tasks in multi-task training. In [40, 83], neural task samplers are developed to schedule the probability of task sampling in the context of few-shot classification. To increase the fidelity of generated tasks, Wu et al. [77] adopt the task representation model and constructs the up-sampling network for meta-training task augmentation. To reduce the required tasks, [36, 83] take the task interpolation strategy and shows that the interpolation strategy outperforms the standard set-up. Distinguished from the above, this work takes more interest in explicitly understanding task identifier structures concerning learning performance and cares about fast adaptation robustness under subpopulation shift constraints. Optimizing the task distribution might reserve the potential to improve generative performance in large models [6].

3 Preliminaries

Notation. Throughout this paper, we use $p(\tau)$ to denote the task distribution with $\mathcal{T}$ the task domain. Here, $\mathcal{D}_\tau$ represents the meta dataset with a sampled task τ. With the model parameter domain $\boldsymbol{\Theta}$ and the support/query dataset construction, e.g., $\mathcal{D}_\tau = \mathcal{D}_\tau^S \cup \mathcal{D}_\tau^Q$, the risk function in meta-learning is a real-value function $\mathcal{L} : \mathcal{T} \times \boldsymbol{\Theta} \mapsto \mathbb{R}$.

As an example, $\mathcal{D}_\tau$ consists of data points $\{(x_i, y_i)\}_{i=1}^{m+n}$ in few shot regression, and it is mostly split into the support dataset $\mathcal{D}_\tau^S$ for fast adaptation and query dataset $\mathcal{D}_\tau^Q$ for evaluation.

3.1 Problem Statement

To begin with, we revisit a couple of commonly-used risk minimization principles for meta-learning as follows.

Standard Meta-Learning Optimization Objective. We consider the meta-learning problem within the expected risk minimization principle in the statistical learning theory [66]. This results in the objective as Eq. (1), and we execute optimization in the form of task batches in implementation.

$$\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \mathbb{E}_{p(\tau)}\Big[\mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta})\Big] \tag{1}$$

Here, $\boldsymbol{\theta}$ refers to the meta-learning model parameters for meta knowledge and fast adaptation. The risk function depends on specific meta-learning methods. For example, in MAML, the form can be $\mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta}) := \mathcal{L}(\mathcal{D}_\tau^Q; \boldsymbol{\theta} - \lambda\nabla_{\boldsymbol{\theta}}\mathcal{L}(\mathcal{D}_\tau^S; \boldsymbol{\theta}))$ in regression, where the gradient update with the learning rate λ in the bracket reflects fast adaptation.

Distributionally Robust Meta Learning Optimization Objective. Recently, tail risk minimization has been adopted for meta-learning, effectively alleviating the effects towards fast adaptation in task distribution shifts [71]. In detail, we can express the optimization objective as Eq. (2) in the presence of the constrained distribution $p_\alpha(\tau; \boldsymbol{\theta})$, which characterizes the $(1-\alpha)$ proportional $\boldsymbol{\theta}$-dependent worst cases in the task space.

$$\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \mathbb{E}_{p_\alpha(\tau;\boldsymbol{\theta})}\Big[\mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta})\Big] \tag{2}$$

It is worth noting that $p_\alpha(\tau; \boldsymbol{\theta})$ is non-differentiable and θ-dependent with no closed-form. Meanwhile, the worst-case optimization for meta-learning in Eq. (3) can be treated as a particular instance of Eq. (2) when α sufficiently approaches 1.

$$\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \max_{\tau\in\mathcal{T}} \mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta}) \tag{3}$$

Through tail risk minimization, the model's adaptation robustness can be enhanced *w.r.t.* the proportional worst scenarios [9, 71].

3.2 Two-Player Stackelberg Game

Before detailing our approach, it is necessary to describe elements in a two-player, non-cooperative Stackelberg game [67].

Let us assume two competitive players are involved in the game $\Gamma := \langle\{\mathcal{P}_1, \mathcal{P}_2\}, \{\boldsymbol{\theta}\in\boldsymbol{\Theta}, \boldsymbol{\phi}\in\boldsymbol{\Phi}\}, \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi})\rangle$, where the meta learner as the leader $\mathcal{P}_1$ makes a decision first in the domain $\boldsymbol{\Theta}$ while the distribution adversary as the follower $\mathcal{P}_2$ tries to deteriorate the leader decision's utility in the domain $\boldsymbol{\Phi}$. We refer to $\mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi})$ as the continuous risk function of the leader $\mathcal{P}_1$, and that of the follower $\mathcal{P}_2$ corresponds to the negative form $-\mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi})$. Without loss of generality, all the players are rational and try to minimize risk functions in the game.

4 Task Robust Meta Learning under Distribution Shift Constraints

This section starts with the game-theoretic framework for meta-learning, followed by approximate optimization. Figure 2 shows a diagram of the constructed Stackelberg game. Then we perform theoretical analysis *w.r.t.* our approach.

4.1 Generate Task Distribution within A Game-Theoretic Framework

As part of an indispensable element in meta-learning, the task distribution is mostly set to be uniform or manually designed from the heuristics. Such a setup hardly identifies a subpopulation of tasks that are tough to resolve in practice and fails to handle task distribution shifts.

In contrast, this paper considers an explicit task distribution to capture along with the learning progress and then automatically creates task distribution shifts for the meta-learner to adapt robustly. Our framework can be categorized as curriculum learning [4], but there places a constraint over the distribution shift in optimization.

Adversarially Task Robust Optimization with Distribution Shift Constraints. Now, we translate the meta-learning problem, namely generative task distributions for robust adaptation, into a min-max optimization problem:

$$\begin{aligned}\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \max_{\boldsymbol{\phi}\in\boldsymbol{\Phi}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}) &:= \mathbb{E}_{p_{\boldsymbol{\phi}}(\tau)}\Big[\mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta})\Big], \\ &\text{s.t. } D_{KL}\Big[p_0(\tau) \parallel p_{\boldsymbol{\phi}}(\tau)\Big] \le \delta,\end{aligned} \tag{4}$$

where the constraint term defines the maximum distribution shift to tolerate in meta training.

Equivalently, we can rewrite the above optimization objective in the form of unconstrained one with the help of a Lagrange multiplier $\lambda \in \mathbb{R}^+$:

$$\begin{aligned}\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \max_{\boldsymbol{\phi}\in\boldsymbol{\Phi}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}) &:= \mathbb{E}_{p_{\boldsymbol{\phi}}(\tau)}\Big[\mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta})\Big] \\ &-\lambda\Big[D_{KL}\Big[p_0(\tau) \parallel p_{\boldsymbol{\phi}}(\tau)\Big] - \delta\Big].\end{aligned} \tag{5}$$

The above can be further simplified as:

$$\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \max_{\boldsymbol{\phi}\in\boldsymbol{\Phi}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}) := \mathbb{E}_{p_{\boldsymbol{\phi}}(\tau)}\Big[\mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta})\Big] + \lambda\mathbb{E}_{p_0(\tau)}\Big[\ln p_{\boldsymbol{\phi}}(\tau)\Big], \tag{6}$$

where the constant terms, e.g., $\lambda\delta \in \mathbb{R}^+$ and $\mathbb{E}_{p_0(\tau)}\Big[\ln p_0(\tau)\Big]$ are eliminated.

As previously mentioned, the role of the distribution adversary attempts to transform the initial task distribution into one that raises challenging task proposals with higher probability. Such a setup drives *the evolution of task distributions via adaptively shifting task sampling chance under constraints*, which can be more crucial for generalization across risky scenarios. The term $D_{KL}\Big[p_0(\tau) \parallel p_{\boldsymbol{\phi}}(\tau)\Big]$ inside Eq. (5) works as regularization to avoid the mode collapse in the generative task distribution. In Figure 2, the goal of the meta learner retains that of traditional meta-learning, while the distribution adversary continually generates the task distribution shifts along optimization processes.

Assumption 1 (Lipschitz Smoothness and Compactness). *The adversarially task robust meta-learning optimization objective $\mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi})$ is assumed to satisfy*

1. *$\mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi})$ with $\forall[\boldsymbol{\theta}, \boldsymbol{\phi}] \in \boldsymbol{\Theta}\times\boldsymbol{\Phi}$ belongs to the class of twice differentiable functions $\mathbb{C}^2$.*
2. *The norm of block terms inside Hesssian matrices $\nabla^2\mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi})$ is bounded, meaning that $\forall[\boldsymbol{\theta}, \boldsymbol{\phi}] \in \boldsymbol{\Theta}\times\boldsymbol{\Phi}$:*
$$\sup\{||\nabla^2_{\boldsymbol{\theta},\boldsymbol{\theta}}\mathcal{J}||, ||\nabla^2_{\boldsymbol{\theta},\boldsymbol{\phi}}\mathcal{J}||, ||\nabla^2_{\boldsymbol{\phi},\boldsymbol{\phi}}\mathcal{J}||\} \le L_{max}.$$
3. *The parameter spaces $\boldsymbol{\Theta} \subseteq \mathbb{R}^{d_1}$ and $\boldsymbol{\Phi} \subseteq \mathbb{R}^{d_2}$ are compact with d_1 and d_2 respectively dimensions of model parameters for two players.*

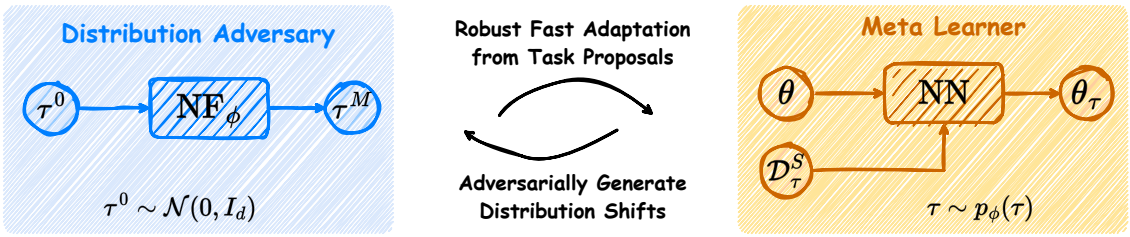

Figure 2: Diagram of Adversarially Task Robust Meta Learning. The proposed framework consists of two players, the distribution adversary and the meta player, in the game of meta-learning. On the left side of the figure: the distribution adversary seeks to transform the distribution from an initial task distribution, e.g., $\mathcal{N}(0, I_d)$ or $\mathcal{U}[a, b]$, via the neural network parameterized by ϕ with the purpose of deteriorating meta player's fast adaptation performance. On the right side of the figure: the meta player parameterized by θ attempts to learn robust strategies for fast adaptation in sampled worst-case tasks (MAML algorithm [15] as an illustration).

Example 1 (Adversarially Task Robust MAML, AR-MAML). *Given the parameterized task distribution $p_{\boldsymbol{\phi}}(\tau)$, the risk function $\mathcal{L}$ and the learning rate γ in the inner loop of MAML [15], the adversarially task robust MAML corresponds to the following optimization problem:*

$$\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \max_{\boldsymbol{\phi}\in\boldsymbol{\Phi}} \mathbb{E}_{p_{\boldsymbol{\phi}}(\tau)} \Big[\mathcal{L}\left(\mathcal{D}_\tau^Q; \boldsymbol{\theta} - \gamma \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathcal{D}_\tau^S; \boldsymbol{\theta})\right) \Big] + \lambda \mathbb{E}_{p_0(\tau)} \Big[\ln p_{\boldsymbol{\phi}}(\tau) \Big] \tag{7}$$

where $\mathcal{D}_\tau^S$ is used for the inner loop with $\mathcal{D}_\tau^Q$ used for the outer loop.

Example 2 (Adversarially Task Robust CNP, AR-CNP). *Given the parameterized task distribution $p_{\boldsymbol{\phi}}(\tau)$, the risk function $\mathcal{L}$, and the conditional neural process [17], the adversarially task robust CNP can be formulated as follows:*

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \max_{\boldsymbol{\phi}\in\boldsymbol{\Phi}} \mathbb{E}_{p_{\boldsymbol{\phi}}(\tau)} \Big[\mathcal{L}(\mathcal{D}_\tau^Q; z, \boldsymbol{\theta}_2) \Big] + \lambda \mathbb{E}_{p_0(\tau)} \Big[\ln p_{\boldsymbol{\phi}}(\tau) \Big], \\ \text{s.t. } z = h_{\boldsymbol{\theta}_1}(\mathcal{D}_\tau^S) \text{ with } \boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2\}, \end{aligned} \tag{8}$$

where $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ are respectively a set encoder and the decoder networks.

Here, we take two typical methods, e.g., MAML [15] and CNP [17], to illustrate the meta learner within the adversarially task robust framework, see Examples 1/2 for details.

Explicit Task Distribution Adversary Construction with Normalizing Flows. Learning to transform the task distribution is treated as a generative process: $\Phi : \mathcal{T} \to \mathcal{T} \subseteq \mathbb{R}^d$ in this paper. Admittedly, there already exist a collection of generative models to achieve the goal of generating task distributions, e.g., variational autoencoders [31, 55], generative adversarial networks [20], and normalizing flows [55].

Among them, we propose to utilize the normalizing flow [55] to achieve *due to its tractability of the exact log-likelihood, flexibility in capturing complicated distributions, and a direct understanding of task structures.* The basic idea of normalizing flows is to transform a simple distribution into a more flexible distribution with a series of invertible mappings $\mathcal{G} = \{g_i\}_{i=1}^M$, where $g_i : \mathcal{T} \to \mathcal{T} \subseteq \mathbb{R}^d$ indicates the smooth invertible mapping. We refer to these mappings implemented in the neural networks as $\mathtt{NN}_{\boldsymbol{\phi}}$ afterward. Specifically, with the base distribution $p_0(\tau)$ and a task sample τ^0, the model applies the above mappings to τ^0 to obtain τ^M.

$$\tau^M = g_M \circ \ldots g_2 \circ g_1(\tau^0) = \mathtt{NN}_{\boldsymbol{\phi}}(\tau^0) \tag{9}$$

In this way, the task distribution of interest is adaptive and adversarially exploits information from the shifted task distributions. The density function after transformations can be easily computed with the help of functions' Jacobians:

$$\ln p_{\boldsymbol{\phi}}(\tau^M) = \ln p_0(\tau^0) - \sum_{i=1}^{M} \ln \left| \det \frac{\partial g_i}{\partial \tau^{i-1}} \right|. \tag{10}$$

Definition 1 ((ℓ_1, ℓ_2)-bi-Lipschitz Function). *An invertible function $g : x \subseteq \mathcal{X} \mapsto x \subseteq \mathcal{X}$, is said to be (ℓ_1, ℓ_2)-bi-Lipschitz if $\forall \{x_1, x_2\} \in \mathcal{X}$, the following conditions hold:*

$$|g(x_1) - g(x_2)| \leq \ell_2 |x_1 - x_2| \quad \text{and} \quad |g^{-1}(x_1) - g^{-1}(x_2)| \leq \ell_1 |x_1 - x_2|.$$

As the normalizing flow function is invertible, the **Definition** 1 is to describe the Lipschitz continuity in bi-directions.

4.2 Solution Concept & Explanations

This work separates players regarding the decision-making order, and the optimization procedure is no longer a simultaneous game. The nature of Stackelberg game enables us to technically express the studied asymmetric bi-level optimization problem as:

$$\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}), \ \text{s.t. } \boldsymbol{\phi} \in \mathcal{S}(\boldsymbol{\theta}) \tag{11}$$

with the $\boldsymbol{\theta}$-dependent conditional subset $\mathcal{S}(\boldsymbol{\theta}) := \{\boldsymbol{\phi} \in \boldsymbol{\Phi} | \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}) \geq \max_{\boldsymbol{\phi}\in\boldsymbol{\Phi}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi})\}$. This suggests the variables $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ are entangled in optimization.

Moreover, we can define the resulting equilibrium as a local minimax point [27] in adversarially task robust meta-learning, due to the non-convex optimization practice.

Definition 2 (Local Minimax Point). *The solution $\{\boldsymbol{\theta}_*, \boldsymbol{\phi}_*\}$ is called local Stackelberg equilibrium when satisfying two conditions: (1) $\boldsymbol{\phi}_* \in \boldsymbol{\Phi}' \subset \boldsymbol{\Phi}$ is the maximum of the function $\mathcal{J}(\boldsymbol{\theta}_*, \cdot)$ with $\boldsymbol{\Phi}'$ a neighborhood; (2) $\boldsymbol{\theta}_* \in \boldsymbol{\Theta}' \subset \boldsymbol{\Theta}$ is the minimum of the function $\mathcal{J}(\boldsymbol{\theta}, g(\boldsymbol{\theta}))$ with $g(\boldsymbol{\theta})$ the implicit function of $\nabla_{\boldsymbol{\phi}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}) = 0$ in the neighborhood $\boldsymbol{\Theta}'$.*

Moreover, there exists a clearer interpretation *w.r.t.* the sequential optimization process and the equilibrium in the **Definition** 2. The meta learner as the leader first optimizes its parameter $\boldsymbol{\theta}$. Then

the distribution adversary as the follower updates the parameter $\boldsymbol{\phi}$ and explicitly generates the task distribution proposal to challenge adaptation performance. In other words, we expect that meta learners can benefit from generative task distribution shifts regarding the adaptation robustness.

Remark 1 (Entropy of the Generated Task Distribution). *Given the generative task distribution $p_{\boldsymbol{\phi}_*}(\tau)$, we can derive its entropy from the initial task distribution $p_0(\tau)$ and normalizing flows $\mathcal{G} = \{g_i\}_{i=1}^M$:*

$$\mathbb{H}\Big[p_{\boldsymbol{\phi}_*}(\tau)\Big] = \mathbb{H}\Big[p_0(\tau)\Big] + \int p_0(\tau)\left[\sum_{i=1}^{M} \ln\left|\det \frac{\partial g_i}{\partial \tau^{i-1}}\right|\right] d\tau. \quad (12)$$

The above implies that the generated task distribution entropy is governed by the change of task identifiers in the probability measure of the task space.

4.3 Strategies for Finding Equilibrium

Given the previously formulated optimization objective, we propose to approach it with the help of estimated stochastic gradients. As noticed, the involvement of adaptive expectation term $p_{\boldsymbol{\phi}}(\tau)$ requires extra considerations in optimization.

Best Response Approximation. Given two players with completely distinguished purposes, the commonly used strategy to compute the equilibrium is the Best Response (BR), which means:

$$\boldsymbol{\theta}_{t+1} = \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}_t) \quad (13a)$$

$$\boldsymbol{\phi}_{t+1} = \arg\max_{\boldsymbol{\phi}\in\boldsymbol{\Phi}} \mathcal{J}(\boldsymbol{\theta}_{t+1}, \boldsymbol{\phi}). \quad (13b)$$

For implementation convenience, we instead apply the gradient updates to the meta player and the distribution adversary, namely stochastic alternating gradient descent ascent (GDA). The operations are entangled and result in the following iterative equations with the index t:

$$\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t - \gamma_1 \nabla_{\boldsymbol{\theta}} \mathcal{J}(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t) \quad (14a)$$

$$\boldsymbol{\phi}_{t+1} \leftarrow \boldsymbol{\phi}_t + \gamma_2 \nabla_{\boldsymbol{\phi}} \mathcal{J}(\boldsymbol{\theta}_{t+1}, \boldsymbol{\phi}_t). \quad (14b)$$

This can be viewed as the gradient approximation for the BR strategy, which leads to at least a local Stackelberg equilibrium for the considered minimax problem [28].

Stochastic Gradient Estimates & Variance Reduction. Addressing the game-theoretic problem is non-trivial especially when it relates to distributions. A commonly-used method is to perform the sample average approximation *w.r.t.* Eq. (14). It iteratively updates the parameters of the meta player and the distribution adversary to approximate the saddle point.

More specifically, we can have the Monte Carlo estimates of the stochastic gradients for the leader $\mathcal{P}_1$:

$$\begin{aligned}\nabla_{\boldsymbol{\theta}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}) &= \int p_{\boldsymbol{\phi}}(\tau) \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta}) d\tau \\ &\approx \frac{1}{K}\sum_{k=1}^{K} \nabla_{\boldsymbol{\theta}} \mathcal{L}(D_{\tau_k}^Q, D_{\tau_k}^S; \boldsymbol{\theta}).\end{aligned} \quad (15)$$

The form of stochastic gradients *w.r.t.* the meta player parameter $\boldsymbol{\theta}$ is the meta-learning algorithm specific or model dependent. We refer the reader to Algorithm ??/?? as examples.

Now, we can derive the estimates with the help of REINFORCE algorithm [76] for the follower $\mathcal{P}_2$ and obtain the score function as:

$$\begin{aligned}\nabla_{\boldsymbol{\phi}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}) \approx \frac{1}{K}\sum_{k=1}^{K} \mathcal{L}(D_{\tau_k}^Q, D_{\tau_k}^S; \boldsymbol{\theta}) \nabla_{\boldsymbol{\phi}} \ln p_{\boldsymbol{\phi}}(\tau_k) \\ + \frac{\lambda}{K}\sum_{k=1}^{K} \nabla_{\boldsymbol{\phi}} \ln p_{\boldsymbol{\phi}}(\tau_k^{-M}),\end{aligned} \quad (16)$$

where the particle $\tau_k \sim p_{\boldsymbol{\phi}}(\tau)$ denotes the task sampled from the generative task distribution, and τ_k^{-M} means the particle sampled from the initial task distribution to enable $\texttt{NN}_{\boldsymbol{\phi}}(\tau_k^{-M}) = \tau_k$.

As validated in [16], the score estimator is an unbiased estimate of $\nabla_{\boldsymbol{\phi}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi})$. However, such a gradient estimator in Eq. (16) mostly exhibits higher variances, which weakens the stability of training processes. To reduce the variances, we utilize the commonly-used trick by including a constant baseline $\mathcal{V} = \mathbb{E}_{p_{\boldsymbol{\phi}}(\tau)}\Big[\mathcal{L}(\mathcal{D}_\tau^Q, \mathcal{D}_\tau^S; \boldsymbol{\theta})\Big] \approx \frac{1}{K}\sum_{k=1}^{K} \mathcal{L}(D_{\tau_k}^Q, D_{\tau_k}^S; \boldsymbol{\theta})$ for the score function, which results in:

$$\begin{aligned}\nabla_{\boldsymbol{\phi}} \mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\phi}) :\approx \frac{1}{K}\sum_{k=1}^{K} [\mathcal{L}(D_{\tau_k}^Q, D_{\tau_k}^S; \boldsymbol{\theta}) - \mathcal{V}] \nabla_{\boldsymbol{\phi}} \ln p_{\boldsymbol{\phi}}(\tau_k) \\ + \frac{\lambda}{K}\sum_{k=1}^{K} \nabla_{\boldsymbol{\phi}} \ln p_{\boldsymbol{\phi}}(\tau_k^{-M}).\end{aligned} \quad (17)$$

Particularly, since the normalizing flow works as the distribution transformation in this work, please refer to Eq. (10) to obtain the derivative of the log-likelihood of the transformed task $\ln p_{\boldsymbol{\phi}}(\tau)$ *w.r.t.* $\boldsymbol{\phi}$ inside Eq. (17). For easier analysis, we characterize the iteration sequence in optimization as $\begin{bmatrix}\boldsymbol{\theta}_0 \\ \boldsymbol{\phi}_0\end{bmatrix} \mapsto \cdots \mapsto \begin{bmatrix}\boldsymbol{\theta}_t \\ \boldsymbol{\phi}_t\end{bmatrix} \mapsto \begin{bmatrix}\boldsymbol{\theta}_{t+1} \\ \boldsymbol{\phi}_{t+1}\end{bmatrix} \mapsto \cdots$.

Remark 2 (Solution as a Fixed Point). *The alternating GDA for solving Eq. (5) results in the fixed point when $\begin{bmatrix}\boldsymbol{\theta}_{H+1} \\ \boldsymbol{\phi}_{H+1}\end{bmatrix} = \begin{bmatrix}\boldsymbol{\theta}_H \\ \boldsymbol{\phi}_H\end{bmatrix}$, or in other words $\begin{bmatrix}\boldsymbol{\theta}_H \\ \boldsymbol{\phi}_H\end{bmatrix}$ is stationary $\nabla \mathcal{J}(\boldsymbol{\theta}_H, \boldsymbol{\phi}_H) = 0$.*

4.4 Theoretical Analysis

Built on the deduction of the local Stackelberg equilibrium's existence and the **Remark** 2, we further perform analysis on the considered equilibrium $\begin{bmatrix}\boldsymbol{\theta}_* \\ \boldsymbol{\phi}_*\end{bmatrix}$, in terms of learning dynamics using the alternating GDA. For notation simplicity, we denote the block terms inside the Hessian matrix $\mathbf{H}_* := \nabla^2 \mathcal{J}(\boldsymbol{\theta}_*, \boldsymbol{\phi}_*)$ around $[\boldsymbol{\theta}_*, \boldsymbol{\phi}_*]^T$ as $\begin{bmatrix}\nabla^2_{\boldsymbol{\theta}\boldsymbol{\theta}}\mathcal{J} & \nabla^2_{\boldsymbol{\theta}\boldsymbol{\phi}}\mathcal{J} \\ \nabla^2_{\boldsymbol{\phi}\boldsymbol{\theta}}\mathcal{J} & \nabla^2_{\boldsymbol{\phi}\boldsymbol{\phi}}\mathcal{J}\end{bmatrix}\Big|_{[\boldsymbol{\theta}_*, \boldsymbol{\phi}_*]^T} := \begin{bmatrix}\mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C}\end{bmatrix}$.

Theorem 1 (Convergence Guarantee). *Suppose that the **Assumption** 1 and the function condition of the (local) Stackelberg equilibrium $\Delta(\mathbf{A}, \mathbf{B}, \mathbf{C}, \gamma_1, \gamma_2) < \frac{1}{2}$ are satisfied, where norms of the corresponding matrix are involved. Then the following statements hold:*

(1) *The resulting iterated parameters $\{\cdots \mapsto [\boldsymbol{\theta}_t, \boldsymbol{\phi}_t]^T \mapsto [\boldsymbol{\theta}_{t+1}, \boldsymbol{\phi}_{t+1}]^T \mapsto \cdots\}$ are Cauchy sequences;*

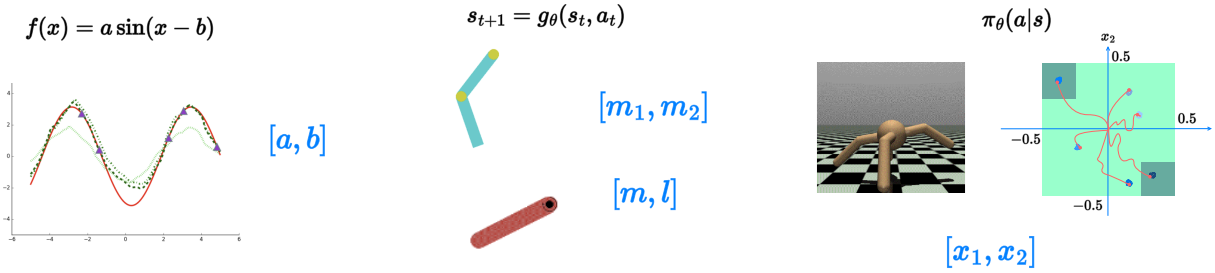

Figure 3: Some Benchmarks in Evaluation. Blue-marked variables in the illustration denote task identifiers that guide the configuration of a specific task. We place distributions over these task identifiers in generating diverse tasks for meta-learning.

(2) *The optimization can guarantee at least the linear convergence to the local Stackelberg equilibrium with the rate* $\sqrt{\Delta}$.

The **Theorem** 1 clarifies learning rates γ_1 and γ_2's influence on convergence and the required second-order derivative conditions of the resulting stationary point $[\boldsymbol{\theta}_*, \boldsymbol{\phi}_*]^T$. And when the game arrives at convergence, the local Stackelberg equilibrium is the best response to these two players, which is at least a local min-max solution to Eq. (5).

Next, we estimate the generalization bound of meta learners when confronting the generated task distribution shifts.

Theorem 2 (Generalization Bound with the Distribution Adversary). *Given the pretrained normalizing flows* $\{g_i\}_{i=1}^M$*, where* g_i *is* (ℓ_a, ℓ_b)*-bi-Lipschitz, and the pretrained meta learner* $\boldsymbol{\theta}_* \in \boldsymbol{\Theta}$*, we can derive the generalization bound with the initial task distribution* p *uniform:*

$$R_p^\omega(\boldsymbol{\theta}_*) \leq \hat{R}_p^\omega(\boldsymbol{\theta}_*) + \Upsilon(\mathcal{T}) \left(\frac{C \ln \frac{2Ke}{C} + \ln \frac{4}{\delta}}{K} \right)^{\frac{3}{8}}, \tag{18}$$

where $C = Pdim(\{\mathcal{L}(\cdot; \boldsymbol{\theta}) : \boldsymbol{\theta} \in \boldsymbol{\Theta}\})$ *denotes the pseudo-dimension in [50],* $R_p^\omega(\boldsymbol{\theta}_*)$ *and* $\hat{R}_p^\omega(\boldsymbol{\theta}_*)$ *are expected and empirical risks.*

We refer the reader to Appendix F for formal **Theorem** 2 and proofs. It reveals the connection between the bound and task complexity $\Upsilon(\mathcal{T})$, and more training tasks from initial distributions decrease the generalization error in adversarially distribution shifts.

5 Experiments

Previous sections recast the adversarially task robust meta-learning to a Stackelberg game, specify the equilibrium, and analyze theoretical properties in distribution generation. This section focuses on the evaluation, and baselines constructed from typical risk minimization principles are reported in Appendix **Table** 3. These include vanilla MAML [15], DRO-MAML [57], TR-MAML [9], DR-MAML [71], and AR-MAML (ours).

Technically, we mainly answer the following **Research Questions** (**RQ**s):

(1) *Does adversarial training help improve few-shot adaptation robustness in case of task distribution shifts?*
(2) *How does the type of the initial task distribution influence the performance of resulting solutions?*
(3) *Can generative modeling the task distribution discover meaningful task structures and afford interpretability?*

Implementation & Examination Setup. As our approach is agnostic to meta-learning methods, we mainly employ AR-MAML as the implementation of this work. Concerning the meta testing distribution, tasks are from the initial task distribution and the adversarial task distribution, respectively. The latter corresponds to the generated task distribution under shift constraints after convergence.

Evaluation Metrics. Here, we use both the average risk and conditional value at risk (CVaR$_\alpha$) in evaluation metrics, where CVaR$_\alpha$ can be viewed as the worst group performance in [57].

5.1 Benchmarks

We consider the few-shot synthetic regression, system identification, and meta reinforcement learning to test fast adaptation robustness with typical baselines. Notably, the task is specified by the generated task identifiers as shown in Figure 3.

Synthetic Regression. The same as that in [15], we conduct experiments in sinusoid functions. The goal is to uncover the function $f(x) = a\sin(x-b)$ with K-shot randomly sampled function points. And the task identifiers are the amplitude a and phase b.

System Identification. Here, we take the Acrobot System [61] and the Pendulum System [35] to perform system identification. In the Acrobot System, we generate different dynamical systems as tasks by varying masses of two pendulums. And the task identifiers are the pendulum mass parameters m_1 and m_2. In the Pendulum System, the system dynamics are distinguished by varying the mass and the length of the pendulum. And the task identifiers are the mass parameter m and the length parameter l. For both benchmarks, we collect the dataset of state transitions with a complete random policy to interact with sampled environments. The goal is to predict state transitions conditioned on randomly sampled context transitions from an unknown dynamical system.

Meta Reinforcement Learning. We evaluate the role of task distributions in meta-learning continuous control. In detail, the Point Robot in [15] and the Ant-Pos Robot in Mujoco [65] are included as navigation environments. We respectively vary goal/position locations as task identifiers within a designed range to generate diverse tasks. The goal is to seek a policy that guides the robot to the target location with a few episodes derived from an environment.

We refer the reader to Appendix I for set-ups, hyper-parameter configurations and additional experimental results.

Table 1: Average mean square errors in 5-shot sinusoid regression/10-shot Acrobot system identification/10-shot Pendulum system identification with reported standard deviations (5 runs). With $\alpha = 0.5$, the best results are in pink (the lower, the better). U/N in benchmarks denote Uniform/Normal as the initial distribution type.

Benchmark	Meta-Test Distribution	Average					CVaR				
		MAML	TR-MAML	DR-MAML	DRO-MAML	AR-MAML	MAML	TR-MAML	DR-MAML	DRO-MAML	AR-MAML
Sinusoid-U	Initial	0.499±0.01	0.539±0.01	0.479±0.01	0.481±0.01	0.459±0.01	0.858±0.01	0.868±0.02	0.793±0.02	0.816±0.02	0.782±0.03
	Adversarial	0.508±0.01	0.548±0.01	0.499±0.01	0.502±0.02	0.405±0.01	0.883±0.02	0.879±0.02	0.836±0.01	0.826±0.03	0.671±0.01
Sinusoid-N	Initial	0.578±0.03	0.628±0.01	0.556±0.01	0.562±0.02	0.554±0.02	1.017±0.05	1.017±0.02	0.932±0.02	0.983±0.03	0.947±0.03
	Adversarial	0.496±0.01	0.511±0.01	0.492±0.02	0.493±0.01	0.404±0.02	0.838±0.03	0.827±0.02	0.807±0.03	0.835±0.01	0.672±0.03
Acrobot-U	Initial	0.244±0.01	0.233±0.00	0.222±0.00	0.237±0.00	0.219±0.01	0.336±0.01	0.320±0.00	0.303±0.00	0.322±0.01	0.298±0.00
	Adversarial	0.243±0.00	0.238±0.01	0.235±0.01	0.244±0.00	0.230±0.00	0.341±0.01	0.320±0.01	0.325±0.01	0.333±0.01	0.306±0.01
Acrobot-N	Initial	0.231±0.00	0.225±0.00	0.227±0.00	0.222±0.00	0.215±0.00	0.321±0.01	0.311±0.00	0.316±0.01	0.309±0.01	0.301±0.01
	Adversarial	0.246±0.00	0.237±0.00	0.241±0.00	0.242±0.00	0.229±0.00	0.338±0.00	0.327±0.01	0.327±0.00	0.332±0.01	0.314±0.01
Pendulum-U	Initial	0.648±0.02	0.694±0.01	0.634±0.01	0.630±0.02	0.627±0.01	0.799±0.03	0.780±0.02	0.744±0.01	0.751±0.03	0.733±0.02
	Adversarial	0.672±0.01	0.724±0.01	0.669±0.01	0.674±0.00	0.660±0.01	0.845±0.02	0.854±0.02	0.808±0.02	0.826±0.01	0.778±0.01
Pendulum-N	Initial	0.596±0.00	0.637±0.01	0.574±0.01	0.582±0.00	0.586±0.01	0.715±0.01	0.720±0.01	0.685±0.01	0.695±0.01	0.694±0.01
	Adversarial	0.664±0.02	0.702±0.01	0.660±0.02	0.677±0.02	0.635±0.01	0.861±0.03	0.837±0.02	0.817±0.03	0.860±0.04	0.777±0.03

5.2 Empirical Result Analysis

Here, we report the experimental results, perform analysis and answer the raised **RQ**s (1)/(2).

Overall Performance: Table 1 shows that AR-MAML mostly outperforms others in the adversarial distribution, seldom sacrificing performance in the initial distribution. Similar to observations in [71], task distributionally robust optimization methods, like DR-MAML and DRO-MAML, not only retain robustness advantage on shifted distribution but also sometimes boost average performance on the initial distribution. Cases with two types of initial task distributions (Uniform/Normal) come to similar conclusions on average and CVaR$_\alpha$ performance. Figures 4/5 show the meta reinforcement learning results for Point Robot and Ant Pos navigation tasks. AR-MAML exhibits similar superiority on both continuous control benchmarks compared to baselines.

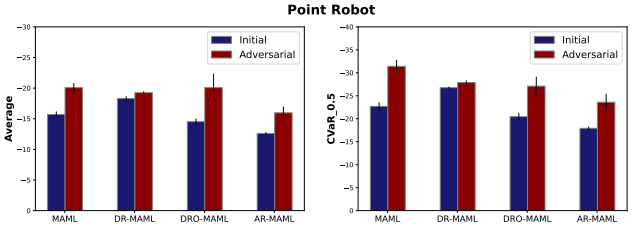

Figure 4: Meta Testing Returns in Point Robot Navigation Tasks (4 runs). The charts report average and CVaR$_\alpha$ returns with $\alpha = 0.5$ in initial and adversarial distributions, with standard error bars indicated by black vertical lines. The higher, the better.

Multiple Tail Risk Robustness: Note that CVaR metrics imply the model's robustness under the subpopulation shift. Figure 6 reports CVaR$_\alpha$ values with various confidence values on pendulum system identification. The AR-MAML's merits in handling the proportional worst cases are consistent across diverse levels. We also illustrate and include these statistics on other benchmarks in Appendix J. Moreover, as suggested in [63], a robust learner seldom encounters a performance gap between a standard (initial) test set and a test set

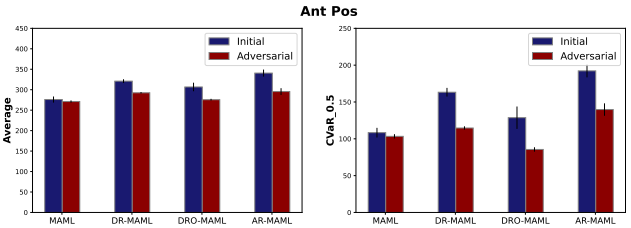

Figure 5: Meta Testing Returns in Ant Pos Tasks (4 runs). The charts report average and CVaR$_\alpha$ returns with $\alpha = 0.5$ in initial and adversarial distributions, with standard error bars indicated by black vertical lines. The higher, the better.

with a distribution shift (adversarial). Figure 7 validates the meta-learners' robustness on sinusoid regression, where AR-MAML's results are more proximal to the $y = x$ line than other baselines.

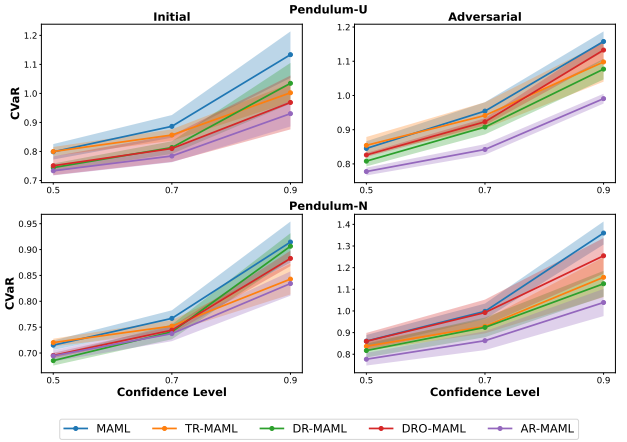

Figure 6: CVaR$_\alpha$ MSEs with Various Confidence Level α. Pendulum-U/N denotes Uniform/Normal as the initial distribution type. The plots report meta testing CVaR$_\alpha$ MSEs in initial and adversarial distributions with standard error in shadow regions.

Random Perturbation Robustness: We also test meta-learners' robustness to random noise from the support dataset. To do so, we

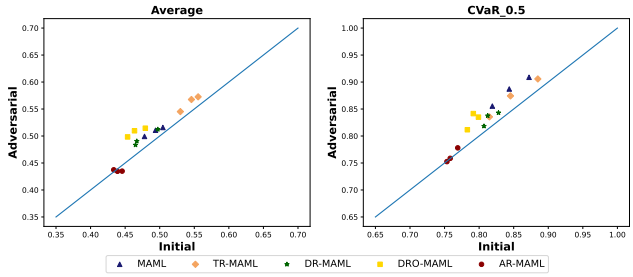

Figure 7: Meta testing MSEs on the initial distribution (x-axis) and on the adversarial distribution (y-axis). The $y = x$ line serves as a baseline for comparison. Models above this line show increased losses when faced with distribution shifts, indicating a decline in performance compared to the standard test set.

take sinusoid regression and inject random noise into the support set, i.e., the noise is drawn from a Gaussian distribution $\mathcal{N}(0, 0.1^2)$ and added to the output y. Figure 8 illustrates that AR-MAML's performance degradation is somewhat less than others on the adversarial distribution. The noise exhibits similar effects on AR-MAML and DR-MAML on the initial distribution, harming performance severely. AR-MAML and DR-MAML still exhibit lower MSEs than other baselines for all cases. This indicates the adversarial training mechanism can also bring more robustness to challenging test scenarios with random noise.

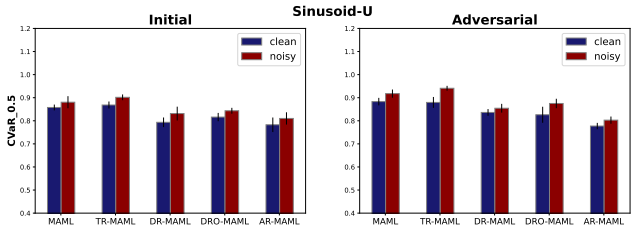

Figure 8: Meta Testing Performance in Clean and Noisy Tasks. The noisy tasks are constructed by adding noise on the outputs of the support dataset. Reported are testing CVaR$_\alpha$ MSEs with $\alpha = 0.5$, where black vertical lines indicate standard error bars.

5.3 Task Structure Analysis

In response to **RQ** (3), we turn to the analysis of the learned distribution adversary. As a result, we visualize the adversarial task probability density.

Explicit Task Distribution: As displayed in Figure 9, our approach enables the discovery of explicit task structures regarding problem-solving. The general learned patterns seem to be regardless of the initial task distributions. In sinusoid regression, more probability mass is allocated in the region with $[3.0, 5.0] \times [0.0, 1.0]$, which reveals more difficulties in adaptation with larger amplitude descriptors. For the Pendulum, the distribution adversary assigns less probability mass to two corner regions, implying that the combination of higher masses and longer pendulums or lower masses and shorter pendulums is easier to predict. Similar phenomena are observed in mass combinations of Acrobat systems. Consistently, the existence of constraint decreases all task distribution entropies to a certain level, which we report in Appendix I. Though such a decrease brings more concentration on some task subsets, AR-MAML still probably fails to cover other challenging combinations in mode collapse.

Initial Task Distributions' Influence on Structures: Comparing the top and the bottom of Figure 9, we notice that the uniform and the normal initial distribution results in similar patterns after normalizing flows' transformations on separate benchmarks. The normal initial distribution can be transformed into smooth ones and captures high-density regions around centroids.

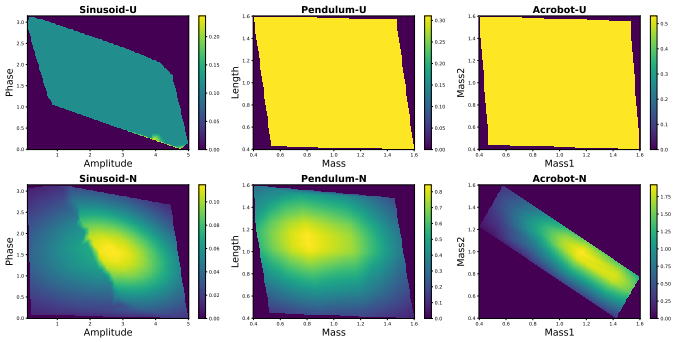

Figure 9: Adversarial Task Probability Distribution. The plots show the adversarial distributions resulting from two different initial distributions: uniform (top row) and normal (bottom row).

5.4 Other Investigations

Here, we conduct additional investigations through the following perspectives.

Impacts of Shift Distribution Constraints: Our studied framework allows the task distribution to shift at a certain level. In Eq. (6), larger λ values tend to cause the generated distribution to collapse into the initial distribution. Consequently, we empirically test the naive and severe adversarial training, e.g., setting $\lambda = \{0.0, 0.1, 0.2\}$ on sinusoid regression. As displayed in Figure 10, the generated distribution with $\lambda = 0.0$ suffers from severe mode collapse, merely covering diagonal regions in the task space. Such a curse is alleviated with increasing λ values. In Figure 11, the meta learner, after heavy distribution shifts, catastrophically fails to generalize well in the initial distribution, illustrating higher adaptation risks in $\lambda = 0.0$.

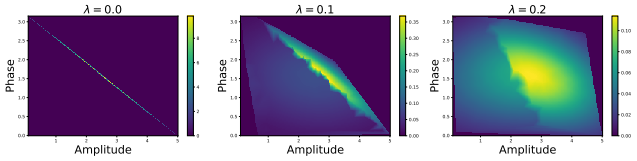

Figure 10: Adversarial Task Probability Distribution on Sinusoid Regression with Various Lagrange Multipliers λ.

Compatibility with Other Meta-learning Methods: Besides the AR-MAML, we also check the effect of adversarially task robust training with other meta-learning methods. Here, AR-CNP in Example 2 is employed in the evaluation. Take the sinusoid regression as an example. Table 2 observes comparable performance between AR-CNP and DR-CNP on the initial task distribution, while results on the adversarial task distribution uncover a significant advantage

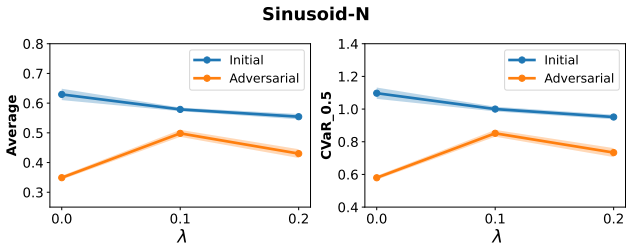

Figure 11: Meta testing MSEs with various lagrange multiplier λ. Reported are testing average and CVaR$_\alpha$ MSEs with $\alpha = 0.5$ with standard error in shadow regions.

over others, particularly on robustness metrics, namely CVaR$_\alpha$ values.

Table 2: Meta testing MSEs in 5-shot sinusoid regression. With $\alpha = 0.5$, the best results are in pink (the lower, the better).

	Average		CVaR	
Method	Initial	Adversarial	Initial	Adversarial
CNP	0.023±0.001	0.026±0.004	0.041±0.003	0.045±0.007
TR-CNP	0.048±0.002	0.050±0.002	0.076±0.004	0.079±0.004
DR-CNP	0.021±0.001	0.023±0.002	0.034±0.003	0.037±0.003
DRO-CNP	0.023±0.001	0.025±0.002	0.039±0.003	0.041±0.004
AR-CNP(Ours)	0.019±0.001	0.018±0.002	0.033±0.001	0.029±0.003

6 Conclusions

Discussions & Society Impacts. This work develops a game-theoretical approach for generating explicit task distributions in an adversarial way and contributes to theoretical understandings. In extensive scenarios, our approach improves adaptation robustness in constrained distribution shifts and enables the discovery of interpretable task structures in optimization.

Limitations & Future Work. The task distribution in this work relies on the task identifier, which can be inaccessible in some cases, e.g., few-shot classification. Also, the adopted strategy to derive the game solution is approximate, leading to suboptimality in optimization. Hence, future efforts can be made to overcome these limitations and facilitate robust adaptation in applications.

Other Supplementary Information

We refer the reader to https://arxiv.org/pdf/2407.19523 and the project homepage https://sites.google.com/view/ar-metalearn for detailed proofs of mentioned theorems and more supplementary information.

Acknowledgments

This work is funded by National Natural Science Foundation of China (NSFC) with the Number # 62306326 and # 62495091. And we thank Dong Liang, Yuhang Jiang, Chen Chen, Daming Shi, other anonymous reviewers, and KDD2025 Area Chairs Prof. Yan Liu and Prof. Auroop R Ganguly for suggestions and helpful discussions.

References

[1] Kelsey Allen, Evan Shelhamer, Hanul Shin, and Joshua Tenenbaum. 2019. Infinite mixture prototypes for few-shot learning. In *International Conference on Machine Learning*. PMLR, 232–241.

[2] Timothée Anne, Jack Wilkinson, and Zhibin Li. 2021. Meta-learning for fast adaptive locomotion with uncertainties in environments and robot dynamics. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 4568–4575.

[3] Yutong Bai, Xinyang Geng, Karttikeya Mangalam, Amir Bar, Alan Yuille, Trevor Darrell, Jitendra Malik, and Alexei A Efros. 2023. Sequential Modeling Enables Scalable Learning for Large Vision Models. arXiv:2312.00785 [cs.CV]

[4] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*. 41–48.

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

[6] Haoang Chi, He Li, Wenjing Yang, Feng Liu, Long Lan, Xiaoguang Ren, Tongliang Liu, and Bo Han. 2024. Unveiling Causal Reasoning in Large Language Models: Reality or Mirage?. In *Neural Information Processing Systems*.

[7] Haoang Chi, Feng Liu, Wenjing Yang, Long Lan, Tongliang Liu, Bo Han, William Cheung, and James Kwok. 2021. TOHAN: A one-step approach towards few-shot hypothesis adaptation. In *Neural Information Processing Systems*, Vol. 34. 20970–20982.

[8] Haoang Chi, Feng Liu, Wenjing Yang, Long Lan, Tongliang Liu, Bo Han, Gang Niu, Mingyuan Zhou, and Masashi Sugiyama. 2022. Meta Discovery: Learning to Discover Novel Classes given Very Limited Data. In *International Conference on Learning Representations*.

[9] Liam Collins, Aryan Mokhtari, and Sanjay Shakkottai. 2020. Task-robust model-agnostic meta-learning. *Advances in Neural Information Processing Systems* 33 (2020), 18860–18871.

[10] Henry Conklin, Bailin Wang, Kenny Smith, and Ivan Titov. 2021. Meta-learning to compositionally generalize. *arXiv preprint arXiv:2106.04252* (2021).

[11] Yan Duan, John Schulman, Xi Chen, Peter L Bartlett, Ilya Sutskever, and Pieter Abbeel. 2016. Rl2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779* (2016).

[12] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. 2020. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1082–1092.

[13] Bernard Faverjon and Pierre Tournassoud. 1987. A local based approach for path planning of manipulators with a high number of degrees of freedom. In *Proceedings. 1987 IEEE international conference on robotics and automation*, Vol. 4. IEEE, 1152–1159.

[14] Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. 2021. Efficiently identifying task groupings for multi-task learning. *Advances in Neural Information Processing Systems* 34 (2021), 27503–27516.

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

[16] Michael C Fu. 2006. Gradient estimation. *Handbooks in operations research and management science* 13 (2006), 575–616.

[17] Marta Garnelo, Dan Rosenbaum, Christopher Maddison, Tiago Ramalho, David Saxton, Murray Shanahan, Yee Whye Teh, Danilo Rezende, and SM Ali Eslami. 2018. Conditional neural processes. In *International Conference on Machine Learning*. PMLR, 1704–1713.

[18] Marta Garnelo, Jonathan Schwarz, Dan Rosenbaum, Fabio Viola, Danilo J Rezende, SM Eslami, and Yee Whye Teh. 2018. Neural processes. *arXiv preprint arXiv:1807.01622* (2018).

[19] Micah Goldblum, Liam Fowl, and Tom Goldstein. 2019. Robust few-shot learning with adversarially queried meta-learners. (2019).

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[21] Jonathan Gordon, Wessel P Bruinsma, Andrew YK Foong, James Requeima, Yann Dubois, and Richard E Turner. 2019. Convolutional Conditional Neural Processes. In *International Conference on Learning Representations*.

[22] Erin Grant, Chelsea Finn, Sergey Levine, Trevor Darrell, and Thomas Griffiths. 2018. Recasting gradient-based meta-learning as hierarchical bayes. *arXiv preprint arXiv:1801.08930* (2018).

[23] David Ha, Andrew Dai, and Quoc V Le. 2016. Hypernetworks. *arXiv preprint arXiv:1609.09106* (2016).

[24] Sepp Hochreiter, A Steven Younger, and Peter R Conwell. 2001. Learning to learn using gradient descent. In *International conference on artificial neural networks*. Springer, 87–94.

[25] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. 2021. Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence* 44, 9 (2021), 5149–5169.

[26] Hongwei Huang, Zhangkai Wu, Wenbin Li, Jing Huo, and Yang Gao. 2021. Local descriptor-based multi-prototype network for few-shot learning. *Pattern Recognition* 116 (2021), 107935.

[27] Chi Jin, Praneeth Netrapalli, and Michael Jordan. 2020. What is local optimality in nonconvex-nonconcave minimax optimization?. In *International conference on machine learning*. PMLR, 4880–4889.

[28] Chi Jin, Praneeth Netrapalli, and Michael I Jordan. 2019. Minmax optimization: Stable limit points of gradient descent ascent are locally optimal. *arXiv preprint arXiv:1902.00618* (2019).

[29] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. 2022. Transformers in vision: A survey. *ACM computing surveys (CSUR)* 54, 10s (2022), 1–41.

[30] Hyunjik Kim, Andriy Mnih, Jonathan Schwarz, Marta Garnelo, Ali Eslami, Dan Rosenbaum, Oriol Vinyals, and Yee Whye Teh. 2018. Attentive Neural Processes. In *International Conference on Learning Representations*.

[31] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[32] David M Kreps. 1989. Nash equilibrium. In *Game Theory*. Springer, 167–177.

[33] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. 2016. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236* (2016).

[34] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436–444.

[35] Kimin Lee, Younggyo Seo, Seunghyun Lee, Honglak Lee, and Jinwoo Shin. 2020. Context-aware dynamics model for generalization in model-based reinforcement learning. In *International Conference on Machine Learning*. PMLR, 5757–5766.

[36] Seanie Lee, Bruno Andreis, Kenji Kawaguchi, Juho Lee, and Sung Ju Hwang. 2022. Set-based meta-interpolation for few-task meta-learning. *Advances in Neural Information Processing Systems* 35 (2022), 6775–6788.

[37] Timothée Lesort, Massimo Caccia, and Irina Rish. 2021. Understanding continual learning settings with data distribution drift analysis. *arXiv preprint arXiv:2104.01678* (2021).

[38] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo. 2019. Revisiting local descriptor based image-to-class measure for few-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7260–7268.

[39] Wenbin Li, Lei Wang, Xingxing Zhang, Lei Qi, Jing Huo, Yang Gao, and Jiebo Luo. 2022. Defensive Few-Shot Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 5 (2022), 5649–5667.

[40] Chenghao Liu, Zhihao Wang, Doyen Sahoo, Yuan Fang, Kun Zhang, and Steven CH Hoi. 2020. Adaptive task sampling for meta-learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*. Springer, 752–769.

[41] Qi Liu, Tao Liu, Zihao Liu, Yanzhi Wang, Yier Jin, and Wujie Wen. 2018. Security analysis and enhancement of model compressed deep learning systems under adversarial attacks. In *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*. IEEE, 721–726.

[42] Yixiu Mao, Hongchang Zhang, Chen Chen, Yi Xu, and Xiangyang Ji. 2023. Supported trust region optimization for offline reinforcement learning. In *International Conference on Machine Learning*. PMLR, 23829–23851.

[43] Bhairav Mehta, Manfred Diaz, Florian Golemo, Christopher J Pal, and Liam Paull. 2020. Active domain randomization. In *Conference on Robot Learning*. PMLR, 1162–1176.

[44] Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *Comput. Surveys* 56, 2 (2023), 1–40.

[45] Tsendsuren Munkhdalai and Hong Yu. 2017. Meta networks. In *International Conference on Machine Learning*. PMLR, 2554–2563.

[46] Shikhar Murty, Tatsunori B Hashimoto, and Christopher D Manning. 2021. Dreca: A general task augmentation strategy for few-shot natural language inference. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1113–1125.

[47] Renkun Ni, Micah Goldblum, Amr Sharaf, Kezhi Kong, and Tom Goldstein. 2021. Data augmentation for meta-learning. In *International Conference on Machine Learning*. PMLR, 8152–8161.

[48] Kevin C Olds. 2015. Global indices for kinematic and force transmission performance in parallel robots. *IEEE Transactions on Robotics* 31, 2 (2015), 494–500.

[49] Martin J Osborne et al. 2004. *An introduction to game theory*. Vol. 3. Oxford university press New York.

[50] David Pollard. 1984. *Convergence of stochastic processes*. David Pollard.

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

[52] Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. 2019. Meta-learning with implicit gradients. *Advances in neural information processing systems* 32 (2019).

[53] Davis Rempe, Jonah Philion, Leonidas J Guibas, Sanja Fidler, and Or Litany. 2022. Generating useful accident-prone driving scenarios via a learned traffic prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 17305–17315.

[54] James Requeima, Jonathan Gordon, John Bronskill, Sebastian Nowozin, and Richard E Turner. 2019. Fast and flexible multi-task classification using conditional neural adaptive processes. *Advances in Neural Information Processing Systems* 32 (2019).

[55] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*. PMLR, 1278–1286.

[56] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. 2018. Meta-Learning with Latent Embedding Optimization. In *International Conference on Learning Representations*.

[57] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. 2019. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731* (2019).

[58] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. 2016. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*. PMLR, 1842–1850.

[59] Jiayi Shen, Xiantong Zhen, Qi Wang, and Marcel Worring. 2023. Episodic Multi-Task Learning with Heterogeneous Neural Processes. *Advances in Neural Information Processing Systems* 36 (2023).

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

[61] Richard S Sutton, Andrew G Barto, et al. 1998. *Introduction to reinforcement learning*. Vol. 135. MIT press Cambridge.

[62] Shuhan Tan, Kelvin Wong, Shenlong Wang, Sivabalan Manivasagam, Mengye Ren, and Raquel Urtasun. 2021. Scenegen: Learning to generate realistic traffic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 892–901.

[63] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. 2020. Measuring Robustness to Natural Distribution Shifts in Image Classification. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[64] Sebastian Shenghong Tay, Chuan Sheng Foo, Urano Daisuke, Richalynn Leong, and Bryan Kian Hsiang Low. 2022. Efficient distributionally robust Bayesian optimization with worst-case sensitivity. In *International Conference on Machine Learning*. PMLR, 21180–21204.

[65] Emanuel Todorov, Tom Erez, and Yuval Tassa. 2012. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 5026–5033.

[66] Vladimir Vapnik. 1999. *The nature of statistical learning theory*. Springer science & business media.

[67] Heinrich Von Stackelberg and Stackelberg Heinrich Von. 1952. *The theory of the market economy*. Oxford University Press.

[68] Lingxiao Wang, Qi Cai, Zhuoran Yang, and Zhaoran Wang. 2020. On the global optimality of model-agnostic meta-learning. In *International conference on machine learning*. PMLR, 9837–9846.

[69] Qi Wang, Marco Federici, and Herke van Hoof. 2023. Bridge the Inference Gaps of Neural Processes via Expectation Maximization. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=A7v2DqLjZdq

[70] Qi Wang, Yanghe Feng, Jincai Huang, Yiqin Lv, Zheng Xie, and Xiaoshan Gao. 2023. Large-scale generative simulation artificial intelligence: The next hotspot. *The Innovation* 4, 6 (2023).

[71] Qi Wang, Yiqin Lv, Yanghe Feng, Zheng Xie, and Jincai Huang. 2023. A Simple Yet Effective Strategy to Robustify the Meta Learning Paradigm. *Advances in Neural Information Processing Systems* 36 (2023).

[72] Qi Wang and Herke Van Hoof. 2020. Doubly stochastic variational inference for neural processes with hierarchical latent variables. In *International Conference on Machine Learning*. PMLR, 10018–10028.

[73] Qi Wang and Herke van Hoof. 2022. Learning expressive meta-representations with mixture of expert neural processes. In *Advances in neural information processing systems*.

[74] Qi Wang and Herke Van Hoof. 2022. Model-based meta reinforcement learning using graph structured surrogate models and amortized policy search. In *International Conference on Machine Learning*. PMLR, 23055–23077.

[75] Wolfram Wiesemann, Daniel Kuhn, and Melvyn Sim. 2014. Distributionally robust convex optimization. *Operations Research* 62, 6 (2014), 1358–1376.

[76] Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* 8, 3 (1992), 229–256.

[77] Yichen Wu, Long-Kai Huang, and Ying Wei. 2022. Adversarial task up-sampling for meta-learning. *Advances in Neural Information Processing Systems* 35 (2022), 31102–31115.

[78] Zehao Xiao, Jiayi Shen, Mohammad Mahdi Derakhshani, Shengcai Liao, and Cees GM Snoek. 2024. Any-Shift Prompting for Generalization over Distributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13849–13860.

[79] Zehao Xiao, Xiantong Zhen, Shengcai Liao, and Cees GM Snoek. 2023. Energy-based test sample adaptation for domain generalization. *arXiv preprint*

arXiv:2302.11215 (2023).

[80] Zehao Xiao, Xiantong Zhen, Ling Shao, and Cees GM Snoek. 2022. Learning to generalize across domains on single test samples. *arXiv preprint arXiv:2202.08045* (2022).

[81] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. 2021. Mining latent classes for few-shot segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*. 8721–8730.

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

[83] Huaxiu Yao, Yu Wang, Ying Wei, Peilin Zhao, Mehrdad Mahdavi, Defu Lian, and Chelsea Finn. 2021. Meta-learning with an adaptive task scheduler. *Advances in Neural Information Processing Systems* 34 (2021), 7497–7509.

[84] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. 2020. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*. PMLR, 1094–1100.

[85] Jesse Zhang, Brian Cheung, Chelsea Finn, Sergey Levine, and Dinesh Jayaraman. 2020. Cautious adaptation for reinforcement learning in safety-critical settings. In *International Conference on Machine Learning*. PMLR, 11055–11065.

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