Prominent Roles of Conditionally Invariant Components in Domain Adaptation: Theory and Algorithms

Keru Wu[*]   KERU.WU@DUKE.EDU
Department of Statistical Science
Duke University
Durham, North Carolina, USA

Yuansi Chen[*]   YUANSI.CHEN@DUKE.EDU
Department of Statistical Science
Duke University
Durham, North Carolina, USA

Wooseok Ha[*]   HAYWSE@KAIST.AC.KR
Department of Mathematical Sciences
Korea Advanced Institute of Science & Technology
Daejeon, Korea

Bin Yu   BINYU@BERKELEY.EDU
Department of Statistics and EECS
University of California, Berkeley
Berkeley, CA, USA

Editor: Mladen Kolar

Abstract

Domain adaptation (DA) is a statistical learning problem that arises when the distribution of the source data used to train a model differs from that of the target data used to evaluate the model. While many DA algorithms have demonstrated considerable empirical success, blindly applying these algorithms can often lead to worse performance on new datasets. To address this, it is crucial to clarify the assumptions under which a DA algorithm has good target performance. In this work, we focus on the assumption of the presence of conditionally invariant components (CICs), which are relevant for prediction and remain conditionally invariant across the source and target data. We demonstrate that CICs, which can be estimated through conditional invariant penalty (CIP), play three prominent roles in providing target risk guarantees in DA. First, we propose a new algorithm based on CICs, importance-weighted conditional invariant penalty (IW-CIP), which has target risk guarantees beyond simple settings such as covariate shift and label shift. Second, we show that CICs help identify large discrepancies between source and target risks of other DA algorithms. Finally, we demonstrate that incorporating CICs into the domain invariant projection (DIP) algorithm can address its failure scenario caused by label-flipping features. We support our new algorithms and theoretical findings via numerical experiments on synthetic data, MNIST, CelebA, Camelyon17, and DomainNet datasets.

---

[*]. Equal contribution

Keywords: Domain adaptation, Distribution shifts, Conditionally invariant components, Domain invariant projection, Anticausal learning

1. Introduction

The classical statistical learning problem assumes that the data used for training and those used for testing are drawn from the same data distribution. While this assumption is often valid, distribution shifts are prevalent in real-world data problems. Distribution shifts happen when the distribution of training (or source) data differs from that of test (or target) data (Koh et al., 2021). For example, when a machine learning model is trained on labeled source data from a few hospitals and then deployed in a new hospital, often there are distributional shifts because the data collection and pre-processing in different hospitals can be different (Veta et al., 2016; Komura and Ishikawa, 2018; Zech et al., 2018). The statistical learning problem that tackles distributional shifts with labeled source data and unlabeled target data is called a *domain adaptation* (DA) problem. A solution to DA is desired especially in situations where obtaining labeled data from target domain is difficult and expensive while unlabeled data is easily available. In this case, without collecting new labeled target data, one may attempt to pre-train the model on related large labeled datasets such as ImageNet (Deng et al., 2009) and adapt the model onto the unlabeled target data such as CT scan images (Cadrin-Chênevert, 2022). However, due to distributional shifts between the large labeled datasets and the target dataset, performance improvement is not always guaranteed (He et al., 2019). Without careful consideration, the presence of distributional shifts can result in a decrease in performance of many classical statistical learning algorithms.

While DA is an important learning problem, a generic cure is hopeless if there is no useful relation between the source and target data that can aid in prediction. In particular, the DA problem is ill-posed in general because for any given algorithm and source data, there will always be some arbitrarily chosen target data such that the algorithm trained on the source data will not perform well. Establishing reasonable assumptions relating the source and target data is critical, and depending on these assumptions, many ways to formulate a DA problem exist.

One common way to formulate a DA problem is to assume that the conditional distribution of label given the covariate, $Y \mid X$, remains the same in both source and target data. When $Y \mid X$ is invariant, it is implied that the covariate distribution changes. This formulation is known as *covariate shift* (Shimodaira, 2000; Quinonero-Candela et al., 2008). Successful approaches to tackle the covariate shift assumption include estimating the likelihood ratio between source and target covariates to correct for this shift (Shimodaira, 2000; Sugiyama et al., 2007; Sugiyama and Kawanabe, 2012). A related but different way of relating the source and target distributions is to assume that the conditional distribution of covariates, given the label, $X \mid Y$, is invariant. In this case, the marginal distributions of the label can differ. For this reason, this formulation is named *label shift* (Lipton et al., 2018). DA solutions typically involve correcting the likelihood ratio of labels using the conditional invariance of $X \mid Y$ (Azizzadenesheli et al., 2019; Tachet des Combes et al., 2020; Garg et al., 2020). Although covariate shift and label shift assumptions have been widely studied and

successfully applied in some cases (Sugiyama et al., 2007; Wu et al., 2021), their applicability is often limited in more practical scenarios.

Moving beyond the scope of covariate shift and label shift, another popular way of formulating DA is to assume the presence of invariant feature mappings (i.e., transformations of covariates X). Then the DA problem is reduced to identifying features that are important for the underlying task and are invariant across the source and target domains. Domain Invariant Projection (DIP) (Baktashmotlagh et al., 2013) was proposed as an attempt to identify these invariant features through projecting the source and target covariates into a common subspace. Subsequent works (Ganin et al., 2016; Tzeng et al., 2017; Hoffman et al., 2018b) have advanced the common subspace approach by incorporating neural network implementation, demonstrating empirical success across many datasets. Despite its empirical success, however, recent work by Johansson et al. (2019); Zhao et al. (2019) revealed that DIP may have a target risk much larger than its source risk, caused by the so-called label-flipping issue. Specifically, in the absence of target labels, in general there is no guarantee that DIP will find the true invariant representations. If DIP fails to do so, its target performance may deviate significantly from its source performance. What is worse is that currently there are no practical ways to check whether DIP has found the true invariant features, which presents a challenge for its practical use.

In this work, we make the assumption on the existence of conditionally invariant components (CICs) (Gong et al., 2016; Heinze-Deml and Meinshausen, 2021)—feature representations which are useful for prediction and whose distribution is invariant *conditioned* on the labels across source and target domains (see Definition 2 for the formal definition of CICs). With access to multiple source domain data that are related to the target domain, it becomes practically plausible to estimate CICs. In this setting, the existence of CICs can be well-justified because any features that are invariant, conditioned on the labels across these heterogeneous source domains, are likely to remain conditionally invariant in the target domain. The idea that takes advantage of the heterogeneity in multiple datasets has its origins in causality, robust statistics (Peters et al., 2016; Bühlmann, 2020) as well as stability-driven statistical analysis in the PCS framework (Yu and Kumbier, 2020). Moreover, in anticausal learning scenarios (Schölkopf et al., 2012) or when datasets are generated through structural causal models (Pearl, 2009; Chen and Bühlmann, 2021), CICs naturally emerge when unperturbed covariates are descendents of the labels.

Under the assumption on the existence of CICs, Conditional Invariant Penalty (CIP) is a widely used algorithm to identify CICs (Gong et al., 2016; Heinze-Deml and Meinshausen, 2021), through enforcing the invariance of conditional feature distributions across multiple source domains. CIP is shown to achieve good target performance under several theoretical settings (Heinze-Deml and Meinshausen, 2021; Chen and Bühlmann, 2021) and empirically (Li et al., 2018a,b; Jiang et al., 2020).

Despite the rapid development on CIP to identify CICs, the understanding of DA algorithms based on CICs beyond simple structural equation models is still limited, and their ability to handle DA problems involving both covariate and label shifts is unclear in the previous literature. Additionally, while the generalized label shift has been proposed in Tachet des Combes et al. (2020), it is not known how to reliably identify CICs via their proposed algorithms. Note that in the DA setting, target labels are unavailable, making it impossible to

3

estimate target performance through validation or cross-validation. Therefore, it is crucial to rigorously quantify target risk guarantees of DA algorithms under assumptions made.

Additionally, while DA algorithms based on CICs exhibit target performance comparable to that on the source data, often they are not the top performers (Heinze-Deml and Meinshausen, 2021; Chen and Bühlmann, 2021). This is mainly due to the fact that these algorithms only use source data, and the requirement for CICs to maintain invariance across multiple source domains may discard many features useful for target prediction. In contrast, DIP leverages a single source data and target covariates, which can lead to better target performance when successful, making it a preferred choice for many practitioners. However, in some cases, DIP can perform significantly worse than the Empirical Risk Minimization (ERM) (Wu et al., 2019; Zhao et al., 2019; Chen and Bühlmann, 2021). Given the potential advantages and risks of DIP, it is natural to ask whether there is a way to detect potential failure of DIP algorithms and improve their target performance with theoretical guarantees.

1.1 Our contributions

In this work, we show that learning CICs enables significant progress in overcoming the aforementioned challenges in DA with theoretical guarantees. Specifically, under the assumption on the existence of CICs and the availability of multiple source datasets, we highlight three prominent roles that CICs can play in enhancing DA algorithms.

First, we show that CICs can enable label shift correction in the presence of both covariate and label distribution shifts. Based on this, we introduce the importance-weighted conditional invariant penalty (IW-CIP) algorithm and analyze its theoretical performance. Under structural equation models, we show that CICs can be correctly identified via both CIP and IW-CIP using labeled data from multiple source domains. Consequently, it is only the finite-sample error gap that accounts for the difference between the target risk of the IW-CIP classifier and that of the optimal conditionally invariant classifier.

Second, we demonstrate how CICs can be used to provide target risk lower bounds for other DA algorithms without requiring access to target labels. Provided that CICs are accurately identified, this lower bound allows for assessing the target performance of any other DA algorithms, including DIP. Consequently, this approach makes it possible to detect algorithm failures using only source data and unlabeled target data.

Lastly, we introduce JointDIP, a new DA algorithm that extends the domain invariant projection (DIP) by incorporating CICs. Under structural equation models, we prove that JointDIP reduces the possibility of label-flipping and enhances the target prediction performance of CICs-based algorithms. Therefore, this algorithm addresses both the conservative nature of CIP and the potential risks associated with DIP. Our findings are supported by numerical experiments on synthetic and real datasets, including MNIST, CelebA, Camelyon17, and DomainNet.

The rest of the paper is organized as follows. Section 2 provides the necessary technical background and formally sets up the domain adaptation problem, along with a comprehensive review of related work in literature than in the introduction. In Section 3, we present the first role of CICs by introducing the IW-CIP algorithm and establish finite-sample target risk bounds to characterize its target risk performance (cf. Theorem 6.A, 6.B). Section 4

describes the other two roles of CICs in DA, demonstrating how they can be used to detect the failure of other DA algorithms (cf. Theorem 12), and introducing the JointDIP algorithm to address the label-flipping issues of DIP (cf. Theorem 15). In Section 5, we complement our theoretical arguments with extensive numerical experiments on synthetic and real datasets, emphasizing the importance of learning CICs as an essential part of domain adaptation pipeline. Finally, we conclude in Section 6 with potential directions for future work.

2. Background and problem setup

In this section, we begin by defining the domain adaptation problem and introducing the concept of conditionally invariant components (CICs). We then outline two baseline DA algorithms: the conditional invariant penalty (CIP) algorithm, which finds conditionally invariant representation across multiple source domains, and the domain invariant projection (DIP) algorithm, which works with a single source domain and takes advantage of additional unlabeled target data. Subsequently, we review related work to better position our contributions within the existing body of literature.

2.1 Domain adaptation problem setup

We consider the domain adaptation problem with M ($M \geq 1$) labeled source environments and one unlabeled target environment. By an *environment* or a *domain*, we mean a dataset $\mathcal{D}$ with i.i.d. samples drawn from a common distribution $\mathcal{P}$. Specifically, for $m \in \{1, \ldots, M\}$, in the m-th source environment, we observe $n^{(m)}$ i.i.d. samples

$$\mathcal{D}^{(m)} = \{(X_k^{(m)}, Y_k^{(m)})\}_{k=1}^{n^{(m)}},$$

drawn from the m-th source data distribution $\mathcal{P}^{(m)}$. Independently of the source data, there are $n^{(\mathfrak{T})}$ i.i.d. samples

$$\mathcal{D}^{(\mathfrak{T})} = \{(X_k^{(\mathfrak{T})}, Y_k^{(\mathfrak{T})})\}_{k=1}^{n^{(\mathfrak{T})}},$$

drawn from the target distribution $\mathcal{P}^{(\mathfrak{T})}$. We denote general random variables drawn from $\mathcal{P}^{(m)}$ and $\mathcal{P}^{(\mathfrak{T})}$ as $(X^{(m)}, Y^{(m)})$ and $(X^{(\mathfrak{T})}, Y^{(\mathfrak{T})})$, respectively. In the domain adaptation setting, all the source data are observed, while only the target covariates $\mathcal{D}_X^{(\mathfrak{T})} = \{X_k^{(\mathfrak{T})}\}_{k=1}^{n^{(\mathfrak{T})}}$ are observed in the target domain. For simplicity, throughout the paper, we assume that each covariate lies in a p-dimensional Euclidean space $\mathbb{R}^p$, and the labels belong to the set $\mathcal{Y} = \{1, \ldots, L\}$ where L represents the total number of classes.

To measure the performance of a DA algorithm, we define the *target population risk* of a classifier $h : \mathbb{R}^p \to \{1, \ldots, L\}$, mapping covariates to labels, via the 0-1 loss as

$$\mathcal{R}^{(\mathfrak{T})}(h) = \mathbb{E}\left[\mathbf{1}_{h(X^{(\mathfrak{T})}) \neq Y^{(\mathfrak{T})}}\right] = \mathbb{P}\left\{h(X^{(\mathfrak{T})}) \neq Y^{(\mathfrak{T})}\right\}. \tag{1}$$

Consequently, $1 - \mathcal{R}^{(\mathfrak{T})}(h)$ is the target population classification accuracy. Similarly, we define the m-th source population risk as

$$\mathcal{R}^{(m)}(h) = \mathbb{E}\left[\mathbf{1}_{h(X^{(m)}) \neq Y^{(m)}}\right] = \mathbb{P}\left\{h(X^{(m)}) \neq Y^{(m)}\right\}. \tag{2}$$

The main goal of the domain adaptation problem is to use source and unlabeled target data to estimate a classifier $h : \mathbb{R}^p \to \{1, \ldots, L\}$, from a set of functions called the hypothesis class $\mathcal{H}$, such that the target population risk is small. To quantify this discrepancy, we compare the target population risk with the *oracle target population risk* $\mathcal{R}^{(\mathfrak{T})}(h_{\text{oracle}})$ that we may aspire to achieve, where

$$h_{\text{oracle}} = \underset{h \in \mathcal{H}}{\arg \min} \, \mathcal{R}^{(\mathfrak{T})}(h). \tag{3}$$

Without specifying any relationship between the source distribution $\mathcal{P}^{(m)}$ and the target distribution $\mathcal{P}^{(\mathfrak{T})}$, there is no hope that the target population risk of a classifier learned from the source and unlabeled target data is close to the oracle target population risk. Throughout the paper, we focus on DA problems where conditionally invariant components (CICs) across all source and target environments are present and they are correlated with the labels. The existence of CICs was first assumed in Gong et al. (2016) and Heinze-Deml and Meinshausen (2021). Under assumptions of arbitrary large interventions and infinite data, Heinze-Deml and Meinshausen (2021) established that their classifier built on CICs achieves distributional robustness. In this paper, instead of discussing distributional robustness of a classifier, we construct classifiers that have target population risks close to the oracle target risk. Before that, we introduce CICs and the best possible classifier built upon CICs.

Definition 2 (Conditionally invariant components (CICs) Gong et al. (2016); Heinze-Deml and Meinshausen (2021)) *Suppose that there exist M source distributions $\{\mathcal{P}^{(m)}\}_{1 \leq m \leq M}$ and a target distribution $\mathcal{P}^{(\mathfrak{T})}$ on $\mathbb{R}^p \times \{1, 2, \ldots, L\}$. We say that a function $\phi : \mathbb{R}^p \to \mathbb{R}^q$ is a conditionally invariant feature mapping, if*

$$\mathcal{P}_{\phi(X)|Y=y}^{(m)} = \mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}, \quad \forall m \in \{1, \ldots, M\}, \ y \in \{1, \ldots, L\}. \tag{4}$$

The corresponding feature representation $\phi(X)$ is called a conditionally invariant component (CIC) if it has a single dimension ($q = 1$), and CICs if it is multidimensional ($q > 1$). When ϕ maps $\mathbb{R}^p$ to $\{1, 2, \ldots, L\}$ and satisfies Eq. (4), we refer to it as a conditionally invariant classifier.

We will use the term "CICs across source distributions" instead if the feature representations is conditionally invariant on all $\mathcal{P}^{(m)}$ but not necessarily on $\mathcal{P}^{(\mathfrak{T})}$. With this definition, our assumption on the "existence of CICs" can be stated as: there exists a conditionally invariant mapping ϕ such that $\phi(X)$ is CIC(s) across M source distributions $\{\mathcal{P}^{(m)}\}_{1 \leq m \leq M}$ and a target distribution $\mathcal{P}^{(\mathfrak{T})}$. In Definition 2, constant representations can be viewed as a trivial case of CICs. However, only CICs that are useful for prediction are beneficial for DA problems. Thus, our assumption on the existence of CICs refers to the existence of CICs useful for prediction. The best possible classifier built upon such CICs is the following optimal classifier.

Definition 3 (Optimal conditionally invariant classifier) *Let Φ and $\mathcal{G}$ are classes of functions where each function $\phi \in \Phi$ maps $\mathbb{R}^p$ to $\mathbb{R}^q$, and each function $g \in \mathcal{G}$ maps $\mathbb{R}^q$*

to $\{1, 2, \ldots, L\}$. *Under the assumption on the existence of CICs, we define the optimal conditionally invariant classifier $h^\star$ as*

$$
\begin{aligned}
h^\star &= g^\star \circ \phi^\star, \\
g^\star, \phi^\star &= \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg\min} \quad \mathcal{R}^{(\mathfrak{T})}(g \circ \phi) \\
&\text{subject to } \mathcal{P}^{(m)}_{\phi(X)|Y=y} = \mathcal{P}^{(\mathfrak{T})}_{\phi(X)|Y=y}, \quad \forall m \in \{1, \ldots, M\}, \ y \in \{1, \ldots, L\}.
\end{aligned}
\tag{5}
$$

The conditionally invariant classifier above is optimal in the sense that it minimizes the target population risk while the learned representation is conditionally invariant across $\mathcal{P}^{(m)}$ ($1 \leq m \leq M$) and $\mathcal{P}^{(\mathfrak{T})}$. When evaluating the target performance of a CICs-based classifier $h = g \circ \phi$, instead of directly comparing it with h_{oracle}, we consider comparing it with $h^\star$ first, and then relating $h^\star$ to h_{oracle}. Intuitively, the target risk difference between $h^\star$ and h_{oracle} will not be significant when the dimension of CICs is sufficiently large (c.f. Proposition 10). In this case, to build a CICs-based classifier with a guaranteed target risk bound compared to h_{oracle}, it suffices to find a classifier which achieves a low target risk compared to $h^\star$.

Another widely-used class of DA algorithms known as domain invariant projection (DIP) (cf. Section 2.2.2) seeks to find a feature mapping ϕ which matches the source and target marginal distribution of $\phi(X)$. Although it has been shown to be successful in some practical scenarios (Ganin et al., 2016; Mao et al., 2017; Hoffman et al., 2018b; Peng et al., 2019), in general there is no guarantee for the low target risk. Both Johansson et al. (2019) and Zhao et al. (2019) provide simple examples where DIP can even perform worse than a random guess, as if features learned by DIP "flip" the labels. We formulate the rationale behind their examples by defining label-flipping features as follows.

Definition 4 (Label-flipping feature) *Without loss of generality, consider the first source distribution $\mathcal{P}^{(1)}$ and the target distribution $\mathcal{P}^{(\mathfrak{T})}$ on $\mathbb{R}^p \times \{1, 2, \ldots, L\}$. We say that a function $f : \mathbb{R}^p \to \mathbb{R}$ is a label-flipping feature mapping, if there exists $y \in \{1, 2, \ldots, L\}$ such that* [1]

$$
\rho\left(f(X^{(1)}), \mathbf{1}_{Y^{(1)}=y}\right) \cdot \rho\left(f(X^{(\mathfrak{T})}), \mathbf{1}_{Y^{(\mathfrak{T})}=y}\right) < 0,
\tag{6}
$$

where $\rho(\cdot, \cdot)$ denotes the correlation between random variables. The corresponding feature $f(X)$ is called a label-flipping feature.

If the label-flipping features exist between a source distribution and the target distribution, they can be inadvertently learned by DIP as part of its learning algorithm for domain invariant representation. This can lead to degraded prediction performance on the target domain, as the sign of the correlation between these features and the labels changes under source and target distributions. We refer to it as the label-flipping issue of DIP.

Next we define an anticausal data generation model that serves as a concrete working example for validating our assumptions and establishing new results. While not all of our

---

1. When Y is binary, the definition is equivalent to $\rho\left(f(X^{(1)}), Y^{(1)}\right) \cdot \rho\left(f(X^{(\mathfrak{T})}), Y^{(\mathfrak{T})}\right) < 0$.

theoretical results depend on this model, it nevertheless aids in illustrating how our methods work.

Definition 5 (General anticausal model) *We say that the data generation model is a general anticausal model, if the source and target distributions are specified as follows. Under the m-th source distribution $\mathcal{P}^{(m)}$, source covariates and label are generated by*

$$Y^{(m)} \sim \text{Categorical}\left(p_1^{(m)}, p_2^{(m)}, \ldots, p_L^{(m)}\right),$$
$$X^{(m)} = f^{(m)}(Y^{(m)}) + \epsilon^{(m)}, \ \epsilon^{(m)} \perp Y^{(m)},$$

where $p_y^{(m)} \in (0,1), \sum_{y=1}^{L} p_y^{(m)} = 1$, and $f^{(m)} : \mathcal{Y} = \{1, \ldots, L\} \to \mathbb{R}^p$ is a deterministic function defining the mechanism between the m-th source covariates $X^{(m)}$ and label $Y^{(m)}$. Under the target distribution $\mathcal{P}^{(\mathfrak{T})}$, target covariates and label are generated independently of the source data by

$$Y^{(\mathfrak{T})} \sim \text{Categorical}\left(p_1^{(\mathfrak{T})}, p_2^{(\mathfrak{T})}, \ldots, p_L^{(\mathfrak{T})}\right),$$
$$X^{(\mathfrak{T})} = f^{(\mathfrak{T})}(Y^{(\mathfrak{T})}) + \epsilon^{(\mathfrak{T})}, \ \epsilon^{(\mathfrak{T})} \perp Y^{(\mathfrak{T})},$$

where $p_y^{(\mathfrak{T})} \in (0,1), \sum_{y=1}^{L} p_y^{(\mathfrak{T})} = 1$, and $f^{(\mathfrak{T})} : \mathcal{Y} = \{1, \ldots, L\} \to \mathbb{R}^p$ is a deterministic function defining the mechanism between the target covariates $X^{(\mathfrak{T})}$ and label $Y^{(\mathfrak{T})}$. The noise terms $\epsilon^{(m)}, \epsilon^{(\mathfrak{T})} \in \mathbb{R}^p$, are generated i.i.d. from a zero-mean distribution $\mathcal{P}_\epsilon$.

Under the general anticausal model, conditioned on the labels, the mechanism functions $f^{(m)}, f^{(\mathfrak{T})}, m = 1, \ldots, M$, determine the difference between the source and target conditional distributions $X \mid Y$ because the noise terms share the same distribution $\mathcal{P}_\epsilon$. This generative model generalizes various perturbations that can occur in an anticausal model (Pearl, 2009). For example, label shift might occur if the marginal distributions of Y differ. Covariate shift, conditioned on the labels, can occur when the deterministic functions $f^{(m)}, f^{(\mathfrak{T})}$ vary. We present an explicit example under this model, including the presence of both CICs and label-flipping features in Appendix A.

Notation To distinguish subscripts from coordinates, we represent the j-th coordinate of a constant vector x as $x_{[j]}$, and similarly, for a random vector X, we use $X_{[j]}$. Next, we introduce several notations for the empirical equivalents of population quantities. For the m-th source and target datasets $\mathcal{D}^{(m)}, \mathcal{D}^{(\mathfrak{T})}, m = 1, \ldots, M$, we use $\widehat{\mathcal{P}}^{(m)}$ and $\widehat{\mathcal{P}}^{(\mathfrak{T})}$ to denote the empirical data distributions, respectively. We define the m-th source and target empirical risk of a classifier $h \in \mathcal{H}$ as

$$\widehat{\mathcal{R}}^{(m)}(h) = \mathbb{E}_{(X,Y) \sim \widehat{\mathcal{P}}^{(m)}}\left[\mathbf{1}_{h(X) \neq Y}\right] = \frac{1}{n^{(m)}} \sum_{k=1}^{n^{(m)}} \mathbf{1}_{h(X_k^{(m)}) \neq Y_k^{(m)}}, \quad \text{and}$$

$$\widehat{\mathcal{R}}^{(\mathfrak{T})}(h) = \mathbb{E}_{(X,Y) \sim \widehat{\mathcal{P}}^{(\mathfrak{T})}}\left[\mathbf{1}_{h(X) \neq Y}\right] = \frac{1}{n^{(\mathfrak{T})}} \sum_{k=1}^{n^{(\mathfrak{T})}} \mathbf{1}_{h(X_k^{(\mathfrak{T})}) \neq Y_k^{(\mathfrak{T})}}.$$

For any mapping ϕ defined on $\mathbb{R}^p$, we use $\mathcal{P}_{\phi(X)}^{(m)}$ and $\mathcal{P}_{\phi(X)|Y=y}^{(m)}$ to denote the m-th source marginal distribution of $\phi(X)$ and the m-th source conditional distribution of $\phi(X)$ given its label $Y = y$. Similarly, we use $\mathcal{P}_{\phi(X)}^{(\mathfrak{T})}$ and $\mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}$ to denote the target marginal distribution of $\phi(X)$ and the target conditional distribution of $\phi(X)$ given its label $Y = y$. The corresponding empirical quantities are denoted by $\widehat{\mathcal{P}}_{\phi(X)}^{(m)}$, $\widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m)}$, $\widehat{\mathcal{P}}_{\phi(X)}^{(\mathfrak{T})}$, and $\widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(\mathfrak{T})}$, respectively.

Letting $\mathcal{P}$ be a distribution on $\mathbb{R}^q$ and $\mathcal{G}$ be a function class where each function $g \in \mathcal{G}$ maps $\mathbb{R}^q$ to $\mathbb{R}$, we recall the Rademacher complexity as

$$\mathfrak{R}_{n,\mathcal{P}}\left(\mathcal{G}\right) := \mathbb{E}_{Z_k \overset{\text{i.i.d.}}{\sim} \mathcal{P}, \sigma_k \overset{\text{i.i.d.}}{\sim} \sigma} \left[\sup_{g \in \mathcal{G}} \left| \frac{1}{n} \sum_{k=1}^{n} \sigma_k g(Z_k) \right| \right], \tag{7}$$

where σ_k's are random variables drawn independently from the Rademacher distribution, i.e., $\mathbb{P}\left\{\sigma_k = 1\right\} = \mathbb{P}\left\{\sigma_k = -1\right\} = 1/2$. Additionally, for any two distributions $\mathcal{P}$ and $\mathcal{Q}$ on $\mathbb{R}^q$ and a class of classifiers $\mathcal{G}$ where each function $g \in \mathcal{G}$ maps $\mathbb{R}^q$ to $\mathcal{Y} = \{1, 2, \cdots, L\}$, we define the $\mathcal{G}$-divergence between these two distributions as

$$D_{\mathcal{G}}\left(\mathcal{P}, \mathcal{Q}\right) := \sup_{g \in \mathcal{G}} \max_{y=1,2,\ldots,L} \left| \mathbb{E}_{Z \sim \mathcal{P}}\left[\mathbf{1}_{g(Z)=y}\right] - \mathbb{E}_{Z \sim \mathcal{Q}}\left[\mathbf{1}_{g(Z)=y}\right] \right|. \tag{8}$$

Note that the $\mathcal{G}$-divergence defined in Eq. (8) can be seen as an extension of the $\mathcal{H}$-divergence introduced in Ben-David et al. (2010) to multiclass classification. It is also an instantiation of the Integral Probability Metrics (Müller, 1997) with a specific choice of function class.

2.2 Two baseline DA algorithms

With the background of domain adaptation in place, we now proceed to introduce two DA algorithms in this subsection. The first algorithm is the conditional invariant penalty (CIP) that finds conditionally invariant representation across multiple source domains. The second algorithm is domain invariant projection (DIP), which works with a single source domain but also requires target covariates. These two DA algorithms serve as baselines for evaluating the methods we introduce in the subsequent sections.

2.2.1 CONDITIONAL INVARIANT PENALTY (CIP)

The *conditional invariant penalty* (CIP) algorithm uses the multiple labeled source environments to learn a feature representation that is conditionally invariant across all source domains (Gong et al., 2016; Heinze-Deml and Meinshausen, 2021). More precisely, the CIP algorithm is a two-stage algorithm which minimizes the average source risk across domains while enforcing the first-stage features to be conditionally invariant.

Population CIP (Gong et al., 2016; Heinze-Deml and Meinshausen, 2021): The population CIP classifier is formulated as a constrained optimization problem with a match-

ing penalty on the conditional:

$$h_{\text{CIP}} = g_{\text{CIP}} \circ \phi_{\text{CIP}},$$

$$g_{\text{CIP}}, \phi_{\text{CIP}} = \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg\min} \quad \frac{1}{M} \sum_{m=1}^{M} \mathcal{R}^{(m)}(g \circ \phi) \tag{9}$$

$$\text{subject to} \quad \mathfrak{D}\left(\mathcal{P}_{\phi(X)|Y}^{(m)}, \mathcal{P}_{\phi(X)|Y}^{(m')}\right) = 0 \text{ for all } m \neq m' \in \{1, \dots, M\},$$

where $\mathcal{R}^{(m)}(\cdot)$ is the m-th population source risk, and $\mathfrak{D}\left(\cdot, \cdot\right)$ is a distributional distance between two distributions such as the maximum mean discrepancy (MMD) Gretton et al. (2012) or generative adversarial networks (GAN) based distance (Ganin et al., 2016). The optimization is over the set of all two-stage functions where the first stage function belongs to Φ and the second stage to $\mathcal{G}$. The constraint on the conditional $\phi(X) \mid Y$ enforces CIP to use CICs across all $\mathcal{P}^{(m)}$ ($1 \leq m \leq M$) to build the final classifier. Here the hope is that if feature mappings are conditionally invariant across the heterogeneous source distributions $\mathcal{P}^{(m)}$, they are likely to be also conditionally invariant under the target distribution $\mathcal{P}^{(\mathfrak{T})}$. As a result, a classifier built on these features would generalize to the target distribution.

While CIP was originally introduced in the context of domain adaptation (Gong et al., 2016) and distributional robustness (Heinze-Deml and Meinshausen, 2021), it is closely related to the problem of *domain generalization*, or *out-of-distribution* (OOD) generalization (Ben-Tal et al., 2009). The main distinction between domain generalization and domain adaptation is the absence of any unlabeled target data during the training phase in domain generalization Wang et al. (2022). Since CIP does not require unlabeled target data for training, it has been effectively applied to domain generalization, demonstrating significant empirical success (Li et al., 2018a,b; Jiang et al., 2020). In this paper, we use the terms "domain generalizaton" and "domain adaptation" interchangeably since our focus is on evaluating and comparing the prediction performance of models on new target domains.

Finite-sample CIP: In finite-sample case, instead of putting a strict constraint in the optimization, CIP adds the conditional invariant penalty on the empirical distributions, using a pre-specified parameter $\lambda_{\text{CIP}} > 0$ to control the strength of regularization as follows:

$$\widehat{h}_{\text{CIP}} = \widehat{g}_{\text{CIP}} \circ \widehat{\phi}_{\text{CIP}},$$

$$\widehat{g}_{\text{CIP}}, \widehat{\phi}_{\text{CIP}} = \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg\min} \quad \frac{1}{M} \sum_{m=1}^{M} \widehat{\mathcal{R}}^{(m)}(g \circ \phi) + \frac{\lambda_{\text{CIP}}}{LM^2} \cdot \sum_{y=1}^{L} \sum_{m \neq m'} \mathfrak{D}\left(\widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m)}, \widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m')}\right). \tag{10}$$

While CIP makes use of CICs across multiple source distributions to construct the classifier, it does not exploit the target covariates that are also available in our DA setting. Indeed, CIP has often been employed as a method for domain generalization, where the goal is to generalize to the target domain without access to unlabeled target data. Because CIP finds invariant features through multiple source domains without considering target covariates during training, the learned features may be overly conservative for generalizing to the target domain. Later, we show how these CICs can be used to improve target prediction performace in our DA setting. Next, we introduce another class of DA algorithm which takes advantage of target covariates.

2.2.2 DOMAIN INVARIANT PROJECTION (DIP)

In contrast to CIP which utilizes multiple source data, *domain invariant projection* (DIP) (Baktashmotlagh et al., 2013; Ganin et al., 2016) uses labeled data from a single source as well as unlabeled data from the target domain to seek a common representation that is discriminative about the source labels. The idea of finding a common representation is realized via matching feature representations across source and target domains. Without loss of generality, we formulate DIP using the first source distribution $\mathcal{P}^{(1)}$, but in principle it can be formulated with any source distribution $\mathcal{P}^{(m)}$.

Population DIP (Baktashmotlagh et al., 2013; Ganin et al., 2016): We define the population DIP as a minimizer of the source risk under the constraint of marginal distribution matching in feature representations:

$$
\begin{aligned}
h_{\text{DIP}} &= g_{\text{DIP}} \circ \phi_{\text{DIP}}, \\
g_{\text{DIP}}, \phi_{\text{DIP}} &= \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg\min} \quad \mathcal{R}^{(1)}(g \circ \phi) \\
&\text{subject to} \quad \mathfrak{D}\left(\mathcal{P}^{(1)}_{\phi(X)}, \mathcal{P}^{(\mathfrak{T})}_{\phi(X)}\right) = 0,
\end{aligned}
\tag{11}
$$

where $\mathfrak{D}\left(\cdot, \cdot\right)$ is a distributional distance between two distributions as in Eq. (9). The constraint ensures that the source marginal distribution in the feature representation space is well aligned with the target marginal distribution. While our formulation only utilizes the single source, DIP also has a multi-source pooled version (Peng et al., 2019) where marginal distributions in the representation space are matched across all $\mathcal{P}^{(m)}$ ($1 \leq m \leq M$) and $\mathcal{P}^{(\mathfrak{T})}$.

Finite-sample DIP: In the finite sample setting, the hard constraint used in population DIP is relaxed to take a regularization form, and therefore

$$
\begin{aligned}
\widehat{h}_{\text{DIP}} &= \widehat{g}_{\text{DIP}} \circ \widehat{\phi}_{\text{DIP}}, \\
\widehat{g}_{\text{DIP}}, \widehat{\phi}_{\text{DIP}} &= \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg\min} \quad \widehat{\mathcal{R}}^{(1)}(g \circ \phi) + \lambda_{\text{DIP}} \cdot \mathfrak{D}\left(\widehat{\mathcal{P}}^{(1)}_{\phi(X)}, \widehat{\mathcal{P}}^{(\mathfrak{T})}_{\phi(X)}\right),
\end{aligned}
\tag{12}
$$

where $\lambda_{\text{DIP}} > 0$ is a regularization parameter that balances between the source risk and the matching penalty across the empirical source and target marginal distributions for the feature representation.

While DIP makes use of the target covariates to learn domain-invariant representations, in general it does not have target risk guarantees. In particular, the matching penalty can force DIP to learn label-flipping features (see Definition 4) because unlike CIP, it aligns features in the marginal representation space. DIP leverages information about target covariates to learn invariant features, but this comes at the cost of potential label-flipping issue (c.f. see Figure 1 and Theorem 15). Furthermore, DIP can fail when the marginal distribution of Y is perturbed under an anticausal data generation model. For example, Chen and Bühlmann (2021) demonstrates that DIP can be perform worse than CIP in the presence of label shift.

2.3 Related work

As a subfield of transfer learning, *domain adaptation* (DA), also known as transductive transfer learning (Redko et al., 2020), aims to develop effective algorithms when distribution shifts exist across training data (source domains) and test data (target domain). In DA, it is usually assumed that we have access to some unlabeled target data, and we aim to build a model specifically for the target domain. Another related terminology *domain generalization* (DG), or out-of-distribution (OOD) generalization (Ben-Tal et al., 2009), instead assumes that unlabeled target data is unobtainable, and seeks a model to generalize appropriately for all possible test domains. In this paper, we do not distinguish these two terms particularly, and simply consider the DA setting, i.e., we have access to an unlabeled target dataset, and our goal is to find a classifier that performs well on the target domain.

Domain adaptation with common feature representations One line of work in DA focuses on relating source and target domains by learning a common feature representation across the domains. Pan et al. (2008, 2010) first proposed to find transferable components across domains in a reproducing kernel Hilbert space, such that data distributions in different domains are close. Similar designs of creating intermediate representations across domains were investigated in (Gopalan et al., 2011; Gong et al., 2012), and later the formal idea of matching probability distributions and extracting invariant information was introduced in (Baktashmotlagh et al., 2013). They named their approach domain invariant projection (DIP), which projects data to a low-dimensional latent space where source and target covariates distributions are matched. This DIP type of approach was then widely employed in research on neural networks for DA (Sun and Saenko, 2016; Ganin et al., 2016; Hoffman et al., 2018b). Specifically, Ganin et al. (2016) introduced the Domain-Adversarial Neural Network (DANN), which leverages Generative Adversarial Network (GAN) based distributional distances and neural networks to learn feature representations. Sun and Saenko (2016) proposed CORAL, a method that aligns the first and second moments of last layer activations in neural networks, whereas Courty et al. (2016) proposed to match source and target domains via regularized optimal transport. Since then, subsequent research has emerged that applies different neural-network-based distances (Long et al., 2017; Courty et al., 2017; Long et al., 2018; Hoffman et al., 2018b; Peng et al., 2019). While these studies have effectively demonstrated the empirical success of their methodologies across various image and text datasets, there is still a lack of understanding about the specific conditions under which these methods can achieve good target performance.

From a theoretical point of view, the rationale for DIP has been rigorously established by Ben-David et al. (2006, 2010), where they prove a target risk bound via Vapnik-Chervonenkis (VC) theory. Further studies have analyzed the source and target risk difference using different divergence measures (Mansour et al., 2009; Cortes et al., 2010; Cortes and Mohri, 2011, 2014; Hoffman et al., 2018a); see the survey by Redko et al. (2020) for a complete review. In all these works, the target risk bound typically includes three components: the source risk of the classifier, a divergence term measuring invariance of the representation, and an optimal joint error term. Then DIP objective can be viewed as minimizing the sum of the first two terms in order to achieve a low target risk. However, Johansson et al. (2019); Zhao et al. (2019) argued that DIP can completely fail in cer-

tain scenarios because the joint error term is not observable and cannot be controlled by DIP. For instance, there can be label-flipping features that achieve perfect source accuracy and invariance of representation, but result in poor performance on the target domain. Addressing how to avoid such cases in DIP has not been adequately explored.

DA methods under conditional invariance and label shift The second class of DA methods arises from exploring invariance solely from multiple labeled source domains, without using target covariates. One such category of invariance is known as conditional invariance, which aims to discover feature representations $\phi(X)$ that are invariant conditional on the label Y across source distributions. It was first introduced in Gong et al. (2016), where the authors proposed to find conditional transferable components after proper location-scale transformations. Later Heinze-Deml and Meinshausen (2021) applied a related approach by classifying features into "core" ones, which are conditionally invariant across domains, and "style" ones, whose distribution may shift significantly. They sought to construct a classifier built upon only the core features by imposing the conditional invariant penalty (CIP). Chen and Bühlmann (2021) further developed a theoretical framework under structural equation models to analyze the effect of CIP when the data generation process is anticausal. The emergence of conditionally invariant features, or CICs, is a natural consequence under anticausal data generation where the covariates, which are descendants of the labels, remain unperturbed. Consequently, most tasks tackled by DA methods utilizing conditional invariance typically involve anticausal problems.

It is also worth noting that under conditional invariance, it is convenient to study the label shift problem, where the marginal distributions of the label Y are shifted across domains. This shift in label distribution is common in many scenarios, for example, the distribution of ArXiv paper categories can be influenced by changes in research topic trends over time (Wu et al., 2021). Lipton et al. (2018) first proposed the label shift correction algorithm, which estimates the amount of label shift by using the conditional invariance of the covariates and the moment matching equation. To ensure numerically stability, several variants of the algorithms have been further introduced. For instance, Azizzadenesheli et al. (2019) formulated an ℓ_2-norm regularized least squares optimization problem, while Tachet des Combes et al. (2020) considered a constrained least squares problem and the generalized label shift assumption. Garg et al. (2020) introduced a maximum likelihood estimation approach and provided a unified view of previous label shift correction algorithms. More recently, Chen et al. (2022) has considered a distributional shift named Sparse Joint Shift (SJS), which allows for both labels and a few covariates to shift, however, the generalized label shift considered in Tachet des Combes et al. (2020) and this paper permits distribution shifts of more general latent feature representations.

Other DA methods Besides conditional invariance, other types of invariance have also attracted increasing attention. For example, Arjovsky et al. (2019) came up with the Invariant Risk Minimization (IRM) method to identify causal features via enforcing the invariance of label Y given the features $\phi(X)$. The idea is that the optimal linear predictor on top of these features $\phi(X)$ will remain the same across all source domains—this is in contrast to the assumption on CICs where there exist conditionally invariant features conditioned on the labels Y. Following IRM, analogous invariance has been explored. For

example, (Krueger et al., 2021; Xie et al., 2020) proposed V-REx, which maintains risk invariance by reducing the variance of risks across source domains. Meanwhile, (Koyama and Yamaguchi, 2020; Shi et al., 2022) maximized the gradient inner product between different domains to leverage invariant gradient direction, with the latter referring to their method as Fish. While these approaches were initially motivated to identify invariant features in causal data models, they have also been applied to anticausal problems. However, as pointed out by Rosenfeld et al. (2021); Kamath et al. (2021), IRM may fail to capture the correct invariance, especially when the model is non-linear.

Counterfactual invariance is another type of invariance introduced in (Veitch et al., 2021; Jiang and Veitch, 2022), where the aim is to seek representations that are counterfactually invariant to a spurious factor of variation. (Wang and Veitch, 2022) show that the formulation of learning counterfactual invariant representations is closely related to different formulations of invariant representation learning algorithms, depending on the underlying causal structure of the data. Aside from the robustness achieved by enforcing specific invariance, a more general framework is distributionally robust optimization (DRO) (Ben-Tal et al., 2013; Duchi et al., 2021). DRO aims to minimize the worst-case loss over an uncertainty set of distributions, such as a ball around the training distribution. However, this approach may result in models that are overly conservative. GroupDRO (Hu et al., 2018; Oren et al., 2019) intends to address this issue by defining the uncertainty set as mixtures of source distributions (or groups). Furthermore, (Sagawa et al., 2020) improve the worst-group performance of groupDRO using an online algorithm to update group weights. It should be noted that related study of invariance and robustness has also appeared in works from a causal inference point of view (Peters et al., 2016; Meinshausen, 2018; Rothenhäusler et al., 2021; Magliacane et al., 2018).

Other DA algorithms have also been introduced based on different assumptions relating source and target data, which do not fall into the above classes of methods relying on invariance. For instance, self training, which originated from the semi-supervised learning literature (Chapelle et al., 2009), recursively adapts a classifier to fit pseudolabels predicted by a previous model using unlabeled data in a new domain (Amini and Gallinari, 2003). Theoretical properties of self training has been studied recently in (Kumar et al., 2020; Wei et al., 2021). Data augmentation serves as another beneficial way of obtaining better performance (Simard et al., 1998; Zhang et al., 2018; Yao et al., 2022), especially when possible perturbations across domains are well understood. In addition, a different perspective of DA stems from meta-learning (Thrun and Pratt, 2012; Finn et al., 2017). This "learning-to-learn" approach aims to distill knowledge of multiple learning episodes to improve future learning performance.

3. Importance-weighted CIP with target risk guarantees

In the previous section, we introduced two baseline DA algorithms, namely CIP and DIP. However, it is crucial to acknowledge the limitations of these algorithms, as they rely on specific assumptions that may not hold in more complex DA scenarios. For instance, while CIP identifies CICs to build the classifier, its ability to generalize to the target domain is limited when the marginal label distributions shift across source and target. DIP faces

similar limitations and is also subject to the additional uncertainty of learning label-flipping features.

In this section, we present the first role of CICs in addressing situations where neither covariate shift nor label shift assumptions hold, and we introduce the importance-weighted conditional invariant penalty (IW-CIP) algorithm. The intuition behind label shift correction with CICs is as follows. It is known that one can correct for label distribution shift if the conditional $X \mid Y$ is invariant and only the label Y distribution changes across source and target (Lipton et al., 2018). However, the assumption on invariance of $X \mid Y$ can be too rigid. Here we assume the existence of datasets from multiple source domains and a conditionally invariant feature mapping ϕ_{inv} such that $\phi_{\mathrm{inv}}(X) \mid Y$ remains invariant across source and target distributions. By identifying a conditionally invariant $\phi_{\mathrm{inv}}(X)$ via CIP, we can apply the label shift correction algorithm to correct the label shift. Once the label shift is corrected, the joint distribution of $(\phi_{\mathrm{inv}}(X), Y)$ becomes invariant under source and target distributions, allowing us to control the target risk of any algorithms built upon the source $(\phi_{\mathrm{inv}}(X), Y)$. IW-CIP is an extension of CIP by incorporating a label shift correction step before building the classifier based on CICs.

The rest of the section is structured as follows. In Section 3.1, we offer a review on label shift correction and its application in our context. Then, we introduce our new algorithm, IW-CIP, in Section 3.2. In Section 3.3, we establish target risk guarantees for IW-CIP.

3.1 Importance weights estimation

When the m-th source distribution and the target distribution share the same conditional $X \mid Y$ but have different label distributions, the main idea of label shift correction in Lipton et al. (2018) lies in that the true importance weights vector $w^{(m)} \in \mathbb{R}_+^L$, defined as

$$
w_{[j]}^{(m)} := \frac{\mathbb{P}\left\{Y^{(\mathfrak{T})} = j\right\}}{\mathbb{P}\left\{Y^{(m)} = j\right\}},
\tag{13}
$$

can be estimated by exploiting the invariance of the conditional $X \mid Y$. In our setting, while the source and target distribution does not share the same conditional $X \mid Y$, according to our assumption, we have a feature mapping ϕ_{inv} whose corresponding feature representation is a CIC by Definition 2, i.e., $\phi_{\mathrm{inv}}(X) \mid Y$ is invariant under source and target distributions. Then by treating $\phi_{\mathrm{inv}}(X)$ as the new features, we can still correct for the label shifts.

More precisely, for $\mathcal{Y} = \{1, \ldots, L\}$, we have the following distribution matching equation between the m-th source domain and the target domain: for any $g \in \mathcal{G}$ mapping to $\mathcal{Y}$, and for any $i, j \in \mathcal{Y}$,

$$
\mathbb{P}\left\{g \circ \phi_{\mathrm{inv}}(X^{(\mathfrak{T})}) = i\right\} = \sum_{j=1}^L \mathbb{P}\left\{g \circ \phi_{\mathrm{inv}}(X^{(\mathfrak{T})}) = i \mid Y^{(\mathfrak{T})} = j\right\} \mathbb{P}\left\{Y^{(\mathfrak{T})} = j\right\}
$$

$$
\stackrel{(*)}{=} \sum_{j=1}^L \mathbb{P}\left\{g \circ \phi_{\mathrm{inv}}(X^{(m)}) = i \mid Y^{(m)} = j\right\} \mathbb{P}\left\{Y^{(\mathfrak{T})} = j\right\}
$$

$$
= \sum_{j=1}^L \mathbb{P}\left\{g \circ \phi_{\mathrm{inv}}(X^{(m)}) = i, Y^{(m)} = j\right\} w_{[j]}^{(m)},
\tag{14}
$$

15

where step $(*)$ follows from the invariance of $\phi_{\text{inv}}(X) \mid Y$. In the matrix-vector form, we can write

$$\mu_{g \circ \phi_{\text{inv}}} = C^{(m)}_{g \circ \phi_{\text{inv}}} w^{(m)}, \tag{15}$$

where μ_h denotes the predicted probability of h under the target covariate distribution, and $C^{(m)}_h$ is the confusion matrix on the $\mathcal{P}^{(m)}$ given by

$$C^{(m)}_h[i, j] = \mathbb{P}\left\{ h(X^{(m)}) = i, Y^{(m)} = j \right\}. \tag{16}$$

It is then sufficient to solve the linear system (15) to obtain $w^{(m)}$. In practice, with finite-sample source data, we use $\widehat{h}_{\text{CIP}} = \widehat{g}_{\text{CIP}} \circ \widehat{\phi}_{\text{CIP}}$ in place of $g \circ \phi_{\text{inv}}$. To obtain the estimated importance weights $\widehat{w}^{(m)}$, we replace $\mu_{\widehat{h}_{\text{CIP}}}$ and $C^{(m)}_{\widehat{h}_{\text{CIP}}}$ with their empirical estimates. In our multiple source environments scenario, we write $w = (w^{(1)}, w^{(2)}, \dots, w^{(M)}) \in \mathbb{R}^{L \times M}$ and $\widehat{w} = (\widehat{w}^{(1)}, \widehat{w}^{(2)}, \dots, \widehat{w}^{(M)}) \in \mathbb{R}^{L \times M}$ to denote the true and estimated importance weights for all source distributions, respectively.

3.2 Our proposed IW-CIP algorithm

When the target label distribution remains unchanged, the population CIP in Eq. (9) is capable of generalizing to target data because of the invariance of the joint distribution $(\phi_{\text{CIP}}(X), Y)$. However, although CIP ensures invariance in the conditional distribution $\phi_{\text{CIP}}(X) \mid Y$, the joint distribution $(\phi_{\text{CIP}}(X), Y)$ may not be invariant when label shift is present. Indeed, CIP may perform poorly on the target data if the target label distribution substantially deviates from the source label distribution (see experiments on synthetic data and rotated MNIST in Section 5). To address such distributional shift, we propose the IW-CIP algorithm, which combines importance weighting for label shift correction with CIP.

We define the m-th weighted source risk for a hypothesis $h \in \mathcal{H}$ and a weight vector $\mathrm{w}^{(m)} \in \mathbb{R}^L$ as follows:

$$\mathcal{R}^{(m)}(h; \mathrm{w}^{(m)}) = \mathbb{E}\left[\mathrm{w}^{(m)}_{[Y^{(m)}]} \cdot \mathbf{1}_{h(X^{(m)}) \neq Y^{(m)}} \right].$$

In particular, if $\mathrm{w}^{(m)} = w^{(m)}$, i.e., $\mathrm{w}^{(m)}_{[j]} = \frac{\mathbb{P}\{Y^{(\mathfrak{T})} = j\}}{\mathbb{P}\{Y^{(m)} = j\}}$ for all $j = 1, \dots, L$, then it is easy to see that $\mathcal{R}^{(m)}(h; \mathrm{w}^{(m)}) = \mathcal{R}^{(\mathfrak{T})}(h)$ as long as the conditional distributions $h(X) \mid Y = y$ are invariant across $\mathcal{P}^{(m)}$ and $\mathcal{P}^{(\mathfrak{T})}$. Hence, with an appropriate choice of the weight vector $\mathrm{w}^{(m)}$, the weighted source risk can serve as a proxy for the target risk. We are ready to introduce the importance-weighted CIP (IW-CIP) algorithm.

Population IW-CIP: The population IW-CIP is obtained in three steps.

Step 1: Obtain a conditionally invariant feature mapping ϕ_{CIP} and the corresponding CIP classifier h_{CIP} via the CIP algorithm in Eq. (9).

Step 2: Use h_{CIP} in place of $g \circ \phi_{\text{inv}}$ in Eq. (15) to obtain importance weights $w^{(m)}$.

Step 3: Compute a function $g \in \mathcal{G}$ as well as a new conditionally invariant feature mapping $\phi \in \Phi$ to minimize the importance-weighted source risks as follows.

$$h_{\text{IW-CIP}} = g_{\text{IW-CIP}} \circ \phi_{\text{IW-CIP}},$$

$$g_{\text{IW-CIP}}, \phi_{\text{IW-CIP}} = \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg\min} \quad \frac{1}{M} \sum_{m=1}^{M} \mathcal{R}^{(m)}(g \circ \phi; w^{(m)}) \tag{17}$$

$$\text{subject to} \quad \mathfrak{D}\left(\mathcal{P}_{\phi(X)|Y}^{(m)}, \mathcal{P}_{\phi(X)|Y}^{(m')}\right) = 0, \ \forall m \neq m' \in \{1, \dots, M\}.$$

IW-CIP enforces the same constraint on the data representation ϕ as in Eq. (9). However, unlike CIP, the objective of IW-CIP is the importance-weighted source risk, which can serve as a proxy for the target risk under the label distribution shifts. Therefore, IW-CIP can generalize better on the target environment in the presence of label shifts.

Our formulation of IW-CIP in Eq. (17) involves retraining the feature mapping $\phi_{\text{IW-CIP}}$, instead of using ϕ_{CIP} obtained in Step 1, despite that both feature mappings are conditionally invariant. From a theoretical perspective, retraining ϕ enables a more simplified derivation of the generalization bound for IW-CIP where we compare the objective of the retrained classifier with that of the optimal conditionally invariant classifier (c.f. see Eq. (43) in the proof of Theorem 6.B). Empirically, we observe that retraining often leads to better empirical performance, providing both theoretical and practical advantages of our formulation.

Finite-sample IW-CIP: The finite-sample IW-CIP is obtained from the population IW-CIP after replacing all the population quantities by the corresponding empirical estimates and turning constraints into a penalty form. We first solve finite-sample CIP from Eq. (10), then estimate importance weights and solve

$$\widehat{h}_{\text{IW-CIP}} = \widehat{g}_{\text{IW-CIP}} \circ \widehat{\phi}_{\text{IW-CIP}},$$

$$\widehat{g}_{\text{IW-CIP}}, \widehat{\phi}_{\text{IW-CIP}} = \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg\min} \quad \frac{1}{M} \sum_{m=1}^{M} \widehat{\mathcal{R}}^{(m)}(g \circ \phi; \widehat{w}^{(m)}) \tag{18}$$

$$+ \frac{\lambda_{\text{IW-CIP}}}{LM^2} \cdot \sum_{y=1}^{L} \sum_{m \neq m'} \mathfrak{D}\left(\widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m)}, \widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m')}\right),$$

where $\widehat{w}^{(m)}$ is an estimate of $w^{(m)}$, obtained by solving $\widehat{\mu}_{\widehat{h}_{\text{CIP}}} = \widehat{C}_{\widehat{h}_{\text{CIP}}}^{(m)} \widehat{w}^{(m)}$. Here $\widehat{\mu}_{\widehat{h}_{\text{CIP}}}$ and $\widehat{C}_{\widehat{h}_{\text{CIP}}}^{(m)}$ are empirical estimates of $\mathbb{P}\left\{\widehat{h}_{\text{CIP}}(X^{(\mathfrak{T})}) = i\right\}$ and $\mathbb{P}\left\{\widehat{h}_{\text{CIP}}(X^{(m)}) = i, Y^{(m)} = j\right\}$, respectively. For any $\phi \in \Phi$, we define a shorthand for the empirical conditional invariant penalty used in the finite-sample IW-CIP by

$$\widehat{\Lambda}_\phi := \frac{\lambda_{\text{IW-CIP}}}{LM^2} \sum_{y=1}^{L} \sum_{m \neq m'} \mathfrak{D}\left(\widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m)}, \widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m')}\right). \tag{19}$$

The multi-step procedure of IW-CIP raises a natural question of whether we can iterate the estimation of CIP and reweighting multiple times, similar to the EM algorithm (Dempster

et al., 1977). However, without target labels, it is unclear whether this IW-CIP can be improved via reiteration. The primary purpose of IW-CIP is to maintain the same level of conditional invariance as CIP while adjusting the classifier to accommodate target label shifts. At the end of Section 3.3.1, we explain why this approach based on iteration does not work theoretically.

3.3 Target risk upper bounds of IW-CIP

In this subsection, we state our main theorems on the target risk upper bounds for IW-CIP. In a nutshell, the target risk bound can be decomposed into multiple terms involving the source risk or the optimal target risk, error in importance weights estimation, and the deviation from conditional invariance of the finite-sample CIP or IW-CIP feature mapping. Consequently, when the importance weights are accurately estimated and the identified features are near conditional invariance, IW-CIP achieves high target accuracy.

To simplify the theoretical analysis that follows, our finite-sample results are stated by considering the case where the whole source dataset is split into three parts of equal size. That is, the ℓ-th ($\ell = 1, 2, 3$) source dataset $\mathcal{D}_\ell = \{\mathcal{D}_\ell^{(1)}, \mathcal{D}_\ell^{(2)}, \ldots, \mathcal{D}_\ell^{(M)}\}$ is denoted by

$$\mathcal{D}_\ell^{(m)} = \{(X_{\ell,k}^{(m)}, Y_{\ell,k}^{(m)})\}_{k=1}^{n^{(m)}} \quad \text{for } m \in \{1, \ldots, M\}. \tag{20}$$

The number of samples for each source dataset is given by $n^{(m)}$. When it is clear which dataset we are referring to, we simply omit the dataset subscript ℓ in covariates and labels by writing $X_k^{(m)}, Y_k^{(m)}$. In our finite-sample theory, the first $\mathcal{D}_1$ is used for solving the finite-sample CIP, the second $\mathcal{D}_2$ is used for estimating importance weights and correcting the label shift, and the last $\mathcal{D}_3$ is used for solving the finite-sample IW-CIP. We do not apply a similar split to the unlabeled target dataset, because it is only used "once" for estimating the importance weights through solving the linear system Eq. (15).

Given importance weights $\mathrm{w}^{(m)} \in \mathbb{R}^L$ ($1 \leq m \leq M$) for each source distribution, we write $\mathrm{w} = (\mathrm{w}^{(1)}, \ldots, \mathrm{w}^{(M)}) \in \mathbb{R}^{L \times M}$ and introduce the following shorthand for the average weighted source risk across M environments,

$$\overline{\mathcal{R}}(h; \mathrm{w}) := \frac{1}{M} \sum_{m=1}^{M} \mathcal{R}^{(m)}(h; \mathrm{w}^{(m)}).$$

Before we introduce our main theorems, we define a key quantity called *deviation from conditional invariance* of a feature mapping, which measures how conditionally invariant it is across source and target distributions.

Definition 6 (Deviation from Conditional Invariance) *Recall the $\mathcal{G}$-divergence given in Eq. (8). For any feature mapping $\phi : \mathbb{R}^p \to \mathbb{R}^q$, we define its deviation from conditional invariance as*

$$\Psi_{\mathcal{G}, \phi} := \max_{\substack{m=1,\ldots,M, \\ y=1,\ldots,L}} D_{\mathcal{G}} \left(\mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}, \mathcal{P}_{\phi(X)|Y=y}^{(m)} \right). \tag{21}$$

18

The deviation from conditional invariance is defined via the maximal $\mathcal{G}$-divergence between any pair of conditionals $\phi(X) \mid Y = y$ in the source and target environments. When the feature representation is exactly conditionally invariant, this quantity attains its minimum value of zero. Our first theorem bounds the difference between target population risk and the average source population risk of a classifier, where $w^{(m)}$, $m = 1, \ldots, M$, are the true importance weights introduced in Eq. (13).

Theorem 6.A *For any classifier $h = g \circ \phi$ where $g \in \mathcal{G}, \phi \in \Phi$ and any estimated importance weights $\widehat{w} = (\widehat{w}^{(1)}, \ldots, \widehat{w}^{(M)}) \in \mathbb{R}^{L \times M}$, the following target risk bound holds:*

$$\mathcal{R}^{(\mathfrak{T})}(h) \leq \overline{\mathcal{R}}(h; \widehat{w}) + \frac{1}{M} \sum_{m=1}^{M} \left\| w^{(m)} - \widehat{w}^{(m)} \right\|_{\infty} + \Psi_{\mathcal{G}, \phi}.$$

The proof of this theorem is given in Appendix B.1. We observe that IW-CIP is designed to minimize the upper bounds in Theorem 6.A. Specifically, it is expected that for the case of IW-CIP, the estimated importance weights are close to the true importance weights and ϕ is close to the conditionally invariant feature mapping. Then the average weighted source risk can closely approximate the target risk, and Theorem 6.A allows us to establish an upper bound for the target risk of IW-CIP via the average weighted source risk.

To provide more specific risk guarantees for the finite-sample IW-CIP, we establish the following bound on the target risk of the finite-sample IW-CIP via the target risk of the optimal conditionally invariant classifier $h^{\star}$, as defined in Definition 3. Here, $\widehat{\Lambda}_{\phi}$ denotes the empirical IW-CIP penalty given in Eq. (19) and $w^{(m)}$'s are the true importance weights given in Eq. (13).

Theorem 6.B *Let $\widehat{w} = (\widehat{w}^{(1)}, \ldots, \widehat{w}^{(M)}) \in \mathbb{R}^{L \times M}$ be the estimated importance weights. Then, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$, the following target risk bound holds for the finite-sample IW-CIP,*[2]

$$\mathcal{R}^{(\mathfrak{T})}(\widehat{h}_{\text{IW-CIP}}) \leq \mathcal{R}^{(\mathfrak{T})}(h^{\star}) + \widehat{\Lambda}_{\phi^{\star}} + \frac{2}{M} \sum_{m=1}^{M} \left\| w^{(m)} - \widehat{w}^{(m)} \right\|_{\infty} + \Psi_{\mathcal{G}, \widehat{\phi}_{\text{IW-CIP}}} + \gamma,$$

where[3]

$$\gamma = \max_{m=1,\ldots,M} \left[4 \mathfrak{R}_{n^{(m)}, \mathcal{P}^{(m)}} \left(\mathcal{H}^{(m)} \right) + 2 \left\| w^{(m)} \right\|_{\infty} \sqrt{\frac{2 \log(M/\delta)}{n^{(m)}}} \right],$$

and $\mathcal{H}^{(m)} := \left\{ f(x, y) = w_{[y]}^{(m)} \mathbf{1}_{g(\phi(x)) \neq y} : g \in \mathcal{G}, \phi \in \Phi \right\}.$

The proof proceeds by decomposing the target risk of IW-CIP into multiple components and bounding each term individually, which is given in Appendix B.2. According to Theorem 6.B, the target risk of $\widehat{h}_{\text{IW-CIP}}$ is bounded by that of the optimal conditionally invariant

---

2. The probability is with respect to the randomness of source samples $(X_k^{(m)}, Y_k^{(m)}) \overset{\text{i.i.d.}}{\sim} \mathcal{P}^{(m)}$ in $\mathcal{D}_3$ $(1 \leq m \leq M, 1 \leq k \leq n^{(m)})$; see Eq. (20).

3. The sample complexity term γ depends on $\delta, \mathcal{G}, \Phi, w^{(m)}, n^{(m)}$, and $\mathcal{P}^{(m)}$ for $m = 1, \ldots, M$.

target classifier $h^\star$ with additional error terms: the empirical IW-CIP penalty of $\phi^\star$, the importance weight estimation error, the deviation from conditional invariance of $\widehat{\phi}_{\text{IW-CIP}}$, and the sample complexity term γ. The empirical IW-CIP penalty term $\widehat{\Lambda}_{\phi^\star}$ measures the conditional invariance of $\phi^\star(X)$ across empirical source environments. Because $\phi^\star$ is a conditionally invariant feature mapping, this term is expected to decrease in large sample scenarios. Similarly, the sample complexity term γ, which is based on Rademacher complexity, also diminishes as the sample size $n^{(m)}$ increases in the source environments. In this case, the theorem shows that accurate estimation of importance weights and minimal deviation from conditional invariance of feature representations of IW-CIP can guarantee IW-CIP to achieve a target risk similar to that of $h^\star$.

Theorem 6.B provides the target risk bound of the finite sample IW-CIP in the most generic settings. In the following subsections, we demonstrate how the remaining terms can be controlled with additional assumptions. Specifically, in Section 3.3.1, we establish refined bounds for the empirical IW-CIP penalty by constraining the choice of IW-CIP penalty (see Proposition 7). Then we bound the weight estimation error using the deviation from conditional invariance of $\widehat{\phi}_{\text{CIP}}$ (see Proposition 8). In Section 3.3.2, under the general anticausal model 5, we bound the deviation from conditional invariance of feature mappings via a form of conditional invariant penalty (see Proposition 9), and bound the target risk of $h^\star$ relative to the oracle classifier h_{oracle} (see Proposition 10).

3.3.1 UPPER BOUNDS ON THE EMPIRICAL IW-CIP PENALTY AND WEIGHT ESTIMATION ERROR

To refine the target risk upper bounds established in Theorem 6.B, we present two propositions. The first proposition shows that the empirical IW-CIP penalty diminishes to zero as the source sample size grows to infinity. The second proposition shows that the weight estimation error can be controlled through the deviation from conditional invariance Ψ—this result is intuitive because Eq. (15) for weight estimation relies on the conditional invariance of CICs. Recalling that $\phi^\star$ is the feature mapping for the optimal conditionally invariant target classifier given in Definition 3, we present the following proposition regarding the empirical IW-CIP penalty term $\widehat{\Lambda}_{\phi^\star}$ in Eq. (19).

Proposition 7 *Assume that the $\mathcal{G}$-divergence in Eq. (8) is used as the distributional distance in the empirical IW-CIP penalty. Then for any $\delta \in (0,1)$, with probability at least $1 - \delta$, the following bound holds,* [4]

$$\widehat{\Lambda}_{\phi^\star} \leq 2\lambda_{\text{IW-CIP}} \left(2 \max_{\substack{m \in \{1,...,M\} \\ y,y' \in \{1,...,L\}}} \mathfrak{R}_{n^{(m)}, \mathcal{P}^{(m)}_{\phi^\star(X)|Y=y}} \left(\mathcal{G}_{y'} \right) + \max_{m=1,2,...,M} \sqrt{\frac{\log\left(2LM/\delta\right)}{2n^{(m)}}} \right),$$

where $\mathcal{G}_y := \left\{ f(z) = \mathbf{1}_{g(z)=y}, g \in \mathcal{G} \right\}$.

This proposition is proved in Appendix B.3. The bound on $\widehat{\Lambda}_{\phi^\star}$ is given by the sum of a Rademacher complexity term and a finite-sample error term, both of which diminish to zero

---

4. The probability is with respect to the randomness of source samples $(X_k^{(m)}, Y_k^{(m)}) \overset{\text{i.i.d.}}{\sim} \mathcal{P}^{(m)}$ in $\mathcal{D}_3$ $(1 \leq m \leq M, 1 \leq k \leq n^{(m)})$; see Eq. (20).

as the sample size grows to infinity—this result is expected given that $\phi^\star$ is conditionally invariant across the population source distributions. While calculating the exact $\mathcal{G}$-divergence may be challenging in practice, this result offers a vanishing bound without additional assumptions on the underlying data generation model or on the class of feature mappings Φ. A more practical and simpler choice of the distributional distance is the squared mean distance, which penalizes the squared difference of conditional means of $\phi(X) \mid Y$ between source distributions. In this case, a refined bound of $\widehat{\Lambda}_{\phi^\star}$ can be obtained with additional assumptions about the data generation model and the class of feature mappings Φ. For more details on the calculation of this bound, see Appendix B.4.

Next, we show that when the true importance weights $w^{(m)}$ in Eq. (13) are estimated using the finite-sample CIP, the weight estimation error can be upper bounded via the deviation from conditional invariance of the feature mapping $\widehat{\phi}_{\mathrm{CIP}}$ learned through CIP. Let $\widehat{w}^{(m)}$ denote the estimated importance weight obtained by solving the linear system in Eq. (15), where the population quantitites are replaced by their empirical estimates and the finite-sample CIP is used. Then the following proposition provides the upper bound on the estimation error $\left\| w^{(m)} - \widehat{w}^{(m)} \right\|_2$.

Proposition 8 *Assume that the confusion matrix in the m-th source distribution $C_{\widehat{h}_{\mathrm{CIP}}}^{(m)}$, given in Eq. (16), is invertible with conditional number κ_m. Then, for any $\delta \in (0,1)$, with probability at least $1 - \delta$, the error of importance weights estimation is bounded by[5]*

$$\left\| w^{(m)} - \widehat{w}^{(m)} \right\|_2 \leq 2\kappa_m \left(\sqrt{L}\Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}} + \sqrt{\frac{L\log(4L/\delta)}{2n^{(\mathfrak{T})}}} + \sqrt{\frac{3\log(4L/\delta)}{n^{(m)}}} \left\| w^{(m)} \right\|_2 \right),$$

as long as $n^{(m)} \geq 12\kappa_m^2 \log(4L/\delta)$.

The proof of this proposition is given in Appendix B.5. Proposition 8 reveals that the finite-sample estimation error of importance weights is determined by the condition number of the confusion matrix of $\widehat{h}_{\mathrm{CIP}}$, the deviation from conditional invariance of $\widehat{\phi}_{\mathrm{CIP}}$, and sample error terms decaying roughly on the order of $1/\sqrt{n^{(m)}}$ or $1/\sqrt{n^{(\mathfrak{T})}}$. The condition number κ_m reflects the performance of the CIP classifier—if $\widehat{h}_{\mathrm{CIP}}$ achieves perfect classification accuracy on the m-th source distribution, the condition number κ_m takes the value of $1/\min_{y=1,2,\ldots,L} \mathbb{P}\left\{ Y^{(m)} = y \right\}$; however, if $\widehat{h}_{\mathrm{CIP}}$ performs poorly on the m-th source distribution, the condition number κ_m can be large, which can lead to inaccurate estimation of importance weights. The bound also introduces additional deviation from invariance term $\Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}}$, similar to $\Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{IW\text{-}CIP}}}$ that appeared in Theorem 6.B.

Proposition 8 also implies why iterating the two-step procedure of IW-CIP may not be beneficial. The estimation error of importance weights depends on the condition number κ_m of the matrix $C_{\widehat{h}_{\mathrm{CIP}}}^{(m)}$ and the deviation from conditional invariance $\Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}}$. If we obtain IW-CIP in the first iteration and use it to update the importance weights, then it is uncertain whether we would see further gains in both κ_m and $\Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}}$. For κ_m, the accuracy of IW-CIP would be lower than that of CIP on the "source" domain, which can lead to a deterioration

---

5. The probability is with respect to the randomness of $(X_k^{(m)}, Y_k^{(m)}) \overset{\text{i.i.d.}}{\sim} \mathcal{P}^{(m)}$ and $(X_k^{(\mathfrak{T})}, Y_k^{(\mathfrak{T})}) \overset{\text{i.i.d.}}{\sim} \mathcal{P}^{(\mathfrak{T})}$ in $\mathcal{D}_2$; see Eq. (20).

in κ_m; for $\Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}}$, it is not feasible to assume that IW-CIP improves conditional invariance compared to CIP, given that both are subject to the same penalty during training.

3.3.2 Deviation from conditional invariance and target risk bound on the optimal conditionally invariant classifier

With Proposition 7 and 8 now established, it remains to control the deviation from conditional invariance for both $\widehat{\phi}_{\mathrm{CIP}}$ and $\widehat{\phi}_{\mathrm{IW\text{-}CIP}}$ and the target risk of the optimal conditionally invariant target classifier $h^\star$ in Theorem 6.B. In this subsection, we quantify both of these terms under additional assumptions about the data generation model. By quantifying these terms, we gain a comprehensive understanding of the upper bounds presented in Theorem 6.B.

To facilitate our analysis, we focus on the general anticausal model as defined in Definition 5 and introduce the following assumption regarding the type of perturbations on the mechanism functions.

Assumption 1 *Suppose that source and target data are generated under the general anti-causal model in Definition 5. Assume that $f^{(m)}, f^{(\mathfrak{T})}$ are perturbed linearly as follows. For each $y \in \{1, \ldots, L\}$, there exist an orthogonal matrix $P_y \in \mathbb{R}^{p \times d_y}$, $0 \le d_y \le p$, and vectors $v_y^{(m)}, v_y^{(\mathfrak{T})} \in \mathbb{R}^{d_y}$, $m = 1, \ldots, M$ with $v_y^{(1)} = 0$, such that*

$$
\begin{aligned}
f^{(m)}(y) &= f^{(1)}(y) + P_y v_y^{(m)}, \ \text{for all } m \in \{1, 2, \ldots, M\}, \ \text{and} \\
f^{(\mathfrak{T})}(y) &= f^{(1)}(y) + P_y v_y^{(\mathfrak{T})}.
\end{aligned}
\tag{22}
$$

In addition, the noise terms follow a normal distribution $\epsilon^{(m)}, \epsilon^{(\mathfrak{T})} \overset{\text{i.i.d.}}{\sim} \mathcal{P}_\epsilon = \mathcal{N}(0, \Sigma)$.

Note that the perturbation matrix P_y is dependent only on the label y, and remains fixed across source and target data generation processes, whereas the vectors $v_y^{(m)}, v_y^{(\mathfrak{T})} \in \mathbb{R}^{d_y}$ are allowed to vary with both the source index m and the labels y. The assumption states that, conditional on the labels, the perturbation in each environment lies within the low-dimensional space spanned by the columns of P_y. Consequently, for a classifier to generalize to the target data, it is important that the classifier only utilizes the covariates that remain invariant, i.e., those which are orthogonal to the columns of P_y.

Before we state our result, we introduce several population terms related to a feature mapping ϕ. Let $\Delta_\phi^{(m)}(y)$ denote the difference of expected mean of $\phi(X)$ conditional on $Y = y$ between the first and the m-th source distributions, and let $\Sigma_\phi^{(m)}(y)$ denote the conditional covariance matrix of ϕ under the m-th source distribution, i.e.,

$$
\begin{aligned}
\Delta_\phi^{(m)}(y) &:= \mathbb{E}\left[\phi(X^{(m)}) \mid Y^{(m)} = y\right] - \mathbb{E}\left[\phi(X^{(1)}) \mid Y^{(1)} = y\right], \ \text{and} \\
\Sigma_\phi^{(m)}(y) &:= \mathrm{Var}\left[\phi(X^{(m)}) \mid Y^{(m)} = y\right].
\end{aligned}
\tag{23}
$$

Assuming that $\Sigma_\phi^{(m)}(y)$ is invertible, we further define

$$
\Pi_\phi(y) := \frac{1}{M-1} \sum_{m=2}^{M} \Delta_\phi^{(m)}(y)^\top \Sigma_\phi^{(m)}(y)^{-1} \Delta_\phi^{(m)}(y).
\tag{24}
$$

$\Pi_\phi(y)$ evaluates the differences of conditional means in the direction of the eigenvector of the covariance matrix. With these notions in hand, we can now establish the following deterministic bound for $\Psi_{\mathcal{G},\phi}$.

Proposition 9 *Suppose that source and target data are generated under Assumption 1. Let $\Phi = \{\phi(x) = Ax + b, A \in \mathbb{R}^{q \times p}, b \in \mathbb{R}^q\}$ be the linear class of feature mappings from $\mathbb{R}^p$ to $\mathbb{R}^q$ ($q \le p$), and let $\mathcal{G}$ be any hypothesis class mapping $\mathbb{R}^q$ to $\mathcal{Y}$. If for each $y \in \{1, 2, \ldots, L\}$, there exists $\zeta_y > 0$ such that for all $m \in \{1, 2, \cdots, M\}$,*

$$\left(v_y^{(\mathfrak{T})} - v_y^{(m)}\right)\left(v_y^{(\mathfrak{T})} - v_y^{(m)}\right)^\top \preceq \frac{\zeta_y}{M-1}\sum_{m=2}^{M} v_y^{(m)}(v_y^{(m)})^\top, \tag{25}$$

then for any $\phi \in \Phi$ such that $A\Sigma A^\top$ is non-singular, its deviation from conditional invariance (21) satisfies

$$\Psi_{\mathcal{G},\phi}^2 \le 2 \max_{y=1,2,\ldots,L} \left\{\zeta_y \Pi_\phi(y)\right\}.$$

The proof of this proposition is given in Appendix B.6. The proof proceeds by connecting the $\mathcal{G}$-divergence with the total variation distance and applying Pinsker's inequality and data processing inequality. Proposition 9 shows that under the general anticausal model with a specific form of linear perturbations, the deviation from conditional invariance $\Psi_{\mathcal{G},\phi}$ is governed by ζ_y and $\Pi_\phi(y)$. The term ζ_y captures perturbations in the underlying target data generation model, which are beyond our control. Therefore, to obtain good conditionally invariant representations, it is important to make the term $\Pi_\phi(y)$ small. Because $\Pi_\phi(y)$ measures the discrepancy in conditional means across source distributions, under Assumption 1, the squared mean distance can be utilized as a penalty in both CIP and IW-CIP.

Concretely, in Proposition 9, we consider a setting where $q = 1$, i.e., A is a row vector, and assume that the covariance matrix of the noise terms is $\Sigma = \sigma^2\mathbf{I}_p$ for some $\sigma^2 > 0$, in the SCM under Assumption 1. Then each term in $\Pi_\phi(y)$ corresponds to

$$\left\|\Delta_\phi^{(m)}(y)\right\|_2^2/(\sigma^2 \cdot \|A\|_2^2) = \left(\Delta_\phi^{(m)}(y)\right)^2/(\sigma^2 \cdot \|A\|_2^2).$$

If we assume $\|A\|_2 = 1$, a reasonable assumption given that the scale of classifier $g \in \mathcal{G}$ can be adjusted accordingly, then $\Pi_\phi(y)$ is the squared difference in conditional means between source environments, which is precisely what CIP penalizes when it uses mean distance. In cases where the CIP algorithm uses a different distributional distance, such as the MMD, $\Pi_\phi(y)$ will implicitly decreases as the differences in conditional distributions between source environments reduce.

Condition (25) is necessary for the validity of Proposition 9. In practice, verifying the correctness of this condition may be challenging. However, we can show that the condition is satisfied with high probability when the perturbations follow a Gaussian distribution and a sufficient number of source distributions are present. Specifically, in Lemma 17, we establish that approximately $M = \mathcal{O}\left(\max_y\{d_y\}\right)$ many source domains are required to ensure the validity of condition (25) with high probability (See Appendix B.7 for more details on the

precise statement of Lemma 17 and its proof). This indicates that the number of source domains needs to be at least linear with respect to the dimension of perturbations.

Applying Proposition 7, 8 and 9 to Theorem 6.B, we know that IW-CIP has a guaranteed target risk compared to the optimal conditionally invariant classifier $h^\star$. The last goal of this subsection is to connect $h^\star$ with the oracle target classifier h_{oracle} defined in Eq. (3). Again, we consider the general anticausal model under Assumption 1, but in a simpler binary classification setting given as follows.

Assumption 2 *Suppose that source and target data are generated under Assumption 1 with binary labels $\mathcal{Y} = \{1, 2\}$ and $p_1^{(\mathfrak{T})} = p_2^{(\mathfrak{T})} = 1/2$. Assume that $\Sigma = \sigma^2 \mathbb{I}_p, P_1 = P_2 = \left(\mathbb{I}_d, \mathbf{0}_{d \times (p-d)}\right)^\top$, and $f^{(1)}(1) = -f^{(1)}(2) = \xi \cdot \sigma \mathbb{1}_p$ for some $\xi > 0$, where $d_1 = d_2 = d$. Assume further that the source perturbations $\{v_\ell^{(m)}\}_{1 \le \ell \le 2, 1 \le m \le M}$ span the whole space of $\mathbb{R}^d$ and the target perturbations follow $v_1^{(\mathfrak{T})}, v_2^{(\mathfrak{T})} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \tau^2 \cdot \sigma^2 \mathbb{I}_d)$ for some $\tau > 0$.*

The assumption illustrates a specific binary classification problem in domain adaptation, where only the first d coordinates of the covariates are perturbed, with the remaining $p - d$ coordinates remaining invariant conditioned on the label. It also assumes that we have observed a sufficient number of perturbations in source domains, which span the entire space of the first d dimensions, while perturbations in the target domains are allowed to vary according to a normal distribution. Under this data generation model, we prove that the risk of $h^\star$ is close to that of h_{oracle} when the dimension of CICs is substantial.

Proposition 10 *Consider the domain adaptation problem under Assumption 2. Let the hypothesis class of classifiers $\mathcal{H}$ be $\mathcal{G} \circ \Phi$, where $\mathcal{G} = \{g : \mathbb{R} \to \{1, 2\}, g(x) = 1 \cdot \mathbf{1}_{x \le 0} + 2 \cdot \mathbf{1}_{x > 0}\}$ consists of one fixed function, and $\Phi = \{\phi : \mathbb{R}^p \to \mathbb{R}, \phi(x) = \beta^\top x + \beta_0, \|\beta\|_2 = 1, \beta \in \mathbb{R}^p, \beta_0 \in \mathbb{R}\}$ consists of linear feature mappings. There exists a constant $c_{\xi, \tau} > 0$ such that for any $\delta \in (0, 1)$, the risk difference between h_{oracle} and $h^\star$ is bounded by*

$$\mathcal{R}^{(\mathfrak{T})}(h^\star) - \mathcal{R}^{(\mathfrak{T})}(h_{\text{oracle}}) \le c_{\xi, \tau} \left(\sqrt{d} + \sqrt{\log(1/\delta)}\right) \exp\left(-\frac{\xi^2(p - d)}{8}\right), \qquad (26)$$

with probability at least $1 - \delta$.[6]

The proof of this proposition is given in Appendix B.8. As the dimension of covariates p goes to infinity, we can see that the risk difference between $h^\star$ and h_{oracle} converges to zero at an exponential rate, provided that the dimension of perturbations d is smaller than αp for some constant $\alpha < 1$. This result shares similarities with those obtained in a regression setting in (Chen and Bühlmann, 2021, Corollary 6), where the authors demonstrate a polynomial decay rate for the risk difference.

Remark 11 *We have established Propositions 7, 8, 9, 10 to control terms appeared in Theorem 6.A, 6.B. These propositions respectively bound the empirical IW-CIP penalty term, error of importance weights estimation, the deviation from conditional invariance,*

---

6. The probability is with respect to the randomness of target perturbations $v_1^{(\mathfrak{T})}, v_2^{(\mathfrak{T})} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \tau^2 \sigma^2 \mathbb{I}_d)$.

and the risk of the optimal conditionally invariant classifier. As a result, Theorem 6.B now guarantees that finite-sample IW-CIP has a target risk close to that of the oracle classifier h_{oracle}.

4. Enhancing DA with CICs: risk detection and improved DIP

While the previous section demonstrates that IW-CIP achieves target predictive performance close to that of the optimal conditionally invariant classifier, classifiers based on CICs can often be conservative, and their performance may not always be optimal. This conservativeness arises because finding invariant features from multiple source domains requires discarding some features that could be useful for target prediction. This can be seen in Proposition 10 where if the dimension of perturbations d is large, and thus $M = \mathcal{O}(d)$ is large, there is a gap between the oracle target risk and the risk of conditionally optimal invariant classifier. Practitioners may seek alternative DA algorithms, such as DIP, which directly find invariant features between a single (or a few) source domains and the target domain. However, in the absence of target labels, these algorithms can fail completely without any indication of failure. Furthermore, it is crucial to improve these algorithms when only unlabeled target data is available.

In this section, we present two additional roles of CICs to effectively address these challenges. First, we investigate how CICs can be used to detect large target risks for other DA algorithms (second role). Second, we examine how CICs enhance the reliability of DIP by addressing its label-flipping issue (third role). The empirical study for the second role is presented in Section 5.1.2 and Section 5.2.2, while the third rold is demonstrated throughout Section 5. For the purpose of this section, we assume the absence of label shift, meaning that the label distributions are invariant across both the source and target domains. If label shift is present, the procedure outlined in Section 3 can be employed to correct it by applying CIP and adjusting for the importance weights.

In practice, assessing the success or failure of a DA algorithm is challenging due to the unavailability of target labels. While some research has been devoted in establishing target risk guarantees of DIP in the limited settings, it is still considered a "risky" algorithm. DIP seeks invariant feature representations across source and target domains, enforcing only the marginal distribution of these feature representations to be invariant across these domains. However, having the marginal distribution be invariant is not sufficient to ensure conditional invariance. Consequently, DIP may only learn representations that maintain marginal invariance across source and target domains but fail to be conditionally invariant given the labels. Because DIP merely minimizes the source risk, those representations may entirely flip the prediction of labels when applied to the target data (Wu et al., 2019; Zhao et al., 2019; Wu et al., 2020); see Figure 1 (a)(b) for the illustrative examples. Furthermore, without target labels, it is difficult to detect this potential label flipping of DIP.

The label-flipping issue in DIP raises concerns about its reliability when applied blindly without additional validation. In the absence of target labels, previous works often rely on assumptions about the data generation process in order for DIP-type of algorithms to achieve low target risk, e.g. linear structural equation models (Chen and Bühlmann, 2021).
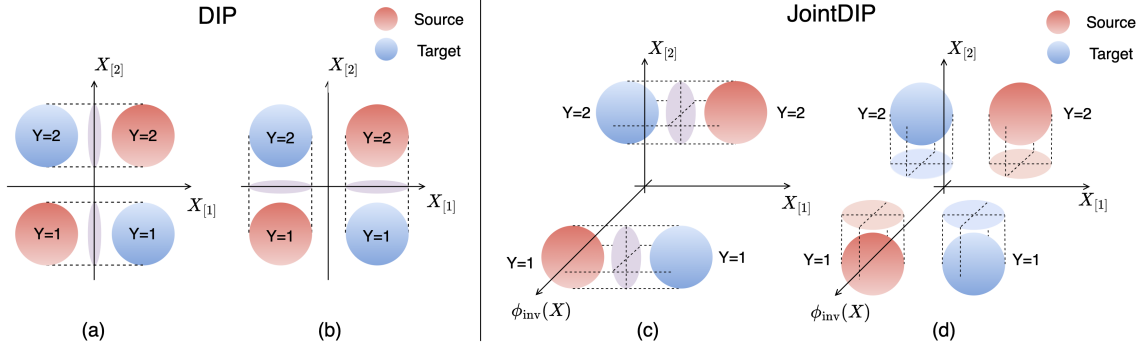
Figure 1: A binary classification example illustrating the difference between DIP and Joint-DIP. (a) DIP correctly matches the source and target covariates by projecting onto the feature $X_{[2]}$, which can generalize to the target distribution. (b) DIP matches the source and target covariates by projecting onto the label-flipping feature $X_{[1]}$, which cannot generalize to the target distribution. (c) JointDIP finds the correct feature $X_{[2]}$ by jointly matching the covariates with a conditional invariant component (CIC) $\phi_{\text{inv}}(X)$. (d) JointDIP discards the label-flipping feature $X_{[1]}$ because the joint distribution of $(X_{[1]}, \phi_{\text{inv}}(X))$ cannot be matched across source and target distributions.

In this paper, given the availability of multiple source domains, we propose the use of CICs to enhance the reliability of DIP algorithms.

Suppose that we have obtained the conditionally invariant feature mapping ϕ_{inv}, and the classifier h_{inv} built upon CICs denoted by

$$h_{\text{inv}} = g_{\text{inv}} \circ \phi_{\text{inv}}. \tag{27}$$

As seen in the previous sections, both ϕ_{inv} and h_{inv} can be approximately obtained by solving CIP—in this case, while finite-sample feature mapping $\widehat{\phi}_{\text{CIP}}$ may not be perfectly conditionally invariant, we expect that they exhibit a small deviation from conditional invariance under the appropriate assumptions, as implied by Proposition 9.[7] Therefore, we assume the exact conditional invariance of ϕ_{inv} for simplicity throughout this section. By leveraging ϕ_{inv} and h_{inv}, we demonstrate that CICs can guide other DA algorithms in two significant ways:

1. Large risk detection: when the classifier h_{inv} built on CICs has low target risk, it can be used to lower-bound the risk of other DA algorithms, even in the absence of target labels. This enables the identification of label-filpping issues in algorithms like DIP.

2. Joint matching: we propose the JointDIP algorithm which uses CICs $\phi_{\text{inv}}(X)$ to learn invariant features between source and target covariates. JointDIP, as a new DA

---

7. Because labeled data is available for multiple source domains, one can verify the conditional invariance of finite-sample CIP by comparing the conditional distributions across these source distributions. If one is willing to assume that CICs exist across source and target, then conditional invariance also holds for the target domain.

algorithm, enhances the reliability of DIP by addressing the label-flipping issue often encountered in DIP when the CICs are generic enough.

For the remainder of the section, when a single source is considered, we assume without loss of generality that the first source domain is used.

4.1 Detect failed DA algorithms using CICs

We present the second role of CICs in DA, specifically, in detecting whether another DA classifier has large target risk. Given the conditionally invarinat classifier h_{inv} as defined in Eq. (27), we prove the following theorem which controls the difference in source and target risks for any classifier.

Theorem 12 *Let h_{inv} be a conditionally invariant classifier and assume that there is no label shift between source and target distributions. For any classifier h, its risk difference in source and target is controlled as follows,* [8]

$$\left| \left[\mathcal{R}^{(\mathfrak{T})}(h) - \mathcal{R}^{(1)}(h) \right] - \left[\mathbb{P}\left\{ h(X^{(\mathfrak{T})}) \neq h_{\mathrm{inv}}(X^{(\mathfrak{T})}) \right\} - \mathbb{P}\left\{ h(X^{(1)}) \neq h_{\mathrm{inv}}(X^{(1)}) \right\} \right] \right| \\ \leq 2\mathcal{R}^{(1)}(h_{\mathrm{inv}}). \quad (28)$$

This result is proved in Appendix C.1. It shows that a large discrepancy between source and target risks of any classifier h can be detected by examining the source risk of h_{inv} and the alignments between the predictions of h and h_{inv} across source and target, which are always available. While Theorem 12 is stated in terms of the first source domain, the result is applicable to any source domain and extends to multiple source domains via averaging. In practice, we may take $\widehat{h}_{\mathrm{CIP}}$ as a proxy for h_{inv} and choose the source domain which gives the tightest bound—specifically, the one that exhibits small discrepancy in prediction alignments between h and h_{inv} across source and target domains, and that shows high accuracy for CIP or IW-CIP algorithms.

By rearranging terms, an empirical version of Eq. (28) can be derived to establish a lower bound for the target risk

$$\mathcal{R}^{(\mathfrak{T})}(h) \geq \widehat{\mathcal{R}}^{(1)}(h) - 2\widehat{\mathcal{R}}^{(1)}(\widehat{h}_{\mathrm{CIP}}) \\ + \left(\frac{1}{n^{(\mathfrak{T})}} \sum_{k=1}^{n^{(\mathfrak{T})}} \mathbf{1}_{h(X_k^{(\mathfrak{T})}) \neq \widehat{h}_{\mathrm{CIP}}(X_k^{(\mathfrak{T})})} - \frac{1}{n^{(1)}} \sum_{k=1}^{n^{(1)}} \mathbf{1}_{h(X_k^{(1)}) \neq \widehat{h}_{\mathrm{CIP}}(X_k^{(1)})} \right). \quad (29)$$

This lower bound on the target risk can be directly translated to an upper bound on accuracy, which can serve as a certificate to detect the failure of any DA classifier h. In Section 5.1.2 and Section 5.2.2, we demonstrate this with numerical examples using CIP to detect the failure of DIP, where we test on both synthetic data and the MNIST dataset.

---

8. In practice, it is difficult to find h_{inv} that is conditionally invariant across source and target distributions, for instance, if we approximate it via CIP, i.e., $h_{\mathrm{inv}} = \widehat{h}_{\mathrm{CIP}}$. In this case, the upper bound has an additional term $L\Psi_{\mathcal{G},\phi_{\mathrm{inv}}}$; see the proof for a generalized version of the theorem.

We observe that the accuracy upper bound derived from Eq. (29) to detect the failures of DIP becomes more accurate when CIP has a higher predictive accuracy on the source domain. If CIP does not perform well across the entire input space, by restricting to a subset of the space where CIP performs well (e.g. region far from decision boundary of a CIP classifier), we can obtain similar results which we will discuss in the subsequent subsection.

4.1.1 DETECT LARGE TARGET RISK BY RESTRICTING TO A SUBSET

While Theorem 12 provides a method for identifying when DA algorithms fail, the effectiveness of this approach may decrease if the invariant classifier h_{inv}, such as CIP, performs poorly in the source domain. In this case, it becomes difficult to detect the failure of DA algorithms across the entire region. However, if a subset of the region can be found where h_{inv} achieves higher accuracy, we can still establish a more precise lower bound on the performance of DA algorithms within this restricted subset.

In particular, the result of Theorem 12 can be extended in a straightforward manner to the case where we condition on the event $X^{(1)} \in \mathcal{A}$ for any subset $\mathcal{A} \subseteq \mathcal{X}$. We write $\mathcal{R}_{\mathcal{A}}^{(1)}(h)$ to denote the source risk conditioned on $\mathcal{A}$, i.e.,

$$\mathcal{R}_{\mathcal{A}}^{(1)}(h) := \mathbb{P}\left\{ h(X^{(1)}) \neq Y^{(1)} \,\Big|\, X^{(1)} \in \mathcal{A} \right\},$$

and similarly for the target risk $\mathcal{R}_{\mathcal{A}}^{(\mathfrak{T})}(h)$. Analogous to Theorem 12, we can obtain the following bound on the risk difference conditioned on set $\mathcal{A}$.

Corollary 13 *Let h_{inv} be a conditionally invariant classifier and assume that there is no label shift between source and target distributions. For any classifier h and any subset $\mathcal{A} \subseteq \mathcal{X}$, we have*

$$\left| \mathcal{R}_{\mathcal{A}}^{(\mathfrak{T})}(h) - \mathcal{R}_{\mathcal{A}}^{(1)}(h) \right| \leq 2\mathcal{R}_{\mathcal{A}}^{(1)}(h_{\text{inv}})$$
$$+ \left| \mathbb{P}_{\mathcal{A}}\left\{ h(X^{(1)}) \neq h_{\text{inv}}(X^{(1)}) \right\} - \mathbb{P}_{\mathcal{A}}\left\{ h(X^{(\mathfrak{T})}) \neq h_{\text{inv}}(X^{(\mathfrak{T})}) \right\} \right|, \quad (30)$$

where for a random vector X, we write

$$\mathbb{P}_{\mathcal{A}}\left\{ h(X) \neq h_{\text{inv}}(X) \right\} := \mathbb{P}\left\{ h(X) \neq h_{\text{inv}}(X) \mid X \in \mathcal{A} \right\}.$$

From Eq. (30), we can obtain a target risk lower bound for h, conditioned on set $\mathcal{A}$, analgous to Eq. (29). If one can identify a region $\mathcal{A} \subseteq \mathcal{X}$ where the classifier h_{inv} has near perfect source accuracy, due to its conditional invariance, the output of h_{inv} can serve as a good proxy of the unobserved target labels. One example of such a choice is to choose $\mathcal{A}$ as the region where h_{inv} gives a high predicted probability. Indeed, our experiments in Section 5.1.2 and Section 5.2.2 compare the tightness of the bounds across the entire space and on a subset $\mathcal{A}$ where h_{inv} exhibits a high predicted probability. We find that Corollary 13 provides a more accurate estimation of the target risk because h_{inv} has higher accuracy when constrained to $\mathcal{A}$. It is important to note that while this approach does not provide insights on performance outside $\mathcal{A}$, it indicates that if certain DA algorithms fail within this subset, they may not be reliable to use overall. Conversely, if they perform well in

$\mathcal{A}$, further investigation into these algorithms guided by domain knowledge is justified to evaluate their applicability to the broader region.

Corollary 13 shows that a desired property for h to have a low target risk is to align its output with the output of h_{inv} on $\mathcal{A}$. We formalize this idea which is a direct consequence of Corollary 13.

Corollary 14 *Let h_{inv} be a conditionally invariant classifier and assume that there is no label shift between source and target distributions. Suppose that $h \in \mathcal{H}$ satisfies*

$$h(x) = h_{\mathrm{inv}}(x) \ for \ all \ x \in \mathcal{A}. \tag{31}$$

Then the target risk of h on $\mathcal{A}$ is bounded by

$$\mathcal{R}_{\mathcal{A}}^{(\mathfrak{T})}(h) \leq 3\mathcal{R}_{\mathcal{A}}^{(1)}(h_{\mathrm{inv}}).$$

In particular, if h_{inv} has a near perfect accuracy on $\mathcal{A}$ under the source distribution, i.e., $\mathcal{R}_{\mathcal{A}}^{(1)}(h_{\mathrm{inv}}) \leq C$ for a small $C \geq 0$, then $\mathcal{R}_{\mathcal{A}}^{(\mathfrak{T})}(h) \leq 3C$.

Corollary 14 guarantees the low target population risk of the classifier on the region where the invariant classifier h_{inv} is confident about its prediction and can serve as a good proxy for the target labels Y. This does *not* necessarily imply that the representation learned via Eq. (31) can generalize to target data outside the region $\mathcal{A}$. Nevertheless, if domain experts all recognize the importance of the region $\mathcal{A}$, Eq. (31) becomes a natural requirement for assessing the quality of any DA classifier.

4.2 JointDIP by matching DIP features jointly with CICs

In this subsection, we demonstrate the third role of CICs in enhancing DIP. According to Theorem 12, if the predictions of h and h_{inv} align well across source and target, the target risk of h won't be too far from its source risk. It turns out that we can enforce this alignment by incorporating the CICs into the DIP matching penalty. This leads to our new algorithm, joint domain invariant projection (JointDIP).

Population JointDIP The population JointDIP minimizes the source risk while matching the joint distributions of ϕ and ϕ_{inv} in the representation space across source and target:[9]

$$\begin{aligned} h_{\text{j-DIP}} &= g_{\text{j-DIP}} \circ \phi_{\text{j-DIP}}, \\ g_{\text{j-DIP}}, \phi_{\text{j-DIP}} &= \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg \min} \quad \mathcal{R}^{(1)}(g \circ \phi) \\ &\text{subject to} \quad \mathfrak{D}\left(\mathcal{P}_{(\phi,\phi_{\mathrm{inv}})(X)}^{(1)}, \mathcal{P}_{(\phi,\phi_{\mathrm{inv}})(X)}^{(\mathfrak{T})}\right) = 0. \end{aligned} \tag{32}$$

Unlike DIP which only matches the marginal distribution of $\phi(X)$, JointDIP takes advantage of CICs to extract invariant feature representations. Intuitively, if ϕ_{inv} is highly correlated with the labels Y, it can effectively serve as a surrogate for the unobserved target Y.

---

9. $(\phi, \phi_{\mathrm{inv}})(X)$ denotes the vector concatenated by $\phi(X)$ and $\phi_{\mathrm{inv}}(X)$.

Thus, matching the joint distribution of feature representations $(\phi(X), \phi_{\text{inv}}(X))$ closely approximates matching the joint distribution of $(\phi(X), Y)$. This helps ensure that ϕ does not inadvertently learn features that flip the labels. Note that if label shift is present, it is not hard to extend the current formulation to an importance-weighted JointDIP (IW-JointDIP) by applying CIP to correct label shift prior to applying JointDIP.

Finite-sample JointDIP   The finite sample formulation of JointDIP enforces the joint invariance of the representations via a regularization term,

$$
\begin{aligned}
\widehat{h}_{\text{j-DIP}} &= \widehat{g}_{\text{j-DIP}} \circ \widehat{\phi}_{\text{j-DIP}}, \\
\widehat{g}_{\text{j-DIP}}, \widehat{\phi}_{\text{j-DIP}} &= \underset{g \in \mathcal{G}, \phi \in \Phi}{\arg \min} \quad \widehat{\mathcal{R}}^{(1)}(g \circ \phi) + \lambda_{\text{j-DIP}} \cdot \mathfrak{D}\left(\widehat{\mathcal{P}}^{(1)}_{(\phi, \phi_{\text{inv}})(X)}, \widehat{\mathcal{P}}^{(\mathfrak{T})}_{(\phi, \phi_{\text{inv}})(X)}\right),
\end{aligned} \tag{33}
$$

where $\lambda_{\text{j-DIP}} > 0$ is a regularization parameter that controls the strength of the joint matching penalty.

To illustrate the advantage of JointDIP over the ordinary DIP, consider a binary classification example under the general anticausal model Definition 5 where the data is generated as in Figure 1 (a)(b), similar to the example in Johansson et al. (2019). Suppose that $\Phi = \{\phi_1(x) = x_{[1]}, \phi_2(x) = x_{[2]}\}$ and $\mathcal{G} = \{g_a(z) = \mathbf{1}_{z>a} + 1, a \in \mathbb{R}\}$. Since DIP only matches the marginal distributions in the representation space, it could choose either ϕ_1 or ϕ_2 as the feature mappings to perfectly match the marginal distributions of the features and achieve zero loss under the source distribution. However, choosing ϕ_1 leads to zero accuracy under the target distribution, as $\phi_1(X) = X_{[1]}$ is the label-flipping feature. Only $\phi_2(X) = X_{[2]}$ is the conditionally invariant feature that can generalize to the target distribution. Now if we have access to CICs through the conditionally invariant feature mapping ϕ_{inv}, and we match these jointly with DIP, we would only get the correct feature $\phi_2(X) = X_{[2]}$, as illustrated in Figure 1 (c)(d). JointDIP would never select $\phi_1(X) = X_{[1]}$ because the joint distribution of $(X_{[1]}, \phi_{\text{inv}}(X))$ is different between the source and target distributions.

4.2.1 THEORETICAL COMPARISON OF CIP, DIP, AND JOINTDIP UNDER GENERAL ANTICAUSAL MODEL

To quantitatively compare the target risk of DA classifiers, we focus on data generated from the general anticausal model defined in Definition 5. Additionally, we introduce Assumption 3 where marginal distribution of Y is uniform under source and target distributions, as follows.

Assumption 3 *Suppose that data is generated according to the general anticausal model defined in Definition 5. Further, assume that the label distribution is uniform under source and target distributions, i.e.,$\forall m \in \{1, 2, \ldots, M\}$ and $\forall y \in \{1, 2, \ldots, L\}$, we have $p_y^{(m)} = p_y^{(\mathfrak{T})} = 1/L$.*

With the above assumptions on the data generation process in place, the following theorem compares the target risk of population CIP, DIP, and JointDIP.

Theorem 15 *Suppose that source and target data are generated under Assumption 3. Let $\Phi = \{\phi(x) = Ax + b, A \in \mathbb{R}^{q \times p}, b \in \mathbb{R}^q\}$ be the class of linear feature mapping from $\mathbb{R}^p$ to $\mathbb{R}^q$, which is used in the optimization of population DIP (11) and population JointDIP (32). Assume that both algorithms match feature distributions exactly, i.e.,*

$$\mathcal{P}^{(1)}_{\phi_{\mathrm{DIP}}(X)} = \mathcal{P}^{(\mathfrak{T})}_{\phi_{\mathrm{DIP}}(X)} \ \text{and} \ \mathcal{P}^{(1)}_{(\phi_{\text{j-DIP}},\phi_{\mathrm{inv}})(X)} = \mathcal{P}^{(\mathfrak{T})}_{(\phi_{\text{j-DIP}},\phi_{\mathrm{inv}})(X)}.$$

Then the following statements hold:

(a) *There exist distributions $\mathcal{P}^{(1)}, \mathcal{P}^{(\mathfrak{T})}$ such that the feature mapping of population DIP, ϕ_{DIP}, flips the labels after matching the marginal distributions in the representation space, i.e.,*

$$\mathcal{P}^{(1)}_{\phi_{\mathrm{DIP}}(X)|Y=y} = \mathcal{P}^{(\mathfrak{T})}_{\phi_{\mathrm{DIP}}(X)|Y=\pi(y)}, \quad \forall y \in \{1, \dots, L\}, \tag{34}$$

for some permutation $\pi \neq \mathbf{I}$ over $\mathcal{Y}$.

(b) *Suppose $\phi_{\mathrm{inv}} : \mathbb{R}^p \to \mathbb{R}^r$ is a linear conditionally invariant feature mapping such that*

$$\mathbb{E}\left[\phi_{\mathrm{inv}}(X^{(1)}) \mid Y^{(1)} = i\right] \neq \mathbb{E}\left[\phi_{\mathrm{inv}}(X^{(1)}) \mid Y^{(1)} = j\right], \quad \forall i \neq j \in \mathcal{Y}. \tag{35}$$

Then the feature mapping of population JointDIP, $\phi_{\text{j-DIP}}$, is conditionally invariant across $\mathcal{P}^{(1)}$ and $\mathcal{P}^{(\mathfrak{T})}$. If additionally $r \leq q$, then the target risk of JointDIP is no greater than that of the optimal classifier built on $\phi^0_{\mathrm{inv}} := (\phi_{\mathrm{inv}}, \mathbf{0}_{q-r})$, i.e.,

$$\mathcal{R}^{(\mathfrak{T})}(h_{\text{j-DIP}}) \leq \min_{g \in \mathcal{G}} \mathcal{R}^{(\mathfrak{T})}(g \circ \phi^0_{\mathrm{inv}}),$$

and when $\phi_{\mathrm{inv}} = \phi^\star$, we have $\mathcal{R}^{(\mathfrak{T})}(h_{\text{j-DIP}}) \leq \mathcal{R}^{(\mathfrak{T})}(h^\star)$.

(c) *Suppose $\phi_{\mathrm{inv}} : \mathbb{R}^p \to \mathbb{R}^r$ is a conditionally invariant feature mapping such that the matrix*

$$C_{\phi_{\mathrm{inv}}}(a) = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ m_1^1(a) & m_2^1(a) & \cdots & m_L^1(a) \\ \vdots & \vdots & \ddots & \vdots \\ m_1^{L-1}(a) & m_2^{L-1}(a) & \cdots & m_L^{L-1}(a) \end{pmatrix} \in \mathbb{R}^{L \times L}, \tag{36}$$

is full rank for some vector $a \in \mathbb{R}^r$, where $m_j^\ell(a) = \mathbb{E}\left[\left(a^\top \phi_{\mathrm{inv}}(X^{(1)})\right)^\ell \mid Y^{(1)} = j\right]$. Then the feature mapping of population JointDIP, $\phi_{\text{j-DIP}}$, is conditionally invariant across $\mathcal{P}^{(1)}$ and $\mathcal{P}^{(\mathfrak{T})}$.

The proof of this result is given in Appendix C.2. The proofs for Theorem 15(a) and Theorem 15(b) rely on the matching property of the two mixing distributions, whereas the proof for Theorem 15(c) analyzes distribution matching in the space of characteristic functions. Theorem 15(a) shows that while DIP uses the additional target covariates information, the corresponding representations can fail to satisfy conditional invariance across source and target distributions; and in fact, if the features correspond to the label-flipping features, as illustrated in Figure 1, it can potentially hurt the target prediction performance.

By contrast, Theorem 15(b) shows that when ϕ_{inv} is a linear conditionally invariant feature mapping, the features obtained by JointDIP are conditionally invariant across $\mathcal{P}^{(1)}$ and $\mathcal{P}^{(\mathfrak{T})}$, therefore avoiding the label-flipping issue of DIP. The condition (35) requires that conditional means of ϕ_{inv} are different for any pair of labels i and j. This condition is a reasonable expectation for any ϕ_{inv} exhibiting a good prediction performance as otherwise there is no way for classifiers built on ϕ_{inv} to distinguish between labels i and j. See Appendix A.2 for an example of the general anticausal model that satisfies the condition (35). Additionally, Theorem 15(b) assures that JointDIP cannot be worse than the optimal classifier built upon ϕ_{inv}. This result is expected because CICs tend to be conservative. For instance, CICs identified via CIP are forced to be conditionally invariant across M many source distributions (and ideally the target distribution), which can potentially eliminate features that are useful for predicting target labels. JointDIP addresses this issue by seeking conditional invariant representations across a single source and target domains to construct a more effective classifier than one solely based on CICs.

Finally, for any conditionally invariant feature mapping ϕ_{inv} which may not necessarily be linear, Theorem 15(c) guarantees that the features obtained by JointDIP are conditionally invariant across $\mathcal{P}^{(1)}$ and $\mathcal{P}^{(\mathfrak{T})}$, as long as the matrix $C_{\phi_{\text{inv}}}(a)$ given in Eq. (36) is full rank. The matrix $C_{\phi_{\text{inv}}}(a)$ is full rank if the CICs under different labels are in a generic position. Comparing with the condition (35) in the linear case, this condition may be more difficult to verify generally as it requires computation of higher order moments. We provide concrete examples in Appendix A.3 where this condition is satisfied. In practice, we can either use $\widehat{\phi}_{\text{CIP}}$ as ϕ_{inv}, or refer to domain experts for suggestions on reasonable CICs.

5. Numerical experiments

In this section, we investigate the target performance of our proposed DA algorithms and compare them with existing methods.[10] In particular, we demonstrate through our experiments the effectiveness of the importance-weights correction in the presence of both covariate and label distribution shifts, the capability of detecting DIP's failure using estimated CICs, and the superior performance of JointDIP over DIP when label-flipping features are present. We consider DA classification tasks across five datasets: synthetic data generated from linear Structural Causal Models (SCMs), the MNIST data (LeCun, 1998) under rotation intervention, the CelebA data (Liu et al., 2015) under color intervention, the Camelyon17 data from WILDS (Koh et al., 2021; Sagawa et al., 2021), and the DomainNet data (Peng et al., 2019) transformed into a binary classification problem. Except for the Camelyon17 and DomainNet data, the domain shifts in other datasets are synthetically created.

We conduct a thorough comparative analysis where our methods are benchmarked against various DA algorithms. In addition to DIP, CIP, IW-CIP, and JointDIP which have been introduced in previous sections, we additionally explore several variants of these algorithms, including *IW-DIP* and *IW-JointDIP*. IW-DIP is the algorithm that applies CIP for importance weighting to correct label shift prior to DIP, while IW-JointDIP applies this importance weighting step before JointDIP. Moreover, for ERM, DIP, and JointDIP, we

---

10. All code to reproduce experiments is available at https://github.com/KeruWu/Roles_CICs.

consider both their single-source versions (e.g. DIP) and their multi-source versions (e.g. DIP-Pool). Contrary to the approach in Section 3.3, where source datasets are split for theoretical analysis, our empirical findings indicate superior performance when using the entire dataset, and thus we do not apply such splits in our experiments. In terms of distributional distances, both squared mean distance and Maximum Mean Discrepancy (MMD) are considered. Lastly, we compare these methods with existing well-known DA algorithms such as Invariant Risk Minimization (IRM) (Arjovsky et al., 2019), V-REx (Krueger et al., 2021), and groupDRO (Sagawa et al., 2020). A detailed description of each DA algorithm, models architectures, and the training setup are presented in Appendix D.

Certain DA algorithms, such as DIP or IW-DIP, rely on a single source domain to learn invariant feature representations and construct the final classifiers (see Appendix D). In this work, we do not focus on how to choose the best single source domain for these algorithms, but instead simply select the last source domain. By design, this single source domain is usually similar to the target domain in terms of the considered covariate shifts, such as mean shift (linear SCMs), rotation angle (MNIST), and color balance (CelebA). However, this domain may have label-flipping features compared to the target domain. We deliberately design these settings to demonstrate that, under such settings, DIP might induce label flipping, while JointDIP avoids this issue.

5.1 Linear structural causal models (SCMs)

We perform experiments on synthetic datasets generated according to linear SCMs under the general anticausal model defined in Definition 5. We first compare the performance of various DA methods, and then show how to use CICs to detect the failure of DIP without access to target labels.

5.1.1 LINEAR SCMs UNDER DIFFERENT INTERVENTIONS

We consider three different types of domain shifts: mean shift, label shift, and a shift to introduce label-flipping features. In all SCMs, we introduce the mean shift across domains. Depending on the presence or absence of label shift and the shift that introduces label-flipping features, we obtain four combinations and their corresponding SCMs. We use $M' = M + 1$ to denote the total number of source and target domains. The last source domain (the M-th domain) is always set as the single source domain for DA algorithms which utilize only one source domain (e.g. DIP). The last domain (the M'-th domain) serves as the target domain, and we generate 1000 samples per domain. Instead of specifying each $f^{(m)}(\cdot)$ and $f^{(\mathfrak{T})}(\cdot)$ in Definition 5, we provide explicit representations of the data generation model for each SCM. For simplicity, we only present the data generation model for source data, and the target data generation model can be obtained by replacing the superscript (m) with $(\mathfrak{T})$.

- SCM I: mean shift exists; no CICs; no label shift; no label-flipping features; $M' = 4$ and $p = 10$. The data generation model is

$$Y^{(m)} \sim \text{Bernoulli}(0.5),$$

$$X^{(m)} = 0.2Y \cdot \mathbb{1}_{10} + 0.25 \cdot \mathcal{N}(0, \mathbb{I}_{10}) + A^{(m)},$$

where $\mathbb{1}_{10} \in \mathbb{R}^{10}$ denotes a vector consisting of ones and $\mathbb{I}_{10} \in \mathbb{R}^{10 \times 10}$ denotes the identity matrix. For all source domains, the mean shift $A^{(m)}$, $1 \le m \le M$, is generated by $0.2 \cdot \mathcal{N}(0, \mathbb{I}_{10})$, while the target domain suffers a large intervention $A^{(\mathfrak{T})} \sim 2 \cdot \text{sign}(\mathcal{N}(0, \mathbb{I}_{10}))$. Note that we do not introduce any CICs in SCM I. This allows us to examine the most extreme case of mean shift where all coordinates of X are perturbed. However, in the following three SCMs (SCM II, III, and IV), we ensure the presence of CICs and expect that certain DA algorithms can exploit these CICs.

- SCM II: mean shift exists; CICs exist; label shift exists; no label-flipping features; $M' = 12$ and $p = 9$. The data generation model is

$$Y^{(m)} \sim \text{Bernoulli}(p^{(m)}),$$
$$X_{[1:6]}^{(m)} = 0.2(Y^{(m)} - 0.5) \cdot \mathbb{1}_6 + 0.25 \cdot \mathcal{N}(0, \mathbb{I}_6) + A^{(m)},$$
$$X_{[7:9]}^{(m)} = 0.2(Y^{(m)} - 0.5) \cdot \mathbb{1}_3 + 0.25 \cdot \mathcal{N}(0, \mathbb{I}_3),$$

where the label distribution $p^{(m)} = 0.5$, for $1 \le m \le M$, is balanced in source domains but perturbed in target domain with $p^{(\mathfrak{T})} = 0.1$. The mean shift only exists in the first six coordinates of X, where $A^{(m)}$, $1 \le m \le M$ is generated by $\mathcal{N}(0, \mathbb{I}_6)$ in the source domains, while the target domain suffers a large intervention $A^{(\mathfrak{T})} \sim 2 \cdot \text{sign}(\mathcal{N}(0, \mathbb{I}_6))$. The last three coordinates of X remain unperturbed, and they serve as CICs.

- SCM III: mean shift exists; CICs exist; no label shift; label-flipping features exist; $M' = 12$ and $p = 18$. The data generation model is

$$Y^{(m)} \sim \text{Bernoulli}(0.5),$$
$$X_{[1:6]}^{(m)} = 0.3(Y^{(m)} - 0.5) \cdot \mathbb{1}_6 + 0.4 \cdot \mathcal{N}(0, \mathbb{I}_6) + A^{(m)},$$
$$X_{[7:12]}^{(m)} = \begin{cases} 0.3(0.5 - Y^{(m)}) \cdot \mathbb{1}_6 + 0.1 \cdot \mathcal{N}(0, \mathbb{I}_6), & \text{m is odd,} \\ 0.3(Y^{(m)} - 0.5) \cdot \mathbb{1}_6 + 0.1 \cdot \mathcal{N}(0, \mathbb{I}_6), & \text{m is even,} \end{cases}$$
$$X_{[13:18]}^{(m)} = 0.3(Y^{(m)} - 0.5) \cdot \mathbb{1}_6 + 0.4 \cdot \mathcal{N}(0, \mathbb{I}_6),$$

The first six coordinates of X suffer mean shift, with $A^{(m)}$, $1 \le m \le M$ and $A^{(\mathfrak{T})} \sim \mathcal{N}(0, \mathbb{I}_d)$ across all domains. In particular, we make the last source domain and target domain share similar mean shift interventions: $A^{(11)} = a_0 + a_1$ and $A^{(12)} = a_0 + a_2$, where $a_0 \sim 0.8 \cdot \mathcal{N}(0, \mathbb{I}_6)$ and $a_1, a_2 \sim 0.6 \cdot \mathcal{N}(0, \mathbb{I}_6)$. This setting potentially enables DIP-based methods to better rely on the last source domain because of the similarity of $X_{[1:6]}$ between this domain and the target domain. However, the 7th to 12th coordinates of X are label-flipping features according to Definition 4: in the first six domains $X_{[7:12]}$ has positive correlation with Y, while the correlation is negative in the remaining six domains, including the target domain. There are no interventions on the last six coordinates of X, and they serve as CICs.

- SCM IV: mean shift exists; CICs exist; label shift exists; label-flipping features exist, $M' = 12$ and $p = 18$. The data generation model of $X \mid Y$ is the same as SCM III,

	SCM I		SCM II	
Mean shift	Y		Y	
CICs	N		Y	
Label shift	N		Y	
Label flip	N		N	
DA Algorithm	src_acc	tar_acc	src_acc	tar_acc
Tar	70.6±13.8	**89.0±0.7**	69.4±3.9	**92.9±1.0**
ERM	89.6±1.0	56.1±12.0	87.5±1.0	57.4±33.3
ERM-Pool	88.3±2.3	54.4±10.2	78.5±1.3	58.6±29.3
DIP	88.3±1.0	**87.6±1.5**	84.5±3.1	62.0±2.9
DIP-Pool	86.7±2.8	**86.4±2.2**	75.8±0.7	60.1±3.1
CIP	87.4±3.3	55.9±12.0	75.2±0.7	75.7±6.5
IW-ERM	54.8±11.6	52.4±10.4	59.3±9.6	54.1±37.7
IW-CIP	53.7±9.9	54.0±11.8	50.3±0.7	**90.4±0.8**
IW-DIP	56.2±12.8	54.3±11.7	71.1±10.8	**92.1±2.7**
JointDIP	87.1±1.3	**86.8±1.9**	82.8±2.7	70.6±6.2
IW-JointDIP	68.4±19.1	68.0±19.4	51.7±3.9	**90.0±1.3**
IRM	87.6±2.2	56.7±10.4	70.9±2.5	71.9±17.9
V-REx	87.3±2.5	55.6±11.5	77.9±1.2	62.9±25.7
groupDRO	88.3±2.3	54.4±10.0	77.8±1.1	64.2±25.7

Table 1: Source and target accuracy in SCM I and SCM II. For each setting of SCMs, DA algorithms are run on 10 different datasets, each generated using 10 different random seeds. Tar represents the oracle method where the model is trained on the labeled target data. The top four methods are highlighted in bold. The best method (excluding Tar) is colored in red.

but we additionally perturb the marginal distribution of Y across source and target distributions. Specifically, the label distribution $Y^{(m)} \sim$ Bernoulli(0.5) is balanced in source domains but perturbed in target domain with $Y^{(\mathfrak{T})} \sim$ Bernoulli(0.3).

We use a linear model in all SCM experiments to predict the labels; see Appendix D.2 for details. Table 1 and 2 compare the target performance of different DA algorithms. In SCM I where only mean shift exists and no CICs exist, DIP gives the best performance, with JointDIP showing comparable accuracy. ERM and other DA algorithms which aim to find invariance across all source domains (e.g. CIP, IRM, V-REx) fail to generalize to the target domain due to the lack of CICs and the substantial mean shift in the target domain. In SCM II where mean shift exists and label shift is added as another intervention, IW-DIP achieves the highest accuracy. Both IW-CIP and IW-JointDIP also achieve over 90% correct predictions. However, IW-ERM which directly applies importance weighting without using CICs completely fails, indicating that identifying conditionally invariant features before applying label correction is necessary in this scenario.

In SCM III where label-flipping features exists, we observe that DIP results in an accuracy lower than a random guess. On the contrary, JointDIP achieves the highest accuracy. To gain a deeper understanding of the classifiers obtained by each algorithm, in Figure 2 we illustrate the L^1 norm of the coefficients of three categories of coordinates: mean-shifted

	SCM III		SCM IV	
Mean shift	Y		Y	
CICs	Y		Y	
Label shift	N		Y	
Label flip	Y		Y	
DA Algorithm	src_acc	tar_acc	src_acc	tar_acc
Tar	70.1±2.9	**97.9±0.4**	69.8±3.0	**96.9±0.7**
ERM	97.9±0.3	58.9±5.0	98.0±0.3	58.7±10.7
ERM-Pool	83.3±0.8	75.3±7.7	83.3±0.7	78.9±6.8
DIP	93.5±3.4	34.5±14.9	94.4±2.7	35.3±14.6
DIP-Pool	84.1±0.5	82.0±1.1	84.4±0.5	82.3±3.8
CIP	82.2±0.4	**81.8±1.3**	82.2±0.4	82.1±1.2
IW-ERM	80.3±9.1	75.1±8.9	77.2±9.0	79.2±4.8
IW-CIP	82.5±0.4	81.2±4.1	81.0±0.9	**83.8±2.2**
IW-DIP	88.4±13.0	37.2±14.9	83.7±8.6	64.2±7.3
JointDIP	88.8±1.3	**85.4±2.1**	88.6±1.9	82.8±1.9
IW-JointDIP	87.8±1.5	**82.9±8.1**	84.1±3.4	**85.1±3.7**
IRM	80.1±1.8	80.2±1.8	84.2±0.6	83.7±3.3
V-REx	83.8±0.8	80.4±7.6	84.3±0.6	83.8±3.7
groupDRO	84.0±0.7	81.3±7.4	83.9±0.6	**84.3±3.2**

Table 2: Source and target accuracy in SCM III and SCM IV. For each setting of SCMs, DA algorithms are run on 10 different datasets, each generated using 10 different random seeds. Tar represents the oracle method where the model is trained on the labeled target data. The top four methods are highlighted in bold. The best method (excluding Tar) is colored in red.

$X_{[1:6]}$, label-flipping $X_{[7:12]}$, and CICs $X_{[13:18]}$. Three key observations can be made from the figure: First, DIP shows large coefficients on the label-flipping features $X_{[7:12]}$, providing insight for its suboptimal performance. Second, DA methods that seek invariant features across all source domains, such as CIP, IRM, V-REx, show small coefficients on both the mean-shifted features and label-flipping features, but large coefficients on the CICs. This result aligns with their fundamental objective of identifying invariant representations across source domains. Lastly, JointDIP demonstrates small coefficients on label-flipping features, but relatively larger coefficients on mean-shifted coordinates, confirming that JointDIP effectively discards label-flipping features $X_{[7:12]}$, retains the invariant features $X_{[13:18]}$ identified by CIP while exploiting the useful mean-shifted features $X_{[1:6]}$.

Lastly in SCM IV where mean shift, label shift, and label flipping features exist simultaneously, we find that IW-JointDIP outperforms other methods in the target accuracy. This can be attributed to its three-step approach—first finding CICs, then correcting label shift via CICs, and finally aligning potential features via JointDIP—which effectively addresses the combination of these three shifts.

Overall, our experiments across various SCM settings reveal that while traditional DA methods like CIP and DIP may struggle with certain types of shifts, JointDIP excels in handling
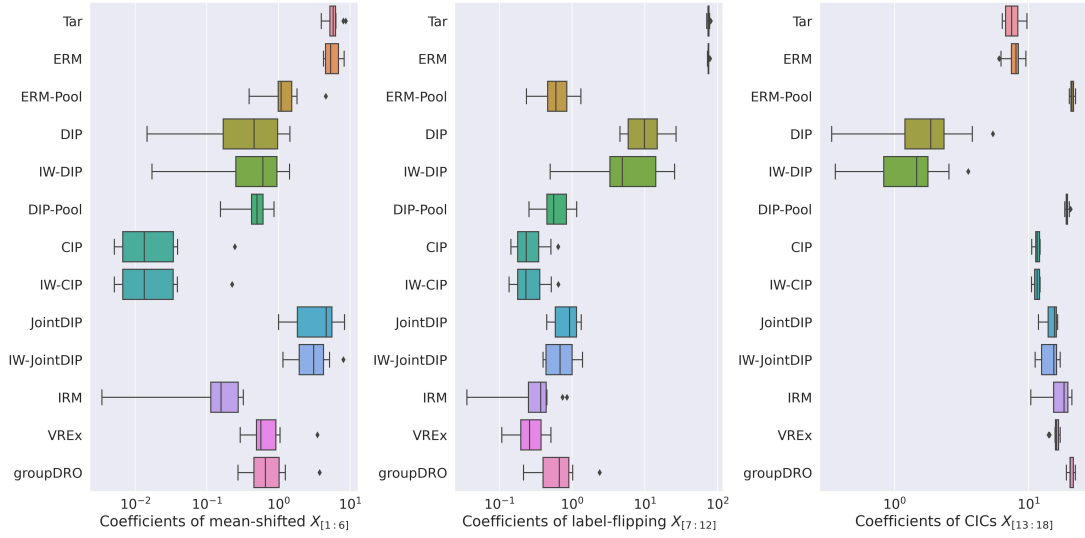
Figure 2: L^1 norm of coefficients in SCM III. Coefficients are grouped into three categories, depending on how corresponding coordinates are perturbed. Contrary to DIP, JointDIP does not have large coefficients on label-flipping features, demonstrating that JointDIP can avoid label-flipping issue observed in DIP.

label-flipping features, and importance-weighted variants of the DA methods perform well in scenarios with label shift.

5.1.2 Detecting failure of DIP in linear SCMs

Previously in Section 4.1, we discussed the second role of CICs to detect the failure of DA algorithms without requiring access to target labels. To demonstrate this second role in practice, here we apply Theorem 12 and Corollary 13 to DIP and JointDIP within the context of SCM III. We take the finite-sample CIP as ϕ_{inv}, and compare the accuracy upper bound derived from Theorem 12 against the actual accuracy of DIP and JointDIP as shown in Figure 3 (a)(b). We vary the DIP penalty parameter in DIP and the JointDIP penalty parameter in JointDIP (both represented by λ). The CIP penalty parameter utilized in CIP and JointDIP is set to the optimal value found via hyperparameter search. The figure illustrates that while the accuracy upper bound from Theorem 12 is valid, it exceeds the true accuracy by a wide margin, which is undesirable.[11] Consequently, it is difficult to directly apply Theorem 12 to test the failure of DIP in SCM III.

We then turn to apply Corollary 13 to compare the accuracy upper bound with the actual accuracy of DIP within region $\mathcal{A}$, where we define $\mathcal{A}$ as the region such that the CIP predicted probability exceeds a threshold q_α. This threshold q_α is defined as the $\alpha \times 100$-th

---

11. This issue might result from the relatively low accuracy of CIP (around 80%) in SCM III. In our subsequent experiments with MNIST, such issue does not appear because CIP has much higher accuracy (over 90%). See Figure 3 (c)(d) for the comparison.
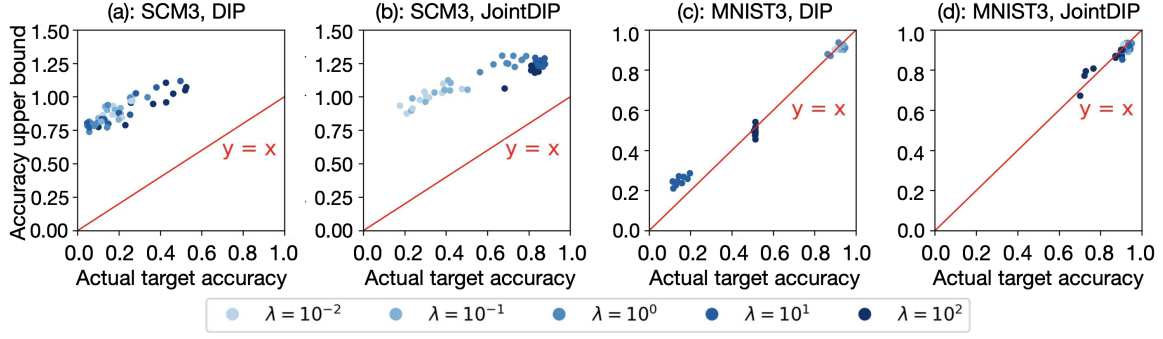
Figure 3: SCM III and MNIST III: target accuracy upper bound obtained via Theorem 12 vs. actual target accuracy. For SCM III, the accuracy of CIP is low, resulting in an upper bound that does not provide a tight estimation of the true target accuracy for both DIP and JointDIP. Conversely, in MNIST III, CIP achieves high accuracy and the accuracy upper bound precisely reflects the true target accuracy. The red line indicates $y = x$.
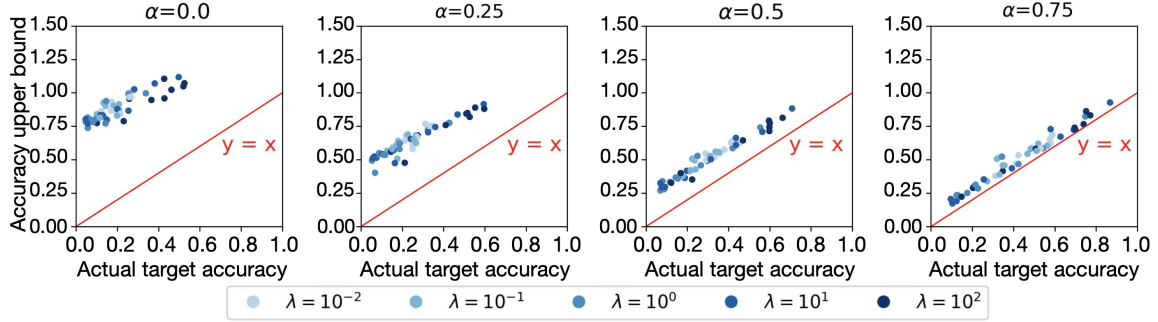

Figure 4: DIP in SCM III: target accuracy upper bound obtained via Corollary 13 vs. actual target accuracy in region $\mathcal{A}$. We define region $\mathcal{A}$ as $\mathcal{A} = Q_\alpha := \{X \in \mathcal{X} \mid \text{CIP predicted probability} \geq q_\alpha\}$, where q_α is the threshold such that $(1 - \alpha) \times 100\%$ of target covariates have a CIP predicted probability greater than q_α. The larger α is, the more confident CIP is about the prediction on the covariate samples in Q_α. The red line indicates $y = x$.

percentile of CIP predicted probability for the target covariates, i.e., $(1 - \alpha) \times 100\%$ of target covariates have a CIP predicted probability greater than q_α. As shown in Figure 4, by increasing α, we observe that our upper bound becomes increasingly precise within region $\mathcal{A}$. For example, DIP with certain values of λ yields an upper bound lower than 0.5, suggesting that DIP flips the label. Although a low accuracy within region $\mathcal{A}$ does not necessarily translate to low accuracy across the entire target domain, in practice we can refer to domain experts and ask them whether such suboptimal performance of DIP
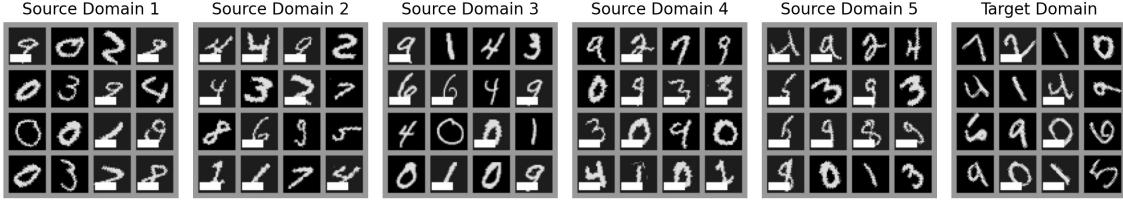
Figure 5: Sample images from MNIST III. In this dataset, two different types of interventions have been applied: a rotation shift and the inclusion of label-flipping features. For rotation shift, each image in the m-th domain is rotated clockwise by $(m \times 15 - 30)°$. As for the label-flipping features, a white bar of 6×16 pixels is added in the bottom-left corner of the images. The correlation between the label and the white bar patch changes across the domains, with positive correlation in the $1^{st}, 3^{rd}$, and 5th domains, and negative correlation in the $2^{nd}, 4^{th}$, and 6th (target) domains.

within region $\mathcal{A}$ is reasonable or not, allowing us to avoid the need to acquire and validate all target labels.

5.2 MNIST under rotation intervention

In this section, we consider binary classification of the MNIST dataset (LeCun, 1998) under rotation interventions, where digits 0-4 are categorized as label 0 and digits 5-9 are categorized as label 1. Similar to our experiments in SCMs, we first evaluate the prediction performance of various DA methods, then discuss how to detect potential failure of DIP without requiring access to target labels.

5.2.1 Rotated MNIST under different interventions

We create five source domains and one target domain (the 6th domain). The 5th domain is fixed as the single source domain for DA algorithms that rely on only one source domain. Each source domain consists of 20% of the images from the MNIST training set, and the target domain includes all images from the MNIST test set. We introduce three different types of interventions as follows:

- Rotation shift: Each image in the m-th domain is rotated clockwise by $(m \times 15 - 30)°$.

- Label shift: In the target domain, we remove 50% of the images labeled as 0.

- Label-flipping features: In the $1^{st}, 3^{rd}$, and 5th domains, 90% of the images with label 1 and 10% of the images with label 0 are patched by a white bar of 6×16 pixels in the bottom-left corner. This creates a correlation of 0.8 between the label and the patch. Conversely, in the $2^{nd}, 4^{th}$, and 6th domains, we add this white bar to 90% of the images with label 0 and 10% of the images with label 1. This creates a correlation of -0.8 between the label and the patch.

	MNIST I		MNIST II	
Rotation shift	Y		Y	
Label shift	N		Y	
Label flip	N		N	
DA Algorithm	src_acc	tar_acc	src_acc	tar_acc
Tar	71.2±0.9	**100.0±0.0**	69.3±1.5	**99.7±0.3**
ERM	99.9±0.1	96.9±0.4	99.9±0.2	**96.9±0.5**
ERM-Pool	99.5±0.4	94.6±0.8	99.5±0.3	94.6±1.3
DIP	100.0±0.1	**97.5±0.3**	99.9±0.2	96.7±0.5
DIP-Pool	99.6±0.2	95.7±0.4	99.6±0.2	94.6±1.5
CIP	99.6±0.3	94.9±0.9	99.6±0.3	94.9±2.1
IW-ERM	99.4±0.4	94.8±0.8	99.4±0.2	94.8±0.5
IW-CIP	99.7±0.2	94.8±0.8	99.2±0.4	94.9±1.2
IW-DIP	99.9±0.3	**97.3±0.4**	99.8±0.2	**97.4±0.2**
JointDIP	99.9±0.1	**97.3±0.3**	99.9±0.2	**96.9±0.6**
IW-JointDIP	99.6±0.6	97.2±0.5	99.5±0.2	96.5±1.3
IRM	99.5±0.5	94.3±1.4	99.6±0.1	94.3±1.4
V-REx	99.4±0.2	94.6±0.8	99.6±0.1	94.5±0.9
groupDRO	99.6±0.3	94.9±1.1	99.5±0.3	95.1±1.2

Table 3: Source and target accuracy in rotated MNIST I and II. For each setting of the rotated MNIST, DA algorithms are run on 10 different datasets, each generated using 10 different random seeds. Tar represents the oracle method where the model is trained on the labeled target data. The top four methods are highlighted in bold. The best method (excluding Tar) is colored in red.

Similar to the settings of SCMs, we consider four DA problems under different combinations of these interventions. The rotation shift intervention is applied to all MNIST experiments, and four different cases are constructed based on whether the label shift or label-flipping features exist: MNIST I includes rotation shift, but does not have label shift and label-flipping features; MNIST II includes rotation shift and label shift, but does not have label-flipping features; MNIST III includes rotation shift and label-flipping features, but does not have label shift; and MNIST IV includes all three interventions—rotation shift, label shift, and the presence of label-flipping feature. Image samples under MNIST III are displayed in Figure 5 for illustration.

We train a Convolutional Neural Network (CNN) model similar to LeNet5 (LeCun et al., 1998) for all MNIST experiments; see Appendix D.2 for more details. Tables 3 and 4 summarize the performance of various DA methods on MNIST. Similar to our findings in linear SCM experiments, we conclude that DIP effectively addresses rotation shift in the absence of label-flipping features (MNIST I), while label correction addresses label shift (MNIST II and IV), and JointDIP mitigates the issue of learning label-flipping features for DIP (MNIST III and IV). Consequently, DA methods which incorporate steps specifically designed to address each shift have the best performance or are among the top-performing methods.

	MNIST III		MNIST IV	
Rotation shift	Y		Y	
Label shift	N		Y	
Label flip	Y		Y	
DA Algorithm	src_acc	tar_acc	src_acc	tar_acc
Tar	67.4±1.1	**100.0±0.1**	66.2±1.9	**99.9±0.1**
ERM	100.0±0.1	87.8±1.9	100.0±0.1	86.8±1.8
ERM-Pool	99.4±0.5	88.7±2.0	99.6±0.2	89.1±2.5
DIP	100.0±0.0	91.0±1.9	100.0±0.0	90.1±1.6
DIP-Pool	99.4±0.5	**92.8±1.2**	99.4±0.4	91.2±2.1
CIP	99.4±0.4	89.1±1.7	99.1±0.9	90.0±1.3
IW-ERM	99.4±0.5	88.0±1.7	98.8±0.8	88.3±3.3
IW-CIP	99.6±0.2	89.8±1.7	99.4±0.4	90.1±1.0
IW-DIP	100.0±0.0	**92.6±1.1**	100.0±0.0	**90.9±1.5**
JointDIP	99.9±0.2	**93.5±1.2**	99.7±0.2	**91.5±1.1**
IW-JointDIP	99.9±0.2	**93.1±1.1**	99.6±0.5	**93.4±0.9**
IRM	99.3±0.7	88.4±1.9	99.1±0.3	89.3±1.4
V-REx	99.1±0.3	89.5±1.4	99.4±0.6	89.8±1.3
groupDRO	99.0±0.2	90.7±1.1	99.0±0.2	90.8±1.9

Table 4: Source and target accuracy in rotated MNIST III and IV. For each setting of the rotated MNIST, DA algorithms are run on 10 different datasets, each generated using 10 different random seeds. Tar represents the oracle method where the model is trained on the labeled target data. The top four methods are highlighted in bold. The best method (excluding Tar) is colored in red.

5.2.2 Detecting failure of DIP in rotated MNIST

Analogous to Section 5.1.2, we apply Theorem 12 and Corollary 13 to MNIST III to detect potential failures of DIP, demonstrating the second role of CICs. Figure 3 (c)(d) illustrate that the theoretical upper bound on target accuracy obtained via Theorem 12 matches closely with the actual target accuracy, and that large values of λ such as 10 and 100 may lead DIP to learn label-flipping features, while no such issues appear in JointDIP. Figure 6 presents the result of applying Corollary 13 with different choices of region $\mathcal{A} = Q_\alpha$ for $\alpha \in \{0, 0.25, 0.5, 0.75\}$. Similar to our findings in Figure 4, we observe that as CIP becomes more confident in region $\mathcal{A}$ (i.e., as α grows), the upper bound becomes more accurate. In Figure 7, we visualize images from the target domain sampled within the region Q_α for different values of α. As α increases, the handwritten digits become more clear and distinguishable. In practice where we suspect label flipping by DIP but cannot access or afford to obtain many target labels, we can investigate target samples where CIP and DIP disagree in the region Q_α with a large value of α, and refer to domain experts to evaluate whether these distinguishable images should be correctly classified by DIP. If DIP fails to do so, our procedure based on Corollary 13 can serve as a diagnostic tool to accurately estimate DIP's target performance and evaluate its validity in region $\mathcal{A}$, especially when Theorem 12 provides an uninformative upper bound.
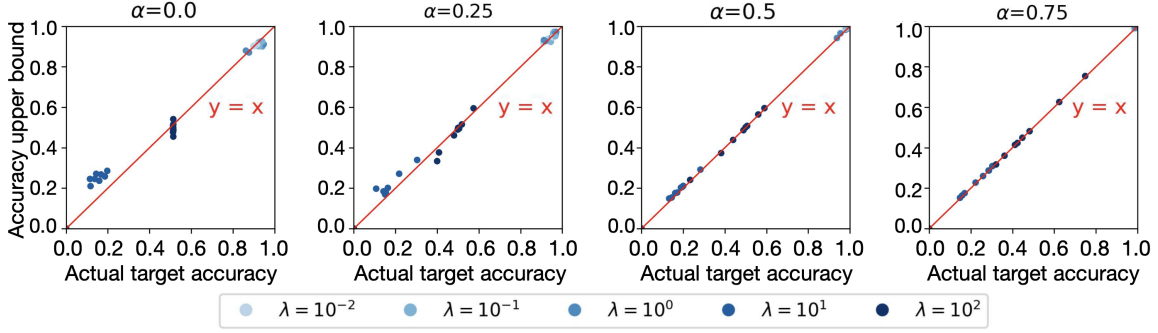
Figure 6: DIP in MNIST III: target accuracy upper bound obtained via Corollary 13 vs. actual target accuracy in region $\mathcal{A}$. We define region $\mathcal{A}$ as $\mathcal{A} = Q_\alpha := \{X \in \mathcal{X} \mid \text{CIP predicted probability} \geq q_\alpha\}$, where q_α is the threshold such that $(1 - \alpha) \times 100\%$ of target covariates have a CIP predicted probability greater than q_α. The larger α is, the more confident CIP is about the prediction on the covariate samples in Q_α. The red line indicates $y = x$.
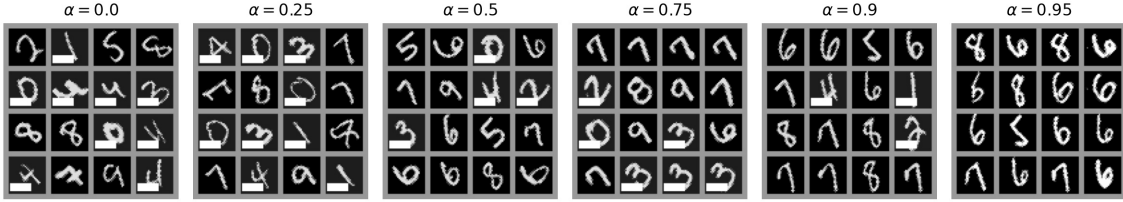

Figure 7: Sample images from MNIST III target domain within region Q_α where CIP disagrees with DIP. The images are shown for varying $\alpha \in \{0, 0.25, 0.5, 0.75, 0.9, 0.95\}$. As α increases, the selected handwritten digits become more distinguishable.

5.3 CelebA under color shift

The CelebA dataset introduced by (Liu et al., 2015) is a large-scale face image dataset with multiple face attributes. It includes over 200K celebrity images, each annotated with 40 binary attributes such as Gender, Eyeglasses, and Smiling. In our experiments, we take Smiling as the label and create six different settings for DA problems. The first three CelebA settings are designed to study the color shift together with label-flipping features, similar to the approach taken with SCM III and MNIST III. The remaining three CelebA settings are developed to examine color shift in conjunction with label shift, similar to the approach taken with SCM II and MNIST II. For each setting, we construct three source domains and one target domain, each consisting of 20K images randomly sampled from CelebA. The last source domain is used as the single source domain for DA algorithms that rely on a single source domain. To predict the labels, we utilize a CNN model architectures across all problem settings (see Appendix D.2 for more details).
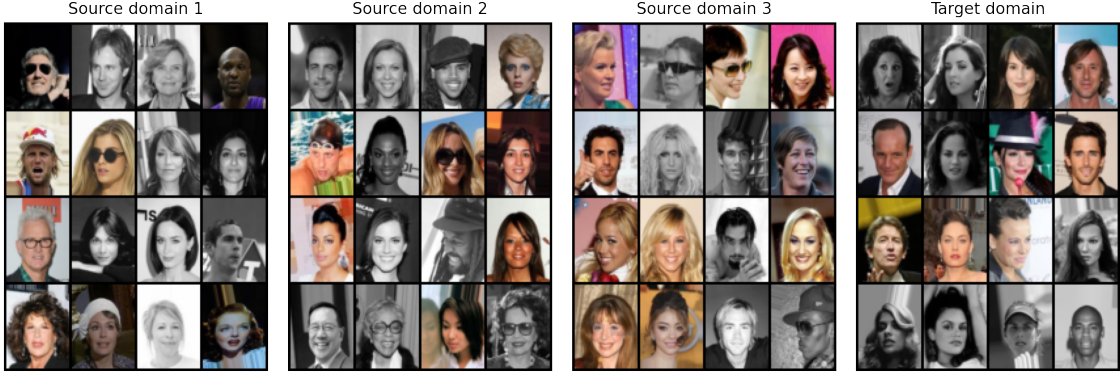
Figure 8: Sample images from CelebA III. In this dataset, we synthetically create a feature Color_Balance, which takes 1 for a full-color image and 0 for a black-and-white image. We also consider the original attribute Mouth_Slightly_Open, which takes 1 if the mouth in the image is slightly open, and 0 otherwise. The correlations between these two features and the label Smiling are varied across domains, as shown in Table 5.

5.3.1 Color shift with label-flipping features

The settings of CelebA I, CelebA II, and CelebA III are presented in Table 5. In these settings, we manipulate two features, Color_Balance and Mouth_Slightly_Open, to generate distribution shift across domains. The Color_Balance feature is a synthetic attribute, which takes a value of 1 for a full-color image and 0 for a black-and-white image. The Mouth_Slightly_Open feature is an original binary attribute annotated in the CelebA dataset, with the value 1 if the mouth in the image is slightly open, and 0 otherwise. We vary these two features so that they are conditionally independent given the label Smiling, and their correlations with this label match the values shown in Table 5. In all three settings, we align the Color_Balance feature of the last source domain and the target domain most closely, while making the correlation between Smiling and Mouth_Slightly_Open reversed. As a result, the Color_Balance feature can somewhat generalize to the target domain for label prediction, whereas the Mouth_Slightly_Open feature serves as a label-flipping feature between the last source domain and the target domain as defined in Definition 4. Figure 8 illustrates image examples from each domain in CelebA III. An ideal DA algorithm that utilizes the last source domain should be able to partially make use of the Color_Balance feature, but should avoid the Mouth_Slightly_Open feature to predict the labels.

Given that there is no label shift in these settings, we evaluate DA methods without the importance-weighted label shift correction (IW) step, and present the results in Table 6. We observe that in CelebA I, where the correlations between Smiling and both the Color_Balance and Mouth_Slightly_Open features vary significantly across domains, Joint-DIP achieves the best performance. In CelebA II and CelebA III where we reduce the variation in correlations between Smiling and one of the features, JointDIP continues to demonstrate comparable accuracy. GroupDRO also shows a good performance, likely due

	ρ(Smiling, Color_Balance)			ρ(Smiling, Mouth_Slightly_Open)		
Domain	CelebA I	CelebA II	CelebA III	CelebA I	CelebA II	CelebA III
Source Domain 1	-0.8	-0.8	0.2	-0.9	-0.5	-0.9
Source Domain 2	-0.6	-0.6	0.4	0.9	0.5	0.9
Source Domain 3	0.6	0.6	0.6	0.9	0.5	0.9
Target Domain	0.8	0.8	0.8	-0.9	-0.5	-0.9

Table 5: Settings of CelebA I, II, III under color shift with label-flipping features. Smiling is the label attribute for prediction. We vary the Color_Balance and the Mouth_Slightly_Open features across domains. The correlation between Color_Balance and Smiling remains relatively stable between the last source domain and the target domain, while Mouth_Slightly_Open serves as the label-flipping feature.

	CelebA I		CelebA II		CelebA III	
DA Algorithm	src_acc	tar_acc	src_acc	tar_acc	src_acc	tar_acc
ERM	97.0±0.2	77.7±1.2	94.4±0.4	**90.3±0.9**	97.0±0.2	77.7±1.2
ERM-Pool	93.3±0.3	77.2±1.4	92.8±0.4	83.6±1.0	93.1±0.2	89.6±0.4
DIP	96.2±0.4	**82.0±2.2**	93.1±0.3	**90.6±1.0**	96.2±0.4	82.0±2.2
DIP-Pool	92.3±0.5	80.4±1.3	90.9±0.4	85.5±0.6	91.6±0.3	**90.4±0.4**
CIP	89.1±0.3	76.6±1.4	92.7±0.6	83.7±1.4	93.0±0.4	**89.7±0.7**
JointDIP	93.0±0.8	**87.3±0.6**	93.1±0.3	**90.4±1.2**	91.4±1.1	88.7±1.0
IRM	90.5±0.7	76.3±3.2	92.6±0.5	83.6±1.4	90.3±0.9	88.8±1.3
V-REx	86.9±1.2	**83.3±1.2**	81.8±0.2	85.1±0.7	91.8±0.3	**91.1±0.8**
groupDRO	91.7±1.0	**84.3±1.0**	91.5±0.5	**86.4±1.5**	92.4±0.4	**91.6±0.5**

Table 6: Source and target accuracy in CelebA I, II, III with color shift and label-flipping features. For each setting of the CelebA, DA algorithms are run on 10 different datasets, each generated using 10 different random seeds. The top four methods are highlighted in bold. The best method is colored in red.

to the substantial shift across domains. Overall, the results show that JointDIP maintains strong and robust performance in the presence of label-flipping features.

5.3.2 COLOR SHIFT WITH LABEL SHIFT

The settings of CelebA IV, CelebA V, and CelebA VI are shown in Table 7. In these settings, we manipulate the Color_Balance feature and additionally introduce interventions on the distribution of Smiling to create label shift between the source and target domains. Specifically, the correlation between Smiling and the Color_Balance feature remains consistent with Table 5, but we no longer vary the Mouth_Slightly_Open feature, thereby eliminating label-flipping features between the source and target domains. Instead, we change the label distribution across domains: the distribution of Smiling is balanced (50% each) in all three source domains, but becomes unbalanced (either 75% and 25%, or 25% and 75%)

	ρ(Smiling, Color_Balance)			P(Smiling = 1)		
Domain	CelebA IV	CelebA V	CelebA VI	CelebA IV	CelebA V	CelebA VI
Src Domain 1	-0.8	-0.8	0.2	0.5	0.5	0.5
Src Domain 2	-0.6	-0.6	0.4	0.5	0.5	0.5
Src Domain 3	0.6	0.6	0.6	0.5	0.5	0.5
Tar Domain	0.8	0.8	0.8	0.75	0.25	0.75

Table 7: Settings of CelebA IV, V, VI under color shift and label shift. Smiling is the label attribute for prediction. We vary the Color_Balance feature and the Smiling label distribution across domains. The correlation between Smiling and the Color_Balance feature remains consistent with Table 5. The distribution of Smiling is balanced (50% each) in all three source domains, but becomes unbalanced (either 75% and 25%, or 25% and 75%) in the target domain.

	CelebA IV		CelebA V		CelebA VI	
DA Algorithm	src_acc	tar_acc	src_acc	tar_acc	src_acc	tar_acc
ERM	94.4±0.6	**94.1±0.5**	94.4±0.6	**93.9±0.9**	94.5±0.6	**94.2±0.5**
ERM-Pool	93.6±0.2	88.7±0.9	93.6±0.2	89.6±0.9	93.8±0.3	93.1±0.5
DIP	94.4±0.4	**93.5±1.2**	94.4±0.5	**94.4±0.9**	94.4±0.4	93.4±1.2
DIP-Pool	93.7±0.6	88.0±0.6	93.6±0.5	89.5±1.2	93.7±0.5	92.9±0.9
CIP	93.4±0.4	88.3±1.0	93.4±0.4	89.7±1.1	93.6±0.2	93.3±0.5
IW-ERM	92.2±0.9	90.1±1.0	93.2±0.4	90.9±0.8	92.1±0.7	**94.4±0.2**
IW-CIP	91.8±0.9	**90.3±0.5**	92.7±1.1	90.6±1.1	91.7±1.3	**94.3±0.5**
IW-DIP	93.0±0.7	**94.8±0.3**	93.5±0.8	**95.1±0.5**	92.4±1.4	**94.7±0.4**
IRM	93.8±0.3	88.7±0.9	93.5±0.4	89.6±0.9	93.7±0.2	93.1±0.6
V-REx	92.5±0.2	89.5±0.9	92.6±0.3	90.2±1.2	93.8±0.2	93.4±0.6
groupDRO	93.3±0.3	90.0±0.8	93.3±0.3	**91.1±1.0**	93.7±0.3	93.0±0.6

Table 8: Source and target accuracy in CelebA IV, V, VI under color shift and label shift. For each setting of the CelebA, DA algorithms are run on 10 different datasets, each generated using 10 different random seeds. The top four methods are highlighted in bold. The best method is colored in red.

in the target domain. We evaluate various DA methods, including those that incorporate the importance-weighted label shift correction (IW) step, as shown in Table 8. Given the absence of label-flipping features in these settings, we do not evaluate JointDIP. We observe that IW-DIP outperforms all other methods. Note that ERM also achieves a high accuracy and outperforms ERM-Pool, indicating that training on all source domains may not be beneficial when substantial distribution shifts, in particular the label shift, exist across domains. Overall, our experimental results highlight the importance of the label shift correction (IW) step in the presence of substantial label shift.

5.4 Camelyon17 from WILDS

In this subsection, we conduct numerical experiments on Camelyon17 to demonstrate the effectiveness of JointDIP on the real data. The Camelyon17 image dataset is one of the WILDS benchmarks (Koh et al., 2021; Sagawa et al., 2021) for domain adaptation, which comprises whole-slide images (WSIs) of breast cancer metastases in lymph node sections from five hospitals. The task involves predicting whether a 96×96 image patch extracted from a WSI contains any tumor pixels within its central 32×32 area. The labels are obtained through manual pixel-level annotations of each WSI. Among the five different hospitals (domains) from which the WSIs are collected, the first three hospitals are treated as source domains (with 302.4K patches) and the remaining two are used as validation domain (with 34.9K patches) and target domain (with 85K patches), respectively. In addition to these labeled examples, WILDS also provides additional unlabeled WSIs from the same five hospitals and patches extracted from them, including 600K unlabeled patches from the target domain. See (Koh et al., 2021; Sagawa et al., 2021) for a detailed description of the Camelyon17 dataset.

Although the 600K unlabeled target patches are collected from the same hospital as the 85K labeled target patches, there is no overlap in the WSIs where these patches are extracted from (Sagawa et al., 2021). Furthermore, the label distribution of unlabeled target patches is heavily skewed towards negative, while labeled target patches are sampled in a class-balanced manner (Sagawa et al., 2021). Consequently, this could create distribution shifts between the "unlabeled target data" and the "labeled target data" in the Camelyon17 dataset. In particular, this can lead DA algorithms, which utilize target covariate information (e.g. DIP or its variants), to identify different latent features depending on the source of target image patches used by these algorithms. In the WILDS leaderboard, methods that leverage data from the target domain utilize these 600K unlabeled patches, rather than treating the 85K labeled target patches as if they were unlabeled. However, if a distribution shift exists between the "unlabeled target data" and the "labeled target data", there is no guarantee that our proposed DA algorithms will improve target performance. Therefore, our experiments examine two approaches to utilize target data for DA methods: (1) using the 600K unlabeled target patches, and (2) leveraging the 85K labeled target patches without label access.

We use the standard DenseNet-121 model architecture (Huang et al., 2017) and follow the training protocol described in Koh et al. (2021); Sagawa et al. (2021). CIP, DIP-Pool, and JointDIP-Pool are implemented, where CIP only uses labeled source data, and DIP-Pool and JointDIP-Pool leverage target covariates (image patches) from the target domain. We find that the pooled versions of DIP and JointDIP increase reliability of the models by utilizing more samples from multiple source domains, compared to their single source versions. Therefore, we choose to implement the pooled version of DIP and JointDIP in our experiments. Similarly, DANN (Ganin et al., 2016) and CORAL (Sun and Saenko, 2016), which were initially developed for the single source version, have been extended in (Sagawa et al., 2021) to take advantage of multiple source domains. Following the standard submission guidelines in WILDS (Koh et al., 2021; Sagawa et al., 2021), we do not use any data augmentation in CIP, but allow DIP-Pool and JointDIP-Pool to utilize the same color augmentation as described in Koh et al. (2021); Sagawa et al. (2021).

DA Algorithm	Target covariates	val accuracy	test accuracy[12]
IRM*		86.2±1.4	64.2±8.1
groupDRO*		85.5±2.2	68.4±7.3
ERM-Pool*	-	84.9±3.1	70.3±6.4
CIP		87.0±1.2	71.1±12.5
Fish*		83.9±1.2	**74.7±7.1**
DANN*		86.9±2.2	68.4±9.2
CORAL*	600K unlabeled target patches	90.4±0.9	77.9±6.6
DIP-Pool		91.6±0.8	81.2±8.2
JointDIP-Pool		91.5±0.6	**82.7±5.2**
DIP-Pool	85K labeled target patches	91.0±0.5	88.7±5.7
JointDIP-Pool	(target labels are not used)	91.7±0.6	**91.9±3.1**

Table 9: Accuracy of DA algorithms on Camelyon17 dataset. The first five DA methods—IRM, GroupDRO, ERM-Pool, CIP, and Fish—only use labeled data from multiple source domains, whereas DANN, CORAL, DIP-Pool, and JointDIP-Pool additionally use target image patches from the target domain, but they do not have access to the target labels. The algorithms are run over 10 trials. Results for algorithms marked with * are obtained from the WILDS benchmark.

Hyperparameters are chosen via a grid search based on the validation accuracy, where we allow a larger range of grids compared to previous experiments (see Appendix D.3 for further details).

The results are presented in Table 9, where we compare CIP with other DA algorithms that enforce different invariance across all source domains, such as IRM (Arjovsky et al., 2019) and Fish (Shi et al., 2022); and we also compare DIP-Pool and JointDIP-Pool with other domain-invariant algorithms using different distributional distances, like DANN (Ganin et al., 2016) and CORAL (Sun and Saenko, 2016). We observe that CIP has a similar performance compared to ERM-Pool, albeit with a slightly larger variance in test accuracy. When covariates of 600K unlabeled target patches are utilized in the DIP matching penalty, DIP-Pool results in an accuracy exceeding 80%. Remarkably, JointDIP-Pool further improves this accuracy by 1.5% while substantially reducing the variance at the same time. In scenarios where covariates of 85K labeled target patches are used in the DIP matching penalty, the accuracy of both methods increases significantly, with JointDIP-Pool improving DIP-Pool by over 3%. This result indicates that jointly matching the covariates with CICs between source and target can help DIP-based algorithms to identify the appropriate features to match across domains. In addition, when comparing the accuracy of two different approaches of using target covariates in the DIP matching penalty, our results demonstrate that distribution shifts exist between the unlabeled and labeled target data in Camelyon17, and leveraging the true target covariates from the 85K labeled target patches proves more effective than using the target covariates from the 600K unlabeled target patches.

---

12. The test accuracy is evaluated on 85K labeled target image patches.

5.5 DomainNet with binary classification

In this subsection, we conduct numerical experiments on the DomainNet dataset. The DomainNet dataset, introduced by (Peng et al., 2019), is a large-scale, multi-domain dataset which contains approximately 600k images collected from 6 different domains: Clipart (illustrative images, such as cartoons), Infograph (infographics containing both graphical and textual information), Painting (artistic depictions in the form of paintings), Quickdraw (simple, hand-drawn images from the Quick Draw game dataset), Real (photographic, real-world images), and Sketch (hand-drawn sketches). Each image is labeled with one of 345 object categories, which are further grouped into 24 divisions (see Tables 10, 11, and 12 of Peng et al. (2019)). In our experiment, to make it easy to interpret the target performance, we transformed the original multi-class prediction problem into a binary classification task. Specifically, for each domain, we selected images with categories corresponding to the "Furniture" and "Mammal" groups, consisting of 35 and 25 categories, respectively, and labeled them as 0 and 1. Merging these categories makes the classification problem simpler and increases the number of images available for each label, so that target performance variation due to small sample size can be reduced. To ensure equal label proportions within each domain, we randomly sub-sampled images to achieve a balanced dataset with 50% labeled as 0 and 50% labeled as 1. This eliminates label shift across all domains and the only source of distribution shift arises from variations in visual styles, contexts, and object representations in the images. As a result, each domain contains 6,844, 6,952, 9,522, 25,000, 30,780, and 10,212 images, respectively. Each domain is used as a target domain in turn, with the remaining domains serving as source domains. This setup results in 6 different cases for evaluating the performance of DA methods.

Our DA algorithms are built on top of a ResNet-101 model architecture (He et al., 2016), pre-trained on the ImageNet dataset. Since there is no label shift in this task, we implement DA methods without importance weight estimation-based label shift correction. Similar to our approach in Section 5.4, we find that the pooled versions of DIP and JointDIP provide more reliable results by leveraging more samples from multiple source domains compared to their single-source versions. Therefore, we implement CIP, DIP-Pool, and JointDIP-Pool using MMD as the distributional distance, along with other DA methods such as IRM, V-REx, and groupDRO. Hyperparameters are chosen via a grid search, as detailed in Appendix D.3.

The results are presented in Table 10. Overall, we observe that JointDIP-Pool consistently achieves the best or comparable performance to the top method across all 6 target domain settings. For certain target domains such as Painting and Quickdraw, it significantly outperforms all other DA methods. This suggests that JointDIP-Pool is more effective at learning invariant features across source and target domains, enabling it to handle diverse visual characteristics when classifying "Furniture" and "Mammal." DIP-Pool also achieves high accuracy, with notably better target performance compared to other DA methods when Infograph is used as the target domain. However, it exhibits high standard deviation in this setting, potentially indicating that DIP-Pool may have learned label-flipping features during some trials. This aligns with observations in Peng et al. (2019), where generalization to Infograph was found to be more challenging, resulting in lower accuracies compared to other settings. V-REx also performs well in certain cases, likely due to its conservative ap-

DA Algorithm	Clipart tar_acc	Infograph tar_acc	Painting tar_acc	Quickdraw tar_acc	Real tar_acc	Sketch tar_acc
ERM-Pool	**89.9±0.7**	64.7±1.8	**81.1±1.3**	69.6±4.2	90.4±2.8	**87.3±0.5**
DIP-Pool	89.7±0.7	**75.9±9.6**	80.4±2.4	**71.1±4.5**	90.9±1.2	**87.4±0.6**
CIP	88.9±1.7	**64.9±1.0**	79.5±1.3	**70.5±2.7**	91.0±1.0	86.7±1.2
JointDIP-Pool	**90.0±0.9**	**66.0±1.6**	**86.5±5.2**	**74.1±2.5**	**91.8±0.9**	**87.2±0.9**
IRM	**90.0±0.8**	64.1±2.4	**81.0±1.7**	69.0±4.9	**91.8±1.0**	**87.2±0.7**
V-REx	**90.0±0.5**	**65.7±1.3**	79.8±2.3	**69.9±4.0**	91.5±0.5	86.8±0.9
groupDRO	86.8±0.4	64.2±2.2	77.9±0.8	68.0±6.1	**91.6±0.9**	84.1±0.8

Table 10: Target accuracies in DomainNet for predicting the labels "Furniture" and "Mammal." Each column corresponds to a specific target domain: Clipart, Infograph, Painting, Quickdraw, Real, and Sketch, where the remaining domains are used as source domains. The algorithms are run over 10 trials with different random seeds. The top four methods are highlighted in bold. The best method is colored in red.

proach by minimizing worst-case performance in the presence of substantial domain shifts. Finally, for challenging target domains such as Infograph and Quickdraw, where average target accuracies are generally lower, DIP-Pool, CIP, JointDIP-Pool, and V-REx demonstrate consistently strong performance, highlighting that these methods can better transfer knowledge between source and target domains in these difficult scenarios.

6. Discussion

In this paper, we highlight three prominent roles of CICs in DA. First, we propose the IW-CIP algorithm based on CICs, which can solve DA problems beyond simple covariate shift and label shift with theoretical guarantees. Specifically, we establish an explicit target risk bound for IW-CIP and further provide fine-grained analysis for each term in the bound. Second, we demonstrate the advantage of using CICs to measure the performance of other DA algorithms. For example, a conditionally invariant classifier built upon CICs can be applied as an approximate proxy for target labels to refute DA algorithms with poor target performance. Finally, to solve the label-flipping issue of DIP, we introduce the JointDIP algorithm which jointly matches the distribution of DIP features and CICs. Both our theoretical results and numerical experiments support the benefits of using CICs in DA.

While CICs across multiple source domains can be beneficial in DA, in practice it may not always be advantageous to use all available source domains for identifying CICs. Particularly, if not all source domains are related to the target domain, or if there aren't any features invariant across all source domains, it would be more efficient to select a subset of source domains for learning CICs. However, choosing the source domains that are related to the target domain or share common features is difficult without access to target labels. In such settings, one feasible approach is to learn invariant features across possible combinations of source domains and refer to domain experts to investigate those features,

which would help determine the selection of source domains. We leave the development of a framework for best source domain selection for future research.

There are other promising directions to pursue following our work. While we provide bounds in Section 4.1 to detect failure of DA algorithms, these bounds may not be tight and potentially can be improved based on other structural assumptions on the data generation model. Moreover, although in this paper we mainly focus on combing CICs with other DA algorithms, it is possible that other forms of invariance can also be applied to improve DA algorithms. For instance, can we use features obtained by IRM (Arjovsky et al., 2019), V-REx (Krueger et al., 2021), or other domain generalization algorithms to aid and enhance existing DA methods such as DIP with target risk guarantees? Finally, this paper mainly considers the unsupervised domain adaptation scenario where target labels are unavailable, however, datasets such as Camelyon17 from WILDS (Koh et al., 2021) may have a small portion of target labels available. Developing DA algorithms with target risk guarantees in such settings is therefore of practical interest.

Acknowledgments

Appendix A. Illustrative examples of general anticausal model

In this section, we provide several examples for the general anticausal model and demonstrate how the conditions in Theorem 15(b) and Theorem 15(c) are satisfied under these models.

A.1 Example with CICs and label-flipping features

We present an example of the general anticausal model, as defined in Definition 5, which incorporates both CICs and label-flipping features.

Example 1 Let $Y \in \mathcal{Y} = \{1, 2\}$ be a binary random variable and balanced in both domains, i.e., $p_1^{(m)} = p_2^{(m)} = p_1^{(\mathfrak{T})} = p_2^{(\mathfrak{T})} = 1/2$. Let $X \in \mathbb{R}^3$ be a three-dimensional covariate vector with mechanism function $f^{(1)}$ and $f^{(\mathfrak{T})}$ as given by

$$f^{(1)}(1) = \begin{pmatrix} -1 \\ -1 \\ -1 \end{pmatrix}, \ f^{(1)}(2) = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix};$$

$$f^{(\mathfrak{T})}(1) = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}, \ f^{(\mathfrak{T})}(2) = \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix},$$

and where the noise terms $\epsilon^{(1)}$ and $\epsilon^{(\mathfrak{T})}$ are drawn from a Gaussian distribution $\mathcal{P}_\epsilon = \mathcal{N}(0, \mathbb{I}_3)$. Examining each coordinate, we can see that the first coordinate $X_{[1]}$ maintains its conditional invariance across source and target, and thus it is a CIC. The second coordinate $X_{[2]}$ is perturbed, but the correlations $\rho(X_{[2]}^{(1)}, Y^{(1)}) = \rho(X_{[2]}^{(\mathfrak{T})}, Y^{(\mathfrak{T})}) = \sqrt{2}/2$ remian unchanged. The third coordinate $X_{[3]}$ is also perturbed, and by Definition 4, it is a label-flipping feature, as $\rho(X_{[3]}^{(1)}, Y^{(1)}) = -\rho(X_{[3]}^{(\mathfrak{T})}, Y^{(\mathfrak{T})}) = \sqrt{2}/2$.

A.2 Demonstration of condition in Theorem 15(b)

The condition (35) in Theorem 15(b) requires that for the linear conditionally invariant feature mapping ϕ_{inv}, CICs must have different conditional means for different labels. In other words, the first-moment of the CICs should suffice to distinguish between labels. We show that under the anticausal model given in the example above, this condition is indeed satisfied.

Example 1 Cont. In Example 1, one possible linear conditionally invariant feature mapping is $\phi_{\mathrm{inv}}(x) = x_{[1]}$, i.e., it projects to the first coordinate of the covariates. Simple algebra shows that

$$\mathbb{E}\left[\phi_{\mathrm{inv}}(X^{(1)}) \mid Y^{(1)} = 1\right] = -1, \text{ and } \mathbb{E}\left[\phi_{\mathrm{inv}}(X^{(1)}) \mid Y^{(1)} = 2 = 1\right],$$

which confirms the condition (35).

A.3 Demonstration of condition in Theorem 15(c)

While the condition (35) for linear ϕ_{inv} is relatively straightforward, verifying the full-rank condition of matrix $C_{\phi_{\text{inv}}}(a)$ for general ϕ_{inv} in Theorem 15(c) can be more challenging due to the computation of higher-moments. Here we provide several examples where a nonlinear conditionally invariant feature mapping ϕ_{inv} meets the full rank condition.

Example 1 Cont. Consider the general anticausal model introduced in Example 1 ($L = 2$) and let

$$
\begin{aligned}
\phi_{\text{inv}}^k(x) &= x_{[1]}^k, \quad k = 0, 1, 2, \ldots, \text{ and} \\
\phi_{\text{inv}}^e(x) &= \exp(x_{[1]}).
\end{aligned}
\tag{37}
$$

Since $X_{[1]}$ is a CIC, we know that $\phi_{\text{inv}}^k(X_{[1]})$ and $\phi_{\text{inv}}^e(X_{[1]})$ are also CICs. Take $a = 1$ in the definition of the matrix $C_{\phi_{\text{inv}}}(a)$. From the standard calculation of moments for Gaussian random variables, we have

$$
C_{\phi_{\text{inv}}^k}(a) = \begin{pmatrix} 1 & 1 \\ \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[(u-1)^k\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[(u+1)^k\right] \end{pmatrix}, \quad C_{\phi_{\text{inv}}^e}(a) = \begin{pmatrix} 1 & 1 \\ e^{-\frac{1}{2}} & e^{\frac{3}{2}} \end{pmatrix}.
$$

We conclude that $C_{\phi_{\text{inv}}^k}(a)$ is full rank if and only if k is odd, and $C_{\phi_{\text{inv}}^e}(a)$ is a full rank matrix. Therefore, $\phi_{\text{inv}}^{2k+1}(x) = x_{[1]}^{2k+1}(k = 0, 1, 2, \ldots)$ and $\phi_{\text{inv}}^e(x) = \exp(x_{[1]})$ satisfy the full rank condition.

Example 2 Consider the general anticausal model from Assumption 3 where $Y \in \mathcal{Y} = \{1, 2, 3\}$ with $L = 3$. Let $X \in \mathbb{R}^2$ be a two-dimensional covariate vector with mechanism functions $f^{(1)}$ and $f^{(\mathfrak{T})}$ defined as follows:

$$
f^{(1)}(1) = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \ f^{(1)}(2) = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \ f^{(1)}(3) = \begin{pmatrix} 3 \\ 1 \end{pmatrix};
$$

$$
f^{(\mathfrak{T})}(1) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \ f^{(\mathfrak{T})}(2) = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \ f^{(\mathfrak{T})}(3) = \begin{pmatrix} 3 \\ -1 \end{pmatrix}.
$$

That is, the first coordinate $X_{[1]}$ is a CIC, while the second coordinate $X_{[2]}$ is perturbed. Assume that the noise distribution $\mathcal{P}_\epsilon = \mathcal{N}(0, \mathbb{I}_2)$ follows a Gaussian distribution. Consider the feature mappings ϕ_{inv}^2 and ϕ_{inv}^e from Eq. (37) as the conditionally invariant feature mappings, and take $a = 1$. Basic algebra shows that

$$
\begin{aligned}
C_{\phi_{\text{inv}}^2}(a) &= \begin{pmatrix} 1 & 1 & 1 \\ \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[(u+1)^2\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[(u+2)^2\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[(u+3)^2\right] \\ \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[(u+1)^4\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[(u+2)^4\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[(u+3)^4\right] \end{pmatrix} \\
&= \begin{pmatrix} 1 & 1 & 1 \\ 2 & 5 & 10 \\ 10 & 43 & 138 \end{pmatrix},
\end{aligned}
$$

and

$$
C_{\phi_{\text{inv}}^e}(a) = \begin{pmatrix} 1 & 1 & 1 \\ \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[e^{u+1}\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[e^{u+2}\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[e^{u+3}\right] \\ \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[e^{2(u+1)}\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[e^{2(u+2)}\right] & \mathbb{E}_{u\sim\mathcal{N}(0,1)}\left[e^{2(u+3)}\right] \end{pmatrix}
$$

$$= \begin{pmatrix} 1 & 1 & 1 \\ e^{\frac{3}{2}} & e^{\frac{5}{2}} & e^{\frac{7}{2}} \\ e^4 & e^6 & e^8 \end{pmatrix},$$

both of which are full-rank, and therefore satisfying the full rank condition in Theorem 15(c).

Example 3 Consider the general anticausal model from Assumption 3 with arbitrary number of classes $L \geq 2$. Let

$$f^{(m)}(y) = \begin{pmatrix} e_y \\ a_y^{(m)} \end{pmatrix}, f^{(\mathfrak{T})}(y) = \begin{pmatrix} e_y \\ a_y^{(\mathfrak{T})} \end{pmatrix},$$

where $e_i \in \mathbb{R}^L$ denotes the zero vector with only the i-th element being 1, and $a_y^{(m)}, a_y^{(\mathfrak{T})} \in \mathbb{R}^{p-L}$ $(p > L)$ are arbitrary vectors. That is, the first L covariates are CICs while the last $p - L$ coordinates can be arbitrarily perturbed. Let

$$\phi_{\mathrm{inv}}^k(X) = \left(X_{[1]}^k, X_{[2]}^k, \cdots, X_{[L]}^k \right)^\top, \quad k = 1, 2, \dots.$$

For a fixed vector $a \in \mathbb{R}^L$, verifying $C_{\phi_{\mathrm{inv}}^k}(a)$ is full rank can be technical depending on the distribution $\mathcal{P}_\epsilon$. Instead, we show *an illustrative explanation* rather than a rigorous proof. First, let's consider $\mathcal{P}_\epsilon = \delta_0(\cdot)$, the Dirac distribution centered at 0. Then we can compute

$$C_{\phi_{\mathrm{inv}}^k}(a) = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ a_{[1]} & a_{[2]} & \cdots & a_{[L]} \\ \vdots & \vdots & \ddots & \vdots \\ a_{[1]}^{L-1} & a_{[2]}^{L-1} & \cdots & a_{[L]}^{L-1} \end{pmatrix},$$

which is a Vandermonde matrix and is full rank as long as we take $a_{[i]} \neq a_{[j]}, i \neq j$. Next we consider a case where $\mathcal{P}_\epsilon$ is a distribution concentrated around 0 (e.g. a Gaussian distribution with very small variance). In this case, the matrix will be slightly perturbed but is still full rank by the continuity of high-order moments and matrix determinants.

Appendix B. Proof of Section 3 and additional results on IW-CIP

In this section we prove our main theorems and propositions given in Section 3, as well as additional results given in Proposition 16 and Lemma 17. Proposition 16 provides the finite-sample bound for the IW-CIP penalty when it is defined by the squared mean distance, and Lemma 17 proves the number of required source domains to validate the conditions in Proposition 9 when the perturbations are randomly generated.

B.1 Proof of Theorem 6.A

Let $w = (w^{(1)}, \dots, w^{(M)})$ where $w^{(m)}$ is the true importance weight for the m-th source domain. Writing $p_y = \mathbb{P}\left\{ Y^{(\mathfrak{T})} = y \right\}$, for any classifier $h = g \circ \phi$, we have

$$\mathcal{R}^{(\mathfrak{T})}(h) - \overline{\mathcal{R}}(h; w)$$

$$= \frac{1}{M} \sum_{m=1}^{M} \left(\mathbb{E} \left[\mathbf{1}_{h(X^{(\mathfrak{T})}) \neq Y^{(\mathfrak{T})}} \right] - \mathbb{E} \left[w_{[Y^{(m)}]}^{(m)} \cdot \mathbf{1}_{h(X^{(m)}) \neq Y^{(m)}} \right] \right)$$

$$= \frac{1}{M} \sum_{m=1}^{M} \sum_{y=1}^{L} \left(\mathbb{P} \left\{ h(X^{(\mathfrak{T})}) \neq y \mid Y^{(\mathfrak{T})} = y \right\} - \mathbb{P} \left\{ h(X^{(m)}) \neq y \mid Y^{(m)} = y \right\} \right) p_y$$

$$= \frac{1}{M} \sum_{m=1}^{M} \sum_{y=1}^{L} \left(\mathbb{P} \left\{ h(X^{(m)}) = y \mid Y^{(m)} = y \right\} - \mathbb{P} \left\{ h(X^{(\mathfrak{T})}) = y \mid Y^{(\mathfrak{T})} = y \right\} \right) p_y,$$

where the second step applies $w_{[y]}^{(m)} = \frac{\mathbb{P}\{Y^{(\mathfrak{T})}=y\}}{\mathbb{P}\{Y^{(m)}=y\}}$ and the law of total expectation. By definition of $D_{\mathcal{G}}(\cdot, \cdot)$, it is then upper bounded by

$$\left| \mathcal{R}^{(\mathfrak{T})}(h) - \overline{\mathcal{R}}(h; w) \right| \leq \frac{1}{M} \sum_{m=1}^{M} \sum_{y=1}^{L} D_{\mathcal{G}} \left(\mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}, \mathcal{P}_{\phi(X)|Y=y}^{(m)} \right) \mathbb{P} \left\{ Y^{(\mathfrak{T})} = y \right\}$$

$$\leq \max_{\substack{m=1,\dots,M, \\ y=1,\dots,L}} D_{\mathcal{G}} \left(\mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}, \mathcal{P}_{\phi(X)|Y=y}^{(m)} \right) \sum_{m=1}^{M} \frac{1}{M} \sum_{y=1}^{L} \mathbb{P} \left\{ Y^{(\mathfrak{T})} = y \right\}$$

$$= \max_{\substack{m=1,\dots,M, \\ y=1,\dots,L}} D_{\mathcal{G}} \left(\mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}, \mathcal{P}_{\phi(X)|Y=y}^{(m)} \right)$$

$$= \Psi_{\mathcal{G},\phi}, \tag{38}$$

where we use $\sum_{y=1}^{L} \mathbb{P}\left\{ Y^{(\mathfrak{T})} = y \right\} = 1$ and $\sum_{m=1}^{M} 1/M = 1$ in the second step. Furthermore, we can bound

$$\left| \mathcal{R}^{(m)}(h; w^{(m)}) - \mathcal{R}^{(m)}(h; \widehat{w}^{(m)}) \right| = \left| \mathbb{E} \left[\left(w_{[Y^{(m)}]}^{(m)} - \widehat{w}_{[Y^{(m)}]}^{(m)} \right) \cdot \mathbf{1}_{h(X^{(m)}) \neq Y^{(m)}} \right] \right|$$

$$\leq \| w^{(m)} - \widehat{w}^{(m)} \|_{\infty}.$$

It follows that

$$\left| \overline{\mathcal{R}}(h; w) - \overline{\mathcal{R}}(h; \widehat{w}) \right| \leq \frac{1}{M} \sum_{m=1}^{M} \left| \mathcal{R}^{(m)}(h; w^{(m)}) - \mathcal{R}^{(m)}(h; \widehat{w}^{(m)}) \right|$$

$$\leq \frac{1}{M} \sum_{m=1}^{M} \| w^{(m)} - \widehat{w}^{(m)} \|_{\infty}. \tag{39}$$

The result now follows from combining Eq. (38) and Eq. (39).

B.2 Proof of Theorem 6.B

Denote the empirical average source risk across M environments by

$$\overline{\widehat{\mathcal{R}}}(h; \mathrm{w}^{(m)}) = \frac{1}{M} \sum_{m=1}^{M} \widehat{\mathcal{R}}^{(m)}(h; \mathrm{w}^{(m)}).$$

Then we can write

$$
\begin{aligned}
&\mathcal{R}^{(\mathfrak{T})}(\widehat{h}_{\text{IW-CIP}}) - \mathcal{R}^{(\mathfrak{T})}(h^\star) \\
&= \underbrace{\mathcal{R}^{(\mathfrak{T})}(\widehat{h}_{\text{IW-CIP}}) - \overline{\mathcal{R}}(\widehat{h}_{\text{IW-CIP}}; w)}_{(A_1)} + \underbrace{\overline{\mathcal{R}}(\widehat{h}_{\text{IW-CIP}}; w) - \widehat{\overline{\mathcal{R}}}(\widehat{h}_{\text{IW-CIP}}; w)}_{(B_1)} + \\
&\quad + \underbrace{\widehat{\overline{\mathcal{R}}}(\widehat{h}_{\text{IW-CIP}}; w) - \widehat{\overline{\mathcal{R}}}(\widehat{h}_{\text{IW-CIP}}; \widehat{w})}_{(C_1)} + \underbrace{\widehat{\overline{\mathcal{R}}}(\widehat{h}_{\text{IW-CIP}}; \widehat{w}) - \widehat{\overline{\mathcal{R}}}(h^\star; \widehat{w})}_{(D)} + \underbrace{\widehat{\overline{\mathcal{R}}}(h^\star; \widehat{w}) - \widehat{\overline{\mathcal{R}}}(h^\star; w)}_{(C_2)} \\
&\quad + \underbrace{\widehat{\overline{\mathcal{R}}}(h^\star; w) - \overline{\mathcal{R}}(h^\star; w)}_{(B_2)} + \underbrace{\overline{\mathcal{R}}(h^\star; w) - \mathcal{R}^{(\mathfrak{T})}(h^\star)}_{(A_2)}.
\end{aligned}
$$

Term A_1 and A_2 From Eq. (38) in the proof of Theorem 6.A, we have

$$
\begin{aligned}
A_1 &= \mathcal{R}^{(\mathfrak{T})}(\widehat{h}_{\text{IW-CIP}}) - \overline{\mathcal{R}}(\widehat{h}_{\text{IW-CIP}}; w) \leq \Psi_{\mathcal{G}, \widehat{\phi}_{\text{IW-CIP}}}, \\
A_2 &= \overline{\mathcal{R}}(h^\star; w) - \mathcal{R}^{(\mathfrak{T})}(h^\star) = 0.
\end{aligned}
\tag{40}
$$

The second equation follows from the definition of $h^\star = g^\star \circ \phi^\star$, i.e., $\phi^\star$ is a conditionally invariant feature mapping and satisfies $\Psi_{\mathcal{G}, \phi^\star} = 0$.

Term B_1 and B_2 For any $h = g \circ \phi$, we have $\left| w^{(m)}_{[Y^{(m)}]} \mathbf{1}_{h(X^{(m)}) \neq Y^{(m)}} \right| \leq \left\| w^{(m)} \right\|_\infty$. Then the generalization bound based on Rademacher complexity (see e.g. (Wainwright, 2019, Theorem 4.10)) shows that, for the m-th source domain, the bound

$$
\sup_{\substack{h = g \circ \phi \\ g \in \mathcal{G}, \phi \in \Phi}} \left| \mathcal{R}^{(m)}(h; w^{(m)}) - \widehat{\mathcal{R}}^{(m)}(h; w^{(m)}) \right| \leq 2 \mathfrak{R}_{n^{(m)}, \mathcal{P}^{(m)}} \left(\mathcal{H}(w^{(m)}, \mathcal{G}, \Phi) \right)
$$

$$
+ \left\| w^{(m)} \right\|_\infty \sqrt{\frac{2 \log(M/\delta)}{n^{(m)}}},
$$

holds with probability at least $1 - \delta/M$. Using the union bound over each $m \in \{1, \ldots, M\}$, we obtain that with probability at least $1 - \delta$,

$$
\begin{aligned}
|B_1| + |B_2| &\leq 2 \sup_{\substack{h = g \circ \phi, \\ g \in \mathcal{G}, \phi \in \Phi}} \frac{1}{M} \sum_{m=1}^{M} \left| \mathcal{R}^{(m)}(h; w^{(m)}) - \widehat{\mathcal{R}}^{(m)}(h; w^{(m)}) \right| \\
&\leq \max_{m=1,\ldots,M} \left[4 \mathfrak{R}_{n^{(m)}, \mathcal{P}^{(m)}} \left(\mathcal{H}(w^{(m)}, \mathcal{G}, \Phi) \right) + 2 \left\| w^{(m)} \right\|_\infty \sqrt{\frac{2 \log(M/\delta)}{n^{(m)}}} \right].
\end{aligned}
\tag{41}
$$

Term C_1 and C_2 For the m-th source domain, let us define

$$
\widehat{\ell}_j^{(m)} = \sum_{k=1}^{n} \mathbf{1}_{Y_k^{(m)} = j} \mathbf{1}_{h(X_k^{(m)}) \neq Y_k^{(m)}}.
$$

Then $\|\widehat{\ell}^{(m)}\|_1 \leq n$ and so for any $h = g \circ \phi$,

$$
\left| \widehat{\mathcal{R}}^{(m)}(h; w^{(m)}) - \widehat{\mathcal{R}}^{(m)}(h; \widehat{w}^{(m)}) \right| = \left| \frac{1}{n} \sum_{k=1}^{n} \left(w^{(m)}_{[Y_k^{(m)}]} - \widehat{w}^{(m)}_{[Y_k^{(m)}]} \right) \mathbf{1}_{h(X_k^{(m)}) \neq Y_k^{(m)}} \right|
$$

$$= \left| \frac{1}{n} \sum_{j=1}^{L} \left(w_{[j]}^{(m)} - \widehat{w}_{[j]}^{(m)} \right) \widehat{\ell}_j^{(m)} \right|$$

$$\leq \frac{1}{n} \left\| w^{(m)} - \widehat{w}^{(m)} \right\|_\infty \left\| \widehat{\ell}^{(m)} \right\|_1 \leq \left\| w^{(m)} - \widehat{w}^{(m)} \right\|_\infty.$$

It follows that

$$|C_1| + |C_2| \leq 2 \sup_{\substack{h=g\circ\phi, \\ g\in\mathcal{G},\phi\in\Phi}} \frac{1}{M} \sum_{m=1}^{M} \left| \widehat{\mathcal{R}}^{(m)}(h; w^{(m)}) - \widehat{\mathcal{R}}^{(m)}(h; \widehat{w}^{(m)}) \right|$$

$$\leq \frac{2}{M} \sum_{m=1}^{M} \left\| w^{(m)} - \widehat{w}^{(m)} \right\|_\infty. \tag{42}$$

Term D By definition of $\widehat{h}_{\text{IW-CIP}}$ from Eq. (18), we know that

$$\widehat{\overline{\mathcal{R}}}(\widehat{h}_{\text{IW-CIP}}; \widehat{w}) + \widehat{\Lambda}_{\widehat{\phi}_{\text{IW-CIP}}} \leq \widehat{\overline{\mathcal{R}}}(h^\star; \widehat{w}) + \widehat{\Lambda}_{\phi^\star},$$

and so rearranging terms,

$$D = \widehat{\overline{\mathcal{R}}}(\widehat{h}_{\text{IW-CIP}}; \widehat{w}) - \widehat{\overline{\mathcal{R}}}(h^\star; \widehat{w}) \leq \widehat{\Lambda}_{\phi^\star} - \widehat{\Lambda}_{\widehat{\phi}_{\text{IW-CIP}}} \leq \widehat{\Lambda}_{\phi^\star}. \tag{43}$$

The theorem then follows from combining Eq. (40), (41), (42), and (43).

B.3 Proof of Proposition 7

For a family of functions $\mathcal{G}$ mapping $\mathcal{Z}$ to $[0,1]$ and a distribution $\mathcal{P}$ on $\mathcal{Z}$, let $\widehat{\mathcal{P}}_n$ denote the empirical distribution of n i.i.d. samples $(Z_1, Z_2, \ldots, Z_n)$ from $\mathcal{P}$. The standard Rademacher complexity theory (see e.g. (Wainwright, 2019, Theorem 4.10)) states that for any $\delta > 0$, with probability at least $1 - \delta$, we have

$$\sup_{g\in\mathcal{G}} \left| \mathbb{E}_{Z\sim\mathcal{P}}[g(Z)] - \mathbb{E}_{Z\sim\widehat{\mathcal{P}}_n}[g(Z)] \right| \leq 2\Re_{n,\mathcal{P}}(\mathcal{G}) + \sqrt{\frac{\log(2/\delta)}{2n}}.$$

By definition of $\mathcal{G}$-divergence, with probability $1 - \delta$ we have

$$D_\mathcal{G}\left(\mathcal{P}, \widehat{\mathcal{P}}_n\right) = \sup_{g\in\mathcal{G}} \max_{y=1,2,\ldots,L} \left| \mathbb{E}_{Z\sim\mathcal{P}}[\mathbf{1}_{g(Z)=y}] - \mathbb{E}_{Z\sim\widehat{\mathcal{P}}_n}[\mathbf{1}_{g(Z)=y}] \right|$$

$$= \max_{y=1,2,\ldots,L} \sup_{g\in\mathcal{G}} \left| \mathbb{E}_{Z\sim\mathcal{P}}[\mathbf{1}_{g(Z)=y}] - \mathbb{E}_{Z\sim\widehat{\mathcal{P}}_n}[\mathbf{1}_{g(Z)=y}] \right|$$

$$\leq \max_{y=1,2,\ldots,L} \left(2\Re_{n,\mathcal{P}}(\mathcal{G}_y) + \sqrt{\frac{\log(2L/\delta)}{2n}} \right), \tag{44}$$

where $\mathcal{G}_y := \{\mathbf{1}_{g(x)=y}, g\in\mathcal{G}\}$. Then we can bound

$$D_\mathcal{G}\left(\widehat{\mathcal{P}}_{\phi^\star(X)|Y=y}^{(m)}, \widehat{\mathcal{P}}_{\phi^\star(X)|Y=y}^{(m')} \right)$$

$$
\overset{(i)}{\leq} D_{\mathcal{G}}\left(\widehat{\mathcal{P}}^{(m)}_{\phi^\star(X)|Y=y}, \mathcal{P}^{(m)}_{\phi^\star(X)|Y=y}\right) + D_{\mathcal{G}}\left(\mathcal{P}^{(m)}_{\phi^\star(X)|Y=y}, \mathcal{P}^{(m')}_{\phi^\star(X)|Y=y}\right)
$$
$$
+ D_{\mathcal{G}}\left(\mathcal{P}^{(m')}_{\phi^\star(X)|Y=y}, \widehat{\mathcal{P}}^{(m')}_{\phi^\star(X)|Y=y}\right)
$$
$$
\overset{(ii)}{=} D_{\mathcal{G}}\left(\widehat{\mathcal{P}}^{(m)}_{\phi^\star(X)|Y=y}, \mathcal{P}^{(m)}_{\phi^\star(X)|Y=y}\right) + D_{\mathcal{G}}\left(\widehat{\mathcal{P}}^{(m')}_{\phi^\star(X)|Y=y}, \mathcal{P}^{(m')}_{\phi^\star(X)|Y=y}\right), \tag{45}
$$

where in step (i) we apply triangular inequality, and in step (ii) we use the fact that $\phi^\star$ is a conditionally invariant feature mapping across source domains to cancel the middle term. Applying union bound, we get

$$
\widehat{\Lambda}_{\phi^\star} = \frac{\lambda_{\text{IW-CIP}}}{LM^2} \sum_{y=1}^{L} \sum_{m \neq m'} D_{\mathcal{G}}\left(\widehat{\mathcal{P}}^{(m)}_{\phi^\star(X)|Y=y}, \widehat{\mathcal{P}}^{(m')}_{\phi^\star(X)|Y=y}\right)
$$
$$
\overset{(i)}{\leq} \frac{2\lambda_{\text{IW-CIP}}}{LM} \sum_{y=1}^{L} \sum_{m=1}^{M} D_{\mathcal{G}}\left(\mathcal{P}^{(m)}_{\phi^\star(X)|Y=y}, \widehat{\mathcal{P}}^{(m)}_{\phi^\star(X)|Y=y}\right)
$$
$$
\overset{(ii)}{\leq} \frac{4\lambda_{\text{IW-CIP}}}{LM} \cdot \sum_{y=1}^{L} \sum_{m=1}^{M} \max_{y'=1,2,\ldots,L} \mathfrak{R}_{n^{(m)}, \mathcal{P}^{(m)}_{\phi^\star(X)|Y=y}}\left(\mathcal{G}_{y'}\right) + \frac{2\lambda_{\text{IW-CIP}}}{M} \sum_{m=1}^{M} \sqrt{\frac{\log(2LM/\delta)}{2n^{(m)}}}
$$
$$
\leq 2\lambda_{\text{IW-CIP}} \left(2 \max_{\substack{m=1,2,\ldots,M \\ y=1,2,\ldots,L \\ y'=1,2,\ldots,L}} \mathfrak{R}_{n^{(m)}, \mathcal{P}^{(m)}_{\phi^\star(X)|Y=y}}\left(\mathcal{G}_{y'}\right) + \max_{m=1,2,\ldots,M} \sqrt{\frac{\log(2LM/\delta)}{2n^{(m)}}} \right),
$$

with probability at least $1 - \delta$. Here step (i) uses Eq. (45) and step (ii) uses Eq. (44). This completes the proof of the proposition.

B.4 Another bound on $\widehat{\Lambda}_{\phi^\star}$ under mean squared distance

In Proposition 7 we established a finite-sample bound for $\widehat{\Lambda}_{\phi^\star}$ when $\mathcal{G}$-divergence is used. If the IW-CIP penalty is defined by the squared mean distance instead, we provide another explicit bound for $\widehat{\Lambda}_{\phi^\star}$ that converges to zero as sample size goes to infinity. To show this, we require additional assumptions on the data generation model and the class of feature mappings Φ.

Proposition 16 *Suppose that source and target data are generated under Assumption 1. Let $\Phi = \{\phi(x) = Ax + b, A \in \mathbb{R}^{q \times p}, b \in \mathbb{R}^q\}$ be the linear class of feature mappings from $\mathbb{R}^p$ to $\mathbb{R}^q$ ($q \leq p$), and let the squared mean distance be the distributional distance in IW-CIP penalty Eq. (19). Suppose $\phi \in \Phi$ is a conditionally invariant feature mapping such that $A\Sigma A^\top$ is non-singular. For any $\delta > 0$ satisfying $\min_{m,y} n^{(m)} p_y^{(m)} \geq 2\log(2ML/\delta)$, we have*

$$
\widehat{\Lambda}_\phi \leq \frac{12\lambda_{\text{IW-CIP}} \cdot \lambda_{\max}(A\Sigma A^\top) \cdot (p + 8\log(2ML/\delta))}{\min_{\substack{m=1,2,\ldots,M \\ y=1,2,\ldots,L}} n^{(m)} p_y^{(m)} - 2\log(2ML/\delta)}, \tag{46}
$$

with probability at least $1-\delta$.[13] Specifically, the statement holds for the optimal conditionally invariant mapping $\phi^\star$ (see Eq. (5)).

Proof The exact from of IW-CIP penalty under squared mean distance is

$$\widehat{\Lambda}_\phi := \frac{\lambda_{\text{IW-CIP}}}{LM^2} \sum_{y=1}^L \sum_{m \neq m'} \left\| \widehat{\mu_\phi^{(m)}(y)} - \widehat{\mu_\phi^{(m')}(y)} \right\|_2^2,$$

where

$$\widehat{\mu_\phi^{(m)}(y)} = \frac{1}{n_y^{(m)}} \sum_{\substack{k=1 \\ Y_k^{(m)}=y}}^{n^{(m)}} \phi(X_k^{(m)}),$$

is the empirical mean of $\phi(X)$ on the m-th source domain. Since ϕ is conditionally invariant, from Eq. (52) we know that for any $y \in \{1, 2, \ldots, L\}$ and for any $m \neq m'$,

$$AP_y v_y^{(m)} = AP_y v_y^{(m')}.$$

In the finite-sample case, when $Y_k^{(m)} = Y_k^{(\mathfrak{T})} = y$, we can write

$$\phi(X_k^{(m)}) = AX_k^{(m)} + b = A\left(f^{(1)}(y) + P_y v_y^{(m)} + \epsilon_k^{(m)}\right) + b,$$

$$\phi(X_k^{(\mathfrak{T})}) = AX_k^{(\mathfrak{T})} + b = A\left(f^{(1)}(y) + P_y v_y^{(\mathfrak{T})} + \epsilon_k^{(\mathfrak{T})}\right) + b,$$

and

$$\widehat{\mu_\phi^{(m)}(y)} - \widehat{\mu_\phi^{(m')}(y)} = \frac{1}{n_y^{(m)}} \sum_{\substack{k=1 \\ Y_k^{(m)}=y}}^{n^{(m)}} \phi(X_k^{(m)}) - \frac{1}{n_y^{(m')}} \sum_{\substack{k=1 \\ Y_k^{(m')}=y}}^{n^{(m')}} \phi(X_k^{(m')})$$

$$= A \left(\underbrace{\frac{1}{n_y^{(m)}} \sum_{\substack{k=1 \\ Y_k^{(m)}=y}}^{n^{(m)}} \epsilon_k^{(m)}}_{e_y^{(m)}} - \underbrace{\frac{1}{n_y^{(m')}} \sum_{\substack{k=1 \\ Y_k^{(m')}=y}}^{n^{(m')}} \epsilon_k^{(m')}}_{e_y^{(m')}} \right), \qquad (47)$$

where $n_y^{(m)}$ is the number of samples with label y in the m-th domain. Since $n_y^{(m)}$ follows a multinomial distribution, we conclude from binomial (multinomial) tail probability bound (e.g. (Chung and Lu, 2006, Theorem 3.2)) that for all $t_y^{(m)} > 0$,

$$\mathbb{P}\left\{n_y^{(m)} \geq n^{(m)} p_y^{(m)}(1 - t_y^{(m)})\right\} \geq 1 - \exp\left(-\frac{n^{(m)} p_y^{(m)} t_y^{(m)2}}{2}\right).$$

---

13. The probability is with respect to the randomness of $(X_k^{(m)}, Y_k^{(m)}) \overset{\text{i.i.d.}}{\sim} \mathcal{P}^{(m)}$, $m = 1, \ldots, M, k = 1, \ldots, n^{(m)}$; see Eq. (20).

When $n_y^{(m)} = k > 0$ is fixed, we have $Ae_y^{(m)} \sim \mathcal{N}(0, \frac{1}{k} A\Sigma A^\top)$, and therefore the standard concentration bound for Gaussian variables (e.g. (Wainwright, 2019, Example 2.11)) implies that for all $t > 0$,

$$\mathbb{P}\left\{\left\|Ae_y^{(m)}\right\|_2^2 \le \frac{1}{k}\lambda_{\max}\left(A\Sigma A^\top\right) \cdot p\left(1+t\right) \,\middle|\, n_y^{(m)} = k\right\} \ge 1 - e^{-p\min\{t,t^2\}/8}.$$

Next we combine two bounds above into a single bound for $\left\|Ae_y^{(m)}\right\|_2^2$. Given any $t_y^{(m)} > 0$ and $t > 0$, we have

$$\mathbb{P}\left\{\left\|Ae_y^{(m)}\right\|_2^2 \le \frac{\lambda_{\max}\left(A\Sigma A^\top\right) \cdot p\left(1+t\right)}{n^{(m)}p_y^{(m)}(1-t_y^{(m)})}\right\}$$

$$= \sum_{k=0}^{n^{(m)}} \mathbb{P}\left\{\left\|Ae_y^{(m)}\right\|_2^2 \le \frac{\lambda_{\max}\left(A\Sigma A^\top\right) \cdot p\left(1+t\right)}{n^{(m)}p_y^{(m)}(1-t_y^{(m)})} \,\middle|\, n_y^{(m)} = k\right\} \mathbb{P}\left\{n_y^{(m)} = k\right\}$$

$$\ge \sum_{k=n^{(m)}p_y^{(m)}(1-t_y^{(m)})}^{n^{(m)}} \mathbb{P}\left\{\left\|Ae_y^{(m)}\right\|_2^2 \le \frac{\lambda_{\max}\left(A\Sigma A^\top\right) \cdot p\left(1+t\right)}{n^{(m)}p_y^{(m)}(1-t_y^{(m)})} \,\middle|\, n_y^{(m)} = k\right\} \mathbb{P}\left\{n_y^{(m)} = k\right\}$$

$$\ge \sum_{k=n^{(m)}p_y^{(m)}(1-t_y^{(m)})}^{n^{(m)}} \mathbb{P}\left\{\left\|Ae_y^{(m)}\right\|_2^2 \le \frac{1}{k}\lambda_{\max}\left(A\Sigma A^\top\right) \cdot p\left(1+t\right) \,\middle|\, n_y^{(m)} = k\right\} \mathbb{P}\left\{n_y^{(m)} = k\right\}$$

$$\ge \sum_{k=n^{(m)}p_y^{(m)}(1-t_y^{(m)})}^{n^{(m)}} \left(1 - e^{-p\min\{t,t^2\}/8}\right) \mathbb{P}\left\{n_y^{(m)} = k\right\}$$

$$\ge \left(1 - e^{-p\min\{t,t^2\}/8}\right)\left(1 - e^{-n^{(m)}p_y^{(m)}t_y^{(m)2}/2}\right)$$

$$\ge 1 - e^{-p\min\{t,t^2\}/8} - e^{-n^{(m)}p_y^{(m)}t_y^{(m)2}/2}.$$

Take $t_y^{(m)} = \sqrt{2\log(2ML/\delta)/(n^{(m)}p_y^{(m)})}$ and $t = 8\log(2ML/\delta)/p + \sqrt{8\log(2ML/\delta)/p}$. Then with probability at least $1 - \delta/ML$, we have

$$\left\|Ae_y^{(m)}\right\|_2^2 \le \frac{\lambda_{\max}\left(A\Sigma A^\top\right)\left(p + 8\log(2ML/\delta) + \sqrt{8p\log(2ML/\delta)}\right)}{n^{(m)}p_y^{(m)} - \sqrt{n^{(m)}p_y^{(m)} \cdot 2\log(2ML/\delta)}}$$

$$\le \frac{3\lambda_{\max}\left(A\Sigma A^\top\right)\left(p + 8\log(2ML/\delta)\right)}{n^{(m)}p_y^{(m)} - 2\log(2ML/\delta)},$$

where we apply the inequality $\sqrt{ab} \le (a+b)/2$ for $a, b > 0$ in the last step. Applying union bound, we get

$$\widehat{\Lambda}_\phi = \frac{\lambda_{\text{IW-CIP}}}{LM^2} \sum_{y=1}^{L} \sum_{m \ne m'} \left\|\widehat{\mu_\phi^{(m)}(y)} - \widehat{\mu_\phi^{(m')}(y)}\right\|_2^2$$

$$= \frac{\lambda_{\text{IW-CIP}}}{LM^2} \sum_{y=1}^{L} \sum_{m \neq m'} \left\| Ae_y^{(m)} - Ae_y^{(m')} \right\|_2^2$$

$$\leq \frac{2\lambda_{\text{IW-CIP}}}{LM^2} \sum_{y=1}^{L} \sum_{m \neq m'} \left(\left\| Ae_y^{(m)} \right\|_2^2 + \left\| Ae_y^{(m')} \right\|_2^2 \right)$$

$$\leq 4\lambda_{\text{IW-CIP}} \cdot \max_{\substack{m=1,2,...,M \\ y=1,2,...,L}} \left\| Ae_y^{(m)} \right\|_2^2$$

$$\leq \frac{12\lambda_{\text{IW-CIP}} \cdot \lambda_{\max}(A\Sigma A^\top) \cdot (p + 8\log(2ML/\delta))}{\min\limits_{\substack{m=1,2,...,M \\ y=1,2,...,L}} n^{(m)} p_y^{(m)} - 2\log(2ML/\delta)},$$

with probability at least $1 - \delta$, proving the result. ∎

B.5 Proof of Proposition 8

We have that for any $i \in \{1, 2, ..., L\}$,

$$\mathbb{P}\left\{ \widehat{h}_{\text{CIP}}(X^{(\mathfrak{T})}) = i \right\} = \sum_{j=1}^{L} \mathbb{P}\left\{ \widehat{h}_{\text{CIP}}(X^{(\mathfrak{T})}) = i \ \middle| \ Y^{(\mathfrak{T})} = j \right\} \cdot \mathbb{P}\left\{ Y^{(\mathfrak{T})} = j \right\}$$

$$= \sum_{j=1}^{L} \left(\mathbb{P}\left\{ \widehat{h}_{\text{CIP}}(X^{(m)}) = i \ \middle| \ Y^{(m)} = j \right\} + \delta_{i,j} \right) \cdot \mathbb{P}\left\{ Y^{(\mathfrak{T})} = j \right\},$$

where $\delta_{i,j} = \mathbb{P}\left\{ \widehat{h}_{\text{CIP}}(X^{(\mathfrak{T})}) = i \ \middle| \ Y^{(\mathfrak{T})} = j \right\} - \mathbb{P}\left\{ \widehat{h}_{\text{CIP}}(X^{(1)}) = i \ \middle| \ Y^{(1)} = j \right\}$. Writing the above equation in matrix form, we have

$$\mu_{\widehat{h}_{\text{CIP}}} = C_{\widehat{h}_{\text{CIP}}}^{(m)} w^{(m)} + \Delta\mu^{(\mathfrak{T})}, \tag{48}$$

where Δ is a matrix with $\delta_{i,j}$ being its (i,j)-th element, and $\mu^{(\mathfrak{T})}$ denotes the target label distribution. Comparing with the linear system in Eq. (15), Eq. (48) has an additional term on the right-hand side because the finite-sample CIP $\widehat{h}_{\text{CIP}}$ does not guarantee exact conditional invariance. Since the importance weight $\widehat{w}^{(m)}$ is obtained by solving the finite-sample version

$$\widehat{\mu}_{\widehat{h}_{\text{CIP}}} = \widehat{C}_{\widehat{h}_{\text{CIP}}}^{(m)} \widehat{w}^{(m)}, \tag{49}$$

combining Eq. (48) and Eq. (49), we have

$$\underbrace{\mu_{\widehat{h}_{\text{CIP}}} - \widehat{\mu}_{\widehat{h}_{\text{CIP}}}}_{(A_1)}$$
$$= C_{\widehat{h}_{\text{CIP}}}^{(m)} \left(w^{(m)} - \widehat{w}^{(m)} \right) + \underbrace{\left(C_{\widehat{h}_{\text{CIP}}}^{(m)} - \widehat{C}_{\widehat{h}_{\text{CIP}}}^{(m)} \right)}_{(A_2)} \widehat{w}^{(m)} + \underbrace{\Delta\mu^{\diamond}}_{(A_3)}. \tag{50}$$

60

Now we bound $(A_1), (A_2)$, and (A_3) separately. For bouding (A_1) and (A_2), we closely follow the proof of Theorem 3 in Lipton et al. (2018).

Term (A_1) By Hoeffding's inequality, we have

$$\mathbb{P}\{|A_{1,i}| > t\} = \mathbb{P}\left\{\left|\frac{1}{n^{(\mathfrak{T})}}\sum_{k=1}^{n^{(\mathfrak{T})}}\mathbf{1}_{\widehat{h}_{\mathrm{CIP}}(\widetilde{X}_k^{(\mathfrak{T})})=i} - \mathbb{E}\left[\mathbf{1}_{\widehat{h}_{\mathrm{CIP}}(X^{(\mathfrak{T})})=i}\right]\right| > t\right\} \le 2\exp\left\{-2n^{(\mathfrak{T})}t^2\right\},$$

for all $t > 0$. Therefore, for any $t_1 > 0$, we have

$$\mathbb{P}\{\|A_1\|_2 > t_1\} \le \mathbb{P}\left\{\exists i \in \{1,...,L\}, |A_{1,i}| > \frac{t_1}{\sqrt{L}}\right\}$$

$$\le \sum_{i=1}^{L}\mathbb{P}\left\{|A_{1,i}| > \frac{t_1}{\sqrt{L}}\right\}$$

$$\le 2L\cdot\exp\left\{-\frac{2n^{(\mathfrak{T})}t_1^2}{L}\right\}.$$

Term (A_2) Note that

$$C_{\widehat{h}_{\mathrm{CIP}}}^{(m)} - \widehat{C}_{\widehat{h}_{\mathrm{CIP}}}^{(m)} = \frac{1}{n^{(m)}}\sum_{k=1}^{n^{(m)}}e_{\widehat{h}_{\mathrm{CIP}}(\widetilde{X}_k^{(m)})}e_{\widetilde{Y}_k^{(m)}}^{\top} - \mathbb{E}\left[e_{\widehat{h}_{\mathrm{CIP}}(X^{(m)})}e_{Y^{(m)}}^{\top}\right],$$

where e_i denotes the zero vector with only the i-th element being 1. Let

$$Z_k = e_{\widehat{h}_{\mathrm{CIP}}(\widetilde{X}_k^{(m)})}e_{\widetilde{Y}_k^{(m)}}^{\top} - \mathbb{E}\left[e_{\widehat{h}_{\mathrm{CIP}}(X^{(m)})}e_{Y^{(m)}}^{\top}\right].$$

Then it is straightforward to see that

$$\begin{cases}\mathbb{E}[Z_k] = 0;\\ \|Z_k\|_2 \le 2;\\ \max\left\{\|\mathbb{E}[Z_kZ_k^{\top}]\|_2, \|\mathbb{E}[Z_k^{\top}Z_k]\|_2\right\} \le 1.\end{cases}$$

By the matrix Bernstein inequality (see e.g. (Tropp, 2015, Theorem 6.1.1)), we have

$$\mathbb{P}\{\|A_2\|_2 \ge t_2\} \le 2L\exp\left\{-\frac{n^{(m)}t_2^2/2}{1 + 2t_2/3}\right\},$$

for all $t_2 > 0$.

Term (A_3) By definition of $D_{\mathcal{G}}(\cdot, \cdot)$ and $\Psi_{\mathcal{G}, \widehat{\phi}_{\mathrm{CIP}}}$, each element of A_3 satisfies

$$|A_{3,i}| \le \sum_{j=1}^{L}|\delta_{i,j}|\cdot\mathbb{P}\left\{Y^{(\mathfrak{T})} = j\right\}$$

61

$$\leq \sum_{j=1}^{L} D_{\mathcal{G}} \left(\mathcal{P}_{\widehat{\phi}_{\mathrm{CIP}}(X)|Y=j}^{(\mathfrak{T})}, \mathcal{P}_{\widehat{\phi}_{\mathrm{CIP}}(X)|Y=j}^{(m)} \right) \cdot \mathbb{P} \left\{ Y^{(\mathfrak{T})} = j \right\} \leq \Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}}.$$

Hence we have

$$\|A_3\|_2 = \sqrt{\sum_{i=1}^{L} |A_{3,i}|^2} \leq \sqrt{L} \Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}}.$$

Applying these bounds to Eq. (50) and using the assumption that $C_{\widehat{h}_{\mathrm{CIP}}}^{(m)}$ has condition number $\kappa^{(m)}$ (i.e., the smallest eigenvalue $\geq 1/\kappa^{(m)}$ because 1 is an eigenvalue of the matrix), we get

$$\left\| w^{(m)} - \widehat{w}^{(m)} \right\|_2 \leq \kappa_m \left(t_1 + t_2 \cdot \left\| \widehat{w}^{(m)} \right\|_2 + \sqrt{L} \Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}} \right)$$
$$\leq \kappa_m \left(t_1 + t_2 \left(\left\| w^{(m)} - \widehat{w}^{(m)} \right\|_2 + \left\| w^{(m)} \right\|_2 \right) + \sqrt{L} \Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}} \right),$$

with probability at least $1 - 2L \exp \left(-2n^{(\mathfrak{T})} t_1^2/L \right) - 2L \exp \left(-3n^{(m)} t_2^2/(6 + 4t_2) \right)$. Finally, let $t_1 = \sqrt{L \log(4L/\delta)/2n^{(\mathfrak{T})}}$ and $t_2 = \sqrt{3 \log(4L/\delta)/n^{(m)}}$. Using the assumption $n^{(m)} \geq 12\kappa_m^2 \log(4L/\delta)$, we get

$$\left\| w^{(m)} - \widehat{w}^{(m)} \right\|_2 \leq 2\kappa_m \left(\sqrt{L} \Psi_{\mathcal{G},\widehat{\phi}_{\mathrm{CIP}}} + \sqrt{\frac{L \log(4L/\delta)}{2n^{(\mathfrak{T})}}} + \sqrt{\frac{3 \log(4L/\delta)}{n^{(m)}}} \left\| w^{(m)} \right\|_2 \right)$$

with probability at least $1 - \delta$. Combining the results then proves the proposition.

B.6 Proof of Proposition 9

Using assumption (25), for any positive semidefinite matrix $S \in \mathbb{R}^{d_y \times d_y}$, we have

$$\left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right)^\top S \left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right) = \mathrm{trace} \left(\left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right)^\top S^{\frac{1}{2}} S^{\frac{1}{2}} \left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right) \right)$$
$$= \mathrm{trace} \left(S^{\frac{1}{2}} \left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right) \left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right)^\top S^{\frac{1}{2}} \right)$$
$$\leq \mathrm{trace} \left(S^{\frac{1}{2}} \left(\frac{\zeta_y}{M-1} \sum_{m=2}^{M} v_y^{(m)} (v_y^{(m)})^\top \right) S^{\frac{1}{2}} \right)$$
$$= \frac{\zeta_y}{M-1} \sum_{m=2}^{M} \mathrm{trace} \left(S^{\frac{1}{2}} v_y^{(m)} (v_y^{(m)})^\top S^{\frac{1}{2}} \right)$$
$$= \frac{\zeta_y}{M-1} \sum_{m=2}^{M} v_y^{(m)\top} S v_y^{(m)}. \tag{51}$$

Inequality above uses the fact that $\mathrm{trace}(A) \leq \mathrm{trace}(B)$ for positive semi-definite matrix A, B satisfying $A \preceq B$. Given that $\epsilon^{(m)}, \epsilon^{(\mathfrak{T})} \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(0, \Sigma)$, for any $y \in \{1, 2, \ldots, L\}$ and

$m \in \{1, 2, \ldots, M\}$ we have

$$
\begin{aligned}
\phi(X^{(m)}) \mid Y^{(m)} = y &\sim \mathcal{N}\left(A\left(f^{(1)}(y) + P_y v_y^{(m)}\right) + b, A\Sigma A^\top\right), \\
\phi(X^{(\mathfrak{T})}) \mid Y^{(\mathfrak{T})} = y &\sim \mathcal{N}\left(A\left(f^{(1)}(y) + P_y v_y^{(\mathfrak{T})}\right) + b, A\Sigma A^\top\right).
\end{aligned}
\tag{52}
$$

Note that $v_y^{(1)} = 0$ by Assumption 1, so we can further simplify terms in Eq. (23) as

$$
\Delta_\phi^{(m)}(y) = AP_y\left(v_y^{(m)} - v_y^{(1)}\right) = AP_y v_y^{(m)} \quad \text{and} \quad \Sigma_\phi^{(m)}(y) = A\Sigma A^\top.
$$

With results above in hand, we can derive

$$
\begin{aligned}
&D_{\mathcal{G}}^2\left(\mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}, \mathcal{P}_{\phi(X)|Y=y}^{(m)}\right) \\
&\leq \sup_{g \in \mathcal{G}} \max_{y'=1,2,\ldots,L} \left| \mathbb{E}_{Z \sim \mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}}\left[\mathbf{1}_{g(Z)=y'}\right] - \mathbb{E}_{Z \sim \mathcal{P}_{\phi(X)|Y=y}^{(m)}}\left[\mathbf{1}_{g(Z)=y'}\right] \right| \\
&\leq \sup_{g \in \mathcal{G}} \sum_{y'=1}^{L} \left| \mathbb{E}_{Z \sim \mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}}\left[\mathbf{1}_{g(Z)=y'}\right] - \mathbb{E}_{Z \sim \mathcal{P}_{\phi(X)|Y=y}^{(m)}}\left[\mathbf{1}_{g(Z)=y'}\right] \right| \\
&= 4\sup_{g \in \mathcal{G}} \mathrm{TV}^2\left(\mathcal{P}_{g(\phi(X))|Y=y}^{(\mathfrak{T})}, \mathcal{P}_{g(\phi(X))|Y=y}^{(m)}\right) \\
&\overset{(i)}{\leq} 2\sup_{g \in \mathcal{G}} \mathrm{KL}\left(\mathcal{P}_{g(\phi(X))|Y=y}^{(\mathfrak{T})}, \mathcal{P}_{g(\phi(X))|Y=y}^{(m)}\right) \\
&\overset{(ii)}{\leq} 2\mathrm{KL}\left(\mathcal{P}_{\phi(X)|Y=y}^{(\mathfrak{T})}, \mathcal{P}_{\phi(X)|Y=y}^{(m)}\right) \\
&\overset{(iii)}{=} \left(AP_y\left(v_y^{(\mathfrak{T})} - v_y^{(m)}\right)\right)^\top \left(A\Sigma A^\top\right)^{-1}\left(AP_y\left(v_y^{(\mathfrak{T})} - v_y^{(m)}\right)\right) \\
&\overset{(iv)}{\leq} \frac{\zeta_y}{M-1} \sum_{m=2}^{M} \left(AP_y v_y^{(m)}\right)^\top \left(A\Sigma A^\top\right)^{-1}\left(AP_y v_y^{(m)}\right) \\
&\overset{(v)}{=} 2\zeta_y \Pi_\phi(y).
\end{aligned}
\tag{53}
$$

Here step (i) applies Pinsker's inequality and step (ii) applies data processing inequality. Step (iii) applies the KL divergence formula between two Gaussians, i.e.,

$$
\mathrm{KL}\left(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_1, \Sigma_2)\right) = \frac{1}{2}\left(\log \frac{|\Sigma_2|}{\Sigma_1}\right) - p + \mathrm{Tr}\left(\Sigma_2^{-1}\Sigma_1\right) + (\mu_1 - \mu_2)^\top \Sigma_2^{-1}(\mu_2 - \mu_1).
$$

Step (iv) uses Eq. (51) with $S = P_y^\top A^\top (A\Sigma A^\top)^{-1}AP_y$ and the last step (vi) follows from definition (24). Now the proposition follows by applying Eq. (53) to the definition of $\Psi_{\mathcal{G},\phi}$ in Eq. (21).

B.7 Number of source domains needed for controlling the deviation from conditional invariance

To validate condition (25) in Proposition 9, we prove the following lemma which states that when the perturbations are generated randomly, the required number of source domains is approximately linear in the dimension of perturbations.

Lemma 17 *Under Assumption 1, further assume that perturbations are generated according to $v_y^{(m)}, v_y^{(\mathfrak{T})} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \tau_y^2 \mathbb{I}_{d_y})$. For any $\delta \in (0,1)$, if the number of source domains M satisfies*

$$M \geq 1 + r \left(\max_y \sqrt{d_y} + \sqrt{2 \log(2L/\delta)} \right)^2,$$

for some constant $r > 1$, then with probability at least $1 - \delta$,[14] condition (25) holds with $\zeta_y = c_r(M + \log M)$ for some constant $c_r > 0$.

Proof We verify that when $v_y^{(m)}, v_y^{(\mathfrak{T})} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \tau_y^2 \mathbb{I}_{d_y})$, condition (25) holds with high probability. Sincet $M \geq \max_{y=1,2,\dots,L}\{d_y\}+1$, Theorem 6.1 and Example 2.11 in Wainwright (2019) show that given $y \in \{1, 2, \dots, L\}$, for any $t_0 \in (0,1), t_1, \dots, t_M > 0$ we have

$$\mathbb{P}\left\{ \lambda_{\min}\left(\frac{1}{M-1} \sum_{m=2}^{M} v_y^{(m)} v_y^{(m)\top} \right) \geq \tau_y^2 \left(1 - t_0 - \sqrt{\frac{d_y}{M-1}} \right) \right\} \geq 1 - e^{-(M-1)t_0^2/2},$$

and

$$\mathbb{P}\left\{ \left\| v_y^{(\mathfrak{T})} - v_y^{(1)} \right\|_2^2 \leq \tau_y^2 d_y (1 + t_1) \right\} \geq 1 - e^{-d_y \min\{t_1, t_1^2\}/8},$$

$$\mathbb{P}\left\{ \left\| v_y^{(\mathfrak{T})} - v_y^{(m)} \right\|_2^2 \leq 2\tau_y^2 d_y (1 + t_m) \right\} \geq 1 - e^{-d_y \min\{t_m, t_m^2\}/8}, \quad \forall m \in \{2, \cdots, M\}.$$

Take $t_0 = \sqrt{\frac{2}{M-1} \log \frac{2L}{\delta}}$ and $t_m = \frac{8}{d_y} \log(\frac{2ML}{\delta}) + \sqrt{\frac{8}{d_y} \log(\frac{2ML}{\delta})}$ for all y such that $d_y > 0$. Applying union bound, the following bound holds for all $y \in \{1, 2, \dots, L\}$ and $m \in \{1, 2, \dots, M\}$ with probability greater than $1 - \delta$:

$$\lambda_{\max}\left(\left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right) \left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right)^\top \right) = \left\| \left(v_y^{(\mathfrak{T})} - v_y^{(m)} \right) \right\|_2^2$$

$$\leq 2\tau_y^2 \left(d_y + 8 \log(2ML/\delta) + \sqrt{8 d_y \log(2ML/\delta)} \right)$$

$$\leq 3\tau_y^2 \left(d_y + 8 \log(2ML/\delta) \right)$$

$$\leq \frac{3 \left(d_y + 8 \log(2ML/\delta) \right)}{1 - \sqrt{2 \left(\log(2L/\delta) \right) / (M-1)} - \sqrt{d_y/(M-1)}} \lambda_{\min}\left(\frac{1}{M-1} \sum_{m=2}^{M} v_y^{(m)} v_y^{(m)\top} \right)$$

$$\leq \frac{3r}{r-1} \left(d_y + 8 \log\left(\frac{2ML}{\delta} \right) \right) \lambda_{\min}\left(\frac{1}{M-1} \sum_{m=2}^{M} v_y^{(m)} v_y^{(m)\top} \right)$$

$$\leq c_r(M + \log M) \cdot \lambda_{\min}\left(\frac{1}{M-1} \sum_{m=2}^{M} v_y^{(m)} v_y^{(m)\top} \right),$$

where we apply the lower bound on M in the last two steps. ∎

---

14. The probability is with respect to the randomness of perturbations $v_y^{(m)}, v_y^{(\mathfrak{T})} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \tau_y^2 \mathbb{I}_{d_y})$.

B.8 Proof of Proposition 10

Let $f_{\mathrm{inv}}(1), f_{\mathrm{inv}}(2) \in \mathbb{R}^{p-d}$ denote the vector of the last $p-d$ coordinates of $f^{(1)}(1), f^{(1)}(2) \in \mathbb{R}^p$ respectively, which are conditionally invariant across source and target. We first prove that the target risks of h_{oracle} and $h^\star$ are given by

$$
\begin{aligned}
\mathcal{R}^{(\mathfrak{T})}(h_{\mathrm{oracle}}) &= r_{p_1^{(\mathfrak{T})}, \sigma}\left(\left\|f^{(\mathfrak{T})}(1) - f^{(\mathfrak{T})}(2)\right\|_2\right), \\
\mathcal{R}^{(\mathfrak{T})}(h^\star) &= r_{p_1^{(\mathfrak{T})}, \sigma}\left(\|f_{\mathrm{inv}}(1) - f_{\mathrm{inv}}(2)\|_2\right),
\end{aligned}
\tag{54}
$$

where $r_{p_1^{(\mathfrak{T})}, \sigma}(x)$ is a continuous decreasing function in x. The risk of the oracle classifier can be derived as follows. For simplicity we write $\mu_1^{(\mathfrak{T})} = f^{(\mathfrak{T})}(1)$ and $\mu_2^{(\mathfrak{T})} = f^{(\mathfrak{T})}(2)$. For any $\phi(x) = \beta^\top x + \beta_0$ with $\|\beta\|_2 = 1$, we know that

$$
\begin{aligned}
\phi(X^{(\mathfrak{T})}) \mid Y^{(\mathfrak{T})} = 1 &\sim \mathcal{N}\left(\beta^\top \mu_1^{(\mathfrak{T})} + \beta_0, \sigma^2\right), \\
\phi(X^{(\mathfrak{T})}) \mid Y^{(\mathfrak{T})} = 2 &\sim \mathcal{N}\left(\beta^\top \mu_2^{(\mathfrak{T})} + \beta_0, \sigma^2\right).
\end{aligned}
$$

Then the target risk of $h = g \circ \phi$ can be computed by

$$
\begin{aligned}
\mathcal{R}^{(\mathfrak{T})}(h) &= \mathbb{E}\left[\mathbf{1}_{h(X^{(\mathfrak{T})}) \neq Y^{(\mathfrak{T})}}\right] \\
&= \mathbb{P}\left\{h(X^{(\mathfrak{T})}) = 2 \mid Y^{(\mathfrak{T})} = 1\right\} \mathbb{P}\left\{Y^{(\mathfrak{T})} = 1\right\} \\
&\qquad\qquad + \mathbb{P}\left\{h(X^{(\mathfrak{T})}) = 1 \mid Y^{(\mathfrak{T})} = 2\right\} \mathbb{P}\left\{Y^{(\mathfrak{T})} = 2\right\} \\
&= \mathbb{P}\left\{\phi(X^{(\mathfrak{T})}) > 0 \mid Y^{(\mathfrak{T})} = 1\right\} \mathbb{P}\left\{Y^{(\mathfrak{T})} = 1\right\} \\
&\qquad\qquad + \mathbb{P}\left\{\phi(X^{(\mathfrak{T})}) \leq 0 \mid Y^{(\mathfrak{T})} = 2\right\} \mathbb{P}\left\{Y^{(\mathfrak{T})} = 2\right\} \\
&= p_1^{(\mathfrak{T})} \cdot \frac{1}{2}\left(1 - \mathrm{erf}\left(\frac{\beta^\top \mu_1^{(\mathfrak{T})} + \beta_0}{\sqrt{2\sigma^2}}\right)\right) + (1 - p_1^{(\mathfrak{T})}) \cdot \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{\beta^\top \mu_2^{(\mathfrak{T})} + \beta_0}{\sqrt{2\sigma^2}}\right)\right) \\
&= \frac{1}{2}\left(1 - p_1^{(\mathfrak{T})} \cdot \mathrm{erf}\left(\frac{\beta^\top \mu_1^{(\mathfrak{T})} + \beta_0}{\sqrt{2\sigma^2}}\right) + (1 - p_1^{(\mathfrak{T})}) \cdot \mathrm{erf}\left(\frac{\beta^\top \mu_2^{(\mathfrak{T})} + \beta_0}{\sqrt{2\sigma^2}}\right)\right),
\end{aligned}
\tag{55}
$$

where $\mathrm{erf}(x) = 2\int_0^x e^{-t^2} dt / \sqrt{\pi}$ is the Gauss error function. The optimization problem of h_{oracle} in Eq. (3) becomes

$$
\begin{aligned}
\beta_{\mathrm{orcale}}, \beta_{0,\mathrm{orcale}} = \underset{\beta, \beta_0}{\arg\min} \quad & -p_1^{(\mathfrak{T})} \cdot \mathrm{erf}\left(\frac{\beta^\top \mu_1^{(\mathfrak{T})} + \beta_0}{\sqrt{2\sigma^2}}\right) + (1 - p_1^{(\mathfrak{T})}) \cdot \mathrm{erf}\left(\frac{\beta^\top \mu_2^{(\mathfrak{T})} + \beta_0}{\sqrt{2\sigma^2}}\right), \\
\text{s.t.} \quad & \|\beta\|_2 = 1.
\end{aligned}
\tag{56}
$$

Setting the gradient of the objective function with respect to β_0 to 0, we get

$$
\beta_0 = \frac{\sigma^2}{\beta^\top(\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})})} \log \frac{p_1^{(\mathfrak{T})}}{1 - p_1^{(\mathfrak{T})}} - \frac{1}{2}\beta^\top(\mu_1^{(\mathfrak{T})} + \mu_2^{(\mathfrak{T})}).
$$

Plugging the expression of β_0 into Eq. (56) and using the method of Lagrange multipliers, we arrive at

$$\frac{\partial}{\partial \beta}\left(-p_1^{(\mathfrak{T})} \cdot \operatorname{erf}\left(\frac{\beta^\top \mu_1^{(\mathfrak{T})} + \beta_0}{\sqrt{2\sigma^2}}\right) + (1 - p_1^{(\mathfrak{T})}) \cdot \operatorname{erf}\left(\frac{\beta^\top \mu_2^{(\mathfrak{T})} + \beta_0}{\sqrt{2\sigma^2}}\right) + \lambda(\|\beta\|_2^2 - 1)\right) = 0$$

$$\implies 2\lambda\beta = p_1^{(\mathfrak{T})} \frac{2}{\sqrt{\pi}} \exp\left(-\frac{(\beta^\top \mu_1^{(\mathfrak{T})} + \beta_0)^2}{2\sigma^2}\right)$$

$$\times \left(\frac{1}{2\sqrt{2\sigma^2}}(\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}) - \frac{\sqrt{\sigma^2/2} \log \frac{p_1^{(\mathfrak{T})}}{1-p_1^{(\mathfrak{T})}}}{(\beta^\top(\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}))^2}(\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})})\right)$$

$$- (1 - p_1^{(\mathfrak{T})})p_1^{(\mathfrak{T})} \frac{2}{\sqrt{\pi}} \exp\left(-\frac{(\beta^\top \mu_2^{(\mathfrak{T})} + \beta_0)^2}{2\sigma^2}\right)$$

$$\times \left(-\frac{1}{2\sqrt{2\sigma^2}}(\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}) - \frac{\sqrt{\sigma^2/2} \log \frac{p_1^{(\mathfrak{T})}}{1-p_1^{(\mathfrak{T})}}}{(\beta^\top(\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}))^2}(\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})})\right).$$

That is, β and $\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}$ are in the same direction. Using $\|\beta\|_2 = 1$, we get

$$\beta_{\text{oracle}} = \frac{\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}}{\left\|\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}\right\|_2},$$

$$\beta_{0,\text{oracle}} = \frac{\sigma^2}{\left\|\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}\right\|_2} \log \frac{p_1^{(\mathfrak{T})}}{1 - p_1^{(\mathfrak{T})}} - \frac{\left\|\mu_1^{(\mathfrak{T})}\right\|_2^2 - \left\|\mu_2^{(\mathfrak{T})}\right\|_2^2}{2\left\|\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}\right\|_2}.$$

Plugging the above equations into Eq. (55), after some algebra we get the oracle target risk as

$$\mathcal{R}^{(\mathfrak{T})}(h_{\text{oracle}}) = r_{p_1^{(\mathfrak{T})},\sigma}\left(\left\|\mu_1^{(\mathfrak{T})} - \mu_2^{(\mathfrak{T})}\right\|_2\right) = r_{p_1^{(\mathfrak{T})},\sigma}\left(\left\|f^{(\mathfrak{T})}(1) - f^{(\mathfrak{T})}(2)\right\|_2\right),$$

where

$$r_{p_1^{(\mathfrak{T})},\sigma}(x) = \frac{1}{2}\left(1 - p_1^{(\mathfrak{T})}\operatorname{erf}\left(\frac{x}{2\sqrt{2}\sigma} + \frac{\sigma}{\sqrt{2}x} \log \frac{p_1^{(\mathfrak{T})}}{1 - p_1^{(\mathfrak{T})}}\right)\right.$$
$$\left. - (1 - p_1^{(\mathfrak{T})})\operatorname{erf}\left(\frac{x}{2\sqrt{2}\sigma} - \frac{\sigma}{\sqrt{2}x} \log \frac{p_1^{(\mathfrak{T})}}{1 - p_1^{(\mathfrak{T})}}\right)\right). \tag{57}$$

Next, turning to $h^\star$, note that $span\{v_1^{(2)}, v_1^{(3)}, \ldots, v_1^{(M)}, v_2^{(2)}, v_2^{(3)}, \ldots, v_2^{(M)}\} = \mathbb{R}^d$, and so the constraint of conditional invariance in Eq. (5) requires that the first d coordinates of $\beta^\star$ are zero, i.e., $\phi^\star$ only uses the last $p - d$ invariant coordinates. Following exactly the same proof as above, we have

$$\mathcal{R}^{(\mathfrak{T})}(h^\star) = r_{p_1^{(\mathfrak{T})},\sigma}\left(\|f_{\text{inv}}(1) - f_{\text{inv}}(2)\|_2\right).$$

Now we verify that $r_{p_1^{(\mathfrak{T})},\sigma}(x)$ is a decreasing function in x, by calculating its gradient.

$$\frac{dr_{p_1^{(\mathfrak{T})},\sigma}(x)}{dx}$$

$$= -\frac{1}{\sqrt{\pi}}p_1^{(\mathfrak{T})}\exp\left(-\left(\frac{x}{2\sqrt{2}\sigma}+\frac{\sigma}{\sqrt{2}x}\log\frac{p_1^{(\mathfrak{T})}}{1-p_1^{(\mathfrak{T})}}\right)^2\right)$$

$$\times\left(\frac{1}{2\sqrt{2}\sigma}-\frac{\sigma}{\sqrt{2}x^2}\log\frac{p_1^{(\mathfrak{T})}}{1-p_1^{(\mathfrak{T})}}\right)$$

$$-\frac{1}{\sqrt{\pi}}(1-p_1^{(\mathfrak{T})})\exp\left(-\left(\frac{x}{2\sqrt{2}\sigma}-\frac{\sigma}{\sqrt{2}x}\log\frac{p_1^{(\mathfrak{T})}}{1-p_1^{(\mathfrak{T})}}\right)^2\right)$$

$$\times\left(\frac{1}{2\sqrt{2}\sigma}+\frac{\sigma}{\sqrt{2}x^2}\log\frac{p_1^{(\mathfrak{T})}}{1-p_1^{(\mathfrak{T})}}\right)$$

$$= -\sqrt{\frac{p_1^{(\mathfrak{T})}(1-p_1^{(\mathfrak{T})})}{2\pi\sigma^2}}\exp\left(-\left(\frac{x}{2\sqrt{2}\sigma}\right)^2-\left(\frac{\sigma}{\sqrt{2}x}\log\frac{p_1^{(\mathfrak{T})}}{1-p_1^{(\mathfrak{T})}}\right)^2\right)<0.$$

This intermediate result implies that the risk difference between $h^\star$ and h_{oracle} originates from the norm differences between $f^{(\mathfrak{T})}(1)-f^{(\mathfrak{T})}(2)$ and $f_{\mathrm{inv}}(1)-f_{\mathrm{inv}}(2)$ only, where the second term is equivalent to the last $p-d$ dimensions of the first term which corresponds to CICs in the general anticausal model. Intuitively, when the dimension of CICs is close to p, the discrepancy between the norms of $f^{(\mathfrak{T})}(1)-f^{(\mathfrak{T})}(2)$ and $f_{\mathrm{inv}}(1)-f_{\mathrm{inv}}(2)$ becomes insignificant, and the disparity in the target risks of $h^\star$ and h_{oracle} tends to be negligible. In particular, we establish the explicit bound for this disparity as follows:

When $p_1^{(\mathfrak{T})}=1/2$, the explicit formula for $r_{p_1^{(\mathfrak{T})},\sigma}(x)$ given in Eq.(57) shows that $r_{p_1^{(\mathfrak{T})},\sigma}(x)=\left(1-\mathrm{erf}\left(x/(2\sqrt{2}\sigma)\right)\right)/2$. Since $v_1^{(\mathfrak{T})}-v_2^{(\mathfrak{T})}\sim\mathcal{N}(0,2\tau^2\sigma^2\mathbb{I}_d)$, the standard Gaussian concentration bound shows that

$$\mathbb{P}\left\{\left\|v_1^{(\mathfrak{T})}-v_2^{(\mathfrak{T})}\right\|_2\leq\sqrt{2\tau^2\sigma^2 d\left(1+\sqrt{\frac{8}{d}\log\frac{1}{\delta}}+\frac{8}{d}\log\frac{1}{\delta}\right)}\right\}\geq 1-\delta.$$

By using the target risks given in Eq. (54), we can calculate

$$\mathcal{R}^{(\mathfrak{T})}(h^\star)-\mathcal{R}^{(\mathfrak{T})}(h_{\mathrm{oracle}})$$

$$= \frac{1}{2}\left(\mathrm{erf}\left(\frac{\|f^{(\mathfrak{T})}(1)-f^{(\mathfrak{T})}(2)\|_2}{2\sqrt{2}\sigma}\right)-\mathrm{erf}\left(\frac{\|f_{\mathrm{inv}}(1)-f_{\mathrm{inv}}(2)\|_2}{2\sqrt{2}\sigma}\right)\right)$$

$$\leq \frac{1}{2}\left(\mathrm{erf}\left(\frac{\|f^{(1)}(1)-f^{(1)}(2)\|_2+\left\|v_1^{(\mathfrak{T})}-v_2^{(\mathfrak{T})}\right\|_2}{2\sqrt{2}\sigma}\right)-\mathrm{erf}\left(\frac{\|f_{\mathrm{inv}}(1)-f_{\mathrm{inv}}(2)\|_2}{2\sqrt{2}\sigma}\right)\right)$$

$$\overset{(i)}{=} \frac{1}{2}\left(\mathrm{erf}\left(\frac{\xi\sigma\sqrt{p}+\left\|v_1^{(\mathfrak{T})}-v_2^{(\mathfrak{T})}\right\|_2}{2\sqrt{2}\sigma}\right)-\mathrm{erf}\left(\frac{\xi\sqrt{p-d}}{2\sqrt{2}}\right)\right)$$

$$\leq \frac{1}{2}\left(\operatorname{erf}\left(\frac{\xi\sqrt{p}+\tau\sqrt{3d+24\log(1/\delta)}}{2\sqrt{2}}\right)-\operatorname{erf}\left(\frac{\xi\sqrt{p-d}}{2\sqrt{2}}\right)\right)$$

$$\overset{(ii)}{\leq} \frac{\xi(\sqrt{p}-\sqrt{p-d})+\tau\sqrt{3d+24\log(1/\delta)}}{\sqrt{8\pi}}\exp\left(-\frac{\xi^2(p-d)}{8}\right)$$

$$= \frac{1}{\sqrt{8\pi}}\left(\frac{cd}{\sqrt{p}+\sqrt{p-d}}+\tau\sqrt{3d+24\log\left(\frac{1}{\delta}\right)}\right)\exp\left(-\frac{\xi^2(p-d)}{8}\right)$$

$$\leq \frac{1}{\sqrt{8\pi}}\left(\xi\sqrt{d}+\tau\sqrt{3d+24\log\left(\frac{1}{\delta}\right)}\right)\exp\left(-\frac{\xi^2(p-d)}{8}\right)$$

$$\leq c_{\xi,\tau}\left(\sqrt{d}+\sqrt{\log(1/\delta)}\right)\exp\left(-\frac{\xi^2(p-d)}{8}\right),$$

with probability at least $1-\delta$. Here, in step (i), we use the fact that $f_{\mathrm{inv}}(1)$ and $f_{\mathrm{inv}}(2)$ are the last $p-d$ invariant coordinates of $f^{(1)}(1)$ and $f^{(1)}(2)$ respectively; and in step (ii) we use the inequality $\int_a^b e^{-t^2}dt \leq e^{-a^2}(b-a)$ for $0 < a < b$. This completes the proof of the proposition.

Appendix C. Proof of Section 4

In this section we prove our theorems given in Section 4.

C.1 Proof of Theorem 12

We prove the following generalized version of the theorem, which does not require that ϕ_{inv} is exactly conditionally invariant. That is, for any $h_{\mathrm{inv}} = g_{\mathrm{inv}} \circ \phi_{\mathrm{inv}}$ where $g_{\mathrm{inv}} \in \mathcal{G}$ and $\phi_{\mathrm{inv}} \in \Phi$, for any other classifier h we have

$$\left|\left[\mathcal{R}^{(\mathfrak{T})}(h)-\mathcal{R}^{(1)}(h)\right]-\left[\mathbb{P}\left\{h(X^{(\mathfrak{T})})\neq h_{\mathrm{inv}}(X^{(\mathfrak{T})})-\mathbb{P}\left\{h(X^{(1)})\neq h_{\mathrm{inv}}(X^{(1)})\right\}\right\}\right]\right|$$
$$\leq 2\mathcal{R}^{(1)}(h_{\mathrm{inv}})+L\Psi_{\mathcal{G},\phi_{\mathrm{inv}}}.$$

If this inequality holds, when ϕ_{inv} is a conditionally invariant feature mapping across source and target domains, we get $\Psi_{\mathcal{G},\phi_{\mathrm{inv}}} = 0$ and arrive at the original Theorem 12.

Proof Define $\delta_{ij} = \mathbb{P}\left\{h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j \mid Y^{(\mathfrak{T})}=i\right\}-\mathbb{P}\left\{h_{\mathrm{inv}}(X^{(1)})=j \mid Y^{(1)}=i\right\}$, then $|\delta_{ij}| \leq \Psi_{\mathcal{G},h_{\mathrm{inv}}}$. Because there is no label shift, for any $i,j \in \{1,2,\dots,L\}$ we have

$$\mathbb{P}\left\{Y^{(\mathfrak{T})}=i,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}-\mathbb{P}\left\{Y^{(1)}=i,h_{\mathrm{inv}}(X^{(1)})=j\right\}=\delta_{ij}\cdot\mathbb{P}\left\{Y^{(\mathfrak{T})}=i\right\}.$$

As a result, we get

$$\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}-\mathbb{P}\left\{h(X^{(1)})=j,h_{\mathrm{inv}}(X^{(1)})=j\right\}$$

$$=\sum_{i=1}^{L}\left[\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j \mid Y^{(\mathfrak{T})}=i,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}\cdot\mathbb{P}\left\{Y^{(\mathfrak{T})}=i,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}\right.$$

$$-\mathbb{P}\left\{h(X^{(1)})=j\ \middle|\ Y^{(1)}=i,h_{\mathrm{inv}}(X^{(1)})=j\right\}\cdot\mathbb{P}\left\{Y^{(1)}=i,h_{\mathrm{inv}}(X^{(1)})=j\right\}\Bigg]$$

$$=\sum_{i=1}^{L}\Bigg(\Bigg[\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}=i,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}$$

$$-\mathbb{P}\left\{h(X^{(1)})=j\ \middle|\ Y^{(1)}=i,h_{\mathrm{inv}}(X^{(1)})=j\right\}\Bigg]\cdot\mathbb{P}\left\{Y^{(1)}=i,h_{\mathrm{inv}}(X^{(1)})=j\right\}$$

$$+\delta_{ij}\cdot\mathbb{P}\left\{Y^{(\mathfrak{T})}=i\right\}\cdot\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}=i,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}\Bigg). \tag{58}$$

Similarly we can calculate

$$\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j,Y^{(\mathfrak{T})}=j\right\}-\mathbb{P}\left\{h(X^{(1)})=j,Y^{(1)}=j\right\}$$

$$=\sum_{i=1}^{L}\Bigg[\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}=j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=i\right\}\cdot\mathbb{P}\left\{Y^{(\mathfrak{T})}=j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=i\right\}$$

$$-\mathbb{P}\left\{h(X^{(1)})=j\ \middle|\ Y^{(1)}=j,h_{\mathrm{inv}}(X^{(1)})=i\right\}\cdot\mathbb{P}\left\{Y^{(1)}=j,h_{\mathrm{inv}}(X^{(1)})=i\right\}\Bigg]$$

$$=\sum_{i=1}^{L}\Bigg(\Bigg[\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}=j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=i\right\}$$

$$-\mathbb{P}\left\{h(X^{(1)})=j\ \middle|\ Y^{(1)}=j,h_{\mathrm{inv}}(X^{(1)})=i\right\}\Bigg]\cdot\mathbb{P}\left\{Y^{(1)}=j,h_{\mathrm{inv}}(X^{(1)})=i\right\}$$

$$+\delta_{ji}\cdot\mathbb{P}\left\{Y^{(\mathfrak{T})}=j\right\}\cdot\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}=j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=i\right\}\Bigg). \tag{59}$$

Subtracting Eq. (59) from Eq. (58), we obtain

$$\Bigg[\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}-\mathbb{P}\left\{h(X^{(1)})=j,h_{\mathrm{inv}}(X^{(1)})=j\right\}\Bigg]$$

$$-\Bigg[\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j,Y^{(\mathfrak{T})}=j\right\}-\mathbb{P}\left\{h(X^{(1)})=j,Y^{(1)}=j\right\}\Bigg]$$

$$=\Bigg[\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}\neq j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}$$

$$-\mathbb{P}\left\{h(X^{(1)})=j\ \middle|\ Y^{(1)}\neq j,h_{\mathrm{inv}}(X^{(1)})=j\right\}\Bigg]\cdot B_j$$

$$-\Bigg[\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}=j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})\neq j\right\}$$

$$-\mathbb{P}\left\{h(X^{(1)})=j\ \middle|\ Y^{(1)}=j,h_{\mathrm{inv}}(X^{(1)})\neq j\right\}\Bigg]\cdot B_j'$$

$$+\sum_{i=1}^{L}\Bigg[\delta_{ij}\cdot\mathbb{P}\left\{Y^{(\mathfrak{T})}=i\right\}\cdot\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}=i,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=j\right\}$$

$$-\delta_{ji}\cdot\mathbb{P}\left\{Y^{(\mathfrak{T})}=j\right\}\cdot\mathbb{P}\left\{h(X^{(\mathfrak{T})})=j\ \middle|\ Y^{(\mathfrak{T})}=j,h_{\mathrm{inv}}(X^{(\mathfrak{T})})=i\right\}\Bigg] \tag{60}$$

where we write

$$B_j = \mathbb{P}\left\{Y^{(1)} \neq j, h_{\text{inv}}(X^{(1)}) = j\right\}, \quad B_j' = \mathbb{P}\left\{Y^{(1)} = j, h_{\text{inv}}(X^{(1)}) \neq j\right\}.$$

Now it follows that

$$\left| \left[\mathbb{P}\left\{h(X^{(1)}) \neq h_{\text{inv}}(X^{(1)})\right\} - \mathbb{P}\left\{h(X^{(\mathfrak{T})}) \neq h_{\text{inv}}(X^{(\mathfrak{T})})\right\} \right] - \left(\mathcal{R}^{(1)}(h) - \mathcal{R}^{(\mathfrak{T})}(h) \right) \right|$$

$$= \left| \left[\mathbb{P}\left\{h(X^{(\mathfrak{T})}) = h_{\text{inv}}(X^{(\mathfrak{T})})\right\} - \mathbb{P}\left\{h(X^{(1)}) = h_{\text{inv}}(X^{(1)})\right\} \right] \right.$$

$$\left. - \left[\mathbb{P}\left\{h(X^{(\mathfrak{T})}) = Y^{(\mathfrak{T})}\right\} - \mathbb{P}\left\{h(X^{(1)}) = Y^{(1)}\right\} \right] \right|$$

$$\overset{(i)}{\leq} \sum_{j=1}^{L} B_j + \sum_{j=1}^{L} B_j' + \left| \sum_{i=1}^{L} \sum_{j=1}^{L} \delta_{ij} \cdot \mathbb{P}\left\{Y^{(\mathfrak{T})} = i\right\} \right.$$

$$\cdot \left(\mathbb{P}\left\{h(X^{(\mathfrak{T})}) = j \mid Y^{(\mathfrak{T})} = i, h_{\text{inv}}(X^{(\mathfrak{T})}) = j\right\} \right.$$

$$\left. \left. - \mathbb{P}\left\{h(X^{(\mathfrak{T})}) = i \mid Y^{(\mathfrak{T})} = j, h_{\text{inv}}(X^{(\mathfrak{T})}) = i\right\} \right) \right|$$

$$\overset{(ii)}{\leq} 2\mathcal{R}^{(1)}(h_{\text{inv}}) + \sum_{i=1}^{L} \sum_{j=1}^{L} |\delta_{ij}| \cdot \mathbb{P}\left\{Y^{(\mathfrak{T})} = i\right\}$$

$$\overset{(iii)}{\leq} 2\mathcal{R}^{(1)}(h_{\text{inv}}) + \sum_{i=1}^{L} \sum_{j=1}^{L} \Psi_{\mathcal{G}, \phi_{\text{inv}}} \cdot \mathbb{P}\left\{Y^{(\mathfrak{T})} = i\right\}$$

$$= 2\mathcal{R}^{(1)}(h_{\text{inv}}) + L\Psi_{\mathcal{G}, \phi_{\text{inv}}},$$

where in step (i) we sum over $j \in \mathcal{Y} = \{1, 2, \ldots, L\}$ using Eq. (60), and in step (ii) we use $\sum_{j=1}^{L} B_j = \sum_{j=1}^{L} B_j' = \mathcal{R}^{(1)}(h_{\text{inv}})$. Step (iii) applies $|\delta_{ij}| \leq \Psi_{\mathcal{G}, \phi_{\text{inv}}}$. ∎

C.2 Proof of Theorem 15

For notation simplicity, for any $\phi \in \Phi$, we omit the bias term and write $\phi(x) = Ax$. All proof below holds after adding the bias term. And we use $u_1, \ldots, u_L$ and $\widetilde{u}_1, \ldots, \widetilde{u}_L$ to denote $f^{(1)}(1), \ldots, f^{(1)}(L)$ and $f^{(\mathfrak{T})}(1), \ldots, f^{(\mathfrak{T})}(L)$, respectively.

Part (a) By Assumption 3, for any $A \in \Phi$, the distributions of $AX^{(1)}$ and $AX^{(\mathfrak{T})}$ are given by the mixture distributions of the form

$$\mathcal{P}_{AX}^{(1)} = \sum_{j=1}^{L} \frac{1}{L} P_{A\epsilon}(Au_j), \quad \mathcal{P}_{AX}^{(\mathfrak{T})} = \sum_{j=1}^{L} \frac{1}{L} P_{A\epsilon}(A\widetilde{u}_j),$$

where $P_{A\epsilon}(u)$ denotes the distribution of $A\epsilon$ with mean shifted by a fixed vector u. Applying Lemma 18, the constraint in DIP, $\mathcal{P}_{AX}^{(1)} = \mathcal{P}_{AX}^{(\mathfrak{T})}$, implies that we can find a permutation π on $\{1, 2, \ldots, L\}$ such that

$$Au_j = A\widetilde{u}_{\pi(j)} \text{ for all } j \in \mathcal{Y}. \tag{61}$$

Hence any representations learned by DIP should meet the constraint given in Eq. (61) for some permutation π. Moreover, letting $\dim\left(\text{span}\{u_1 - \widetilde{u}_{\pi(1)}, \ldots, u_L - \widetilde{u}_{\pi(L)}\}\right) = k$, Lemma 19 shows that any matrix A satisfying Eq. (61) has the row space orthogonal to $\text{span}\{u_1 - \widetilde{u}_{\pi(1)}, \ldots, u_L - \widetilde{u}_{\pi(L)}\}$ and $\text{rank}(A) \leq p - k$.

Now consider all the pairs of (A_π, π) that meets (61) and the row space of A_π is the orthogonal complement to $\text{span}\{u_1 - \widetilde{u}_{\pi(1)}, \ldots, u_L - \widetilde{u}_{\pi(L)}\}$. By definition of DIP in Eq. (11), we obtain $\phi_{\text{DIP}} = A_{\pi_\star}$, where

$$\pi_\star = \underset{\{\pi \text{ is permutation}\}}{\arg\min} \ \underset{g \in \mathcal{G}, A_\pi}{\min} \ \mathcal{R}^{(1)}(g \circ A_\pi).$$

We can clearly find examples where $\pi_\star \neq \mathbf{I}$, for instance, as shown in Figure 1. This occurs as long as the label-flipping features offer lower source risk than the conditionally invariant features while it cannot generalize to the target domain. In this case, the conditional distributions of $\phi_{\text{DIP}}(X^{(1)})$ and $\phi_{\text{DIP}}(X^{(\mathfrak{T})})$, given the labels, are aligned after the target labels are permuted by $\pi_\star$.

Part (b) Next, we consider JointDIP defined in Eq. (32). If ϕ_{inv} is a linear mapping, say $\phi_{\text{inv}}(x) = Bx$ for some matrix $B \in \mathbb{R}^{r \times p}$, then writing $C = (A^\top, B^\top)^\top$ to denote the concatenation of A and B, we have

$$\mathcal{P}_{(\phi_{\text{j-DIP}}, \phi_{\text{inv}})(X)}^{(1)} = \mathcal{P}_{CX}^{(1)} = \sum_{j=1}^{L} \frac{1}{L} P_{C\epsilon}(Cu_j),$$

$$\mathcal{P}_{(\phi_{\text{j-DIP}}, \phi_{\text{inv}})(X)}^{(\mathfrak{T})} = \mathcal{P}_{CX}^{(\mathfrak{T})} = \sum_{j=1}^{L} \frac{1}{L} P_{C\epsilon}(C\widetilde{u}_j).$$

Under the JointDIP matching penalty, by Lemma 18 we similarly have

$$Cu_j = C\widetilde{u}_{\pi(j)} \text{ for all } j, \tag{62}$$

for some permutation π. If $\pi \neq \mathbf{I}$, Eq. (62) suggests that $Bu_j = B\widetilde{u}_{\pi(j)} = B\widetilde{u}_{j'}$ for some $j' \neq j$, which contradicts the condition (35). Therefore, we must have $\pi = \mathbf{I}$ (note that it is a valid solution because ϕ_{inv} is a CIC), which shows that $\phi_{\text{j-DIP}}$ is always conditionally invariant across source and target domains.

Moreover, given that there is no label shift and $\phi_{\text{j-DIP}}$ is conditionally invariant across $\mathcal{P}^{(1)}$ and $\mathcal{P}^{(\mathfrak{T})}$, we have $\mathcal{R}^{(\mathfrak{T})}(g \circ \phi_{\text{j-DIP}}) = \mathcal{R}^{(1)}(g \circ \phi_{\text{j-DIP}})$ for all $g \in \mathcal{G}$. Then the optimization objective of population JointDIP becomes equivalent to minimizing $\mathcal{R}^{(\mathfrak{T})}(g \circ \phi_{\text{j-DIP}})$. Since $\phi_{\text{inv}}^0 = (\phi_{\text{inv}}, \mathbf{0}_{q-r})$ is a feasible solution to the JointDIP constraint, we conclude that JointDIP is at least as good as the optimal classifier built upon ϕ_{inv}^0, i.e.,

$$\mathcal{R}^{(\mathfrak{T})}(h_{\text{j-DIP}}) \leq \min_{g \in \mathcal{G}} \mathcal{R}^{(\mathfrak{T})}(g \circ \phi_{\text{inv}}^0).$$

Part (c) In JointDIP, the joint distributions of $AX \in \mathbb{R}^q$ and $\phi_{\text{inv}}(X) \in \mathbb{R}^r$ are matched across the source and target domains. Let

$$T(X) = \begin{pmatrix} AX \\ \phi_{\text{inv}}(X) \end{pmatrix} \in \mathbb{R}^{q+r}.$$

The characteristic function of $T(X^{(1)})$ under the source distribution is then:

$$
\begin{aligned}
c^{(1)}(t) = c^{(1)}(t_1, t_2) &= \mathbb{E}\left[e^{it^\top T(X^{(1)})} \right] \\
&= \frac{1}{L} \sum_{j=1}^{L} \mathbb{E}\left[e^{it^\top T(X^{(1)})} \mid Y^{(1)} = j \right] \\
&= \frac{1}{L} \sum_{j=1}^{L} \mathbb{E}\left[e^{it_1^\top A X^{(1)} + it_2^\top \phi_{\text{inv}}(X^{(1)})} \mid Y^{(1)} = j \right] \\
&= \frac{1}{L} \sum_{j=1}^{L} e^{it_1^\top A u_j} \cdot \underbrace{\mathbb{E}\left[e^{it_1^\top A \epsilon^{(1)} + it_2^\top \phi_{\text{inv}}(X^{(1)})} \mid Y^{(1)} = j \right]}_{:=b_j(t_1, t_2)},
\end{aligned}
$$

where the second step uses the tower property, and the last step follows from the general anticausal model, where conditional on $Y^{(1)} = j$, $X^{(1)} = u_j + \epsilon^{(m)}$ under the source distribution. Similarly, we can compute the characteristic function of T under the target distribution:

$$
\begin{aligned}
c^{(\mathfrak{T})}(t) &= c^{(\mathfrak{T})}(t_1, t_2) \\
&= \frac{1}{L} \sum_{j=1}^{L} e^{it_1^\top A \widetilde{u}_j} \cdot \mathbb{E}\left[e^{it_1^\top A \epsilon^{(\mathfrak{T})} + it_2^\top \phi_{\text{inv}}(X^{(\mathfrak{T})})} \mid Y^{(\mathfrak{T})} = j \right] \\
&= \frac{1}{L} \sum_{j=1}^{L} e^{it_1^\top A \widetilde{u}_j} \cdot b_j(t_1, t_2).
\end{aligned}
$$

Here the last equality holds because $\epsilon^{(\mathfrak{T})}$ and $\phi_{\text{inv}}(X^{(\mathfrak{T})})$ share the same conditional distributions under the target distribution as $\epsilon^{(1)}$ and $\phi_{\text{inv}}(X^{(1)})$ under the source distribution. The JointDIP matching penalty enforces $c^{(1)}(t) = c^{(\mathfrak{T})}(t)$ for all t, or

$$\sum_{j=1}^{L} e^{it_1^\top A u_j} b_j(t_1, t_2) = \sum_{j=1}^{L} e^{it_1^\top A \widetilde{u}_j} b_j(t_1, t_2) \text{ for all } t_1 \in \mathbb{R}^q, t_2 \in \mathbb{R}^r.$$

If we take partial derivaties of both sides with respect to t_1,

$$
\begin{aligned}
&\left(\sum_{j=1}^{L} A u_j e^{it_1^\top A u_j} b_j(t_1, t_2) \right) i + \sum_{j=1}^{L} e^{it_1^\top A u_j} \frac{\partial b_j(t_1, t_2)}{\partial t_1} \\
&= \left(\sum_{j=1}^{L} A \widetilde{u}_j e^{it_1^\top A u_j} b_j(t_1, t_2) \right) i + \sum_{j=1}^{L} e^{it_1^\top A \widetilde{u}_j} \frac{\partial b_j(t_1, t_2)}{\partial t_1}.
\end{aligned}
$$

Setting $t_1 = 0$, it follows that

$$\sum_{j=1}^{L} Au_j \cdot b_j(0, t_2) = \sum_{j=1}^{L} A\widetilde{u}_j \cdot b_j(0, t_2),$$

or equivalently,

$$\sum_{j=1}^{L} Au_j \cdot \mathbb{E}\left[e^{it_2^\top \phi_{\text{inv}}(X^{(1)})} \,\Big|\, Y^{(1)} = j \right] = \sum_{j=1}^{L} A\widetilde{u}_j \cdot \mathbb{E}\left[e^{it_2^\top \phi_{\text{inv}}(X^{(1)})} \,\Big|\, Y^{(1)} = j \right]. \qquad (63)$$

For any vector $a \in \mathbb{R}^r$, let $t_2 = ka$ for $k \in \mathbb{R}$. In Eq. (63), take derivative with respect to k up to $L - 1$ times and set $k = 0$, then we get

$$\sum_{j=1}^{L} Au_j \cdot \mathbb{E}\left[\left(a^\top \phi_{\text{inv}}(X^{(1)}) \right)^u \,\Big|\, Y^{(1)} = j \right] = \sum_{j=1}^{L} A\widetilde{u}_j \cdot \mathbb{E}\left[\left(a^\top \phi_{\text{inv}}(X^{(1)}) \right)^u \,\Big|\, Y^{(1)} = j \right],$$

for $u = 0, 1, \ldots, L - 1$, where we use the fact that ϕ_{inv} is a conditionally invariant feature mapping. Writing the above linear system in a matrix form, we have

$$\underbrace{\begin{pmatrix} 1 & 1 & \cdots & 1 \\ m_1^1(a) & m_2^1(a) & \cdots & m_L^1(a) \\ \vdots & \vdots & \ddots & \vdots \\ m_1^{L-1}(a) & m_2^{L-1}(a) & \cdots & m_L^{L-1}(a) \end{pmatrix}}_{=C_{\phi_{\text{inv}}}(a)} \begin{pmatrix} (Au_1 - A\widetilde{u}_1)^\top \\ (Au_2 - A\widetilde{u}_2)^\top \\ \vdots \\ (Au_L - A\widetilde{u}_L)^\top \end{pmatrix} = 0, \quad \forall \ell \in \{1, 2, \ldots, L\},$$

where $m_j^l(a) = \mathbb{E}\left[\left(a^\top \phi_{\text{inv}}(X^{(1)}) \right)^l \,\Big|\, Y^{(1)} = j \right]$. By assumption that $C_{\phi_{\text{inv}}}(a)$ is full rank, we get $Au_j = A\widetilde{u}_j$ for all $j \in \{1, \ldots, L\}$, i.e.,

$$\mathcal{P}^{(1)}_{\phi_{\text{j-DIP}}(X)|Y=y} = \mathcal{P}^{(\mathfrak{T})}_{\phi_{\text{j-DIP}}(X)|Y=y}, \quad \forall y \in \{1, 2, \ldots, L\},$$

proving the result.

Lemma 18 *Consider the mixture distributions P_1, P_2 given by*

$$P_1 = \sum_{j=1}^{L} w_j P(u_j) \text{ and } P_2 = \sum_{j=1}^{L} \widetilde{w}_j P(\widetilde{u}_j),$$

where $w_j, \widetilde{w}_j$ are the mixing weights such that $\sum_j w_j = \sum_j \widetilde{w}_j = 1$ and $w_j, \widetilde{w}_j \geq 0$, and $u_j, \widetilde{u}_j \in \mathbb{R}^p$ are the fixed vectors for the j-th components, and $P(u_j), P(\widetilde{u}_j)$ denote the distribution P location shifted by $u_j, \widetilde{u}_j$ respectively. If $P_1 = P_2$, then there exists a permutation π such that $u_j = \widetilde{u}_{\pi j}$ for all j.

Proof Let $\nu = \sum_{j=1}^{L} w_j \delta_{u_j}$ and $\widetilde{\nu} = \sum_{j=1}^{L} \widetilde{w}_j \delta_{\widetilde{u}_j}$ be the mixing distributions of P_1 and P_2 respectively. Let $X \sim P_1$ and $\widetilde{X} \sim P_2$. Then we can write $X = U + Z$ and $\widetilde{X} = \widetilde{U} + \widetilde{Z}$ where

$$\begin{cases} U \sim \nu, Z \sim P \text{ independent of } U; \\ \widetilde{U} \sim \widetilde{\nu}, \widetilde{Z} \sim P \text{ independent of } \widetilde{U}. \end{cases}$$

73

The characteristic functions of X and $\widetilde{X}$ can be expressed as

$$\varphi_X(t) = \varphi_U(t)\varphi_Z(t), \quad \varphi_{\widetilde{X}}(t) = \varphi_{\widetilde{U}}(t)\varphi_{\widetilde{Z}}(t).$$

Since $\varphi_X(t) = \varphi_{\widetilde{X}}(t)$ and $\varphi_Z(t) = \varphi_{\widetilde{Z}}(t)$, it follows that $\varphi_U(t) = \varphi_{\widetilde{U}}(t)$, or equivalently $\nu = \widetilde{\nu}$.

Next let $S = \{u_1, \ldots, u_L\}$ and $\widetilde{S} = \{\widetilde{u}_1, \ldots, \widetilde{u}_L\}$ to denote the support sets of ν and $\widetilde{\nu}$. We can easily extend Lemma 2 in Wu and Yang (2020) to the high-dimensional setting to show that

$$d_H(S, \widetilde{S}) \le L \cdot W_1(\nu, \widetilde{\nu}) = 0,$$

where $d_H(S, \widetilde{S})$ is the Hausdorff distance between S and $\widetilde{S}$, defined as

$$d_H(S, \widetilde{S}) = \max\left\{\sup_{x \in S}\inf_{\widetilde{x} \in \widetilde{S}}\|x - \widetilde{x}\|_1, \sup_{\widetilde{x} \in \widetilde{S}}\inf_{x \in S}\|x - \widetilde{x}\|_1\right\},$$

and where W_1 denotes the 1-Wasserstein distance. By definition of $d_H(S, \widetilde{S})$, this proves that we can find a permutation π such that $u_j = \widetilde{u}_{\pi j}$ for all j, thus proving the lemma. ∎

Lemma 19 *Suppose that $\{u_1, \ldots, u_L\}$ and $\{\widetilde{u}_1, \ldots, \widetilde{u}_L\} \subseteq \mathbb{R}^p$ are collections of vectors such that*

$$\dim\left(\mathrm{span}\{u_1 - \widetilde{u}_1, \ldots, u_L - \widetilde{u}_L\}\right) = k < p.$$

Then for any $A \in \mathbb{R}^{q \times p}$ satisfying $Au_j = A\widetilde{u}_j$ for all $j = 1, \ldots, L$, we have $\mathrm{rank}(A) \le p - k$. The equality is achieved if and only if the rows of A span $\mathrm{span}\{u_1 - \widetilde{u}_1, \ldots, u_L - \widetilde{u}_L\}^\perp$.

Proof This lemma is a simple consequence of linear algebra. By the condition of the lemma, $A(u_j - \widetilde{u}_j) = 0$ for all $j = 1, \ldots, L$. This implies that the row space of A is orthogonal to the subspace spanned by $\{u_1 - \widetilde{u}_1, \ldots, u_L - \widetilde{u}_L\}$, i.e.,

$$\mathrm{row}(A) \perp \mathrm{span}\{u_1 - \widetilde{u}_1, \ldots, u_L - \widetilde{u}_L\}.$$

In particular, $\dim(\mathrm{row}(A)) \le p - k$, with equality achieved if and only if $\mathrm{row}(A)$ is the orthogonal complement of $\mathrm{span}\{u_1 - \widetilde{u}_1, \ldots, u_L - \widetilde{u}_L\}$. By the rank theorem, $\mathrm{rank}(A) = \dim(\mathrm{row}(A)) \le p - k$, proving the lemma. ∎

DA method	$\widehat{w}^{(m)}$ Estimation	Minimization Objective
Tar	-	$\widehat{\mathcal{R}}^{(\mathfrak{T})}(g \circ \phi)$
ERM	-	$\widehat{\mathcal{R}}^{(1)}(g \circ \phi)$
ERM-Pool	-	$\frac{1}{M} \sum_{m=1}^{M} \widehat{\mathcal{R}}^{(m)}(g \circ \phi)$
DIP	-	$\widehat{\mathcal{R}}^{(\mathfrak{T})}(g \circ \phi) + \lambda_{\text{DIP}} \cdot \mathfrak{D}\left(\widehat{\mathcal{P}}_{\phi(X)}^{(1)}, \widehat{\mathcal{P}}_{\phi(X)}^{(\mathfrak{T})}\right)$
DIP-Pool	-	$\frac{1}{M} \sum_{m=1}^{M} \left\{ \widehat{\mathcal{R}}^{(m)}(g \circ \phi) + \lambda_{\text{DIP}} \cdot \mathfrak{D}\left(\widehat{\mathcal{P}}_{\phi(X)}^{(m)}, \widehat{\mathcal{P}}_{\phi(X)}^{(\mathfrak{T})}\right) \right\}$
CIP	-	$\frac{1}{M} \sum_{m=1}^{M} \widehat{\mathcal{R}}^{(m)}(g \circ \phi) + \frac{\lambda_{\text{CIP}}}{LM^2} \cdot \sum_{y=1}^{L} \sum_{m \neq m'} \mathfrak{D}\left(\widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m)}, \widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m')}\right)$
IW-ERM	ERM-Pool	$\frac{1}{M} \sum_{m=1}^{M} \widehat{\mathcal{R}}^{(m)}(g \circ \phi; \widehat{w}^{(m)})$
IW-CIP	CIP	$\frac{1}{M} \sum_{m=1}^{M} \widehat{\mathcal{R}}^{(m)}(g \circ \phi; \widehat{w}^{(m)}) + \frac{\lambda_{\text{CIP}}}{LM^2} \cdot \sum_{y=1}^{L} \sum_{m \neq m'} \mathfrak{D}\left(\widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m)}, \widehat{\mathcal{P}}_{\phi(X)|Y=y}^{(m')}\right)$
IW-DIP	CIP	$\widehat{\mathcal{R}}^{(1)}(g \circ \phi; \widehat{w}^{(1)}) + \lambda_{\text{DIP}} \cdot \mathfrak{D}\left(\widehat{\mathcal{P}}_{\widehat{w}^{(1)}(Y)\phi(X)}^{(1)}, \widehat{\mathcal{P}}_{\phi(X)}^{(\mathfrak{T})}\right)$
JointDIP	-	$\widehat{\mathcal{R}}^{(1)}(g \circ \phi) + \lambda_{\text{j-DIP}} \cdot \mathfrak{D}\left(\widehat{\mathcal{P}}_{(\phi, \widehat{\phi}_{\text{CIP}})(X)}^{(1)}, \widehat{\mathcal{P}}_{(\phi, \widehat{\phi}_{\text{CIP}})(X)}^{(\mathfrak{T})}\right)$
JointDIP-Pool	-	$\frac{1}{M} \sum_{m=1}^{M} \left\{ \widehat{\mathcal{R}}^{(m)}(g \circ \phi) + \lambda_{\text{j-DIP}} \cdot \mathfrak{D}\left(\widehat{\mathcal{P}}_{(\phi, \widehat{\phi}_{\text{CIP}})(X)}^{(m)}, \widehat{\mathcal{P}}_{(\phi, \widehat{\phi}_{\text{CIP}})(X)}^{(\mathfrak{T})}\right) \right\}$
IW-JointDIP	CIP	$\widehat{\mathcal{R}}^{(1)}(g \circ \phi; \widehat{w}^{(1)}) + \lambda_{\text{j-DIP}} \cdot \mathfrak{D}\left(\widehat{\mathcal{P}}_{\widehat{w}^{(1)}(Y) \cdot (\phi, \widehat{\phi}_{\text{CIP}})(X)}^{(1)}, \widehat{\mathcal{P}}_{(\phi, \widehat{\phi}_{\text{CIP}})(X)}^{(\mathfrak{T})}\right)$
IRM	-	$\frac{1}{M} \sum_{m=1}^{M} \left\{ \widehat{\mathcal{R}}^{(m)}(g \circ \phi) + \lambda_{\text{IRM}} \cdot \left\| \nabla_{w|w=1.0} \widehat{\mathcal{R}}^{(m)}(w \cdot g \circ \phi) \right\|_2^2 \right\}$
V-REx	-	$\frac{1}{M} \sum_{m=1}^{M} \widehat{\mathcal{R}}^{(m)}(g \circ \phi) + \lambda_{\text{VREx}} \cdot \text{Var}\left(\widehat{\mathcal{R}}^{(m)}(g \circ \phi)\right)$
groupDRO	-	$\sup_{\sum_{m=1}^{M} q_m = 1, q_m \geq 0} \sum_{m=1}^{M} q_m \widehat{\mathcal{R}}^{(m)}(g \circ \phi)$

Table 11: DA algorithms used in numerical experiments.

Appendix D. Experiment details

This section provides details on various DA algorithms, training procedures, network architectures, and hyperparameter selection strategies that we employed in our numerical experiments (Section 5).

D.1 DA algorithms in numerical experiments

Table 11 provides a summary of the algorithms that we used in our numerical experiments. Both mean distance and maximum mean discrepancy (MMD) were used in DIP and CIP, but for JointDIP, we only considered MMD distance, because joint matching with mean distance is essentially the same as the original DIP. In addition to the DA algorithms

Task	Epochs	Optimizer	Batch Size	NN Architecture	Feature Layer	$\mathfrak{D}(\cdot, \cdot)$ [15]
SCM	50	Adam(lr=1e-2)	100	Linear Model	Last layer	mean
SCM_binary[16]	50	Adam(lr=1e-2)	100	Linear$(10, 10)$ or Linear$(10, 10)$ ReLU Linear$(10, 2)$	Last layer	mean
MNIST	20	Adam(lr=1e-3)	256	Conv2d$(1, 20, 5, 1)$ ReLU MaxPool2d(2) Conv2d$(20, 50, 5, 1)$ ReLU MaxPool2d(2) Linear$(800, 500)$ ReLU Linear$(500, 2)$	Last layer	MMD
CelebA	10	Adam(lr=1e-3)	64	Conv2d$(3, 16, 5, 1)$ ReLU MaxPool2d(2) Conv2d$(16, 32, 5, 1, 1)$ ReLU MaxPool2d(2) Conv2d$(32, 64, 5, 1)$ ReLU MaxPool2d(2) Linear$(960, 256)$ ReLU Linear$(256, 2)$	Last layer	MMD
Camelyon17	5	SGD(lr=1e-3, weight_decay=0.01, momentum=0.9)	32	DenseNet-121 (pretrained=False)	Flattened CNN output	MMD
DomainNet	10	Adam(lr=1e-3)	16	ResNet-101 (pretrained=True)	Last layer	MMD

Table 12: Training details of each classification task.

introduced in this work, we also include IRM (Arjovsky et al., 2019), V-REx (Krueger et al., 2021), and groupDRO (Sagawa et al., 2020) in our experiments.

D.2 Training details and network architectures

Except for Camelyon17 and DomainNet, the datasets we used in Section 5 are either generated or perturbed synthetically. To avoid any potential bias that can result from repeatedly running DA algorithms on a single, fixed synthetic dataset, for each classification task, we use 10 different seeds to construct 10 different datasets, and each DA algorithm is run once per seed.

---

15. This column only applies to CIP and DIP. We always use MMD distance when implementing JointDIP.
16. This setting is used for the additional experiments in Appendix E.

Algorithm	Hyperparameter	Grid Search Space
DIP (mean / MMD, Pool)	λ_{DIP}^{*}	$10^{\{-2,-1,0,1,2\}}$
CIP (mean / MMD)	λ_{CIP}^{*}	$10^{\{-2,-1,0,1,2\}}$
IW-DIP (mean / MMD, Pool)	λ_{CIP}	$10^{\{-2,-1,0,1,2\}}$
	$\lambda_{\text{IW-DIP}}$	$10^{\{-2,-1,0,1,2\}}$
IW-CIP (mean / MMD)	λ_{CIP}	$10^{\{-2,-1,0,1,2\}}$
	$\lambda_{\text{IW-CIP}}$	$10^{\{-2,-1,0,1,2\}}$
JointDIP (Pool)	λ_{CIP}^{*}	$10^{\{-2,-1,0,1,2\}}$
	$\lambda_{\text{JointDIP}}^{*}$	$10^{\{-2,-1,0,1,2\}}$
IW-JointDIP	λ_{CIP}	$10^{\{-2,-1,0,1,2\}}$
	$\lambda_{\text{JointDIP}}$	$10^{\{-2,-1,0,1,2\}}$
IRM	$\lambda_{\text{IRM}}^{**}$	$10^{\{-1,0,1,2,3,4\}}$
	iterations annealing***	$0, 10, 100, 1000, 3000$
V-REx	$\lambda_{\text{VREx}}^{**}$	$10^{\{-1,0,1,2,3,4\}}$
	iterations annealing***	$0, 10, 100, 1000, 3000$
groupDRO	eta	$10^{\{-2,-1,0,1\}}$

Table 13: Hyperparameter search space. In the experiments on Camelyon17, the search space of hyperparameters marked with * is defined by $10^{\{-3,-2,-1,0,1,2\}}$. In the experiment on DomainNet, the search space of hyperparameters marked with * is defined by $10^{\{-2,-1,0,1,2,3\}}$, for hyperparameters marked with ** by $10^{\{-1,0,1\}}$, and for hyperparameters marked with *** by $\{0, 10, 100\}$.

Table 12 shows details on the training parameters and model architectures used in all the experiments in Section 5. For multi-stage DA algorithms such as JointDIP (where CIP is run before solving JointDIP optimization problem), the number of epochs is the same in each training stage. As for the feature layer used for CIP and DIP matching penalty, we utilize the last layer of the neural network in experiments on SCM, MNIST, CelebA, and DomainNet; for experiments on Camelyon17, we use the flattened output from CNN in DenseNet-121 following the practice in Koh et al. (2021).

D.3 Hyperparameter selection

In our experiments on SCM, MNIST, and CelebA, instead of searching for different hyperparameters for each seed, which would be time-consuming, we choose the same hyperparameters across all seeds. In Appendix D.4, we compare between two hyperparameter selection strategies, i.e., using different hyperparameters versus using the same hyperparameters across seeds on SCMs, and find that these two strategies yield similar results.

In order to select hyperparameters which allow for the highest average accuracy, we use a simple grid search. Three different ways of hyperparameter selection are presented in Gulrajani and Lopez-Paz (2021): training-domain validation set, leave-one-out cross validation, and test-domain validation set (oracle). The training-domain validation set approach is not suitable for us as it assumes the target domain is very similar to the training domains, which does not apply to our case. The leave-one-out cross validation approach is cum-

	SCM I		SCM II	
Mean shift	Y		Y	
CICs	N		Y	
Label shift	N		Y	
Label-flip	N		N	
Hyperparam	Different	Same	Different	Same
DIP-mean	**88.2±1.9**	**87.6±1.5**	73.3±10.7	62.0±2.9
DIP-MMD	86.2±2.2	86.6±2.2	72.0±16.2	59.9±3.3
DIP-Pool-mean	86.2±2.8	86.4±2.2	70.8±10.1	60.1±3.1
DIP-Pool-MMD	86.2±1.9	85.5±3.1	74.2±13.5	61.3±9.3
CIP-mean	56.6±11.8	55.9±12.0	79.3±6.4	75.7±6.5
CIP-MMD	63.2±14.1	60.4±14.8	84.1±9.6	73.5±7.0
IW-CIP-mean	54.1±11.8	54.0±11.8	90.8±1.4	90.4±0.8
IW-CIP-MMD	64.4±12.7	56.8±13.6	89.5±4.3	90.4±0.8
IW-DIP-mean	54.3±11.7	54.3±11.7	**92.7±2.1**	**92.1±2.7**
IW-DIP-MMD	75.2±16.2	68.0±19.4	91.5±2.9	90.1±1.1
JointDIP	85.8±2.2	86.8±1.9	77.7±12.5	70.6±6.2
IW-JointDIP	72.7±17.3	68.0±19.4	91.3±2.8	90.0±1.3
IRM	66.1±12.1	56.7±10.4	86.7±5.4	71.9±17.9
V-REx	63.0±13.1	55.6±11.5	72.7±26.3	62.9±25.7
groupDRO	54.4±10.0	54.4±10.0	65.2±26.6	64.2±25.7

Table 14: Target accuracy in linear SCM I and II under different hyperparameter tuning strategies.

bersome and would also suffer if the validation domain is significantly different from the target domain. The test-domain validation set approach has often been adopted in several domain adaptation papers (Ganin et al., 2016; Arjovsky et al., 2019; Krueger et al., 2021) when an appropriate validation set is unavailable. However, this approach contradicts the domain adaptation principle where most target labels should be unavailable. Therefore, we restrict access to just 10% of labeled samples from the target domain in each seed, and check the accuracy of these labeled samples only at the final checkpoint during hyperparameter selection.

Note that Camelyon17 provides a validation set. Hence, we adhere to the standard submission guidelines from WILDS for this dataset, selecting hyperparameters based on accuracy scores on the validation set Koh et al. (2021); Sagawa et al. (2021).

D.4 Different hyperparameter vs. same hyperparameter across all seeds

We carry out a comparison of two strategies for selecting hyperparameters on SCMs: (1) using the same hyperparameter across all seeds, and (2) using different hyperparameters for different seeds. See Table 14 and Table 15. We observe that in most cases, the second approach tends to give a better performance. However, this difference typically doesn't affect the order of the top-performing algorithms; the best performing algorithm tends to remain the same regardless of the hyperparameter selection strategies. For the sake of simplicity and time efficiency, we therefore adopt the first approach in our final results.

	SCM III		SCM IV	
Mean shift	Y		Y	
CICs	Y		Y	
Label shift	N		Y	
Label-flip	Y		Y	
Hyperparam	Different	Same	Different	Same
DIP-mean	36.3±11.8	34.5±14.9	35.6±14.0	35.3±14.6
DIP-MMD	77.5±7.9	63.3±31.7	74.0±4.2	60.2±30.1
DIP-Pool-mean	81.9±1.8	82.0±1.1	81.9±3.8	82.3±3.8
DIP-Pool-MMD	82.5±1.4	83.3±0.8	82.2±4.1	82.5±3.8
CIP-mean	82.2±1.5	81.8±1.3	83.2±1.8	82.1±1.2
CIP-MMD	81.7±2.7	81.7±2.6	83.6±2.8	82.7±1.6
IW-CIP-mean	80.3±4.2	81.2±4.1	83.1±2.3	83.8±2.2
IW-CIP-MMD	80.9±3.5	80.5±6.3	84.1±1.9	83.5±2.8
IW-DIP-mean	37.6±12.9	37.2±14.9	66.6±5.2	64.2±7.3
IW-DIP-MMD	70.9±13.8	65.1±23.6	81.5±2.1	80.1±2.6
JointDIP	**85.3±2.8**	**85.4±2.1**	81.6±3.9	82.8±1.9
IW-JointDIP	82.9±3.3	82.9±8.1	**85.4±1.9**	**85.1±3.7**
IRM	82.0±1.9	80.2±1.8	84.0±2.2	83.7±3.3
V-REx	80.8±7.6	80.4±7.6	84.0±4.0	83.8±3.7
groupDRO	82.5±6.7	81.3±7.4	84.7±3.8	84.3±3.2

Table 15: Target accuracy in linear SCM III and IV under different hyperparameter tuning strategies.

Appendix E. Additional experiments

In this section, we provide additional experiments on SCMs and MNIST to investigate the effect of the number of source domains on the generalization performance of CIP and its variants. Additionally, we compare the performance of our algorithms with simple heuristic alternatives through numerical experiments on MNIST and CelebA.

E.1 Effect of number of source domains in linear SCMs

To show that a sufficient number of source domains is necessary for finding CICs, we conduct experiments on linear SCMs to compare the results of utilizing fewer versus more source domains. Table 16 reports the risk difference $\widehat{\mathcal{R}}^{(\mathfrak{T})}(h) - \widehat{\mathcal{R}}(h; \widehat{w})$ between the source and target domains for four different CIP-based methods. We observe that the risk difference tends to be closer to 0 for larger values of M in SCM II, III, and IV, whereas smaller values of M lead to a larger risk difference. This finding suggests that CIP and IW-CIP may underperform when the number of source domains M is small, because they can only find feature representations that are conditionally invariant across the source domains, but with limited domains, it becomes challenging to obtain feature representations that are conditionally invariant across both source and target domains. Note that in SCM I, the number of domains does not make a significant difference, because CICs do not exist in SCM I.

	SCM I		SCM II ($\times 10^{-2}$)	
DA Algorithm	$M = 2$	$M = 3$	$M = 2$	$M = 11$
CIP-mean	1.7 ± 2.0	2.1 ± 1.4	11.3 ± 57.9	-0.1 ± 0.9
CIP-MMD	1.7 ± 1.9	1.3 ± 1.1	-5.9 ± 7.6	3.6 ± 9.6
IW-CIP-mean	1.9 ± 2.2	2.0 ± 0.8	89.0 ± 120.0	-0.6 ± 10.9
IW-CIP-MMD	2.0 ± 2.3	1.7 ± 1.0	41.7 ± 60.0	-4.7 ± 9.0

	SCM III ($\times 10^{-2}$)		SCM IV ($\times 10^{-2}$)	
DA Algorithm	$M = 2$	$M = 11$	$M = 2$	$M = 11$
CIP-mean	22.2 ± 26.3	1.3 ± 3.6	14.2 ± 25.3	0.1 ± 0.9
CIP-MMD	6.2 ± 7.1	2.1 ± 5.2	2.0 ± 4.7	0.3 ± 2.1
IW-CIP-mean	41.1 ± 50.1	2.6 ± 6.9	25.6 ± 29.9	5.1 ± 6.7
IW-CIP-MMD	30.2 ± 31.6	6.6 ± 16.8	11.4 ± 20.8	5.3 ± 9.6

Table 16: Risk difference $\widehat{\mathcal{R}}^{(\mathfrak{T})}(h) - \widehat{\overline{\mathcal{R}}}(h; \widehat{w})$ between source and target in linear SCMs. These differences are evaluated under varying number of source domains M.

The minimum number of source domains necessary for finding CICs depends on the types of interventions and the hypothesis class in use. Proposition 9 and Lemma 17 show that the number of source domains needs to be greater than the dimension of the perturbations, which is the case for linear SCMs above. In this context, we also consider another SCM under binary interventions, as described below. The purpose is to demonstrate that the required number of source domains that enables successful finding of CICs via CIP can vary when the hypothesis class changes.

- SCM binary: The data generation model is

$$Y^{(m)} = \text{Bernoulli}(0.5),$$
$$X^{(m)}_{[1:5]} = 0.2(Y^{(m)} - 0.5) \cdot \mathbb{1}_5 + 0.4 \cdot \mathcal{N}(0, \mathbb{I}_5),$$
$$X^{(m)}_{[6:10]} = 0.2(Y^{(m)} - 0.5) \cdot \mathbb{1}_5 + 0.4 \cdot \mathcal{N}(0, \mathbb{I}_5) + A^{(m)},$$

where $A^{(m)}$ is randomly sampled from $\{a \in \mathbb{R}^5 \mid a_i \in \{-1, 1\}\}$ $(1 \leq m \leq M)$ without replacement in source domains, and the target domain suffers larger intervention with $A^{(\mathfrak{T})} = (2, 2, 2, 2, 2)^\top$.

We use two different models in this experiment: a linear model and a two-layer neural network. For the linear model, as shown in Proposition 9 and Lemma 17, the number of source domains should be at least on the order of the dimension of intervention plus one, that is, $M \geq 6$. We vary the number of source domains M from 2 to 7 and evaluate the performance of CIP-mean and CIP-MMD in Table 17 (there is no label shift between source and target domains). We find that when $M \geq 6$ the target accuracy and risk are close to those in the source domains. We also observe that using more source domains than the theoretically required minimum number of source domains can further improve performance in practice.

	CIP-mean		
#source domains	src_acc	tar_acc	risk_diff
$M = 2$	85.1 ± 1.5	50.6 ± 1.6	1.8 ± 1.0
$M = 3$	84.7 ± 2.6	54.3 ± 7.9	$9.1 \pm 6.3 \times 10^{-1}$
$M = 4$	82.4 ± 2.3	62.3 ± 11.2	$4.1 \pm 5.1 \times 10^{-1}$
$M = 5$	81.9 ± 2.0	63.4 ± 12.2	$2.9 \pm 4.3 \times 10^{-1}$
$M = 6$	80.6 ± 1.5	72.2 ± 10.8	$7.7 \pm 13.2 \times 10^{-2}$
$M = 7$	80.0 ± 0.5	78.6 ± 1.6	$6.7 \pm 9.9 \times 10^{-3}$

	CIP-MMD		
#source domains	src_acc	tar_acc	risk_diff
$M = 2$	84.5 ± 1.0	51.7 ± 3.4	2.4 ± 1.4
$M = 3$	83.1 ± 1.7	58.4 ± 9.9	1.3 ± 1.2
$M = 4$	81.9 ± 1.5	67.9 ± 11.5	$5.2 \pm 7.0 \times 10^{-1}$
$M = 5$	81.7 ± 1.5	69.8 ± 11.4	$3.8 \pm 5.6 \times 10^{-1}$
$M = 6$	81.3 ± 0.9	73.0 ± 8.7	$2.5 \pm 4.7 \times 10^{-1}$
$M = 7$	81.1 ± 0.7	73.9 ± 5.0	$1.1 \pm 1.1 \times 10^{-1}$

Table 17: Performance of CIP under a linear model on SCM-binary. The number of source domains varies from $M = 2$ to $M = 7$.

	CIP-mean		
#src domains	src_acc	tar_acc	risk_diff
$M = 2$	85.3 ± 1.7	50.8 ± 1.7	2.8 ± 1.8
$M = 4$	84.1 ± 1.3	55.5 ± 8.7	$9.3 \pm 7.0 \times 10^{-1}$
$M = 8$	82.5 ± 1.4	63.8 ± 12.1	$3.1 \pm 4.2 \times 10^{-1}$
$M = 16$	80.8 ± 0.3	76.4 ± 4.3	$2.7 \pm 2.3 \times 10^{-2}$
$M = 32$	80.5 ± 0.4	78.4 ± 1.2	$1.6 \pm 1.1 \times 10^{-2}$

	CIP-MMD		
#src domains	src_acc	tar_acc	risk_diff
$M = 2$	86.5 ± 1.9	50.6 ± 1.6	2.7 ± 1.1
$M = 4$	85.2 ± 1.3	53.5 ± 8.4	2.1 ± 1.0
$M = 8$	83.3 ± 1.4	59.7 ± 11.3	$9.5 \pm 7.2 \times 10^{-1}$
$M = 16$	81.1 ± 0.5	72.1 ± 7.4	$1.5 \pm 1.8 \times 10^{-1}$
$M = 32$	80.6 ± 0.3	78.7 ± 1.2	$2.6 \pm 1.6 \times 10^{-2}$

Table 18: Performance of CIP under a two-layer neural network on SCM-binary. The number of source domains varies from $M = 2^1$ to $M = 2^5$.

For the two-layer neural network, we vary M from 2^1 to 2^5. In Table 18, we observe that when $M = 32$, the target accuracy and target risk are close to the source accuracy and source risk. This suggests that if the hypothesis class is nonlinear and complex, CIP may need more domains to accurately identify CICs. In this case, $M = 2^5 = 32$ source domains covering all possible binary interventions are required.

	MNIST I ($\times 10^{-2}$)		MNIST II ($\times 10^{-4}$)	
DA Algorithm	$M = 3$	$M = 5$	$M = 3$	$M = 5$
CIP-mean	43.4 ± 8.9	16.0 ± 1.9	31.5 ± 3.4	17.1 ± 5.1
CIP-MMD	60.1 ± 4.1	17.9 ± 4.9	65.3 ± 13.6	15.6 ± 3.2
IW-CIP-mean	58.2 ± 7.5	15.7 ± 1.4	33.3 ± 4.9	11.5 ± 2.0
IW-CIP-MMD	62.9 ± 5.6	20.3 ± 6.9	62.5 ± 13.3	14.1 ± 2.2

	MNIST III ($\times 10^{-4}$)		MNIST IV ($\times 10^{-4}$)	
DA Algorithm	$M = 3$	$M = 5$	$M = 3$	$M = 5$
CIP-mean	45.0 ± 9.6	26.8 ± 3.8	44.4 ± 8.7	33.7 ± 5.7
CIP-MMD	47.5 ± 6.8	35.2 ± 4.1	50.1 ± 10.6	36.0 ± 6.3
IW-CIP-mean	40.3 ± 5.5	27.5 ± 4.4	43.5 ± 6.5	27.6 ± 2.9
IW-CIP-MMD	42.7 ± 5.9	36.3 ± 4.3	46.5 ± 5.6	35.3 ± 5.9

Table 19: Risk difference $\widehat{\mathcal{R}}^{(\mathfrak{T})}(h) - \widehat{\overline{\mathcal{R}}}(h; \widehat{w})$ between source and target in MNIST experiments. These differences are evaluated under varying number of source domains M.

E.2 Effect of number of source domains in MNIST

Next we compare the performance of CIP-based methods on MNIST with varying number of source domains ($M = 3$ versus $M = 5$). For $M = 3$, we use the middle three domains with rotation angles of $-15°, 0°$, and $15°$ from Section 5.2; for $M = 5$, we use all source domains mentioned in our experiments from Section 5.2. Table 19 shows the risk difference between source and target domains for four different CIP-based methods. Similar to our findings in the SCM experiments, the results demonstrate that using a higher number of source domains helps in identifying CICs and reducing the risk difference between source and target.

E.3 Comparison with other heuristic baselines

We introduced JointDIP, a method specifically designed to address the label-flipping issue observed with DIP when label-flipping features exist between source and target domains. Our theoretical results support this claim by showing that, under certain conditions, JointDIP avoids learning label-flipping features and achieves strong generalization to the target domain, where DIP may still struggle. The advantage of JointDIP is further confirmed through numerical experiments conducted in Section 5.

One may wonder if there are simpler alternatives to designing DA algorithms that prevent label-flipping issues while being easier to implement than JointDIP, which incorporates CIP features into the matching penalty. To explore this, we consider the following two alternative methods.

- **CIP-FT-DIP:** This method first implements CIP to learn CICs and the corresponding classifier based on CICs. The learned features and classifier are then used as an initialization for DIP, followed by fine-tuning using the DIP objective.

- **CIP-Sum-DIP:** This method combines the CIP and DIP matching penalties into a single objective by optimizing their sum. By doing so, it learns features that simultaneously satisfy the marginal invariance of covariates between source and target domains and the conditional distributional invariance across multiple source domains. Unlike CIP-FT-DIP, this approach integrates both objectives into one optimization problem instead of a two-step algorithm.

At first glance, both CIP-FT-DIP and CIP-Sum-DIP might appear capable of addressing label-flipping issues in the presence of label-flipping features. However, the following simple example illustrates that these algorithms can still learn label-flipping features. Consider a scenario where the source and target data are generated according to the following SCMs:

$$\begin{cases} Y^{(1)} \sim \mathrm{Rad}(\{-1, +1\}), \\ X_{[1]}^{(1)} = Y^{(1)} + \epsilon_{[1]}^{(1)}, \\ X_{[2]}^{(1)} = 10Y^{(1)} + \epsilon_{[2]}^{(1)}, \\ X_{[3]}^{(1)} = Y^{(1)} + \epsilon_{[3]}^{(1)}, \end{cases} \quad \begin{cases} Y^{(2)} \sim \mathrm{Rad}(\{-1, +1\}), \\ X_{[1]}^{(2)} = Y^{(2)} + \epsilon_{[1]}^{(2)}, \\ X_{[2]}^{(2)} = 10Y^{(2)} + 2 + \epsilon_{[2]}^{(2)}, \\ X_{[3]}^{(2)} = -2Y^{(2)} + \epsilon_{[3]}^{(2)}, \end{cases} \quad \begin{cases} Y^{(\mathfrak{T})} \sim \mathrm{Rad}(\{-1, +1\}), \\ X_{[1]}^{(\mathfrak{T})} = Y^{(\mathfrak{T})} + \epsilon_{[1]}^{(\mathfrak{T})}, \\ X_{[2]}^{(\mathfrak{T})} = -10Y^{(\mathfrak{T})} + \epsilon_{[2]}^{(\mathfrak{T})}, \\ X_{[3]}^{(\mathfrak{T})} = Y^{(\mathfrak{T})} + \epsilon_{[3]}^{(\mathfrak{T})}, \end{cases}$$

where $\mathrm{Rad}(\{-1, +1\})$ denotes the Rademacher distribution defined on $\{-1, +1\}$ with equal probability, and $\epsilon^{(1)}, \epsilon^{(2)}, \epsilon^{(\mathfrak{T})} \sim \mathcal{N}(0, \mathbb{1}_3)$. In this example, $X_{[1]}$ is a CIC that CIP can correctly identify. $X_{[2]}$ represents a label-flipping feature between the first source and target domains, while $X_{[3]}$ is a feature that helps prediction between the first source and the target domain. Here the first source domain is used for DIP.

Assume that we use the feature selector as the feature mapping. For CIP-FT-DIP, if we initialize with CIP by selecting feature $X_{[1]}$ and then fine-tune with the DIP objective, $X_{[2]}$, which satisfies the DIP penalty for marginal invariance and is more predictive of Y compared to $X_{[1]}$ and $X_{[3]}$, will quickly be learned during fine-tuning. As a result, CIP-FT-DIP ends up learning $X_{[2]}$, despite being initialized with $X_{[1]}$, thereby inheriting the label-flipping issue of DIP. For CIP-Sum-DIP, enforcing features to simultaneously satisfy both CIP and DIP penalties restricts the selection to $X_{[1]}$, leading to results similar to CIP. However, this approach ignores the potential benefits of incorporating $X_{[3]}$, limiting its effectiveness in improving target performance. In contrast, JointDIP, when applied to the first source and target domains, can learn both $X_{[1]}$ and $X_{[3]}$, allowing to avoid the label-flipping issue associated with $X_{[2]}$ while leveraging $X_{[3]}$ to enhance target performance. As a result, JointDIP outperforms both CIP-FT-DIP and CIP-Sum-DIP in this scenario.

We further conduct experiments on MNIST and CelebA to numerically compare the two methods, CIP-FT-DIP and CIP-Sum-DIP, against other DA methods such as DIP, CIP, and JointDIP. For CIP-FT-DIP, we first run CIP to learn the weights, which are then used to initialize the architecture before fine-tuning with the DIP objective. For CIP-Sum-DIP, the DIP objective is modified by adding the CIP matching penalty, combining the two objectives into a single optimization process. The training details, including architectures, batch sizes, and other settings, follow the same configurations as in our main experiments (see Appendix D.2). For hyperparameter selection of λ_{CIP} and λ_{DIP}, we vary over $\{0.01, 0.1, 1\}$ for both parameters and select the best-performing one using the protocol described in Appendix D.3.

	MNIST I	MNIST II	MNIST III	MNIST IV
Rotation shift	Y	Y	Y	Y
Label shift	N	Y	N	Y
Label-flipping features	N	N	Y	Y
DA Algorithm	tar_acc	tar_acc	tar_acc	tar_acc
DIP*	97.5±0.3	96.7±0.5	91.0±1.9	90.1±1.6
CIP*	94.9±0.9	94.9±2.1	89.1±1.7	90.0±1.3
JointDIP*	97.3±0.3	96.9±0.6	93.5±1.2	91.5±1.1
CIP-FT-DIP	95.1±1.4	92.6±0.8	91.0±1.4	89.3±1.5
CIP-Sum-DIP	91.6±0.9	89.3±1.6	80.8±3.1	79.5±3.2

Table 20: Target accuracy on rotated MNIST for CIP-FT-DIP and CIP-Sum-DIP. For comparison with other DA algorithms, we include results from Table 3, Table 4, marked with *. The algorithms are run on 10 different datasets, each generated using 10 different random seeds.

The results on MNIST are presented in Table 20. In this experiment, MNIST II and IV involve label shift interventions, and we do not implement the importance-weighting based label shift correction (IW) step for CIP-FT-DIP and CIP-Sum-DIP. To ensure a fair comparison, we also exclude IW-DIP, IW-CIP, and IW-JointDIP. We observe that CIP-FT-DIP performs comparably to CIP across various settings but shows slightly worse performance than DIP. In cases where label-flipping features are present, such as in MNIST III and IV, the performance drops compared to settings without label-flipping features. This behavior is similar to that of DIP, indicating that CIP-FT-DIP is not as effective as DIP in preventing the learning of label-flipping features. CIP-Sum-DIP exhibits similar behaviors but performs significantly worse overall compared to other DA methods. A potential explanation is that the CIP and DIP matching penalties may have interactions that make the method more sensitive to the choice of hyperparameter settings. As a result, CIP-Sum-DIP likely requires more careful hyperparameter tuning than other DA methods that rely on a single matching penalty.

The results on CelebA are presented in Table 21. In this setting, there is no label shift, but the features are manipulated such that the Color_Balance feature maintains a stable correlation with the label between the last source and target domains, while the Mouth_Slightly_Open feature serves as a label-flipping feature. CIP-FT-DIP and CIP-Sum-DIP show comparable or slightly worse performance than DIP and CIP on CelebA I and CelebA II. Interestingly, both methods outperform all other DA methods on CelebA III by a significant margin. On CelebA III, the correlation between the Mouth_Slightly_Open feature and the label is high, allowing DIP to rely on this feature for predictions, which leads to poor prediction performance. JointDIP demonstrates higher accuracy, highlighting its reliability compared to DIP. Moreover, the performance of CIP-FT-DIP and CIP-Sum-DIP in this setting indicates that combining CIP with DIP can further enhance the reliability of algorithms in mitigating the influence of label-flipping features in this example.

	CelebA I	CelebA II	CelebA III
DA Algorithm	tar_acc	tar_acc	tar_acc
DIP*	82.0±2.2	90.6±1.0	82.0±2.2
CIP*	76.6±1.4	83.7±1.4	89.7±0.7
JointDIP*	87.3±0.6	90.4±1.2	88.7±1.0
CIP-FT-DIP	81.5±2.8	79.7±2.6	92.3±0.4
CIP-Sum-DIP	77.6±2.1	79.5±1.8	92.9±0.3

Table 21: Target accuracy on CelebA I, II, III for CIP-FT-DIP and CIP-Sum-DIP. For comparison with other DA algorithms, we include results from Table 6, marked with *. For each setting of the CelebA, DA algorithms are run on 10 different datasets, each generated using 10 different random seeds.

References

M. Amini and P. Gallinari. Semi-supervised learning with an explicit label-error model for misclassified data. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 555–560, 2003.

M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

K. Azizzadenesheli, A. Liu, F. Yang, and A. Anandkumar. Regularized learning for domain adaptation under label shifts. In *International Conference on Learning Representations (ICLR)*, 2019.

M. Baktashmotlagh, M. T. Harandi, B. C. Lovell, and M. Salzmann. Unsupervised domain adaptation by domain invariant projection. In *IEEE International Conference on Computer Vision (ICCV)*, pages 769–776, 2013.

S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. *Advances in Neural Information Processing Systems (NeurIPS)*, 19, 2006.

S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1–2):151–175, 2010.

A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*, volume 28. Princeton University Press, 2009.

A. Ben-Tal, D. Den Hertog, A. De Waegenaere, B. Melenberg, and G. Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

P. Bühlmann. Invariance, causality and robustness. *Statistical Science*, 35(3):404–426, 2020.

A. Cadrin-Chênevert. Moving from imagenet to radimagenet for improved transfer learning and generalizability. *Radiology: Artificial Intelligence*, 4(5):e220126, 2022.

O. Chapelle, B. Schölkopf, and A. Zien. Semi-supervised learning. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

L. Chen, M. Zaharia, and J. Y. Zou. Estimating and explaining model performance when both covariates and labels shift. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:11467–11479, 2022.

Y. Chen and P. Bühlmann. Domain adaptation under structural causal models. *Journal of Machine Learning Research*, 22(261):1–80, 2021.

F. Chung and L. Lu. Concentration inequalities and martingale inequalities: A survey. *Internet Mathematics*, 3(1):79–127, 2006.

C. Cortes and M. Mohri. Domain adaptation in regression. In *International Conference on Algorithmic Learning Theory (ALT)*, pages 308–323. Springer, 2011.

C. Cortes and M. Mohri. Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science*, 519:103–126, 2014.

C. Cortes, Y. Mansour, and M. Mohri. Learning bounds for importance weighting. *Advances in Neural Information Processing Systems (NeurIPS)*, 23, 2010.

N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9): 1853–1865, 2016.

N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. IEEE, 2009.

J. C. Duchi, P. W. Glynn, and H. Namkoong. Statistics of robust optimization: A generalized empirical likelihood approach. *Mathematics of Operations Research*, 46(3):946–969, 2021.

C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, pages 1126–1135. PMLR, 2017.

Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

S. Garg, Y. Wu, S. Balakrishnan, and Z. Lipton. A unified view of label shift estimation. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:3290–3300, 2020.

B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2066–2073. IEEE, 2012.

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

R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *IEEE International Conference on Computer Vision (ICCV)*, pages 999–1006. IEEE, 2011.

A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

I. Gulrajani and D. Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations (ICLR)*, 2021.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

K. He, R. Girshick, and P. Dollár. Rethinking imagenet pre-training. In *IEEE International Conference on Computer Vision (ICCV)*, pages 4918–4927, 2019.

C. Heinze-Deml and N. Meinshausen. Conditional variance penalties and domain shift robustness. *Machine Learning*, 110(2):303–348, 2021.

J. Hoffman, M. Mohri, and N. Zhang. Algorithms and theory for multiple-source adaptation. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018a.

J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning (ICML)*, pages 1989–1998. PMLR, 2018b.

W. Hu, G. Niu, I. Sato, and M. Sugiyama. Does distributionally robust supervised learning give robust classifiers? In *International Conference on Machine Learning (ICML)*, pages 2029–2037. PMLR, 2018.

G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4700–4708, 2017.

X. Jiang, Q. Lao, S. Matwin, and M. Havaei. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In *International Conference on Machine Learning (ICML)*, pages 4816–4827. PMLR, 2020.

Y. Jiang and V. Veitch. Invariant and transportable representations for anti-causal domain shifts. *Advances in Neural Information Processing Systems*, 35:20782–20794, 2022.

F. D. Johansson, D. Sontag, and R. Ranganath. Support and invertibility in domain-invariant representations. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 527–536. PMLR, 2019.

P. Kamath, A. Tangella, D. Sutherland, and N. Srebro. Does invariant risk minimization capture invariance? In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 4069–4077. PMLR, 2021.

P. W. Koh, S. Sagawa, H. Marklund, S. M. Xie, M. Zhang, A. Balsubramani, W. Hu, M. Yasunaga, R. L. Phillips, I. Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, pages 5637–5664. PMLR, 2021.

D. Komura and S. Ishikawa. Machine learning methods for histopathological image analysis. *Computational and Structural Biotechnology Journal*, 16:34–42, 2018.

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

D. Krueger, E. Caballero, J.-H. Jacobsen, A. Zhang, J. Binas, D. Zhang, R. Le Priol, and A. Courville. Out-of-distribution generalization via risk extrapolation (REx). In *International Conference on Machine Learning (ICML)*, pages 5815–5826. PMLR, 2021.

A. Kumar, T. Ma, and P. Liang. Understanding self-training for gradual domain adaptation. In *International Conference on Machine Learning (ICML)*, pages 5468–5479. PMLR, 2020.

Y. LeCun. The mnist database of handwritten digits. *http://yann.lecun.com/exdb/mnist/*, 1998.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Y. Li, M. Gong, X. Tian, T. Liu, and D. Tao. Domain generalization via conditional invariant representations. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 32, 2018a.

Y. Li, X. Tian, M. Gong, Y. Liu, T. Liu, K. Zhang, and D. Tao. Deep domain generalization via conditional invariant adversarial networks. In *European Conference on Computer Vision (ECCV)*, pages 624–639, 2018b.

Z. Lipton, Y.-X. Wang, and A. Smola. Detecting and correcting for label shift with black box predictors. In *International Conference on Machine Learning (ICML)*, pages 3122–3130. PMLR, 2018.

Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3730–3738, 2015.

M. Long, H. Zhu, J. Wang, and M. I. Jordan. Deep transfer learning with joint adaptation networks. In *International Conference on Machine Learning (ICML)*, pages 2208–2217. PMLR, 2017.

M. Long, Z. Cao, J. Wang, and M. I. Jordan. Conditional adversarial domain adaptation. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

S. Magliacane, T. Van Ommen, T. Claassen, S. Bongers, P. Versteeg, and J. M. Mooij. Domain adaptation by using causal inference to predict invariant conditional distributions. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *Conference on Learning Theory (COLT)*, 2009.

X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2794–2802, 2017.

N. Meinshausen. Causality from a distributional robustness point of view. In *IEEE Data Science Workshop (DSW)*, pages 6–10. IEEE, 2018.

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

Y. Oren, S. Sagawa, T. B. Hashimoto, and P. Liang. Distributionally robust language modeling. In *Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4227–4237, 2019.

S. J. Pan, J. T. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 8, pages 677–682, 2008.

S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2010.

J. Pearl. *Causality*. Cambridge University Press, 2009.

X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang. Moment matching for multi-source domain adaptation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1406–1415, 2019.

J. Peters, P. Bühlmann, and N. Meinshausen. Causal inference by using invariant prediction: Identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):947–1012, 2016.

J. Quinonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence. *Dataset Shift in Machine Learning*. MIT Press, 2008.

I. Redko, E. Morvant, A. Habrard, M. Sebban, and Y. Bennani. A survey on domain adaptation theory: Learning bounds and theoretical guarantees. *arXiv preprint arXiv:2004.11829*, 2020.

E. Rosenfeld, P. Ravikumar, and A. Risteski. The risks of invariant risk minimization. In *International Conference on Learning Representations (ICLR)*, 2021.

D. Rothenhäusler, N. Meinshausen, P. Bühlmann, and J. Peters. Anchor regression: Heterogeneous data meet causality. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 83(2):215–246, 2021.

S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020.

S. Sagawa, P. W. Koh, T. Lee, I. Gao, S. M. Xie, K. Shen, A. Kumar, W. Hu, M. Yasunaga, H. Marklund, et al. Extending the wilds benchmark for unsupervised adaptation. *arXiv preprint arXiv:2112.05090*, 2021.

B. Schölkopf, D. Janzing, J. Peters, E. Sgouritsa, K. Zhang, and J. Mooij. On causal and anticausal learning. In *International Conference on Machine Learning (ICML)*, pages 459–466, 2012.

Y. Shi, J. Seely, P. H. S. Torr, N. Siddharth, A. Hannun, N. Usunier, and G. Synnaeve. Gradient matching for domain generalization. In *International Conference on Learning Representations (ICLR)*, 2022.

H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

P. Y. Simard, Y. A. LeCun, J. S. Denker, and B. Victorri. Transformation invariance in pattern recognition: Tangent distance and tangent propagation. In *Neural Networks: Tricks of the Trade*, pages 239–274. Springer, 1998.

M. Sugiyama and M. Kawanabe. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT Press, 2012.

M. Sugiyama, M. Krauledat, and K.-R. Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(5), 2007.

B. Sun and K. Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision (ECCV)*, pages 443–450. Springer, 2016.

R. Tachet des Combes, H. Zhao, Y.-X. Wang, and G. J. Gordon. Domain adaptation with conditional distribution matching and generalized label shift. *Advances in Neural Information Processing Systems*, 33:19276–19289, 2020.

S. Thrun and L. Pratt. *Learning to Learn*. Springer Science & Business Media, 2012.

J. A. Tropp. An introduction to matrix concentration inequalities. *Foundations and Trends in Machine Learning*, 8(1–2):1–230, 2015.

E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7167–7176, 2017.

V. Veitch, A. D'Amour, S. Yadlowsky, and J. Eisenstein. Counterfactual invariance to spurious correlations: Why and how to pass stress tests. In *Conference on Neural Information Processing Systems (NeurIPS)*, pages 16196–16208, 2021.

M. Veta, P. J. Van Diest, M. Jiwa, S. Al-Janabi, and J. P. W. Pluim. Mitosis counting in breast cancer: Object-level interobserver agreement and comparison to an automatic method. *PLOS ONE*, 11(8):e0161286, 2016.

M. J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*, volume 48. Cambridge University Press, 2019.

J. Wang, C. Lan, C. Liu, Y. Ouyang, T. Qin, W. Lu, Y. Chen, W. Zeng, and S. Y. Philip. Generalizing to unseen domains: A survey on domain generalization. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):8052–8072, 2022.

Z. Wang and V. Veitch. A unified causal view of domain invariant representation learning. *arXiv preprint arXiv:2208.06987*, 2022.

C. Wei, K. Shen, Y. Chen, and T. Ma. Theoretical analysis of self-training with deep networks on unlabeled data. In *International Conference on Learning Representations (ICLR)*, 2021.

R. Wu, C. Guo, Y. Su, and K. Q. Weinberger. Online adaptation to label distribution shift. *Advances in Neural Information Processing Systems*, 34:11340–11351, 2021.

X. Wu, Y. Guo, J. Chen, Y. Liang, S. Jha, and P. Chalasani. Representation Bayesian risk decompositions and multi-source domain adaptation. *arXiv preprint arXiv:2004.10390*, 2020.

Y. Wu and P. Yang. Optimal estimation of Gaussian mixtures via denoised method of moments. *The Annals of Statistics*, 48(4):1981–2007, 2020.

Y. Wu, E. Winston, D. Kaushik, and Z. Lipton. Domain adaptation with asymmetrically-relaxed distribution alignment. In *International Conference on Machine Learning (ICML)*, pages 6872–6881. PMLR, 2019.

C. Xie, H. Ye, F. Chen, Y. Liu, R. Sun, and Z. Li. Risk variance penalization. *arXiv preprint arXiv:2006.07544*, 2020.

H. Yao, Y. Wang, S. Li, L. Zhang, W. Liang, J. Zou, and C. Finn. Improving out-of-distribution robustness via selective augmentation. In *International Conference on Machine Learning (ICML)*, pages 25407–25437. PMLR, 2022.

B. Yu and K. Kumbier. Veridical data science. *Proceedings of the National Academy of Sciences*, 117(8):3920–3929, 2020.

J. R. Zech, M. A. Badgeley, M. Liu, A. B. Costa, J. J. Titano, and E. K. Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLOS Medicine*, 15(11):e1002683, 2018.

H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. Mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018.

H. Zhao, R. T. Des Combes, K. Zhang, and G. Gordon. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning (ICML)*, pages 7523–7532. PMLR, 2019.