Conditional Generative Learning from Invariant Representations in Multi-Source: Robustness and Efficiency

Guojun Zhu[1,2] **Sanguo Zhang**[1,2] **Mingyang Ren**[3,†]
[1]School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing, China
[2]Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing, China
[3]School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai, China
[†]Corresponding author: mingyangren@sjtu.edu.cn

Abstract

Multi-source generative models have gained significant attention due to their ability to capture complex data distributions across diverse domains. However, existing approaches often struggle with limitations such as negative transfer and an over-reliance on large pre-trained models. To address these challenges, we propose a novel method that effectively handles scenarios with outlier source domains, while making weaker assumptions about the data, thus ensuring broader applicability. Our approach enhances robustness and efficiency, supported by rigorous theoretical analysis, including non-asymptotic error bounds and asymptotic guarantees. In the experiments, we validate our methods through numerical simulations and real-world data experiments, showcasing their practical effectiveness and adaptability.

1 INTRODUCTION

A fundamental problem in statistics and machine learning is modeling the relationship between a response Y and a covariate X. Regression models, which estimate the conditional mean or median of Y given X, are commonly used for this task. However, when the conditional distribution is multimodal or asymmetric, these methods fall short in capturing the full complexity of the relationship between Y and X. To gain a complete understanding, it is necessary to model the entire conditional distribution, a task at which conditional generative models excel (Zhou et al., 2023; Liu et al., 2021), particularly when based on well-established architectures like Generative Adversarial Networks (GAN) (Goodfellow et al., 2014) and Wasserstein GAN (WGAN) (Arjovsky et al., 2017). Conditional generative models also play a central role in many important areas, including natural language processing, computer vision, and biomedical applications, where deeper insights into data distributions enable more flexible and informed decision-making.

In real-world applications of conditional generative models, beyond the target dataset of interest, data is also collected from multi-source domains that might differ from the target domain. For example, in biomedical studies, patient data can be sourced from different hospitals or regions, while in financial modeling, market conditions may vary across different time periods. Pooling these datasets together without accounting for domain differences can lead to suboptimal performance. Transfer learning has emerged as a powerful approach to handle such domain discrepancies by enabling knowledge transfer from multi-source domains to the target domain, which has gained increasing attention across various fields (Tian et al., 2023; He et al., 2024).

While transfer learning has been extensively studied for a wide range of models, including high-dimensional linear models (Bastani, 2021; Li et al., 2022), generalized linear models (Tian and Feng, 2023), functional regression Lin and Reimherr (2022), semi-supervised classification (Zhou et al., 2024) and and basis-type models (Cai and Pu, 2024), applying it to conditional generative models poses unique challenges. Unlike parametric or semi-parametric models, where the common approach is to directly transfer parameters, conditional generative models are non-parametric and require a different method. Besides, they capture entire distribution rather than just mean or median, making it crucial to characterize the bias between the empiri-

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

cal distribution and the true distribution with the help of reliable source domains.

While multi-source transfer learning for conditional generative models has gained attention, existing approaches that rely on fine-tuning pre-trained models face several limitations. One major issue is that these methods often over-rely on the large-scale pre-trained models, such as those used in image generation tasks like StyleGAN, which was trained on massive datasets like Flickr-Faces-HQ (Karras et al., 2019). For traditional datasets, such as tabular medical data, such pre-trained models simply do not exist. This makes the application of these methods impractical in many real-world tasks. Moreover, fine-tuning pre-trained models introduces theoretical challenges, as the complex adjustments required to align the generator and discriminator make it difficult to derive rigorous theoretical guarantees (Han et al., 2021). Additionally, pre-trained models are often praised for their strong generalization capabilities, which can make negative transfer, where the model's performance degrades due to irrelevant or misleading information from source domains, a less frequently discussed issue. However, their out-of-distribution generalization still falls short, highlighting a robustness gap (Harun et al., 2024).

These gaps motivate the need for novel method that do not rely on pre-trained models. Our approach seeks to address this by developing transfer learning frameworks for conditional generative models that are more robust, broadly applicable, and theoretically sound. We specifically consider settings where not all source domains are assumed to have a strong similarity with the target domain, allowing for the presence of outlier source domains. Additionally, our method handles high-dimensional covariates X and response variables Y, without imposing strict assumptions. While these factors present significant challenges, developing such a method would lead to a framework that is more general and widely applicable.

To address these challenges, we propose a novel method that leverages *low-dimensional domain-invariant representations* to transfer knowledge effectively across multiple reliable source domains, even in the presence of outlier source domains. Our approach ensures that the conditional generative model remains both robust and efficient by using a criterion to select reliable source domains. We investigate both cases where the reliable source domains are known and unknown, providing a comprehensive solution to this problem.

Our contributions can be summarized as follows:

- Considering more challenging data settings, we propose a novel algorithm to learn the conditional generator, even in the presence of outlier source domains.
- We fill a theoretical gap by deriving non-asymptotic error upper bounds and asymptotic properties for the algorithm. This advances the theoretical understanding of both single-source and multi-source conditional generative models.
- Our method outperforms other approaches in both numerical simulations and real-world image data experiments.

Notation. For a vector $\boldsymbol{u}$, $\|\boldsymbol{u}\|_1, \|\boldsymbol{u}\|_2$ stands for its ℓ_1-norm and ℓ_2-norm, respectively. For a function $\psi : \mathcal{X} \to \mathbb{R}$, $\|\psi\|_\infty$ is defined to be $\max_{x\in\mathcal{X}} |\psi(x)|$. For two positive real sequences $\{a_n\}_{n=1}^\infty$ and $\{b_n\}_{n=1}^\infty$, $a_n \lesssim b_n$ means there exists a universal constant $C > 0$ such that $a_n \leq Cb_n$ for all n. For any $N \in \mathbb{N}_+, [N]$ is defined to be $\{1, \ldots, N\}$. The notation O is the 'big-O' notation. $\langle \cdot, \cdot \rangle_F$ is the Frobenius dot product. $\mathbb{E}$ is the expectation taken over all randomness.

2 RELATED WORK

Here we give a review of related work in the literature.

Theoretical Insights for GAN. Early theoretical work by Liang (2021) critically analyzed how well GAN learn distributions, laying a foundation for performance analysis. Chen et al. (2020) further provided important statistical guarantees for adversarial training. Huang et al. (2022) analyzed approximation error in GAN and its impact on learning. Building on this, Liu et al. (2021) explored the wasserstein generative learning approach, improving GAN applicability. Zhou et al. (2023) developed a conditional sampling method using KL divergence, offering insights into weak convergence. Expanding on wasserstein method, Song et al. (2023) introduced Wasserstein Generative Regression, showing the versatility of GAN in regression problem. Besides, Tan et al. (2024) proposed an adaptive generator architecture to enhance scalability, while Suh and Cheng (2024) provided a broad overview of GAN developments. However, none of these works address the theoretical challenges of multi-source conditional generative models. Our approach fills this gap by offering a comprehensive theoretical framework for multi-source GAN.

Domain adaptation. Domain adaptation tackles the distribution shift between source and target, often relying on representation-based methods. Asymmetric approaches transform the features of the source domain to match those of the target domain (Hoffman et al., 2014; Kandemir, 2015; Courty et al., 2017), while symmetric methods project both domains into

a shared latent space, aligning their distributions. Notable examples include DeepJDOT (Damodaran et al., 2018) and WDGRL (Shen et al., 2018), both using optimal transport to achieve domain alignment. Despite their effectiveness, these methods have not been applied to conditional generative models. Our work is the first to extend optimal transport techniques to multi-source conditional generative modeling.

Few-shot Generative Model. Few-shot generative models have shown promise in generating high-quality images from limited data using pre-trained models. Wang et al. (2018) introduced GAN transfer techniques, while Wang et al. (2020) presented MineGAN, which mines relevant knowledge from pre-trained models to generate images with few samples. Li et al. (2020) proposed elastic weight consolidation to retain critical information during model adaptation, and Zhao et al. (2022) offered a framework for few-shot generation methods, maximizing mutual information to preserve diversity. Moreover, Tian and Shen (2024) explored diffusion models and introduced a shared embedding conditioning mechanism. However, their knowledge transfer approach primarily focuses on extracting source-shared information during pre-training, without any target domain interaction. While these methods achieve practical success, they heavily rely on large-scale pre-training, limiting their applicability to rare datasets. Our approach eliminates the need for pre-trained models, enabling more flexible and robust multi-source transfer.

3 PROBLEM SET-UP

Suppose there are T sources in total, and we have collected n_t i.i.d. pairs $\left\{\boldsymbol{x}_i^{(t)}, \boldsymbol{y}_i^{(t)}\right\}_{i=1}^{n_t}$ from the t-th source, where $\boldsymbol{x}_i^{(t)} \in \mathcal{X} \subset \mathbb{R}^d$ is drawn according to distribution $P_X^{(t)}$ over $\mathcal{X}$, and then $\boldsymbol{y}_i^{(t)} \in \mathcal{Y} \subset \mathbb{R}^q$ is drawn according to the conditional distribution $P^{(t)}_{Y|X=\boldsymbol{x}_i^{(t)}}$, $t \in [T]$. Besides, we also have collected n_0 i.i.d. pairs $\left\{\boldsymbol{x}_i^{(0)}, \boldsymbol{y}_i^{(0)}\right\}_{i=1}^{n_0}$ from the target. There exists a subset of reliable sources $S \subseteq [T]$, such that for all $t \in S$, we assume a low-dimensional subspace $\mathcal{Z} \subset \mathbb{R}^r$, $r << d$, and a common nonlinear mapping $R : \mathcal{X} \mapsto \mathcal{Z}$ that is shared across different domains, which has the properties described in the following part.

Similarity Measure. We denote $\boldsymbol{z}_i^{(t)} = R\left(\boldsymbol{x}_i^{(t)}\right)$, which follows $P_Z^{(t)}$. The low-dimensional representation $\boldsymbol{z}_i^{(t)}$ retains all the necessary information for learning the conditional distribution of $\boldsymbol{y}_i^{(t)}$. Besides, to the best of our knowledge, this low-dimensional representation is generally not unique (Li, 2018). Our goal is to

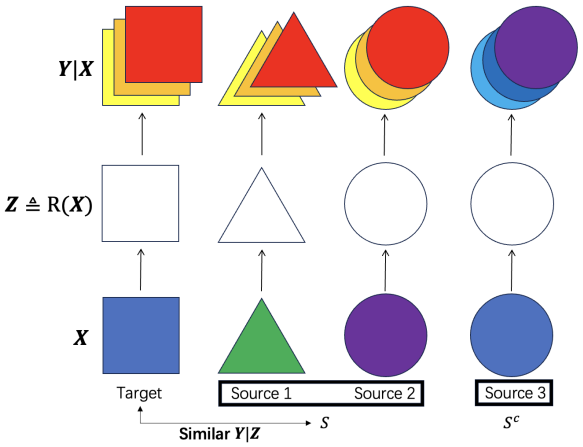


Figure 1: A simple visualization of our setting

learn *domain-invariant representations* that not only facilitate the learning of the conditional distribution but also reduce the distribution discrepancy between the source and target domains. However, in the process of reducing joint distribution differences, R may degenerate. Therefore, we are more concerned with the alignment of conditional distributions. We next define a new similarity measure between the source and target domains in terms of the integral probability metric(IPM) (Müller, 1997), in the sense that

$$d_{\mathcal{F}_B^1}\left(P_{Y|Z}^{(t)}, P_{Y|Z}^{(0)}\right) = \sup_{f \in \mathcal{F}_B^1} \left\{\mathbb{E}_{P_{Y|Z}^{(t)}} f(\boldsymbol{y})) - \mathbb{E}_{P_{Y|Z}^{(0)}} f(\boldsymbol{y}))\right\},$$

where $\mathcal{F}_B^1$ is the uniformly bounded 1-Lipschitz function class,

$$\mathcal{F}_B^1 = \{f : \mathbb{R}^q \mapsto \mathbb{R}, |f(\boldsymbol{u}) - f(\boldsymbol{v})| \leq \|\boldsymbol{u} - \boldsymbol{v}\|_2, \\ \boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^q \text{ and } \|f\|_\infty \leq B\}. \quad (1)$$

We define the source domain as *reliable* if the similarity measure between the source and target domains is sufficiently small in expectation. Specifically, for some $h > 0$, we require:

$$\forall t \in S, \mathbb{E}_{P_Z^{(t)}} d_{\mathcal{F}_B^1}\left(P_{Y|Z}^{(t)}, P_{Y|Z}^{(0)}\right) \leq h. \quad (2)$$

To precisely determine how small h must be for the source to be considered *reliable*, we set $h = O\left(\max\left\{n^{-1/(r+q)}, n_0^{-1/r}\right\}\right)$, where $n = \sum_{t \in S \cup \{0\}} n_t$. It ensures the optimal convergence rate is achieved, making the source reliable.

After representation learning, we are interested in finding a function $G : \mathbb{R}^m \times \mathcal{Z} \mapsto \mathcal{Y}$ such that the conditional distribution of $G(\eta, Z)$ given $Z = \boldsymbol{z}$ equals the conditional distribution of Y given $Z = \boldsymbol{z}$ in the target domain. Since $\eta \sim P_\eta$ is independent of Z, this is

equivalent to finding a G such that

$$G(\eta, \boldsymbol{z}) \sim P^{(0)}_{Y|Z=\boldsymbol{z}}, \boldsymbol{z} \in \mathcal{Z}. \tag{3}$$

Because of this property, we shall refer to G as a conditional generator. The existence of such a G is guaranteed by the noise-outsourcing lemma (Theorem 5.10 in Kallenberg (1997)). For ease of reference, we state it here with a slight modification.

Lemma. (Noise-outsourcing lemma). *Suppose $\mathcal{Y}$ is a standard Borel space. Then there exist a random vector $\eta \sim N(\mathbf{0}, \mathbf{I}_m)$ for a given $m \geq 1$ and a Borel-measurable function $G : \mathbb{R}^m \times \mathcal{Z} \to \mathcal{Y}$ such that η is independent of Z and*

$$(Z, Y) = (Z, G(\eta, Z)) \text{ almost surely.} \tag{4}$$

The noise distribution P_η is taken to be $N(\mathbf{0}, \mathbf{I}_m)$. Because η and X are independent, a G satisfies formula (3) if and only if it also satisfies formula (4). Therefore, to construct the conditional generator, we can find a G such that the joint distribution of $(Z, G(\eta, Z))$ matches the joint distribution of (Z, Y). This is the basis of the proposed generative approach described below.

Finally, we review the core idea of reliable source domains. In terms of S, property (2) naturally holds when $P^{(t)}_{Y|Z} = P^{(0)}_{Y|Z}$, which is a relatively strong assumption of many works(Fernando et al., 2013; Long et al., 2014; Gong et al., 2016). For clarity, we provide Figure 1 as a visualization of the case where $T = 3$ and $|S| = 2$. To better introduce the case where the reliable sources subset S is unknown, we first assume in Section 4 that the subset S is known. The case that S is unknown will be dealt with in Section 5.

4 ORACLE TRANSFER-WGAN

In this section, we assume that S is known. In practice, we use neural networks, denoted as $\hat{G}$ and $\hat{R}$, to approximate the functions G and R, respectively. We denote $\hat{\boldsymbol{z}}^{(t)}_i = \hat{R}\left(\boldsymbol{x}^{(t)}_i\right)$, which is drawn from the distribution $P^{(t)}_{\hat{Z}}$. We consider aggregating the target domain with all reliable source domains in S, pooling their samples for training. This approach can reduce the learning bias of the conditional generative model. However, it introduces a new problem: we are actually approximating a mixture distribution, given by[1]

$$P_{\hat{Z},Y} = \sum_{t \in S \cup \{0\}} \frac{n_t}{n} P^{(t)}_{\hat{Z},Y}.$$

[1]For convenience, we denote that any distribution notation without the domain index (t) refers to a mixture distribution.

The metric we use to compare the model performance with the ground truth is the integral probability metric(IPM): $d_{\mathcal{F}^1_B}\left(P_{\hat{Z},\hat{G}}, P^{(0)}_{\hat{Z},Y}\right)$[2]. Using the mixture distribution, we can decompose this IPM distance into **learning bias** $d_{\mathcal{F}^1_B}\left(P_{\hat{Z},\hat{G}}, P_{\hat{Z},Y}\right)$ and **transfer bias** $d_{\mathcal{F}^1_B}\left(P_{\hat{Z},Y}, P^{(0)}_{\hat{Z},Y}\right)$. Transfer learning provides a way to address the *curse of dimensionality* in learning bias by utilizing data from multiple sources, but it unexpectedly introduces transfer bias. Therefore, our objective is to balance learning bias and transfer bias while taking advantage of the properties of domain-invariant representations.

Building on the work of (Liu et al., 2021), we adopt the WGAN architecture which formulates the training process as a min-max optimization problem. Specifically, the generator aims to minimize the Wasserstein distance between the generated and real data distributions, while the discriminator attempts to maximize it. However, since it relies on min-max optimization, it is prone to instability during training. Alternating the training of $\hat{R}$ and $\hat{G}$ simultaneously tends to exacerbate this instability in practice. To enhance the stability of WGAN training, we split the process into two stages, as motivated by (Wang et al., 2024). In the first stage, the primary objective is to identify domain-invariant representations, with regularization applied to minimize distributional differences. In the second stage, using only these identified representations, we conduct a refined estimation.

Stage 1. To improve the stability of the discriminator's training, this stage uses $(\boldsymbol{x}^{(t)}_i, \boldsymbol{y}^{(t)}_i)$ as the input samples for the discriminator. We have $d_{\mathcal{F}^1_B}\left(P_{X,\hat{G}}, P_{X,Y}\right) \leq W_1\left(P_{X,\hat{G}}, P_{X,Y}\right)$, where W_1 is the 1-Wasserstein distance, the Kantorovich-Rubinstein theorem shows that the dual form of the 1-Wasserstein distance can be written as a form of integral probability metric(IPM) (Villani et al., 2009),

$$W_1\left(P_{X,\hat{G}}, P_{X,Y}\right) = \sup_{D \in \mathcal{F}^1_{\text{Lip}}} \left\{\mathbb{E}_{P_X P_\eta} D(X, G(\eta, R(X))) - \mathbb{E}_{P_{X,Y}} D(X, Y)\right\},$$

$$\mathcal{F}^1_{\text{Lip}} = \left\{f : \mathbb{R}^{d+q} \to \mathbb{R}, \frac{|f(\boldsymbol{u}) - f(\boldsymbol{v})|}{\|\boldsymbol{u} - \boldsymbol{v}\|_2} \leq 1, \forall \boldsymbol{u}, \boldsymbol{v}\right\}.$$

Thus, finding the conditional generator and the representation can be formulated as a minimax problem,

$$\underset{G,R}{\text{argmin}} \underset{D \in \mathcal{F}^1_{\text{Lip}}}{\text{argmax}} \mathcal{L}_1(R, G, D; S),$$

which we incorporate regularization into the objective function, based on the original form of the 1-

[2]We omit the argument $\hat{G}(\eta, \hat{Z})$ and refer to it as $\hat{G}$.

Wasserstein distance between the source domains and the target domain.

$$\mathcal{L}_1(R,G,D;S) = \mathbb{E}_{P_X P_\eta} D(X, G(\eta, R(X))) - \mathbb{E}_{P_{X,Y}} D(X,Y) + \sum_{t\in S} \lambda_t \inf_{\gamma} \int \|(R(X^{(t)}),Y^{(t)}) - (R(X^{(0)}),Y^{(0)})\|_1 d\gamma,$$

where λ_t represents weights for different source domains, $\gamma \in \Pi\left(P^{(t)}_{X,Y}, P^{(0)}_{X,Y}\right)$ describes the space of joint probability distributions with marginals $P^{(t)}_{X,Y}$ and $P^{(0)}_{X,Y}$. We avoid using the dual form for regularization because we do not want to introduce additional neural networks.

Let $\eta_i^{(t)}$ be independently generated from P_η. The empirical version of $\mathcal{L}_1(R,G,D;S)$ is

$$\widehat{\mathcal{L}}_1(R,G,D;S) = \frac{1}{n}\left[\sum_{\substack{t\in S\cup\{0\}\\ i=1}}^{n_t} D\left(\boldsymbol{x}_i^{(t)}, G\left(\eta_i, R\left(\boldsymbol{x}_i^{(t)}\right)\right)\right) - D\left(\boldsymbol{x}_i^{(t)}, \boldsymbol{y}_i^{(t)}\right)\right] + \sum_{t\in S}\lambda_t \min_{\gamma\in\Pi\left(P^{n_t}_{X,Y}, P^{n_0}_{X,Y}\right)} \left\langle \gamma, \mathbf{C}_R^{(t)}\right\rangle_F,$$

where $P^{n_t}_{X,Y}, P^{n_0}_{X,Y}$ are the empirical distributions and $\mathbf{C}_R^{(t)} = (\mathbf{C}^{(t)}_{R,ij})_{i,j=1}^{n_t,n_0}$ is a cost matrix $\in \mathbb{R}^{n_t\times n_0}$,

$$\mathbf{C}^{(t)}_{R,ij} = \|(R(\boldsymbol{x}_i^{(t)}), \boldsymbol{y}_i^{(t)}) - (R(\boldsymbol{x}_j^{(0)}), \boldsymbol{y}_j^{(0)})\|_1.$$

Although this avoids introducing additional neural networks, it still requires solving an optimization problem with γ. Efficient computational schemes have been proposed with stochastic versions using the dual formulation of the problem Genevay et al. (2016); Seguy et al. (2017), allowing for the tackling of small to medium-sized problems.

We use a feedforward neural network $G_{\boldsymbol{\theta}_1}$ with parameter $\boldsymbol{\theta}_1$ for estimating the conditional generator G in Stage 1, a second network $D_{\boldsymbol{\phi}_1}$ with parameter $\boldsymbol{\phi}_1$ for estimating the discriminator D in Stage 1 and a third network $R_{\boldsymbol{\omega}}$ with parameter $\boldsymbol{\omega}$ for estimating the representation R. We estimate $\boldsymbol{\theta}_1$, $\boldsymbol{\phi}_1$ and $\boldsymbol{\omega}$ by solving the minimax problem,

$$(\hat{\boldsymbol{\omega}}, \hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\phi}}_1) = \operatorname*{argmin}_{\boldsymbol{\omega},\boldsymbol{\theta}_1} \operatorname*{argmax}_{\boldsymbol{\phi}_1} \widehat{\mathcal{L}}_1\left(R_{\boldsymbol{\omega}}, G_{\boldsymbol{\theta}_1}, D_{\boldsymbol{\phi}_1}; S\right).$$

The estimated representation is $\hat{R} = R_{\hat{\omega}}$ which will be used in Stage 2. At this stage, the estimated $G_{\boldsymbol{\theta}_1}$ and $D_{\boldsymbol{\phi}_1}$ are not the final results.

Stage 2. To further refine estimation, network retraining is conducted using the identified representation $\hat{R}$. This stage uses $(\hat{R}(\boldsymbol{x}_i^{(t)}), \boldsymbol{y}_i^{(t)})$ as the input samples for the discriminator which is different from the stage 1. We consider the minimax problem with no regularization:

$$\operatorname*{argmin}_{G} \operatorname*{argmax}_{D\in\mathcal{F}^1_{\text{Lip}}} \mathcal{L}_2(G,D;S),$$

where

$$\mathcal{L}_2(G,D;S) = \mathbb{E}_{P_X P_\eta} D(\hat{R}(X), G(\eta, \hat{R}(X))) - \mathbb{E}_{P_{X,Y}} D(\hat{R}(X), Y).$$

The empirical version of $\mathcal{L}_2(G,D;S)$ is

$$\widehat{\mathcal{L}}_2(G,D;S) = \frac{1}{n}\left[\sum_{\substack{t\in S\cup\{0\}\\ i=1}}^{n_t} D\left(\hat{R}(\boldsymbol{x}_i^{(t)}), G\left(\eta_i, \hat{R}\left(\boldsymbol{x}_i^{(t)}\right)\right)\right) - D\left(\hat{R}(\boldsymbol{x}_i^{(t)}), \boldsymbol{y}_i^{(t)}\right)\right].$$

We use a feedforward neural network $G_{\boldsymbol{\theta}}$ with parameter $\boldsymbol{\theta}$ for estimating the conditional generator G, a second network $D_{\boldsymbol{\phi}}$ with parameter $\boldsymbol{\phi}$ for estimating the discriminator D. We estimate $\boldsymbol{\theta}$, $\boldsymbol{\phi}$ by solving the minimax problem,

$$(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\phi}}) = \operatorname*{argmin}_{\boldsymbol{\theta}} \operatorname*{argmax}_{\boldsymbol{\phi}} \widehat{\mathcal{L}}_2\left(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\phi}}; S\right).$$

The estimated conditional generator and discriminator are $\hat{G} = G_{\hat{\theta}}, \hat{D} = D_{\hat{\phi}}$. Due to space limitations, the detailed algorithm implementation and the **non-asymptotic error bound** can be found in the Appendix.

Remark. As illustrated in the Appendix, compared to the results in Liu et al. (2021), where the convergence rate without utilizing other source domains is $n_0^{-1/(d+q)}$, we improve this to $n^{-1/(r+q)} + n_0^{-1/r}$ by incorporating representation learning and leveraging information from the source domains when S is known. When both d and q are high-dimensional, $r \ll d$ signifies that the representation dimension is much smaller than the original data dimension, allowing for a more **efficient** convergence.

5 SELECTED TRANSFER-WGAN

In the previous section, we introduced an oracle algorithm based on a known subset S of reliable source domains. This leads to an intriguing and practically significant question: Can we develop a data-driven, adaptive selection criterion to estimate the subset $\hat{S}$?

Recall that our previous definition of S,

$$\forall t \in S, \mathbb{E}_{P^{(t)}_{\hat{Z}}} d_{\mathcal{F}^1_B}\left(P^{(t)}_{Y|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) \le h.$$

Estimating the conditional distributions $P^{(t)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}$ and $P^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}$ can be challenging due to the differences in covariates $\boldsymbol{x}$. This often results in an insufficient number of samples for $\hat{Z}=\hat{\boldsymbol{z}}$, leading to significant biases compared to the ground truth. Therefore, using this distance directly for selection is not feasible.

To address this issue, we aim to utilize the joint distribution as a bridge. By doing so, we can constrain the 1-Wasserstein distance of the joint distribution instead of conditional distribution. This approach allows for a more feasible and practical method of selection. Then we can follow the oracle method mentioned in Section 4 by using $\hat{S}$ as a substitute for unknown S.

5.1 Selection Criterion

At first, we are unaware of which source domain is intrinsically similar to the target domain with the low-dimensional representation. Therefore, in the initial step, we should train a *full model* using all the source domains. We also observe that representation learning, compared to the subsequent conditional generative model learning, is less affected by outlier source domains. This phenomenon has also been studied and confirmed by Ortego et al. (2021). Thus, we can utilize the representation neural network $\tilde{R}=R_{\tilde{\boldsymbol{\omega}}}$ and representations $\tilde{\boldsymbol{z}}_i^{(t)}=\tilde{R}\left(\boldsymbol{x}_i^{(t)}\right)$ obtained from the full model to construct our selection criterion.

Stage 1. In this stage, our goal is to estimate $\hat{S}$ using the representation $\tilde{R}$ trained in the full model. To avoid confusion with the previous content, we present the training loss function of the full model here:

$$(\tilde{\boldsymbol{\theta}},\tilde{\boldsymbol{\phi}},\tilde{\boldsymbol{\omega}})=\underset{\boldsymbol{\theta},\boldsymbol{\omega}}{\operatorname{argmin}}\ \underset{\boldsymbol{\phi}}{\operatorname{argmax}}\ \widehat{\mathcal{L}}_1\left(R_{\boldsymbol{\omega}},G_{\boldsymbol{\theta}},D_{\boldsymbol{\phi}};[T]\right).$$

Then, for some constant $C>0$, we estimate $\hat{S}$ as:

$$\left\{t:W_1(P^{n_t}_{\tilde{Z},Y},P^{n_0}_{\tilde{Z},Y})\le C\left(\max\left\{n^{-1/(r+q)},n_0^{-1/r}\right\}\right)\right\},$$

where $P^{n_t}_{\tilde{Z},Y},P^{n_0}_{\tilde{Z},Y}$ are empirical distributions.

Stage 2. In this stage, our goal is to estimate $\hat{\boldsymbol{\theta}},\hat{\boldsymbol{\phi}},\hat{\boldsymbol{\omega}}$ using $\hat{S}$. We simply replace S in the Oracle Transfer-WGAN with $\hat{S}$. The detailed algorithm implementation can be referenced in the Appendix.

5.2 Theoretical results

In this section, we summarize the key theoretical results. Due to space constraints, Assumptions 2-6 and the conditions of the theorems have been moved to the Appendix. Additionally, we choose $h=O\left(\max\left\{n^{-1/(r+q)},n_0^{-1/r}\right\}\right)$. To clearly define a reliable source domain, we assume that the outlier source domains are sufficiently distant from the target domain.

Assumption 1. The similarity measure between the outlier sources and the target domain is assumed to be of a much larger order than h. Specifically, we assume

$$\forall t\in S^c,\mathbb{E}_{P^{(t)}_{\tilde{Z}}}d_{\mathcal{F}_B^1}\left(P^{(t)}_{Y|\tilde{Z}},P^{(0)}_{Y|\tilde{Z}}\right)=O(h^\alpha),\alpha>1.$$

From this, we can derive the following two theorems:

Theorem 5.1 *Suppose that $P_{\tilde{Z},Y},P_{\hat{Z},Y}$ are supported on $[-U,U]^{r+q}$ for some $U>0$ and satisfies Assumptions 1, 5-6 provided in Appendix, we have*

$$P(\hat{S}=S)\to 1,\text{when } n_t,n_0\to+\infty.$$

Theorem 5.2 *Suppose that $P_{\tilde{Z},Y},P_{\hat{Z},Y}$ are supported on $[-U,U]^{r+q}$ for some $U>0$ and satisfies Assumptions 1, 3-6 provied in Appendix, we have*

$$\mathbb{E}_{\hat{G}}\mathbb{E}_{P^{(0)}_{\hat{Z}}}d_{\mathcal{F}_B^1}\left(P_{\hat{G}|\hat{Z}},P^{(0)}_{Y|\hat{Z}}\right)\precsim n^{-1/(r+q)}+n_0^{-1/(r+q)}.$$

Remark 1. As a result, although the first term in the upper bound is dominated by the second term, and the efficiency improvement is only from $n_0^{-1/(d+q)}$ to $n_0^{-1/(r+q)}$, this is due to the fact that when S is unknown, we must also consider the bias introduced by the estimate $\hat{S}$ in the presence of limited samples. This represents a trade-off where **efficiency is sacrificed to ensure robustness**. Since $r\ll d$, there is still a significant improvement.

Remark 2. Since we only know the order of h, the corresponding constant remains unknown. In practical applications, we can adjust the constant after determining the order and use it as a threshold. Additionally, we can sort $W_1\left(P^{n_t}_{\tilde{Z},Y},P^{n_0}_{\tilde{Z},Y}\right)$ for $t\in[T]$, where smaller values indicate more reliable source domains for transfer, allowing for a more **robust** selection.

6 EXPERIMENTS

In this section, we present the key settings and results of three different experiments[3], with additional details provided in the appendix.

6.1 Numerical simulation

We focus on the problem of estimating the conditional mean and standard deviation in nonparametric conditional density models. Since our approach is the first to eliminate the need for a pretrained model,

[3]Our code is publicly available at https://github.com/zgj19stat/STWGAN

and no pretrained models are available for this task, we compare the proposed Selected Transfer-WGAN method (referred to as **STWGAN** in Table 1) with two baselines: **Target-Only**, a method trained exclusively on the target domain without representation learning, and **Pool**, an ablation variant where $\lambda_t = 0$. We have placed additional method comparisons and experimental results in the Appendix. We simulated data from the following three models:

Model 1 (M1). A nonlinear model:

$$Y = X_1 + \exp(X_2 + X_3/3) + \sin(X_4 + X_5) + \varepsilon,$$

where $\varepsilon \sim N(0, X_1^2)$.

Model 2 (M2). A model with a multiplicative error:

$$Y = (2 + X_1^2/3 + X_2^2 + X_3^2 + X_4^2 + X_5^2)/3 \times \varepsilon,$$

where $\varepsilon \sim N(X_3, 1)$.

Model 3 (M3). A mixture model:

$$\begin{aligned} Y =& \mathbb{I}_{\{U \le 1/3\}} N\left(-3 - X_1/3 - X_2^2, 0.25\right) \\ &+ \mathbb{I}_{\{U > 1/3\}} N\left(3 + X_1/3 + X_2^2, 1\right), \end{aligned}$$

where $U \sim \text{Unif}(0,1)$ and is independent of X.

In each model, the covariate vector X is generated from $N(\boldsymbol{\mu}^{(t)}, \mathbf{I}_{100})$ in the t-th domain. So the ambient dimension of X is 100, but (M1) and (M2) only depend on the first 5 components of X and (M3) only depends on the first 2 components of X. To further demonstrate the robustness and efficiency of our approach, we consider 5 different source domains, with the corresponding values of $\boldsymbol{\mu}^{(t)}$ provided in appendix. Additionally, to demonstrate the impact of different outlier source domains, we introduce posterior drift in the fourth and fifth source domain.

Similar to the experiments conducted by Liu et al. (2021); Zhou et al. (2023), we consider the mean squared error (MSE) of the estimated conditional mean $\mathbb{E}(Y|X)$ and the estimated conditional standard deviation $\text{SD}(Y|X)$. We employ a test data set $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K\}$ of size $K = 2000$. For the proposed method, we first generate samples $\{\eta_1, \ldots, \eta_J\}$ of size $J = 10000$ from the reference distribution P_η and calculate conditional samples $\left\{\hat{G}(\eta_j, \boldsymbol{x}_k), j = 1, \ldots, J, k = 1, \ldots, K\right\}$. The estimated conditional standard deviation is calculated as the sample standard deviation of the conditional samples. The MSE of the estimated conditional mean is $\text{MSE(mean)} = (1/K)\sum_{k=1}^{K}\{\widehat{\mathbb{E}}(Y|X = \boldsymbol{x}_k) - \mathbb{E}(Y|X = \boldsymbol{x}_k)\}^2$. Similarly, the MSE of the estimated conditional standard deviation is $\text{MSE(sd)} = (1/K)\sum_{k=1}^{K}\{\widehat{\text{SD}}(Y|X = \boldsymbol{x}_k) - \text{SD}(Y|X = \boldsymbol{x}_k)\}^2$.

Based on Figure 2, in all three data simulated models, the first source domain is considered a reliable source domain, while the others are identified as outlier source domains. The MSE(mean) and MSE(sd) are summarized in Table 1. Comparing with the models trained with only target domain, **STWGAN** has the smallest MSEs error in most cases.

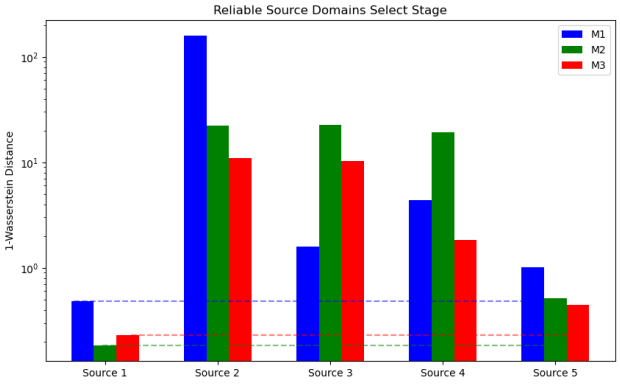


Figure 2: A simple visualization of STWGAN Stage 1.

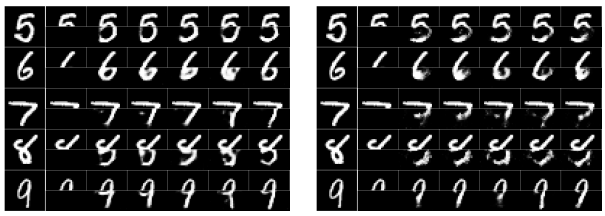
Table 1: Mean squared error (MSE) of the estimated conditional mean, the estimated standard deviation and the corresponding simulation standard errors (in parentheses). The experimental results are presented in three parts, arranged from top to bottom, corresponding to $n_t = 20000, 40000, 60000$ and $n_0 = 10000$. **Our complete experimental results are provided in the Appendix.**

		STWGAN	Target-only	Pool
M1	Mean	**15.77**(1.29)	21.49(1.24)	16.90(1.63)
	SD	4.43(1.48)	8.21(2.84)	**1.89**(0.45)
M2	Mean	**4.40**(1.10)	9.51(3.63)	6.75(2.35)
	SD	1.95(0.30)	**1.39**(0.14)	1.84(0.18)
M3	Mean	**2.22**(0.99)	25.75(4.10)	3.07(1.42)
	SD	**0.47**(0.10)	10.14(5.20)	0.75(0.10)
M1	Mean	**10.64**(2.07)	17.06(1.91)	16.94(2.94)
	SD	6.69(4.47)	7.69(3.33)	**1.37**(0.22)
M2	Mean	**3.12**(1.14)	7.10(3.01)	5.15(1.77)
	SD	1.90(0.38)	**1.53**(0.23)	2.10(0.34)
M3	Mean	**2.09**(1.39)	26.89(7.13)	2.32(1.55)
	SD	**0.54**(0.13)	7.72(4.06)	0.61(0.51)
M1	Mean	**10.73**(1.16)	24.40(2.84)	17.56(1.69)
	SD	2.84(1.59)	9.84(2.56)	**1.61**(0.56)
M2	Mean	**2.22**(1.30)	7.73(4.01)	6.96(1.42)
	SD	2.37(1.16)	**1.51**(0.20)	2.05(0.13)
M3	Mean	**1.68**(1.34)	20.97(3.40)	2.32(1.55)
	SD	**0.56**(0.06)	5.67(2.67)	0.61(0.51)

6.2 Image reconstruction: MNIST dataset

We now illustrate the application of the proposed method to high-dimensional data problems. We use the MNIST handwritten digits dataset(Deng, 2012), which contains 60000 images for training and 10000 images for testing. The images are stored in 28×28 matrices with gray color intensity from 0 to 1. We use STWGAN to help reconstruct the missing part of an image. Specifically, we consider a scenario in which only the upper or left half of an image is observed, and the task is to reconstruct the missing part. In this setting, let $X \in \mathbb{R}^{28\times 14}$ represent the observed upper or left half of the image, while $Y \in \mathbb{R}^{28\times 14}$ denotes the missing part. We refer to the two experiments corresponding to different missing parts as "upper2lower" and "left2right".

In practice, we construct the target domain and a source domain directly within the MNIST dataset. We select 5,000 images of digits 5-9 from the training set to serve as the target domain, and 50,000 images of digits 0-9 as the source domain. The experimental results are illustrated in Figure 3.



(a) STWGAN (b) Target Only

(c) STWGAN (d) Target Only

Figure 3: Comparison of STWGAN and Target Only: (a) and (b) show results of upper2lower, while figures (c) and (d) show results of left2right.

In Figure 3, the first column of each panel displays the true images, the second column shows the generator condition X, and the remaining columns present the generated images. Digits "6", "7", and "9" are reconstructed effectively with the aid of the source domain. However, when only the upper part of the digits is provided, digit "8" may be misinterpreted as "6" due to its similar structure, resulting in poorer reconstruction quality. Nonetheless, in terms of stroke consistency, the images generated after transfer exhibit greater realism.

6.3 Image-to-Image translation

We demonstrate the application of the proposed method to the task of image-to-image translation using the edges2shoes and edges2handbags datasets (Isola et al., 2017). The edges2shoes dataset includes over 40000 training images derived from the UT Zappos50K dataset (Yu and Grauman, 2017), while the edges2handbags dataset comprises more than 130000 training images from the iGAN project (Zhu et al., 2016). Each dataset consists of a real image of shoes or handbags paired with a corresponding edge map of the object, where the edges were generated using the HED edge detector (Xie and Tu, 2015). In both datasets, the edge maps and real images are stored as tensors with dimensions $1\times 286\times 286$ and $3\times 286\times 286$, respectively. Due to the smaller sample size in the edges2shoes dataset, we selected 40000 samples from it to serve as the target domain and selected 120000 samples from the edges2handbags dataset as the source domain.

(a) shoes2edges (b) edges2shoes

Figure 4: Comparison of STWGAN and Target Only.

We then conduct two sets of experiments. In the first experiment, we use the edge map as $Y \in \mathbb{R}^{81,796}$ and the real image as $X \in \mathbb{R}^{254,388}$, with the results shown

in the figure (4.a). In the second experiment, we reverse the X and Y with the results also presented in the figure (4.b). It can be observed that the STWGAN method effectively transfers knowledge of complex patterns from the edges2handbags dataset, resulting in more accurate edge representations on shoes and enhancing the richness of patterns in the generated shoe images. For example, in the case of sneakers with intricate edge details, our method often produces brighter, more vibrant, and realistic images, while other approaches tend to generate duller, grayish color patterns, lacking in vibrancy and detail.

7 CONCLUSIONS

We proposed STWGAN, a robust transfer approach designed to address the challenges of multi-source conditional generation models. This is achieved through a two-stage training process that maintains the training stability of WGAN. Our algorithm does not rely on pre-trained models from large datasets and provides both non-asymptotic error bounds and asymptotic guarantees. Future work will focus on two key aspects: (1) investigating how neural networks can learn complex dimensionality reduction structures while preserving essential information, and (2) exploring the integration of a regularization term into diffusion models to design domain-shared coupling flows, enabling effective utilization of source domain information.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China No.12171454, the Fundamental Research Funds for the Central Universities and the Shanghai Sailing Program (24YF2721900), SJTU Startup Grant. We would like to thank the reviewers who provided valuable comments.

References

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR.

Bastani, H. (2021). Predicting with proxies: Transfer learning in high dimension. *Management Science*, 67(5):2964–2984.

Cai, T. T. and Pu, H. (2024). Transfer learning for nonparametric regression: Non-asymptotic minimax analysis and adaptive procedure. *arXiv preprint arXiv:2401.12272*.

Chen, M., Liao, W., Zha, H., and Zhao, T. (2020). Distribution approximation and statistical estimation guarantees of generative adversarial networks. *arXiv preprint arXiv:2002.03938*.

Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. (2017). Joint distribution optimal transportation for domain adaptation. *Advances in neural information processing systems*, 30.

Damodaran, B. B., Kellenberger, B., Flamary, R., Tuia, D., and Courty, N. (2018). Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 447–463.

Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142.

Fernando, B., Habrard, A., Sebban, M., and Tuytelaars, T. (2013). Unsupervised visual domain adaptation using subspace alignment. In *Proceedings of the IEEE international conference on computer vision*, pages 2960–2967.

Genevay, A., Cuturi, M., Peyré, G., and Bach, F. (2016). Stochastic optimization for large-scale optimal transport. *Advances in neural information processing systems*, 29.

Gong, M., Zhang, K., Liu, T., Tao, D., Glymour, C., and Schölkopf, B. (2016). Domain adaptation with conditional transferable components. In *International conference on machine learning*, pages 2839–2848. PMLR.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.

Han, X., Zhang, Z., Ding, N., Gu, Y., Liu, X., Huo, Y., Qiu, J., Yao, Y., Zhang, A., Zhang, L., et al. (2021). Pre-trained models: Past, present and future. *AI Open*, 2:225–250.

Harun, M. Y., Lee, K., Gallardo, J., Krishnan, G., and Kanan, C. (2024). What variables affect out-of-distribution generalization in pretrained models? *arXiv preprint arXiv:2405.15018*.

He, B., Liu, H., Zhang, X., and Huang, J. (2024). Representation transfer learning for semiparametric regression. *arXiv preprint arXiv:2406.13197*.

Hoffman, J., Rodner, E., Donahue, J., Kulis, B., and Saenko, K. (2014). Asymmetric and category invariant feature transformations for domain adaptation. *International journal of computer vision*, 109:28–41.

Huang, J., Jiao, Y., Li, Z., Liu, S., Wang, Y., and Yang, Y. (2022). An error analysis of generative ad-

versarial networks for learning distributions. *Journal of machine learning research*, 23(116):1–43.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.

Kallenberg, O. (1997). *Foundations of modern probability*, volume 2. Springer.

Kandemir, M. (2015). Asymmetric transfer learning with deep gaussian processes. In *International Conference on Machine Learning*, pages 730–738. PMLR.

Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410.

Li, B. (2018). *Sufficient dimension reduction: Methods and applications with R*. Chapman and Hall/CRC.

Li, S., Cai, T. T., and Li, H. (2022). Transfer learning for high-dimensional linear regression: Prediction, estimation and minimax optimality. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(1):149–173.

Li, Y., Zhang, R., Lu, J., and Shechtman, E. (2020). Few-shot image generation with elastic weight consolidation. *arXiv preprint arXiv:2012.02780*.

Liang, T. (2021). How well generative adversarial networks learn distributions. *Journal of Machine Learning Research*, 22(228):1–41.

Lin, H. and Reimherr, M. (2022). Transfer learning for functional linear regression with structural interpretability. *arXiv preprint arXiv:2206.04277*.

Liu, S., Zhou, X., Jiao, Y., and Huang, J. (2021). Wasserstein generative learning of conditional distribution. *arXiv preprint arXiv:2112.10039*.

Long, M., Wang, J., Ding, G., Sun, J., and Yu, P. S. (2014). Transfer joint matching for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1410–1417.

Müller, A. (1997). Integral probability metrics and their generating classes of functions. *Advances in applied probability*, 29(2):429–443.

Ortego, D., Arazo, E., Albert, P., O'Connor, N. E., and McGuinness, K. (2021). Multi-objective interpolation training for robustness to label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6606–6615.

Seguy, V., Damodaran, B. B., Flamary, R., Courty, N., Rolet, A., and Blondel, M. (2017). Large-scale optimal transport and mapping estimation. *arXiv preprint arXiv:1711.02283*.

Shen, J., Qu, Y., Zhang, W., and Yu, Y. (2018). Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Song, S., Wang, T., Shen, G., Lin, Y., and Huang, J. (2023). Wasserstein generative regression. *arXiv preprint arXiv:2306.15163*.

Suh, N. and Cheng, G. (2024). A survey on statistical theory of deep learning: Approximation, training dynamics, and generative models. *arXiv preprint arXiv:2401.07187*.

Tan, Z., Zhou, L., and Lin, H. (2024). Generative adversarial learning with optimal input dimension and its adaptive generator architecture. *arXiv preprint arXiv:2405.03723*.

Tian, X. and Shen, X. (2024). Enhancing accuracy in generative models via knowledge transfer. *arXiv preprint arXiv:2405.16837*.

Tian, Y. and Feng, Y. (2023). Transfer learning under high-dimensional generalized linear models. *Journal of the American Statistical Association*, 118(544):2684–2697.

Tian, Y., Gu, Y., and Feng, Y. (2023). Learning from similar linear representations: Adaptivity, minimaxity, and robustness. *arXiv preprint arXiv:2303.17765*.

Villani, C. et al. (2009). *Optimal transport: old and new*, volume 338. Springer.

Wang, T., Huang, J., and Ma, S. (2024). Penalized generative variable selection. *arXiv preprint arXiv:2402.16661*.

Wang, Y., Gonzalez-Garcia, A., Berga, D., Herranz, L., Khan, F. S., and Weijer, J. v. d. (2020). Minegan: effective knowledge transfer from gans to target domains with few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9332–9341.

Wang, Y., Wu, C., Herranz, L., Van de Weijer, J., Gonzalez-Garcia, A., and Raducanu, B. (2018). Transferring gans: generating images from limited data. In *Proceedings of the European conference on computer vision (ECCV)*, pages 218–234.

Xie, S. and Tu, Z. (2015). Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1395–1403.

Yu, A. and Grauman, K. (2017). Semantic jitter: Dense supervision for visual comparisons via syn-

thetic images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5570–5579.

Zhao, Y., Ding, H., Huang, H., and Cheung, N.-M. (2022). A closer look at few-shot image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9140–9150.

Zhou, D., Liu, M., Li, M., and Cai, T. (2024). Doubly robust augmented model accuracy transfer inference with high dimensional features. *Journal of the American Statistical Association*, (just-accepted):1–26.

Zhou, X., Jiao, Y., Liu, J., and Huang, J. (2023). A deep generative approach to conditional sampling. *Journal of the American Statistical Association*, 118(543):1837–1848.

Zhu, J.-Y., Krähenbühl, P., Shechtman, E., and Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. In *Proceedings of European Conference on Computer Vision (ECCV)*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   See Appendix for full description.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   We have included the complete assumptions and theoretical results in the Appendix.
   (b) Complete proofs of all theoretical results. [Yes]
   We provide proofs for all the proposed theoretical results in the Appendix.
   (c) Clear explanations of any assumptions. [Yes]
   For Assumptions 1-6, we provide clear explanations and analyses in the Appendix.

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [No]
   We will open source the complete code after acceptance.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   Most training details are included in the Appendix, while the values of some commonly used hyperparameters that are not mentioned can be found in the code.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]
   Our description on this aspect is included in Section 1.1 of the Appendix.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary Materials of Conditional Generative Learning from Invariant Representations in Multi-Source: Robustness and Efficiency

1 IMPLEMENT AND EXPERIMENT DETAILS

In this section, we provide additional experimental and algorithmic details not covered in the main text. Section 1.1 includes definitions of the neural networks and the core pseudocode for the algorithms. Sections 1.2 to 1.4 cover the three experiments discussed in the main text, while Section 1.5 includes an additional experiment.

1.1 Implement

In terms of implementation details, we first provide a brief overview of feedforward neural networks (FNN) utilizing the rectified linear unit (ReLU) activation function. The ReLU function is defined as $\sigma(\boldsymbol{x}) := \max(\boldsymbol{x}, 0)$ and is applied component-wise to the input $\boldsymbol{x}$. A neural network can be represented as a composite function given by

$$\zeta(\boldsymbol{x}) = \mathcal{L}_H \circ \sigma \circ \mathcal{L}_{H-1} \circ \sigma \circ \cdots \circ \sigma \circ \mathcal{L}_1 \circ \sigma \circ \mathcal{L}_0(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathbb{R}^{p_0},$$

where $\mathcal{L}_i(\boldsymbol{x}) = \boldsymbol{W}_i\boldsymbol{x} + \boldsymbol{b}_i$ represents the i-th linear transformation with a weight matrix $\boldsymbol{W}_i \in \mathbb{R}^{p_{i+1} \times p_i}$ and a bias vector $\boldsymbol{b}_i \in \mathbb{R}^{p_{i+1}}$. Here, p_i denotes the width of the i-th layer for $i = 0, 1, \ldots, H$. The overall architecture of the network is characterized by its width, denoted as $W = \max\{p_1, \ldots, p_H\}$, and its depth, represented by H. To facilitate discussion, we denote a neural network with input dimension p_0, output dimension p_{H+1}, a maximum width of W, and a maximum depth of H as $\mathcal{NN}(p_0, p_{H+1}, W, H)$. This notation encapsulates the structural parameters of the network, allowing for a more concise representation in subsequent discussions regarding training, optimization, and performance evaluation.

We now specify the function classes below:

- For the generator network class $G_{\boldsymbol{\theta}}$: Let $\mathcal{G} \equiv \mathcal{NN}(r + m, q, W_G, H_G)$ be a class of ReLU-activated FNNs, $G_{\boldsymbol{\theta}} : \mathbb{R}^{d+m} \to \mathbb{R}^q$, with parameter $\boldsymbol{\theta}$, width W_G, and depth H_G.
- For the discriminator network class $D_{\boldsymbol{\phi}}$: Let $\mathcal{D} \equiv \mathcal{NN}(r + q, 1, W_D, H_D) \cap \mathcal{F}^1_{\text{Lip}}$ be a class of ReLU-activated FNNs, $f_\phi : \Omega \to \mathbb{R}$, with parameter ϕ, width $W_{\mathcal{D}}$, and depth $H_{\mathcal{D}}$.
- For the representation network class $R_{\boldsymbol{\omega}}$: Let $\mathcal{R} \equiv \mathcal{NN}(d, r, W_R, H_R)$ be a class of ReLU-activated FNNs, $R_{\boldsymbol{\omega}} : \mathbb{R}^d \to \mathbb{R}^r$, with parameter $\boldsymbol{\omega}$, width W_R, and depth H_R.

Algorithm 1 outlines the core component of Stage 1 in the Oracle Transfer-WGAN. To ensure that the discriminator belongs to the class of 1-Lipschitz functions, we apply the gradient penalty algorithm (Gulrajani et al., 2017). For convenience, in each minibatch, we select n_b samples from the target domain and $u_t \times n_b$ samples from the t-th source domain, where $t \in S$ and $u_0 = 1, u_t \in \mathbb{N}^+$. We denote $n_p = n_b \times \sum_{t \in S \cup \{0\}} u_t$. Additionally, it is worth mentioning that for the EOT algorithm, we use the Sinkhorn algorithm(Cuturi, 2013) to compute the 1-Wasserstein distance. The computation for Stage 2 of the Oracle Transfer-WGAN is omitted, as it is a simplified version of Algorithm 1. Algorithm 2 outlines the core component of Stage 1 in the Selected Transfer-WGAN. The computation for Stage 2 of the Selected Transfer-WGAN is omitted, as it is a simplified version of Algorithm 1. We implemented these algorithms in Pytorch.

Besides, we conduct training using a single NVIDIA GeForce RTX 4090 GPU with most training runs taking between 1-10 hours, depending on the model size and the specific experiments conducted.

Algorithm 1 Oracle Transfer-WGAN

Require: Tuning parameter λ_t, λ_{gp}; Target minibatch size $n_b \leq n_0$; Minibatch ratio u_t; Noise dimension m;

1: **for** number of training iterations in stage 1 **do**

2: $\forall t \in S \cup \{0\}$, Sample $\{(\boldsymbol{x}_{bi}^{(t)}, \boldsymbol{y}_{bi}^{(t)}\}_{i=1}^{n_b \times u_t}$ from $\{(\boldsymbol{x}_i^{(t)}, \boldsymbol{y}_i^{(t)}\}_{i=1}^{n_t}$ and $\{\eta_{bi}^{(t)}\}_{i=1}^{n_b \times u_t}$ from $N(\mathbf{0}, \boldsymbol{I}_m)$

3: Update the discriminator $D_{\boldsymbol{\phi}_1}$ by descending its stochastic gradient:

$$\nabla_{\boldsymbol{\phi}_1}\Bigg[\frac{1}{n_p}\sum_{t\in S\cup\{0\},i=1}^{n_b\times u_t}\Bigg(-D_{\boldsymbol{\phi}_1}\left(\boldsymbol{x}_{bi}^{(t)}, G_{\boldsymbol{\theta}_1}\left(\eta_i, R_{\boldsymbol{\omega}}\left(\boldsymbol{x}_{bi}^{(t)}\right)\right)\right) + D_{\boldsymbol{\phi}_1}\left(\boldsymbol{x}_{bi}^{(t)}, \boldsymbol{y}_{bi}^{(t)}\right) + \lambda_{gp}\left(\left\|\nabla_{\boldsymbol{\phi}_1} D_{\boldsymbol{\phi}_1}\left(\boldsymbol{x}_{bi}^{(t)}, \boldsymbol{y}_{bi}^{(t)}\right)\right\|_2 - 1\right)^2\Bigg)\Bigg].$$

4: Update the generator $G_{\boldsymbol{\theta}_1}$ by descending its stochastic gradient:

$$\nabla_{\boldsymbol{\theta}_1}\left[\frac{1}{n_p}\sum_{t\in S\cup\{0\},i=1}^{n_b\times u_t} D_{\boldsymbol{\phi}_1}\left(\boldsymbol{x}_i^{(t)}, G_{\boldsymbol{\theta}_1}\left(\eta_i, R_{\boldsymbol{\omega}}\left(\boldsymbol{x}_i^{(t)}\right)\right)\right)\right].$$

5: Solving the optimal transport problem using the Large-scale EOT algorithm:

$$\forall t \in S, \min_{\gamma\in\Pi\left(P_{X,Y}^{n_b\times u_t}, P_{X,Y}^{n_b}\right)} \left\langle \gamma, \mathbf{C}_{R_{\boldsymbol{\omega}}}^{(t)} \right\rangle_F.$$

6: Update the representation $R_{\boldsymbol{\omega}}$ by descending its stochastic gradient:

$$\nabla_{\boldsymbol{\omega}}\Bigg[\frac{1}{n_p}\sum_{t\in S\cup\{0\},i=1}^{n_b\times u_t}\left(D_{\boldsymbol{\phi}_1}\left(\boldsymbol{x}_i^{(t)}, G_{\boldsymbol{\theta}_1}\left(\eta_i, R_{\boldsymbol{\omega}}\left(\boldsymbol{x}_i^{(t)}\right)\right)\right) - D_{\boldsymbol{\phi}_1}\left(\boldsymbol{x}_i^{(t)}, \boldsymbol{y}_i^{(t)}\right)\right) + \sum_{t\in S}\lambda_t \min_{\gamma\in\Pi\left(P_{X,Y}^{n_b\times u_t}, P_{X,Y}^{n_b}\right)} \left\langle \gamma, \mathbf{C}_{R_{\boldsymbol{\omega}}}^{(t)} \right\rangle_F\Bigg].$$

7: **end for**

Algorithm 2 Selected Transfer-WGAN

Require: Tuning parameter λ_t, λ_{gp}; Target minibatch size $n_b \leq n_0$; Minibatch ratio u_t; Noise dimension m; Threshold M;

1: **for** number of training iterations in stage 1 **do**

2: $\forall t$, Sample $\{(\boldsymbol{x}_{bi}^{(t)}, \boldsymbol{y}_{bi}^{(t)}\}_{i=1}^{n_b \times u_t}$ from $\{(\boldsymbol{x}_i^{(t)}, \boldsymbol{y}_i^{(t)}\}_{i=1}^{n_t}$ and $\{\eta_{bi}^{(t)}\}_{i=1}^{n_b \times u_t}$ from $N(\mathbf{0}, \boldsymbol{I}_m)$

3: Update the *full model* neural networks $D_{\tilde{\boldsymbol{\phi}}}, G_{\tilde{\boldsymbol{\theta}}}$ and $R_{\tilde{\boldsymbol{\omega}}}$ similar to Algorithm 1.

4: **end for**

5: Calculating the 1-Wasserstein distance using the EOT algorithm:

$$\forall t, \min_{\gamma\in\Pi\left(P_{X,Y}^{n_t}, P_{X,Y}^{n_0}\right)} \left\langle \gamma, \mathbf{C}_{R_{\tilde{\boldsymbol{\omega}}}}^{(t)} \right\rangle_F.$$

6: Estimate $\hat{S}$ with threshold M:

$$\hat{S} \leftarrow \{t : W_1(P_{R_{\tilde{\boldsymbol{\omega}}}(X),Y}^{n_t}, P_{R_{\tilde{\boldsymbol{\omega}}}(X),Y}^{n_0}) \leq M\}.$$

1.2 Nonparametric conditional density estimation

We will first present the network architecture, followed by the details of the simulated dataset. For the proposed method, the conditional generator G is parameterized using a neural network in $\mathcal{NN}(r+m, q, 512, 2)$. The discriminator D is parameterized using a neural network in $\mathcal{NN}(r+q, 1, 128, 2)$ and the representation R is parameterized using a neural network in $\mathcal{NN}(d, r, 512, 2)$. In practice, we experimented with different values of r and found that choosing a value slightly larger than the ground-true yields the best results. Thus, we set $r = 10$ for models (M1) and (M2), and $r = 5$ for model (M3). The noise vector η is drawn from $N(\mathbf{0}, \mathbf{I}_3)$ in models (M1) and (M2), and from $N(0,1)$ in model (M3). We set the other hyperparameters as follows: $\forall t \in S, \lambda_t = 0.1, \text{epochs} = 300, \text{batch size} = 64$, and the optimizer is Adam, with an initial learning rate of 0.0001.

Table 1: The value of $\mu^{(t)}$, where the index (t) represents the domain, with (0) denoting the target domain.

(t)	$\boldsymbol{\mu}^{(t)}$	posterior drift
(0)	$(2,1,0,\ldots,0)^\top$	-
(1)	$(0,0,0,\ldots,0)^\top$	No
(2)	$(5,5,5,\ldots,5)^\top$	No
(3)	$(-5,\ldots,-5)^\top$	No
(4)	$(2,1,0,\ldots,0)^\top$	Yes
(5)	$(2,1,0,\ldots,0)^\top$	Yes

We consider the posterior drift in the fourth and fifth source domains across different data generation models. For convenience, the two source domains will be referred to as (4) and (5) below.

Model 1 (M1). A nonlinear model with an additive error term:

$$\begin{aligned}(4): &\quad Y = 5X_1 + \exp(X_2 + X_3/3 + 2) + \cos(X_4 + X_5) + \varepsilon + 5, \varepsilon \sim N(0, X_1^2),\\ (5): &\quad Y = X_1/5 + \exp(X_2 + X_3/3 - 2) + \cos(X_4 + X_5) + \varepsilon - 5, \varepsilon \sim N(0, X_1^2),\end{aligned}$$

Model 2 (M2). A model with a multiplicative Gaussian error term:

$$\begin{aligned}(4): &\quad Y = (7 + X_1^3/3 + X_2^3 + X_3^3 + X_4^3 + X_5^3) \times \varepsilon + 5,\\ (5): &\quad Y = (-3 + X_1 + X_2 + X_3 + X_4 + X_5) \times \varepsilon - 5,\end{aligned}$$

where $\varepsilon \sim N(X_3, 1)$

Model 3 (M3). A mixture of two normal distributions:

$$\begin{aligned}(4): &\quad Y = \mathbb{I}_{\{U \le 1/3\}} N\left(-8 - X_1^3 - X_2, 0.25\right) + \mathbb{I}_{\{U > 1/3\}} N\left(8 + X_1^3 + X_2, 1\right),\\ (5): &\quad Y = \mathbb{I}_{\{U \le 1/3\}} N\left(2 - X_1 - X_2, 0.25\right) + \mathbb{I}_{\{U > 1/3\}} N\left(-2 + X_1 + X_2, 1\right),\end{aligned}$$

where $U \sim \text{Unif}(0,1)$ and is independent of X.

In addition to the three methods, we also considered a more straightforward approach using pre-trained fine-tuning models, referred to as **PT-FT**. In this method, we trained the model on the source domain for 450 epochs before continuing the training on the target domain. Additionally, it is worth noting that we did not use the representation network in both the Target-only and PT-FT methods. These are the complete experimental results, where we considered different sample sizes for the source domains while keeping $n_0 = 10,000$ fixed, as shown in Table 2.

1.3 Image reconstruction: MNIST dataset

Regarding this experiment, we provide the details of the network architecture here. To maintain maximum consistency between theoretical conditions and experimental setup, we opted not to use convolutional neural

Table 2: Mean squared error (MSE) of the estimated conditional mean, the estimated standard deviation and the corresponding simulation standard errors (in parentheses).

			STWGAN	Target-only	Pool	PT-FT
$n_t = 20,000$	M1	Mean	**15.77**(1.29)	21.49(1.24)	16.90(1.63)	77.87(11.27)
		SD	4.43(1.48)	8.21(2.84)	**1.89**(0.45)	2.17(1.22)
	M2	Mean	4.40(1.10)	9.51(3.63)	6.75(2.35)	**3.83**(2.82)
		SD	1.95(0.30)	**1.39**(0.14)	1.84(0.18)	2.08(0.33)
	M3	Mean	**2.22**(0.99)	25.75(4.10)	3.07(1.42)	3.07(1.02)
		SD	**0.47**(0.10)	10.14(5.20)	0.75(0.10)	9.94(1.69)
$n_t = 40,000$	M1	Mean	**10.64**(2.07)	17.06(1.91)	16.94(2.94)	81.76(11.55)
		SD	6.69(4.47)	7.69(3.33)	**1.37**(0.22)	1.66(0.46)
	M2	Mean	**3.12**(1.14)	7.10(3.01)	5.15(1.77)	5.01(2.81)
		SD	1.90(0.38)	**1.53**(0.23)	2.10(0.34)	2.67(1.04)
	M3	Mean	**2.09**(1.39)	26.89(7.13)	2.32(1.55)	2.96(0.82)
		SD	**0.54**(0.13)	7.72(4.06)	0.61(0.51)	8.09(2.72)
$n_t = 60,000$	M1	Mean	**10.73**(1.16)	24.40(2.84)	17.56(1.69)	85.43(14.43)
		SD	2.84(1.59)	9.84(2.56)	1.61(0.56)	**1.60**(0.81)
	M2	Mean	**2.22**(1.30)	7.73(4.01)	6.96(1.42)	5.27(3.10)
		SD	2.37(1.16)	**1.51**(0.20)	2.05(0.13)	3.46(1.36)
	M3	Mean	**1.68**(1.34)	20.97(3.40)	2.32(1.55)	2.66(1.03)
		SD	**0.56**(0.06)	5.67(2.67)	0.61(0.51)	8.41(2.19)

network (CNN) when handling the image dataset. Instead, we continued to use a simple feedforward neural network (FNN). The proposed method utilizes a conditional generator G parameterized by a neural network $\mathcal{NN}(m+r, q, 512, 2)$, a discriminator D parameterized by a neural network $\mathcal{NN}(d+q, 1, 1024, 3)$, and a representation network R parameterized by a neural network $\mathcal{NN}(d, r, 512, 2)$. The random noise vector $\eta \sim N(\mathbf{0}, \mathbf{I}_{10})$, and we set $r = 50$. We set the other hyperparameters as follows: $\forall t \in S, \lambda_t = 0.1$, epochs $= 300$, batch size $= 64$, and the optimizer used is RMSprop.

Furthermore, for a comprehensive evaluation, we employ two widely-used metrics: Fréchet Inception Distance (FID) and Learned Perceptual Image Patch Similarity (LPIPS). While FID measures the similarity between generated and real images by comparing their feature distributions in a pre-trained Inception network, LPIPS quantifies perceptual similarity by leveraging deep features from a pre-trained convolutional neural network. Notably, both metrics are not limited to evaluating image generation tasks but are also effective for assessing image reconstruction tasks such as inpainting and super-resolution(Chung et al., 2022). In the MNIST left2right experiment, we generated 25,000 images using the trained model to calculate the FID and LPIPS scores shown in table 3.

Table 3: Comparison of STWGAN and Target-only on FID and LPIPS scores in left2right experiment.

	STWGAN	Target-only
FID	**77.19**	96.98
LPIPS	**0.0698**	0.0717

1.4 Image-to-Image translation

In this experiment, considering the issue of image size, we followed the neural network setup from previous work and chose not to continue using FNNs. This setup aligns with the image experiment settings used in all current FNN-related theoretical studies. Our innovation lies in incorporating a novel regularization term into the method. In this work, we build upon the architecture proposed by Isola et al. (2017), making modifications only to the Generator network, where we adopt the UNet256 structure. Specifically, we treat the first half of their Unet-based Generator (Ronneberger et al., 2015) as our Representation network, where the vector generated at the lowest layer of the "U"-shaped structure is considered the domain-invariant representation. Therefore, we

set $r = 512$. The second half of the Unet is regarded as our Generator network. The rest of the architecture, including the Discriminator, loss functions, and other hyperparameters, remain unchanged from the original setup.

1.5 Conditional prediction

In this experiment, we consider the abalone dataset as Liu et al. (2021), which is available in the UCI Machine Learning Repository(Dua et al., 2017), includes physical measurements of abalone and their corresponding number of rings, which are used to determine age. The age determination process involves cutting the shell, staining it, and counting the rings under a microscope, which is labor-intensive. To simplify age prediction, other easily measurable attributes are utilized. The dataset comprises 9 variables: sex, length, diameter, height, whole weight, shucked weight, viscera weight, shell weight, and rings. All variables except sex are continuous. In this experiment, the number of rings is treated as the response variable $Y \in \mathbb{R}$, while the remaining measurements form the covariate vector $X \in \mathbb{R}^9$. The categorical variable sex represents three groups: female, male, and infant. We treat the infant group as the target and the others as sources for conditional prediction of rings.

For more baseline, we refer to the latest paper that presents the method **MSSG**(Lai et al., 2024). This method focuses on a two-source scenario, where the target dataset is created by concatenating the two source datasets, with no third dataset involved. Considering our task is conditional prediction, we use mean squared error(MSE) as evaluation. For the proposed method, the conditional generator G is parameterized using a neural network in $\mathcal{NN}(r+m, q, 512, 2)$. The discriminator D is parameterized using a neural network in $\mathcal{NN}(r+q, 1, 128, 2)$ and the representation R is parameterized using a neural network in $\mathcal{NN}(d, r, 512, 2)$. The results are listed in table 4 ($n_t = 1000, T = |S| = 2, |\hat{S}| = 2$). In this experiment, **Pool** performs well because the infant group is likely a combination of the male and female groups, which also aligns with the problem scenario of the **MSSG** algorithm. The algorithm does not require target data, therefore, the quantity of n_0 in the experiment does not affect the **MSSG** results.

Table 4: Mean squared error (MSE) of the age prediction and the corresponding standard errors (in parentheses).

	STWGAN	MSSG	Target-only	Pool	PT-FT
n_0=100	**3.72(0.25)**	6.184(0.082)	4.80(0.61)	3.74(0.13)	3.93(0.28)
n_0=500	3.21(0.17)	6.184(0.082)	3.59(0.17)	**2.93(0.06)**	6.55(0.33)

2 A HIGH LEVEL DESCRIPTION OF THE ERROR ANALYSIS

Below we first present a high level description of the error analysis. For the estimator $\hat{G}$ of the conditional generator, we are interested in bounding the error $d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P^{(0)}_{\hat{Z},Y}\right)$. Our basic idea is to decompose this error into terms that are easier to analyze. Given that we have proposed algorithms for different scenarios, but the analysis is quite similar, we will use the oracle algorithm as an example.

Let $\left\{\left(\boldsymbol{x}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}, \eta_i^{(t)\prime}\right), i = 1, \cdots n_t, t = 0, \cdots, T\right\}$ be ghost samples that are independent of the original samples. Here we introduce ghost samples as a technical tool for bounding the stochastic error term $\mathcal{E}_3, \mathcal{E}_4$ defined below. We consider $(\hat{G}, \hat{D}, \hat{R})$ based on the empirical version of $\mathcal{L}(G, D, R)$ that depends on the original samples $\left(\boldsymbol{x}_i^{(t)}, \boldsymbol{y}_i^{(t)}, \eta_i^{(t)}\right)$ given in Algorithm 1 and $\left(\hat{G}', \hat{D}', \hat{R}'\right)$ based on the loss function of the ghost samples $\left(\boldsymbol{x}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}, \eta_i^{(t)\prime}\right)$.

Recall the error $d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P^{(0)}_{\hat{Z},Y}\right)$ is defined by

$$d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P^{(0)}_{\hat{Z},Y}\right) = \sup_{f \in \mathcal{F}_B^1} \{\mathbb{E} f(\hat{Z}, \hat{G}) - \mathbb{E}_{P^{(0)}_{\hat{Z},Y}} f(\hat{Z}, Y)\}. \tag{1}$$

We consider mixture distribution $P_{\hat{Z},Y} = \sum_{t \in S \cup \{0\}} \frac{n_t}{n} P^{(t)}_{\hat{Z},Y}, n = \sum_{t \in S \cup \{0\}} n_t$ and denote $\hat{\boldsymbol{z}}_i^{(t)} = \hat{R}(\boldsymbol{x}_i^{(t)}), \hat{\boldsymbol{z}}_i^{(t)\prime} =$

$\hat{R}'(\boldsymbol{x}_i^{(t)\prime})$. Then we decompose (1) as follows:

$$
\begin{aligned}
&d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P_{\hat{Z},Y}^{(0)}\right) \leq d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P_{\hat{Z},Y}\right) + d_{\mathcal{F}_B^1}\left(P_{\hat{Z},Y}, P_{\hat{Z},Y}^{(0)}\right) \\
&= d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P_{\hat{Z},Y}\right) + \sum_{t\in S\cup\{0\}} \frac{n_t}{n} d_{\mathcal{F}_B^1}\left(P_{\hat{Z},Y}^{(t)}, P_{\hat{Z},Y}^{(0)}\right) \\
&\leq \sup_{f\in\mathcal{F}_B^1}\left\{\mathbb{E}f(\hat{Z},\hat{G}) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \hat{G}'\right)\right\} \\
&\quad + \sup_{f\in\mathcal{F}_B^1}\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \hat{G}'\right) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}\right)\right\} \\
&\quad + \sup_{f\in\mathcal{F}_B^1}\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}\right) - \mathbb{E}_{P_{\hat{Z},Y}} f(\hat{Z},Y)\right\} + \sum_{t\in S}\frac{n_t}{n} d_{\mathcal{F}_B^1}\left(P_{\hat{Z},Y}^{(t)}, P_{\hat{Z},Y}^{(0)}\right) \\
&:= \mathcal{E}_4 + A + \mathcal{E}_5 + B,
\end{aligned}
$$

where $A = \sup_{f\in\mathcal{F}_B^1}\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \hat{G}'\right) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}\right)\right\}$, $B = \sum_{t\in S\cup\{0\}}\frac{n_t}{n} d_{\mathcal{F}_B^1}\left(P_{\hat{Z},Y}^{(t)}, P_{\hat{Z},Y}^{(0)}\right)$,

$$
\mathcal{E}_4 = \sup_{f\in\mathcal{F}_B^1}\left\{\mathbb{E}f(\hat{Z},\hat{G}) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \hat{G}'\right)\right\}, \tag{2}
$$

and

$$
\mathcal{E}_5 = \sup_{f\in\mathcal{F}_B^1}\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}\right) - \mathbb{E}_{P_{\hat{Z},Y}} f(\hat{Z},Y)\right\}. \tag{3}
$$

By Lemma 3.1, we have

$$
\begin{aligned}
A &\leq 2\sup_{f\in\mathcal{F}_B^1}\inf_{\boldsymbol{\phi}}\|f - D_\phi\|_\infty + \sup_{\boldsymbol{\phi}}\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \hat{G}'\right) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}\right)\right\} \\
&= 2\sup_{f\in\mathcal{F}_B^1}\inf_{\phi}\|f - D_{\boldsymbol{\phi}}\|_\infty + \inf_{\theta}\sup_{\phi}\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, G_{\boldsymbol{\theta}}\right) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}\right)\right\} \\
&:= 2\mathcal{E}_1 + \mathcal{E}_7,
\end{aligned}
$$

where

$$
\mathcal{E}_1 = \sup_{f\in\mathcal{F}_B^1}\inf_{\boldsymbol{\phi}}\|f - D_{\boldsymbol{\phi}}\|_\infty, \tag{4}
$$

and

$$
\mathcal{E}_7 = \inf_{\theta}\sup_{\phi}\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, G_{\boldsymbol{\theta}}\right) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime}\right)\right\}. \tag{5}
$$

By lemma 3.2 and Assumption 3, we have

$$
\begin{aligned}
B &\leq \sum_{t\in S}\frac{n_t}{n}\left[\mathbb{E}_{P^{(t)}_{\hat{Z}}} d_{\mathcal{F}^1_B}\left(P^{(t)}_{Y|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) + K d_{\mathcal{F}^1_B}\left(P^{(t)}_{\hat{Z}}, P^{(0)}_{\hat{Z}}\right)\right] \\
&\leq \sum_{t\in S}\frac{n_t}{n}\left\{\mathbb{E}_{P^{(t)}_{\hat{Z}}} d_{\mathcal{F}^1_B}\left(P^{(t)}_{Y|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) + K\left[d_{\mathcal{F}^1_B}\left(P^{n_t}_{\hat{Z}}, P^{(t)}_{\hat{Z}}\right) + d_{\mathcal{F}^1_B}\left(P^{n_0}_{\hat{Z}}, P^{(0)}_{\hat{Z}}\right)\right] + K d_{\mathcal{F}^1_B}\left(P^{n_t}_{\hat{Z}}, P^{n_0}_{\hat{Z}}\right)\right\} \\
&\leq \sum_{t\in S}\frac{n_t}{n}\left\{\mathbb{E}_{P^{(t)}_{\hat{Z}}} d_{\mathcal{F}^1_B}\left(P^{(t)}_{Y|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) + K\left[\sup_{f\in\mathcal{F}^1_B}\left\{\frac{1}{n}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}^{(t)\prime}_i\right) - \mathbb{E}_{P^{(t)}_{\hat{Z}}} f(\hat{Z})\right\}\right.\right. \\
&\quad \left.\left. + \sup_{f\in\mathcal{F}^1_B}\left\{\frac{1}{n}\sum_{i=1}^{n_0} f\left(\hat{\boldsymbol{z}}^{(0)\prime}_i\right) - \mathbb{E}_{P^{(0)}_{\hat{Z}}} f(\hat{Z})\right\}\right] + K d_{\mathcal{F}^1_B}\left(P^{n_t}_{\hat{Z}}, P^{n_0}_{\hat{Z}}\right)\right\} \\
&\leq h + \mathcal{E}_6 + \mathcal{E}_8,
\end{aligned}
$$

where

$$
\mathcal{E}_6 = K\sum_{t\in S}\frac{n_t}{n}\left[\sup_{f\in\mathcal{F}^1_B}\left\{\frac{1}{n}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}^{(t)\prime}_i\right) - \mathbb{E}_{P^{(t)}_{\hat{Z}}} f(\hat{Z})\right\} + \sup_{f\in\mathcal{F}^1_B}\left\{\frac{1}{n}\sum_{i=1}^{n_0} f\left(\hat{\boldsymbol{z}}^{(0)\prime}_i\right) - \mathbb{E}_{P^{(0)}_{\hat{Z}}} f(\hat{Z})\right\}\right], \tag{6}
$$

and

$$
\mathcal{E}_8 = K\sum_{t\in S}\frac{n_t}{n} d_{\mathcal{F}^1_B}\left(P^{n_t}_{\hat{Z}}, P^{n_0}_{\hat{Z}}\right). \tag{7}
$$

We combine the $\mathcal{E}_7$ and $\mathcal{E}_8$ into $\mathcal{E}_2$, because they describe how powerful the generator class and representation class are in realizing the empirical version of the noise outsourcing lemma and reducing distributional differences,

$$
\begin{aligned}
\mathcal{E}_7 + \mathcal{E}_8 \leq \mathcal{E}_2 := \inf_{\boldsymbol{\theta}}\sup_{\boldsymbol{\phi}}&\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{z}}^{(t)\prime}_i, G_{\boldsymbol{\theta}}\right) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{z}}^{(t)\prime}_i, \boldsymbol{y}^{(t)\prime}_i\right)\right\} \\
&+ \inf_{\boldsymbol{\omega}}\left\{K\sum_{t\in S}\frac{n_t}{n} d_{\mathcal{F}^1_B}\left(P^{n_t}_{R_{\boldsymbol{\omega}}(X)}, P^{n_0}_{R_{\boldsymbol{\omega}}(X)}\right)\right\},
\end{aligned} \tag{8}
$$

where this inequality holds because it is easy to see that

$$
\arg\min_{\boldsymbol{\omega}} K\sum_{t\in S}\frac{n_t}{n} d_{\mathcal{F}^1_B}\left(P^{n_t}_{R_{\boldsymbol{\omega}}(X)}, P^{n_0}_{R_{\boldsymbol{\omega}}(X)}\right) \subset \arg\min_{\boldsymbol{\omega}} \mathcal{L}_1(R_{\boldsymbol{\omega}}, G_{\hat{\boldsymbol{\theta}}_1}, D_{\hat{\boldsymbol{\phi}}_1}, S).
$$

We also combine the $\mathcal{E}_5$ and $\mathcal{E}_6$ into $\mathcal{E}_3$ because they both describe the distance between the distribution and its empirical distribution,

$$
\begin{aligned}
\mathcal{E}_3 := \mathcal{E}_5 + \mathcal{E}_6 = &\sup_{f\in\mathcal{F}^1_B}\left\{\frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}^{(t)\prime}_i, \boldsymbol{y}^{(t)\prime}_i\right) - \mathbb{E}_{P_{\hat{Z},Y}} f(\hat{Z}, Y)\right\} \\
&+ K\sum_{t\in S}\frac{n_t}{n}\left[\sup_{f\in\mathcal{F}^1_B}\left\{\frac{1}{n}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}^{(t)\prime}_i\right) - \mathbb{E}_{P^{(t)}_{\hat{Z}}} f(\hat{Z})\right\} + \sup_{f\in\mathcal{F}^1_B}\left\{\frac{1}{n}\sum_{i=1}^{n_0} f\left(\hat{\boldsymbol{z}}^{(0)\prime}_i\right) - \mathbb{E}_{P^{(0)}_{\hat{Z}}} f(\hat{Z})\right\}\right].
\end{aligned} \tag{9}
$$

By their definitions, we can see that $\mathcal{E}_1, \mathcal{E}_2$ are approximation errors; $\mathcal{E}_3, \mathcal{E}_4$ are stochastic errors. We summarize the above derivation in the following lemma.

Lemma 2.1. Let $\hat{G} = G_{\hat{\theta}}, \hat{R} = R_{\hat{\omega}}$ be the minimax solution in oracle algorithm. Then the bounded Lipschitz distance between $P_{\hat{Z},\hat{G}}$ and $P^{(0)}_{\hat{Z},Y}$ can be decomposed as follows.

$$
d_{\mathcal{F}^1_B}\left(P_{\hat{Z},\hat{G}}, P^{(0)}_{\hat{Z},Y}\right) \leq 2\mathcal{E}_1 + \mathcal{E}_2 + \mathcal{E}_3 + \mathcal{E}_4 + h. \tag{10}
$$

Theorems 4.1-5.2 are proved based on the error decomposition (10).

3 THEORETICAL RESULTS

In this section, we present the supporting lemmas and prove the theorems. First, in Section 3.1, we outline the assumptions and non-asymptotic error bounds of the Oracle Transfer-WGAN algorithm. Then, in Section 3.2, we present the lemmas required for our proofs, followed by a discussion of certain equivalences in Section 3.3. In Section 3.4, we provide non-asymptotic upper bound estimates for the decomposed errors introduced in Section 2. Finally, in Section 3.5, we complete the proofs of all the theorems.

3.1 Assumption and non-asymptotic error bound of oracle transfer-WGAN

For the sake of analysis, we make the following mild assumptions. To avoid confusion, we denote $\hat{Z} = \hat{R}(X)$ and any random variable without domain index (t) should be understood as referring to all domains, including both reliable source and target domains. Additionally, we believe that the assumptions regarding $\tilde{Z}$ also hold for $\hat{Z}$, as the latter eliminates the influence of outlier source domains during training.

Assumption 1. The similarity measure between the outlier sources and the target domain is assumed to be of a much larger order than h. Specifically, we assume

$$\forall t \in S^c, \mathbb{E}_{P_{\tilde{Z}}^{(t)}} d_{\mathcal{F}_B^1}\left(P_{Y|\tilde{Z}}^{(t)}, P_{Y|\tilde{Z}}^{(0)}\right) = O(h^\alpha), \alpha > 1.$$

Assumption 2. For some $\delta > 0, (\hat{Z}, Y)$ satisfies the first-order moment tail condition, for any $n \geq 1$,

$$\mathbb{E}\left[\|(\hat{Z}, Y)\| \mathbb{I}_{\{\|(\hat{Z},Y)\| > \log n\}}\right] = O\left(n^{-(\log n)^\delta/(r+q)}\right).$$

Assumption 3. The noise distribution P_η is absolutely continuous with respect to the Lebesgue measure.

Assumption 4. The IPM distance between the conditional distributions of reliable source domains and the target domain is bounded in expectation, for some $h = O\left(\max\left\{n^{-1/r+q}, n_0^{-1/r}\right\}\right)$,

$$\forall t \in S \subset [T], \quad \mathbb{E}_{P_{\hat{Z}}^{(t)}} d_{\mathcal{F}_B^1}\left(P_{Y|\hat{Z}}^{(t)}, P_{Y|\hat{Z}}^{(0)}\right) \leq h.$$

Assumption 5. The conditional distribution of the target domain satisfies a certain Lipschitz condition under the Total Variation (TV) distance, for some $K > 1$:

$$\forall \hat{\boldsymbol{z}}_1, \hat{\boldsymbol{z}}_2, d_{TV}\left(P_{Y|\hat{Z}=\hat{\boldsymbol{z}}_1}^{(0)}, P_{Y|\hat{Z}=\hat{\boldsymbol{z}}_2}^{(0)}\right) \leq \frac{K-1}{2B} \left\|\hat{\boldsymbol{z}}_1 - \hat{\boldsymbol{z}}_2\right\|_1,$$

where $d_{TV}(\cdot, \cdot)$ is the TV distance.The TV distance is a measure of the difference between two probability distributions. It is defined as: $d_{TV}(P, Q) = \frac{1}{2}\int |p(x) - q(x)| \mathrm{d}x$, where $p(x)$ and $q(x)$ are the probability density functions of the distributions P and Q, respectively.

Assumption 6. The conditional distribution of the target domain satisfies a certain Lipschitz condition under the 1-Wasserstein distance, for some $K > 1$:

$$\forall \tilde{\boldsymbol{z}}_1, \tilde{\boldsymbol{z}}_2 \in \mathcal{Z}, W_1\left(P_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_1}^{(0)}, P_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_2}^{(0)}\right) \leq (K-1) \left\|\tilde{\boldsymbol{z}}_1 - \tilde{\boldsymbol{z}}_2\right\|_1,$$

where $W_1(\cdot, \cdot)$ is the 1-Wasserstein distance. Gibbs and Su (2002) conducted a detailed comparison between the Total Variation (TV) distance and the 1-Wasserstein distance, revealing that there is no strict dominance between the two measures.

In addition to Assumption 1 presented in the main text, Assumptions 2 and 3 are standard conditions commonly found in the literature. Assumption 2 is a technical assumption used to handle the case where the support of $P_{\hat{Z},Y}$ is an unbounded set. When the support of $P_{\hat{Z},Y}$ is bounded, this assumption is naturally satisfied, which aligns with most practical scenarios. Assumption 3 is a standard assumption, typically satisfied when P_η is taken as the standard normal distribution. Assumption 4 constrains the differences in the conditional distributions between all reliable source domains and the target domain to be bounded in expectation, considering it as part of

the bias introduced by transfer. The assumption naturally holds when $P^{(t)}_{Y|\hat{Z}} = P^{(0)}_{Y|\hat{Z}}$ or when h is a uniform upper bound of $d_{\mathcal{F}_B^1}\left(P^{(t)}_{Y|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right)$ with respect to $\hat{Z}$. Many works directly assume that the conditional distributions are identical, which is a relatively strong assumption (Fernando et al., 2013; Long et al., 2014; Gong et al., 2016). In contrast, Assumption 3 is more relaxed. Assumption 5-6 is a technical assumption requiring the conditional distribution on target domain to satisfy a certain uniform continuity.

For the generator network $G_{\boldsymbol{\theta}}$, we require that

$$\|G_{\boldsymbol{\theta}}\|_\infty \leq \log n. \tag{11}$$

This condition is satisfied by adding an additional clipping layer ℓ after the original output layer of the network,

$$\ell(a) = a \wedge c_n \vee (-c_n) = \sigma\left(a + c_n\right) - \sigma\left(a - c_n\right) - c_n,$$

where $c_n = \log n$. We truncate the value of $\|G_\theta\|$ to an increasing cube $[-\log n, \log n]^q$ so that the support of the evaluation function to $[-\log n, \log n]^{r+q}$. This restricts the evaluation function class to a $2\log n$ domain.

Based on this, we derive the following three theorems, numbered according to the original section sequence, with their proofs provided in Section 3.5.

Theorem 4.1 *Suppose Assumptions 2-5 hold. Let (W_D, H_D) of $D_{\boldsymbol{\phi}}$, (W_G, H_G) of $G_{\boldsymbol{\theta}}$ and (W_R, H_R) of $R_{\boldsymbol{\omega}}$ be specified such that $W_D H_D = \lceil\sqrt{n}\rceil, W_G^2 H_G = c_1 qn$ and $W_R^2 H_R = c_2 rn$ for some constants $12 \leq c_1, c_2 \leq 384$. Let $n = \sum_{t\in S\cup\{0\}} n_t$, we have:*

$$\mathbb{E}_{\hat{G}} d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P^{(0)}_{\hat{Z},Y}\right) \precsim n^{-1/(r+q)}\log n + n_0^{-1/r}.$$

When $P_{Z,Y}$ has a bounded support, we can drop the logarithm factor in the first term.

Theorem 4.2 *Suppose that $P_{\hat{Z},Y}$ is supported on $[-U, U]^{r+q}$ for some $U > 0$ and Assumptions 3-5 hold. Let (W_D, H_D) of $D_{\boldsymbol{\phi}}$, (W_G, H_G) of $G_{\boldsymbol{\theta}}$ and (W_R, H_R) of $R_{\boldsymbol{\omega}}$ be specified such that $W_D H_D = \lceil\sqrt{n}\rceil, W_G^2 H_G = c_1 qn$ and $W_R^2 H_R = c_2 rn$ for some constants $12 \leq c_1, c_2 \leq 384$. Let the output of $G_{\boldsymbol{\theta}}$ be on $[-U, U]^q$. Let $n = \sum_{t\in S\cup\{0\}} n_t$, we have:*

$$\mathbb{E}_{\hat{G}} d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P^{(0)}_{\hat{Z},Y}\right) \precsim n^{-1/(r+q)} + n_0^{-1/r}.$$

Theorem 4.3 *Under the conditions of Theorem 4.2, we have*

$$\mathbb{E}_{\hat{G}}\mathbb{E}_{P^{(0)}_{\hat{Z}}} d_{\mathcal{F}_B^1}\left(P_{\hat{G}|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) \precsim n^{-1/(r+q)} + n_0^{-1/r}.$$

Remark. The most challenging part is addressing the term $n_0^{-1/r}$ even after transfer. The problem lies in the fact that learning a generative model requires learning the complete distribution information, not just simple statistics like conditional mean and variance. Thus, we cannot avoid the process of approximating the true distribution with the empirical distribution from samples, which introduces significant bias.

3.2 Proofs of the lemmas

Now, we will introduce the following lemmas and prove some of our propositions.

Lemma 3.1(Liu et al. (2021) Lemma 3.1). *For any symmetric function classes $\mathcal{F}$ and $\mathcal{H}$, denote the approximation error $\mathcal{E}(\mathcal{H}, \mathcal{F})$ as*

$$\mathcal{E}(\mathcal{H}, \mathcal{F}) := \sup_{h\in\mathcal{H}} \inf_{f\in\mathcal{F}} \|h - f\|_\infty,$$

then for any probability distributions μ and ν,

$$d_{\mathcal{H}}(\mu, \nu) - d_{\mathcal{F}}(\mu, \nu) \leq 2\mathcal{E}(\mathcal{H}, \mathcal{F}).$$

This inequality can be extended to an empirical version by using empirical measures.

Lemma 3.2 *Suppose Assumption 5 holds. We have:*

$$\forall t \in [T], d_{\mathcal{F}_B^1}\left(P^{(t)}_{\hat{Z},Y}, P^{(0)}_{\hat{Z},Y}\right) \leq \mathbb{E}_{P^{(t)}_{\hat{Z}}} d_{\mathcal{F}_B^1}\left(P^{(t)}_{Y|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) + K d_{\mathcal{F}_B^1}\left(P^{(t)}_{\hat{Z}}, P^{(0)}_{\hat{Z}}\right).$$

Proof. First, we can expand it according to the definition,

$$\begin{aligned}
d_{\mathcal{F}_B^1}\left(P^{(t)}_{\hat{Z},Y}, P^{(0)}_{\hat{Z},Y}\right) &= \sup_{f\in\mathcal{F}_B^1}\left\{\mathbb{E}_{P^{(t)}_{\hat{Z},Y}} f(\hat{\boldsymbol{z}},\boldsymbol{y}) - \mathbb{E}_{P^{(0)}_{\hat{Z},Y}} f(\hat{\boldsymbol{z}},\boldsymbol{y})\right\} \\
&= \sup_{f\in\mathcal{F}_B^1} \int\int f(\hat{\boldsymbol{z}},\boldsymbol{y}) p^{(t)}_{\hat{Z},Y}(\hat{\boldsymbol{z}},\boldsymbol{y}) - f(\hat{\boldsymbol{z}},\boldsymbol{y}) p^{(0)}_{\hat{Z},Y}(\hat{\boldsymbol{z}},\boldsymbol{y}) d\boldsymbol{y} d\hat{\boldsymbol{z}} \\
&= \sup_{f\in\mathcal{F}_B^1} \int\int f(\hat{\boldsymbol{z}},\boldsymbol{y}) p^{(t)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) - f(\hat{\boldsymbol{z}},\boldsymbol{y}) p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) p^{(0)}_{\hat{Z}}(\hat{\boldsymbol{z}}) d\boldsymbol{y} d\hat{\boldsymbol{z}} \\
&= \sup_{f\in\mathcal{F}_B^1} \int\int f(\hat{\boldsymbol{z}},\boldsymbol{y}) \left[p^{(t)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) - p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y})\right] p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) \\
&\quad + f(\hat{\boldsymbol{z}},\boldsymbol{y}) p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) \left[p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) - p^{(0)}_{\hat{Z}}(\hat{\boldsymbol{z}})\right] d\boldsymbol{y} d\hat{\boldsymbol{z}} \\
&\leq \underbrace{\sup_{f\in\mathcal{F}_B^1} \int\int f(\hat{\boldsymbol{z}},\boldsymbol{y}) \left[p^{(t)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) - p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y})\right] p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) d\boldsymbol{y} d\hat{\boldsymbol{z}}}_{:=L_1} \\
&\quad + \underbrace{\sup_{f\in\mathcal{F}_B^1} \int\int f(\hat{\boldsymbol{z}},\boldsymbol{y}) p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) \left[p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) - p^{(0)}_{\hat{Z}}(\hat{\boldsymbol{z}})\right] d\boldsymbol{y} d\hat{\boldsymbol{z}}}_{:=L_2},
\end{aligned}$$

where the inequality is trivial, we have

$$L_1 = \sup_{f\in\mathcal{F}_B^1} \int\int f(\hat{\boldsymbol{z}},\boldsymbol{y}) \left[p^{(t)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) - p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y})\right] p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) d\boldsymbol{y} d\hat{\boldsymbol{z}},$$

and

$$L_2 = \sup_{f\in\mathcal{F}_B^1} \int\int f(\hat{\boldsymbol{z}},\boldsymbol{y}) p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) \left[p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) - p^{(0)}_{\hat{Z}}(\hat{\boldsymbol{z}})\right] d\boldsymbol{y} d\hat{\boldsymbol{z}}.$$

For L_1, when $f(\hat{\boldsymbol{z}},\boldsymbol{y}) \in \mathcal{F}_B^1$, for its component embedding, $\forall \hat{\boldsymbol{z}}_0$, $f(\hat{\boldsymbol{z}},\boldsymbol{y})|_{\hat{\boldsymbol{z}}=\hat{\boldsymbol{z}}_0} \in \mathcal{F}_B^1$. Therefore, we can make the following scaling:

$$\begin{aligned}
L_1 &\leq \int \sup_{f|_{\hat{\boldsymbol{z}}}\in\mathcal{F}_B^1} \left\{\int f(\hat{\boldsymbol{z}},\boldsymbol{y}) \left[p^{(t)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) - p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y})\right] d\boldsymbol{y}\right\} p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) d\hat{\boldsymbol{z}} \\
&= \int d_{\mathcal{F}_B^1}\left(P^{(t)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}, P^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}\right) p^{(t)}_{\hat{Z}}(\hat{\boldsymbol{z}}) d\hat{\boldsymbol{z}},
\end{aligned}$$

where the first inequality is trivial.

For L_2, let $f_2(\hat{\boldsymbol{z}}) := \int f(\hat{\boldsymbol{z}},\boldsymbol{y}) p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}(\boldsymbol{y}) d\boldsymbol{y}$. Next, we will prove that it is a K-Lipschitz continuous function. Consider $\forall \hat{\boldsymbol{z}}_1, \hat{\boldsymbol{z}}_2$, we have

$$\begin{aligned}
|f_2(\hat{\boldsymbol{z}}_1) - f_2(\hat{\boldsymbol{z}}_2)| &= \left|\int f(\hat{\boldsymbol{z}}_1,\boldsymbol{y}) p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_1}(\boldsymbol{y}) - f(\hat{\boldsymbol{z}}_2,\boldsymbol{y}) p^{(0)}_{Y|\hat{Z}=z_2}(\boldsymbol{y}) d\boldsymbol{y}\right| \\
&= \left|\int \left[f(\hat{\boldsymbol{z}}_1,\boldsymbol{y}) - f(\hat{\boldsymbol{z}}_2,\boldsymbol{y})\right] p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_1}(\boldsymbol{y}) + f(\hat{\boldsymbol{z}}_2,\boldsymbol{y}) \left[p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_1}(\boldsymbol{y}) - p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_2}(\boldsymbol{y})\right] d\boldsymbol{y}\right| \\
&\leq \int |f(\hat{\boldsymbol{z}}_1,\boldsymbol{y}) - f(\hat{\boldsymbol{z}}_2,\boldsymbol{y})| p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_1}(\boldsymbol{y}) d\boldsymbol{y} + \int \left|f(\hat{\boldsymbol{z}}_2,\boldsymbol{y}) \left[p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_1}(\boldsymbol{y}) - p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_2}(\boldsymbol{y})\right]\right| d\boldsymbol{y} \\
&\leq \int_y \|\hat{\boldsymbol{z}}_1 - \hat{\boldsymbol{z}}_2\|_1 p^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_1}(\boldsymbol{y}) d\boldsymbol{y} + 2B\mathcal{D}_{TV}\left(P^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_1}, P^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}_2}\right) \\
&\leq K \|\hat{\boldsymbol{z}}_1 - \hat{\boldsymbol{z}}_2\|_1,
\end{aligned}$$

where the first inequality is an absolute value inequality, the second inequality considers $f(\hat{\boldsymbol{z}}, \boldsymbol{y}) \in \mathcal{F}_B^1$and the third inequality is based on Assumption 5. Furthermore, since $|f_2(\hat{\boldsymbol{z}})| \leq \sup_y |f(\hat{\boldsymbol{z}}, \boldsymbol{y})| \leq B$, it follows that $\frac{f_2}{K} \in \mathcal{F}_{\frac{B}{K}}^1$. Therefore, we have

$$L_2 \leq \sup_{f_2 \in \mathcal{F}_B^K} \int_z f_2(\hat{\boldsymbol{z}}) \left(P_{\hat{Z}}^{(t)}(\hat{\boldsymbol{z}}) - P_{\hat{Z}}^{(0)}(\hat{\boldsymbol{z}}) \right) d\hat{\boldsymbol{z}} = K d_{\mathcal{F}_{\frac{B}{K}}^1} \left(P_{\hat{Z}}^{(t)}, P_{\hat{Z}}^{(0)} \right) \leq K d_{\mathcal{F}_B^1} \left(P_{\hat{Z}}^{(t)}, P_{\hat{Z}}^{(0)} \right).$$

where the last inequality is trivial since $K > 1$. □

Lemma 3.3(Lu and Lu (2020) Proposition 3.1). *Assume that probability distribution π on $\mathbb{R}^d$ satisfies that $M_3 = \mathbb{E}_\pi |X|^3 < \infty$, and let $\hat{\pi}_n$ be its empirical distribution. Then*

$$\mathbb{E} d_{W_1} (\hat{\pi}_n, \pi) \precsim \sqrt{d} n^{-\frac{1}{d}},$$

where d_{W_1} is the 1-Wasserstein distance.

Remark. When $\left(\hat{Z}, Y\right)$ satisfies Assumption 2, it also satisfies the condition of Lemma 3.3. Let $V = \left(\hat{Z}, Y\right)$, and by Markov's inequality,

$$\Pr(\|V\| > \log n) \leq \frac{\mathbb{E}\|V\| \mathbb{I}\{\|V\| > \log n\}}{\log n} = O\left(n^{-\frac{(\log n)^\delta}{r+q}} / \log n \right).$$

Thus,

$$\mathbb{E}\|V\|^3 = \int_0^\infty 3t^2 P(\|V\| > t) dt = \int_0^\infty O(1) 3t \exp\left(-\frac{t^{1+\delta}}{r+q} \right) dt < \infty.$$

Lemma 3.4(Shen et al. (2019) Theorem 4.3). *Let f be a Lipschitz continuous function defined on $B_\infty^{r+q}(R)$. For arbitrary $W_D, H_D \in \mathbb{N}_+$, there exists a function D_ϕ implemented by a ReLU feedforward neural network with width no more than W_D and depth no more than H_D such that*

$$\|f - D_{\boldsymbol{\phi}}\|_\infty \precsim R\sqrt{d+q} \left(W_D H_D \right)^{-\frac{2}{r+q}}.$$

Lemma 3.5(Liu et al. (2021)). *Suppose probability measure ν supported on $\mathbb{R}$ is absolutely continuous w.r.t. Lebesgue measure, and probability meansure μ is supported on $\mathbb{R}^q$.η_i and y_i are i.i.d. samples from ν and μ, respectively for $i \in [n]$. Then there exist generator ReLU FNN $G : \mathbb{R} \mapsto \mathbb{R}^q$ maps η_i to y_i for all i. Moreover, such G can be obtained by properly specifying $W_G^2 L_G = cqn$ for some constant $12 \leq c \leq 384$.*

Lemma 3.6(Srebro and Sridharan, 2010). *Assume $\sup_{f \in \mathcal{F}} \|f\|_\infty \leq B$. For any distribution μ and its empirical distribution $\hat{\mu}_n$, the empirical Rademacher complexity $\hat{\mathcal{R}}_n(\mathcal{F})$, we have*

$$\mathbb{E}\left[d_{\mathcal{F}} \left(\hat{\mu}_n, \mu \right) \right] \leq 2\mathbb{E}\hat{\mathcal{R}}_n(\mathcal{F}) \leq \mathbb{E} \inf_{0<\delta<B} \left(4\delta + \frac{12}{\sqrt{n}} \int_\delta^B \sqrt{\log \mathcal{N}\left(\epsilon, \mathcal{F}, L_\infty\left(P_n \right) \right)} d\epsilon \right).$$

Lemma 3.7(Wellner et al. (2013), Theorem 2.7.1). *Let $\mathcal{X}$ be a bounded, convex subset of $\mathbb{R}^d$ with nonempty interior. There exists a constant c_d depending only on d such that*

$$\log \mathcal{N}\left(\epsilon, \mathcal{F}^1(\mathcal{X}), \|\cdot\|_\infty \right) \leq c_d \lambda\left(\mathcal{X}^1 \right) \left(\frac{1}{\epsilon} \right)^d,$$

for every $\epsilon > 0$, where $\mathcal{F}^1(\mathcal{X})$ is the 1-Lipschitz function class defined on $\mathcal{X}$, and $\lambda\left(\mathcal{X}^1\right)$ is the Lebesgue measure of the set $\{x : \|x - \mathcal{X}\| < 1\}$.

Lemma 3.8 Suppose Assumption 6 holds. We have,

$$\forall t \in [T], \mathbb{E}_{P_{\tilde{Z}}^{(t)}} d_{\mathcal{F}_B^1}(P_{Y|\tilde{Z}}^{(t)}, P_{Y|\tilde{Z}}^{(0)}) \leq W_1(P_{\tilde{Z},Y}^{(t)}, P_{\tilde{Z},Y}^{(0)}) + K W_1(P_{\tilde{Z}}^{(t)}, P_{\tilde{Z}}^{(0)}),$$

where $W_1(\cdot, \cdot)$ denotes the 1-Wasserstein distance

Proof. Firstly, we consider a trivial result, $d_{\mathcal{F}_B^1}(P^{(t)}_{Y|\tilde{Z}}, P^{(0)}_{Y|\tilde{Z}}) \leq W_1(P^{(t)}_{Y|\tilde{Z}}, P^{(0)}_{Y|\tilde{Z}})$. By the definition of 1-Wasserstein distance, we have $W_1\left(P^{(t)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}, P^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}\right) = \inf_{\gamma_{\tilde{\boldsymbol{z}}}} \int \|Y^{(t)} - Y^{(0)}\| d\gamma_{\tilde{\boldsymbol{z}}}$, where the $\inf_{\gamma_{\tilde{\boldsymbol{z}}}}$ is taken over the set of all the couplings of $P^{(t)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}$ and $P^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}$. Adding a coordinate while preserving the norm, we have

$$W_1\left(P^{(t)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}, P^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}\right) = \inf_{\gamma_{\tilde{\boldsymbol{z}}}} \int \|(\tilde{\boldsymbol{z}}, Y^{(t)}) - (\tilde{\boldsymbol{z}}, Y^{(0)})\| d\gamma_{\tilde{\boldsymbol{z}}}.$$

Therefore, if we denote $p^{(t)}_{\tilde{Z}}(\tilde{\boldsymbol{z}}) \times p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}(\boldsymbol{y})$ as the new joint density $Q^{(t)}_{\tilde{Z},Y}$, then we have

$$\begin{aligned}
\mathbb{E}_{P^{(t)}_{\tilde{Z}}} d_{\mathcal{F}_B^1}(P^{(t)}_{Y|\tilde{Z}}, P^{(0)}_{Y|\tilde{Z}}) &\leq \mathbb{E}_{P^{(t)}_{\tilde{Z}}} W_1(P^{(t)}_{Y|\tilde{Z}}, P^{(0)}_{Y|\tilde{Z}}) \\
&= \int \inf_{\gamma_{\tilde{\boldsymbol{z}}}} \int \|(\tilde{\boldsymbol{z}}, Y^{(t)}) - (\tilde{\boldsymbol{z}}, Y^{(0)})\| d\gamma_{\tilde{\boldsymbol{z}}} dP^{(t)}_{\tilde{Z}} \\
&\leq \inf_{\pi} \int \|(\tilde{Z}, Y)^{P^{(t)}} - (\tilde{Z}, Y)^{Q^{(t)}}\| d\pi \\
&= W_1(P^{(t)}_{\tilde{Z},Y}, Q^{(t)}_{\tilde{Z},Y}),
\end{aligned} \tag{12}$$

where $(\tilde{Z}, Y)^{P^{(t)}}$ and $(\tilde{Z}, Y)^{Q^{(t)}}$ denote random variables following the distributions $P^{(t)}_{\tilde{Z},Y}$ and $Q^{(t)}_{\tilde{Z},Y}$, respectively. The $\inf_{\pi}$ is taken over the set of all couplings of $P^{(t)}_{\tilde{Z},Y}$ and $Q^{(t)}_{\tilde{Z},Y}$. The $\inf_{\gamma_{\tilde{\boldsymbol{z}}}}$ is taken over the set of all couplings of $P^{(t)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}$ and $P^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}$. Then considering the triangle inequality of distance, we have,

$$W_1(P^{(t)}_{\tilde{Z},Y}, Q^{(t)}_{\tilde{Z},Y}) \leq W_1(P^{(t)}_{\tilde{Z},Y}, P^{(0)}_{\tilde{Z},Y}) + W_1(P^{0)}_{\tilde{Z},Y}, Q^{(t)}_{\tilde{Z},Y}). \tag{13}$$

We can expand the second term $W_1(P^{0)}_{\tilde{Z},Y}, Q^{(t)}_{\tilde{Z},Y})$ in dual form,

$$W_1(P^{(0)}_{\tilde{Z},Y}, Q^{(t)}_{\tilde{Z},Y}) = \sup_{f \in \mathcal{F}^1} \int \int f(\tilde{\boldsymbol{z}}, \boldsymbol{y}) p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}(\boldsymbol{y}) \left[p^{(t)}_{\tilde{Z}}(\tilde{\boldsymbol{z}}) - p^{(0)}_{\tilde{Z}}(\tilde{\boldsymbol{z}})\right] d\boldsymbol{y} d\tilde{\boldsymbol{z}}.$$

Since Assumption 6 holds, similar to the proof of Lemma 3.2, we also define $f_2(\tilde{\boldsymbol{z}}) := \int f(\tilde{\boldsymbol{z}}, \boldsymbol{y}) p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}}(\boldsymbol{y}) d\boldsymbol{y}$. Next, we will prove that it is a K-Lipschitz continuous function. Consider $\forall \tilde{\boldsymbol{z}}_1, \tilde{\boldsymbol{z}}_2$, we have

$$\begin{aligned}
|f_2(\tilde{\boldsymbol{z}}_1) - f_2(\tilde{\boldsymbol{z}}_2)| &= \left| \int f(\tilde{\boldsymbol{z}}_1, \boldsymbol{y}) p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_1}(\boldsymbol{y}) - f(\tilde{\boldsymbol{z}}_2, \boldsymbol{y}) p^{(0)}_{Y|\tilde{Z}=z_2}(\boldsymbol{y}) d\boldsymbol{y} \right| \\
&= \left| \int \left[f(\tilde{\boldsymbol{z}}_1, \boldsymbol{y}) - f(\tilde{\boldsymbol{z}}_2, \boldsymbol{y})\right] p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_1}(\boldsymbol{y}) + f(\tilde{\boldsymbol{z}}_2, \boldsymbol{y}) \left[p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_1}(\boldsymbol{y}) - p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_2}(\boldsymbol{y})\right] d\boldsymbol{y} \right| \\
&\leq \int |f(\tilde{\boldsymbol{z}}_1, \boldsymbol{y}) - f(\tilde{\boldsymbol{z}}_2, \boldsymbol{y})| p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_1}(\boldsymbol{y}) d\boldsymbol{y} + W_1\left(P^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_1}, P^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_2}\right) \\
&\leq \int_y \|\tilde{\boldsymbol{z}}_1 - \tilde{\boldsymbol{z}}_2\|_1 p^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_1}(\boldsymbol{y}) d\boldsymbol{y} + W_1\left(P^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_1}, P^{(0)}_{Y|\tilde{Z}=\tilde{\boldsymbol{z}}_2}\right) \\
&\leq K \|\tilde{\boldsymbol{z}}_1 - \tilde{\boldsymbol{z}}_2\|_1,
\end{aligned}$$

where the first inequality is an absolute value inequality, the second inequality considers $f(\tilde{\boldsymbol{z}}, \boldsymbol{y}) \in \mathcal{F}^1$ and the third inequality is based on Assumption 6. Then, we have $\frac{f_2}{K} \in \mathcal{F}^1$, we finally get,

$$W_1(P^{(0)}_{\tilde{Z},Y}, Q^{(t)}_{\tilde{Z},Y}) \leq K W_1(P^{(t)}_{\tilde{Z}}, P^{(0)}_{\tilde{Z}}), \tag{14}$$

where we can combine formula (12)-(14) to complete the proof. □

3.3 An equivalent statement

We hope that functions in the evaluation class $\mathcal{F}^1$ are defined on a bounded domain so we can apply existing neural nets approximation theorems to bound the approximation error $\mathcal{E}_1$. It motivates us to first show that

proving the desired convergence rate is equivalent to establishing the same convergence rate but with the domain restricted function class $\mathcal{F}_n^1 := \left\{ f|_{B_\infty(2\log n)} : f \in \mathcal{F}^1 \right\}$ as the evaluation class under Assumption 2. Suppose Assumption 2 holds. By the Markov inequality we have

$$P(\|(\hat{Z}, Y)\| > \log n) \leq \frac{\mathbb{E}\|(\hat{Z}, Y)\| \mathbb{I}_{\{\|(\hat{Z},Y)\| > \log n\}}}{\log n} = O\left(n^{-\frac{(\log n)^\delta}{r+q}} / \log n \right), \tag{15}$$

The bounded Lipschitz distance is defined as

$$d_{\mathcal{F}^1}\left(P_{\hat{Z},Y}, P_{\hat{Z},\hat{G}}\right) = \sup_{f \in \mathcal{F}^1} \mathbb{E} f(\hat{Z}, Y) - \mathbb{E} f(\hat{Z}, \hat{G}).$$

The first term above can be decomposed as

$$\mathbb{E} f(\hat{Z}, Y) = \mathbb{E} f(\hat{Z}, Y) \mathbb{I}_{\left\{\|(\hat{Z},Y)\| \leq \log n\right\}} + \mathbb{E} f(\hat{Z}, Y) \mathbb{I}_{\left\{\|(\hat{Z},Y)\| > \log n\right\}}. \tag{16}$$

For any $f \in \mathcal{F}^1$ and a fixed point $\|(\hat{\boldsymbol{z}}_0, \boldsymbol{y}_0)\| < \log n$, due to the Lipschitzness of f, the second term above satisfies

$$\begin{aligned}
&\left| \mathbb{E} f(\hat{Z}, Y) \mathbb{I}_{\left\{\|(\hat{Z},Y)\| > \log n\right\}} \right| \\
\leq &\left| \mathbb{E} f(\hat{Z}, Y) \mathbb{I}_{\left\{\|(\hat{Z},Y)\| > \log n\right\}} - \mathbb{E} f\left(\hat{\boldsymbol{z}}_0, \boldsymbol{y}_0\right) \mathbb{I}_{\left\{\|(\hat{Z},Y)\| > \log n\right\}} \right| + \left| \mathbb{E} f\left(\hat{\boldsymbol{z}}_0, \boldsymbol{y}_0\right) \mathbb{I}_{\left\{\|(\hat{Z},Y)\| > \log n\right\}} \right| \\
\leq &\mathbb{E} \left\| (\hat{Z}, Y) - (\hat{\boldsymbol{z}}_0, \boldsymbol{y}_0) \right\| \mathbb{I}_{\left\{\|(\hat{Z},Y)\| > \log n\right\}} + BP(\|(\hat{Z}, Y)\| > \log n) \\
= &O\left(n^{-\frac{(\log n)^\delta}{r+q}} \right),
\end{aligned}$$

where the second inequality is due to lipschitzness and boundedness of f, and the last inequality is due to Assumption 2 and formula (15). The second term in formula (16) can be dealt similarly due to Condition (11) for the network G_θ. Hence, restricting the evaluation class to $\mathcal{F}_n^1$ will not affect the convergence rate in the main results, i.e. $O\left(n^{-\frac{1}{r+q}}\right)$. Due to this fact, to keep notation simple, we denote $\mathcal{F}_n^1$ as $\mathcal{F}^1$ in the following sections.

3.4 Bounding the error terms

Bounding $\mathcal{E}_1$. The discriminator approximation error (4) describes how well the discriminator neural network class is in the task of approximating functions from the Lipschitz class $\mathcal{F}^1$. There has been much recent work on the approximation power of deep neural networks. The lemma 3.4 is a quantitative and non-asymptotic result from Shen et al. (2019). When balancing the errors, we can let the discriminator structure be $W_D H_D \geq \sqrt{n}$ and $R = 2\log n$ so that $\mathcal{E}_1$ is of the order $n^{1/(r+q)} \log n$, which is the same order of the statistical errors.

Bound $\mathcal{E}_2$. The generator and representation approximation error (8) describes how powerful the generator class and representation class are in realizing the empirical version of the noise outsourcing lemma and reducing distributional differences. If we can find ReLU FNNs $G_{\boldsymbol{\theta}_0}, R_{\boldsymbol{\omega}_0}$ such that $G_{\boldsymbol{\theta}_0}\left(\eta_i^{(t)\prime}, R_{\boldsymbol{\omega}_0}\left(\boldsymbol{x}_i^{(t)\prime}\right)\right) = \boldsymbol{y}_i^{(t)\prime}$ for all $t \in S \cup \{0\}, i \in [n_t]$, and $R_{\boldsymbol{\omega}_0}\left(\boldsymbol{x}_j^{(t)\prime}\right) = R_{\boldsymbol{\omega}_0}\left(\boldsymbol{x}_i^{(0)\prime}\right)$ for all $t \in S, i \in [n_0], j \in S_i^{(t)}$ where $S_i^{(t)}$ is a subset of $[n_t]$ and $|S_i^{(t)}|$ are equal to i, then $\mathcal{E}_2 = 0$. The existence of such neural networks are guaranteed by the Lemma 3.5, where the structure of the generator network is to be set as $W_G^2 H_G = c_1 q n, W_R^2 H_R = c_2 r n$ for some constant $12 < c_1, c_2 < 384$. Note that Lemma 3.5 holds under the condition that the range of $G_{\boldsymbol{\theta}}$ covers the support of P_Y. Since we imposed Condition (11), this is not always satisfied. However, Assumption 2 controls the probability of the bad set where $\mathcal{E}_2 \neq 0$ and we can show that the desired convergence rate is not affected by the bad set.

Bound $\mathcal{E}_3$. The statistical error (9) quantifies how close the empirical distribution and the true target are under bounded Lipschitz distance. The lemma 3.3 is a quantitative and non-asymptotic result form Lu and Lu, (2020). The finite moment condition is satisfied due to Assumption 1 and $E|X|^3 = \int_0^\infty 3t^2 P(|X| > t) dt$. Recall that $d_{\mathcal{F}_B^1}(\hat{\pi}_n, \pi) \leq d_{W_1}(\hat{\pi}_n, \pi)$, hence we have

$$\mathbb{E}\mathcal{E}_3 \precsim n^{-1/(r+q)} + \sum_{t \in S} (n_t/n) n_t^{-1/r} + n_0^{-1/r}.$$

Bound $\mathcal{E}_4$. Similar to $\mathcal{E}_3$, the statistical error (2) describes the distance between the mixture distribution of $(\hat{R}(X), \hat{G})$ and its empirical distribution. We need to introduce the empirical Rademacher complexity $\hat{\mathcal{R}}_n(\mathcal{F})$ to quantify it. Define the empirical Rademacher complexity of function class $\mathcal{F}$ as

$$\hat{\mathcal{R}}_n(\mathcal{F}) := \mathbb{E}_\epsilon \sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \epsilon_i f\left(X_i, \hat{G}\right),$$

where $\epsilon = (\epsilon_1, \ldots, \epsilon_n)$ are i.i.d. Rademacher variables, i.e. uniform $\{-1, 1\}$.

In $\mathcal{E}_4 = \sup_{f\in\mathcal{F}_B^1}\left\{\mathbb{E}f(\hat{Z}, \hat{G}) - \frac{1}{n}\sum_{t\in S\cup\{0\}}\sum_{i=1}^{n_t} f\left(\hat{\boldsymbol{z}}_i^{(t)\prime}, \hat{G}'\right)\right\}$, we used the discriminator network $\hat{G}', \hat{R}'$ obtained from the ghost samples for the empirical distribution. The reason is that symmetrization requires two distributions being the same. In our settings, $\left(\hat{R}(X_i), \hat{G}\left(\hat{R}(X_i), \eta_i\right)\right)$ and $\left(\hat{R}(X_i'), \hat{G}\left(\hat{R}(X_i'), \eta_i'\right)\right)$ do not have the same distribution, but $\left(\hat{R}(X_i), \hat{G}\left(\hat{R}(X_i), \eta_i\right)\right)$ and $\left(\hat{R}'(X_i'), \hat{G}'\left(\hat{R}'(X_i'), \eta_i'\right)\right)$ do. Recall that we have restricted $\mathcal{F}_B^1$ to $B_\infty(2\log n)$. Since $\mathcal{N}\left(\epsilon, \mathcal{F}, L_\infty\left(P_n\right)\right) \leq \mathcal{N}\left(\epsilon, \mathcal{F}, \|\cdot\|_\infty\right)$, now it suffices to bound the covering number $\mathcal{N}\left(\epsilon, \mathcal{F}^1\big|_{B_\infty(2\log n)}, \|\cdot\|_\infty\right)$. Applying lemmas 3.6, 3.7 and taking $\delta = C\sqrt{r+q}n^{-1/(r+q)}\log n$ for some constant $C > 0$, we have

$$\mathbb{E}\mathcal{E}_4 = O\left((r+q)^{1/2} n^{-1/(r+q)} \log n\right).$$

3.5 Proofs of the theorems

In this section, unless specified otherwise, we denote $n = \sum_{t\in S\cup\{0\}} n_t$. We numbered the theorems according to the original section sequence.

Theorem 4.1 *Suppose Assumptions 2-5 hold. Let (L_D, H_D) of $D_{\boldsymbol{\phi}}$, (L_G, H_G) of $G_{\boldsymbol{\theta}}$ and (L_R, H_R) of $R_{\boldsymbol{\omega}}$ be specified such that $W_D H_D = \lceil\sqrt{n}\rceil, W_G^2 H_G = c_1 qn$ and $W_R^2 H_R = c_2 rn$ for some constants $12 \leq c_1, c_2 \leq 384$. We have:*

$$\mathbb{E}_{\hat{G}} d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P_{\hat{Z},Y}^{(0)}\right) \precsim n^{-1/(r+q)}\log n + n_0^{-1/r}.$$

Proof. By taking $W_D H_D = \lceil\sqrt{n}\rceil$ and $R = 2\log n$ in Lemma 3.4, we get $\mathcal{E}_1 \precsim n^{-1/(r+q)}\log n$. Lemma 3.5 states that $\mathcal{E}_2 = 0$ as long as the range of G_θ covers all the Y_i', i.e. $\max_{t\in S\cup\{0\}, 1\leq i\leq n_t}\left\|\boldsymbol{y}_i^{(t)\prime}\right\|_\infty \leq \log n$. Hence the nice set $H := \left\{\max_{t\in S\cup\{0\}, 1\leq i\leq n_t}\left\|(\hat{\boldsymbol{z}}_i^{(t)\prime}, \boldsymbol{y}_i^{(t)\prime})\right\| \leq \log n\right\}$ is where $\mathcal{E}_2 = 0$, and $P\left(H^c\right) = 1 - P(H) \leq 1 - \left(1 - C\frac{n^{-\frac{(\log n)^\delta}{r+q}}}{\log n}\right)^n \leq Cn^{-\frac{(\log n)^\delta}{r+q}}/\log n$. Also, we have $\mathbb{E}\mathcal{E}_3 \precsim n^{-1/(r+q)} + \sum_{t\in S}(n_t/n)n_t^{-1/r} + n_0^{-1/r}$ and $\mathbb{E}\mathcal{E}_4 \precsim n^{-\frac{1}{r+q}}\log n$ by Lemma 3.3,3.6 and 3.7, respectively. Therefore, by Lemma 2.1, we have

$$\begin{aligned}
\mathbb{E}d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P_{\hat{Z},Y}^{(0)}\right) &\leq \mathbb{E}d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P_{\hat{Z},Y}^{(0)}\right)\mathbb{I}_H + \mathbb{E}d_{\mathcal{F}_B^1}\left(P_{X,\hat{G}}, P_{X,Y}^{(0)}\right)\mathbb{I}_{H^c} \\
&\leq \left(2\mathcal{E}_1 + \mathcal{E}_2 + \mathbb{E}\mathcal{E}_3 + \mathbb{E}\mathcal{E}_4 + h\right)\mathbb{I}_H + 2BP\left(H^c\right) \\
&\precsim n^{-1/(r+q)}\log n + 0 + \sum_{t\in S}(n_t/n)n_t^{-1/r} + n_0^{-1/r} + n^{-1/(r+q)}\log n + h + n^{-\frac{(\log n)^\delta}{r+q}}/\log n \\
&\precsim n^{-1/(r+q)}\log n + n_0^{-1/r}.
\end{aligned}$$

This completes the proof of Theorem 4.1. □

Theorem 4.2 *Suppose that $P_{\hat{Z},Y}$ is supported on $[-U, U]^{r+q}$ for some $U > 0$ and Assumptions 3-5 hold. Let (L_D, H_D) of $D_{\boldsymbol{\phi}}$, (L_G, H_G) of $G_{\boldsymbol{\theta}}$ and (L_R, H_R) of $R_{\boldsymbol{\omega}}$ be specified such that $W_D H_D = \lceil\sqrt{n}\rceil, W_G^2 H_G = c_1 qn$ and $W_R^2 H_R = c_2 rn$ for some constants $12 \leq c_1, c_2 \leq 384$. Let the output of $G_{\boldsymbol{\theta}}$ be on $[-U, U]^q$. We have:*

$$\mathbb{E}_{\hat{G}} d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P_{\hat{Z},Y}^{(0)}\right) \precsim n^{-1/(r+q)} + n_0^{-1/r}.$$

Proof. By taking $W_D L_D = \lceil\sqrt{n}\rceil$ and $R = M$ in Lemma 3.4, we get $\mathcal{E}_1 \precsim n^{-1/(r+q)}$. Since the range of $G_{\boldsymbol{\theta}}$ covers all the $\boldsymbol{y}_i^{(t)\prime}$, we have $\mathcal{E}_2 = 0$. Also, we have $\mathbb{E}\mathcal{E}_3 \precsim \sum_{t\in S}(n_t/n)n_t^{-1/r} + n_0^{-1/r}$ by previous results. Similar

to the procedure for obtaining the convergence rate of $\mathbb{E}\mathcal{E}_4$, we get $\mathbb{E}\mathcal{E}_4 \precsim n^{-1/(r+q)}$. In all by Lemma 2.1, we have

$$\begin{aligned}\mathbb{E}d_{\mathcal{F}_B^1}\left(P_{\hat{Z},\hat{G}}, P^{(0)}_{\hat{Z},Y}\right) &\leq 2\mathcal{E}_1 + \mathcal{E}_2 + \mathbb{E}\mathcal{E}_3 + \mathbb{E}\mathcal{E}_4 + h \\ &\precsim n^{-1/(r+q)} + 0 + \sum_{t\in S}(n_t/n)n_t^{-1/r} + n_0^{-1/r} + n^{-1/(r+q)} + h \\ &\precsim n^{-1/(r+q)} + n_0^{-1/r}.\end{aligned}$$

This completes the proof of Theorem 4.2. □

Theorem 4.3 *Under the same conditions of Theorem 4.2, we have*

$$\mathbb{E}_{\hat{G}}\mathbb{E}_{P^{(0)}_{\hat{Z}}} d_{\mathcal{F}_B^1}\left(P_{\hat{G}|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) \precsim n^{-1/(r+q)} + n_0^{-1/r}.$$

Proof. By choosing a suitably large B, we can demonstrate that, on the domain $[-U,U]^{r+q}$, the distance $d_{\mathcal{F}_B^1}(\cdot,\cdot)$ is equivalent to the 1 -Wasserstein distance $W_1(\cdot,\cdot)$. By the similar process in Lemma 3.8, we have

$$\begin{aligned}\mathbb{E}_{P^{(0)}_{\hat{Z}}} d_{\mathcal{F}_B^1}\left(P_{\hat{G}|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) &= \mathbb{E}_{P^{(0)}_{\hat{Z}}} W_1\left(P_{\hat{G}|\hat{Z}}, P^{(0)}_{Y|\hat{Z}}\right) \\ &= \int \inf_{\gamma_{\hat{z}}} \int \left\| \left(\hat{\boldsymbol{z}}, Y^{(t)}\right) - \left(\hat{\boldsymbol{z}}, Y^{(0)}\right) \right\| d\gamma_{\hat{z}} dP^{(0)}_{\hat{Z}} \\ &\leq \inf_{\pi} \int \left\| (\hat{Z}, Y)^{P^{(0)}_{\hat{Z}} P_{\hat{G}|\hat{Z}}} - (\hat{Z}, Y)^{P^{(0)}_{\hat{Z}} P^{(0)}_{Y|\hat{Z}}} \right\| d\pi \\ &= W_1\left(P^{(0)}_{\hat{Z}} P_{\hat{G}|\hat{Z}}, P^{(0)}_{\hat{Z}} P^{(0)}_{Y|\hat{Z}}\right) \\ &\leq W_1\left(P^{(0)}_{\hat{Z}} P_{\hat{G}|\hat{Z}}, P_{\hat{Z}} P_{\hat{G}|\hat{Z}}\right) + W_1(P_{\hat{Z}} P_{\hat{G}|\hat{Z}}, P^{(0)}_{\hat{Z}} P^{(0)}_{Y|\hat{Z}}) \\ &= d_{\mathcal{F}_B^1}\left(P^{(0)}_{\hat{Z}} P_{\hat{G}|\hat{Z}}, P_{\hat{Z}} P_{\hat{G}|\hat{Z}}\right) + d_{\mathcal{F}_B^1}(P_{\hat{Z}} P_{\hat{G}|\hat{Z}}, P^{(0)}_{\hat{Z}} P^{(0)}_{Y|\hat{Z}}),\end{aligned}$$

where the $\inf_\pi$ is taken over the set of all couplings of $P^{(0)}_{\hat{Z}} P_{\hat{G}|\hat{Z}}$ and $P^{(0)}_{\hat{Z}} P^{(0)}_{Y|\hat{Z}}$. The $\inf_{\gamma_{\hat{\boldsymbol{z}}}}$ is taken over the set of all couplings of $P_{\hat{G}|\hat{Z}=\hat{\boldsymbol{z}}}$ and $P^{(0)}_{Y|\hat{Z}=\hat{\boldsymbol{z}}}$. Recall that we denote the mixture distribution by $P_{\hat{Z}}$, where the domain index (t) is removed.

Let's now address $d_{\mathcal{F}_B^1}\left(P^{(0)}_{\hat{Z}} P_{\hat{G}|\hat{Z}}, P_{\hat{Z}} P_{\hat{G}|\hat{Z}}\right)$. Expanding this expression, we arrive at

$$\begin{aligned}d_{\mathcal{F}_B^1}\left(P^{(0)}_{\hat{Z}} P_{\hat{G}|\hat{Z}}, P_{\hat{Z}} P_{\hat{G}|\hat{Z}}\right) &= \sup_{f\in\mathcal{F}_B^1} \int\int f(\hat{\boldsymbol{z}}, \hat{G}) p_{\hat{G}|\hat{Z}=\hat{\boldsymbol{z}}}(\hat{G}) \left(p^{(0)}_{\hat{Z}}(\hat{\boldsymbol{z}}) - p_{\hat{Z}}(\hat{\boldsymbol{z}})\right) d\hat{\boldsymbol{z}} d\hat{G} \\ &\leq \int d_{\mathcal{F}_B^1}\left(P^{(0)}_{\hat{Z}}, P_{\hat{Z}}\right) p_{\hat{G}|\hat{Z}=\hat{\boldsymbol{z}}}(\hat{G}) d\hat{G} \\ &\leq d_{\mathcal{F}_B^1}\left(P^{(0)}_{\hat{Z}}, P_{\hat{Z}}\right).\end{aligned}$$

By following the proof of Theorem 4.2, we eventually obtain the inequality $d_{\mathcal{F}_B^1}\left(P^{(0)}_{\hat{Z}}, P_{\hat{Z}}\right) \precsim \sum_{t\in S}(n_t/n)n_t^{-1/r} + n_0^{-1/r}$, which is of the same order as certain terms in $\mathbb{E}_{\hat{G}} d_{\mathcal{F}_B^1}(P_{\hat{Z}} P_{\hat{G}|\hat{Z}}, P^{(0)}_{\hat{Z}} P^{(0)}_{Y|\hat{Z}})$. The latter corresponds to the upper bound proposed in Theorem 4.2.

This completes the proof of Theorem 4.3. □

Theorem 5.1 *Suppose that $P_{\tilde{Z},Y}$ is supported on $[-U,U]^{r+q}$ for some $U>0$ and Assumptions 1, 5-6 hold for $\tilde{Z}$. Let $(W_{\tilde{D}}, H_{\tilde{D}})$ of $D_{\tilde{\boldsymbol{\phi}}}$, $(W_{\tilde{G}}, H_{\tilde{G}})$ of $G_{\tilde{\boldsymbol{\theta}}}$ and $(W_{\tilde{R}}, H_{\tilde{R}})$ of $R_{\tilde{\boldsymbol{\omega}}}$ be specified such that $W_{\tilde{D}} H_{\tilde{D}} = \lceil \sqrt{n} \rceil$, $W_{\tilde{G}}^2 H_{\tilde{G}} = c_1 qn$ and $W_{\tilde{R}}^2 H_{\tilde{R}} = c_2 rn$ for some constants $12 \leq c_1, c_2 \leq 384$, where $n = \sum_{t\in[T]\cup\{0\}} n_t$. Let the output of $G_{\tilde{\boldsymbol{\theta}}}$ be on $[-U,U]^q$ and selection threshold $C\left(\max\left\{n^{-1/(r+q)}, n_0^{-1/r}\right\}\right) = h$, we have:*

$$P(\hat{S} = S) \to 1.$$

where it equcals to $P(\forall t \in \hat{S} \longleftrightarrow$ *the t-th source domain hold Assumption 4)* $\to 1$.

Proof. By Lemma 3.3, 3.5 and 3.8, $\forall t \in \hat{S}$, we have,

$$\begin{aligned}\mathbb{E}_{P_{\tilde{Z}}^{(t)}} d_{\mathcal{F}_B^1}\left(P_{Y|\tilde{Z}}^{(t)}, P_{Y|\tilde{Z}}^{(0)}\right) \leq & W_1\left(P_{\tilde{Z},Y}^{(t)}, P_{\tilde{Z},Y}^{(0)}\right) + KW_1\left(P_{\tilde{Z}}^{(t)}, P_{\tilde{Z}}^{(0)}\right) \\ \leq & W_1\left(P_{\tilde{Z},Y}^{n_t}, P_{\tilde{Z},Y}^{n_0}\right) + W_1\left(P_{\tilde{Z},Y}^{(t)}, P_{\tilde{Z},Y}^{n_t}\right) + W_1\left(P_{\tilde{Z},Y}^{(0)}, P_{\tilde{Z},Y}^{n_0}\right) \\ & + KW_1\left(P_{\tilde{Z}}^{n_t}, P_{\tilde{Z}}^{n_0}\right) + KW_1\left(P_{\tilde{Z}}^{(t)}, P_{\tilde{Z}}^{n_t}\right) + KW_1\left(P_{\tilde{Z}}^{(0)}, P_{\tilde{Z}}^{n_0}\right) \\ \precsim & h + n_t^{-1/(r+q)} + n_0^{-1/(r+q)} + 0 + n_t^{-1/r} + n_0^{-1/r} \to h,\end{aligned} \tag{17}$$

where the first inequality follows from Lemma 3.8, and the second inequality is straightforward. Similarly, by the process used to bound $\mathcal{E}_2$, we also have $W_1(P_{\tilde{Z}}^{n_t}, P_{\tilde{Z}}^{n_0}) = 0$. Thus, the remainder of the third inequality follows from Lemma 3.3. Hence, we have $P(\hat{S} \subset S) \to 1$.

Moreover, since $P_{\tilde{Z},Y}$ is supported on $[-U,U]^{r+q}$ for some $U > 0$, when B is sufficiently large, we have $d_{\mathcal{F}_B^1}(\cdot,\cdot) = d_{\mathcal{F}^1}(\cdot,\cdot) = W_1(\cdot,\cdot)$. Therefore, by Lemma 3.2, 3.3 and 3.5, for all $t \in S$, we have,

$$\begin{aligned}W_1\left(P_{\tilde{Z},Y}^{n_t}, P_{\tilde{Z},Y}^{n_0}\right) &\leq W_1\left(P_{\tilde{Z},Y}^{(t)}, P_{\tilde{Z},Y}^{(0)}\right) + W_1\left(P_{\tilde{Z},Y}^{(t)}, P_{\tilde{Z},Y}^{n_t}\right) + W_1\left(P_{\tilde{Z},Y}^{(0)}, P_{\tilde{Z},Y}^{n_0}\right) \\ &= d_{\mathcal{F}_B^1}\left(P_{\tilde{Z},Y}^{(t)}, P_{\tilde{Z},Y}^{(0)}\right) + W_1\left(P_{\tilde{Z},Y}^{(t)}, P_{\tilde{Z},Y}^{n_t}\right) + W_1\left(P_{\tilde{Z},Y}^{(0)}, P_{\tilde{Z},Y}^{n_0}\right) \\ &\leq \mathbb{E}_{P_{\tilde{Z}}^{(t)}} d_{\mathcal{F}_B^1}\left(P_{Y|\tilde{Z}}^{(t)}, P_{Y|\tilde{Z}}^{(0)}\right) + Kd_{\mathcal{F}_B^1}\left(P_{\tilde{Z}}^{(t)}, P_{\tilde{Z}}^{(0)}\right) + W_1\left(P_{\tilde{Z},Y}^{(t)}, P_{\tilde{Z},Y}^{n_t}\right) + W_1\left(P_{\tilde{Z},Y}^{(0)}, P_{\tilde{Z},Y}^{n_0}\right) \\ &\precsim h + 0 + n_t^{-1/(r+q)} + n_0^{-1/(r+q)} \to h,\end{aligned}$$

where the second inequality follows from Lemma 3.2, and the remainder is similar to formula (17). If Assumption 4 holds for the representation $\tilde{Z}$, then we also consider Assumption 4 to hold for the re-trained $\hat{Z}$ after selection, since the training of $\hat{Z}$ excludes outlier source domains. Thus, we have $P(S \subset \hat{S}) \to 1$.

This completes the proof of Theorem 5.1. □

Theorem 5.2 *Suppose that* $P_{\tilde{Z},Y}, P_{\hat{Z},Y}$ *is supported on* $[-U,U]^{r+q}$ *for some* $U > 0$ *and Assumptions 1, 3-6 hold. Let* $(W_{\tilde{D}}, H_{\tilde{D}})$ *of* $D_{\tilde{\boldsymbol{\phi}}}$, $(W_{\tilde{G}}, H_{\tilde{G}})$ *of* $G_{\tilde{\boldsymbol{\theta}}}$ *and* $(W_{\tilde{R}}, H_{\tilde{R}})$ *of* $R_{\tilde{\boldsymbol{\omega}}}$ *be specified such that* $W_{\tilde{D}} H_{\tilde{D}} = \lceil \sqrt{n} \rceil$, $W_{\tilde{G}}^2 H_{\tilde{G}} = c_1 qn$ *and* $W_{\tilde{R}}^2 H_{\tilde{R}} = c_2 rn$ *for some constants* $12 \leq c_1, c_2 \leq 384$, *where* $n = \sum_{t\in[T]\cup\{0\}} n_t$. *Let the output of* $G_{\tilde{\boldsymbol{\theta}}}$ *be on* $[-U,U]^q$ *and selection threshold* $C\left(\max\left\{n^{-1/(r+q)}, n_0^{-1/r}\right\}\right) = h$, *we have:*

$$\mathbb{E}_{\hat{G}} \mathbb{E}_{P_{\hat{Z}}^{(0)}} d_{\mathcal{F}_B^1}\left(P_{\hat{G}|\hat{Z}}, P_{Y|\hat{Z}}^{(0)}\right) \precsim n^{-1/(r+q)} + n_0^{-1/(r+q)}.$$

Proof. Combining Theorem 4.3 and the formula (17) in Theorem 5.1, the proof of this theorem is straightforward. □

References

Chung, H., Kim, J., Mccann, M. T., Klasky, M. L., and Ye, J. C. (2022). Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*.

Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.

Dua, D., Graff, C., et al. (2017). Uci machine learning repository.

Fernando, B., Habrard, A., Sebban, M., and Tuytelaars, T. (2013). Unsupervised visual domain adaptation using subspace alignment. In *Proceedings of the IEEE international conference on computer vision*, pages 2960–2967.

Gibbs, A. L. and Su, F. E. (2002). On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435.

Gong, M., Zhang, K., Liu, T., Tao, D., Glymour, C., and Schölkopf, B. (2016). Domain adaptation with conditional transferable components. In *International conference on machine learning*, pages 2839–2848. PMLR.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.

Lai, W.-C., Kirchler, M., Yassin, H., Fehr, J., Rakowski, A., Olsson, H., Starke, L., Millward, J. M., Waiczies, S., and Lippert, C. (2024). Heterogeneous medical data integration with multi-source stylegan. In *Medical Imaging with Deep Learning*.

Liu, S., Zhou, X., Jiao, Y., and Huang, J. (2021). Wasserstein generative learning of conditional distribution. *arXiv preprint arXiv:2112.10039*.

Long, M., Wang, J., Ding, G., Sun, J., and Yu, P. S. (2014). Transfer joint matching for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1410–1417.

Lu, Y. and Lu, J. (2020). A universal approximation theorem of deep neural networks for expressing probability distributions. *Advances in neural information processing systems*, 33:3094–3105.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer.

Shen, Z., Yang, H., and Zhang, S. (2019). Deep network approximation characterized by number of neurons. *arXiv preprint arXiv:1906.05497*.

Srebro, N. and Sridharan, K. (2010). Note on refined dudley integral covering number bound. http://ttic.uchicago.edu/karthik/dudley.pdf.

Wellner, J. et al. (2013). *Weak convergence and empirical processes: with applications to statistics*. Springer Science & Business Media.