Model Transferability With Responsive Decision Subjects

Yang Liu^{*1}**, Yatong Chen**^{*2}**, Zeyu Tang**$^{\dagger 3}$**, Kun Zhang**$^{\dagger 4 \ddagger 5}$
*Computer Science and Engineering, University of California, Santa Cruz
†Department of Philosophy, Carnegie Mellon University
‡Machine Learning Department, Mohamed bin Zayed University of Artificial Intelligence
[1]yangliu@ucsc.edu, [2]ychen592@ucsc.edu, [3]zeyutang@cmu.edu,
[4]kunz1@cmu.ed, [5]kun.zhang@mbzuai.ac.ae

Abstract

This paper studies model transferability when human decision subjects respond to a deployed machine learning model. In our setting, an agent or a user corresponds to a sample (X, Y) drawn from a distribution $\mathcal{D}$ and will face a model h and its classification result $h(X)$. Agents can modify X to adapt to h, which will incur a distribution shift on (X, Y). Therefore, when training h, the learner will need to consider the subsequently "induced" distribution when the output model is deployed. Our formulation is motivated by applications where the deployed machine learning models interact with human agents, and will ultimately face *responsive* and *interactive* data distributions. We formalize the discussions of the transferability of a model by studying how the model trained on the available source distribution (data) would translate to the performance on the induced domain. We provide both upper bounds for the performance gap due to the induced domain shift, as well as lower bound for the trade-offs that a classifier has to suffer on either the source training distribution or the induced target distribution. We provide further instantiated analysis for two popular domain adaptation settings with *covariate shift* and *target shift*.

1 Introduction

Decision makers are increasingly required to be transparent on their decision making to offer the "right to explanation" [13, 35, 42] [1]. Being transparent also invites potential adaptations from the population, leading to potential shifts. We are motivated by settings where the deployed machine learning models interact with human agents, which will ultimately face data distributions that reflect how human agents respond to the models. For instance, when a model is used to decide loan applications, candidates may adapt their features based on the model specification in order to maximize their chances of approval; thus the loan decision classifier observes a data distribution caused by its own deployment (e.g., see Figure 1 for a demonstration). Similar observations can be articulated for application in insurance sector (i.e. developing policy s.t. customers' behaviors might adapt to lower premium [16]), education sector (i.e. developing courses when students are less incentivized to cheat [22]) and so on.

This paper investigates model transferability when the underlying distribution shift is induced by the model being deployed. What we would like is to have some guarantee on the *transferability* of a classifier — that is, how training on the available source distribution $\mathcal{D}_S$ translates to performance on the induced domain $\mathcal{D}(h)$, which depends on the model h being deployed. A key concept in our

[1] See Appendix A.1 (supplemental material) for more detailed discussions.

Preprint. Under review.

FEATURE	WEIGHT	||	ORIGINAL VALUE		ADAPTED VALUE
Income	2	||	$ 6,000	⟶	$ 6,000
Education Level	3	||	College	⟶	College
Debt	**-10**	||	$40,000	⟶	**$20,000**
Savings	**5**	||	$20,000	⟶	**$0**

Figure 1: An example of an agent who originally has both savings and debt, observes that the classifier penalizes debt (weight -10) more than it rewards savings (weight +5), and concludes that their most efficient adaptation is to use their savings to pay down their debt.

setting is the *induced risk*, defined as the error a model incurs on the distribution induced by itself:

$$\text{Induced Risk}: \quad \text{Err}_{\mathcal{D}(h)}(h) := \mathbb{P}_{\mathcal{D}(h)}(h(X) \neq Y) \tag{1}$$

Most relevant to the above formulation is the strategic classification literature [17, 7]. In this literature, agents are modeled as rational utility maximizers and game theoretical solutions were proposed to characterize the induced risk. However, our results are motivated by the following challenges in more general scenarios:

- **Modeling assumptions being restrictive** In many practical situations, it is often hard to faithfully characterize agents' utilities. Furthermore, agents might not be fully rational when they response. All the uncertainties can lead to a far more complicated distribution change in (X, Y), as compared to often-made assumptions that agents only change X but not Y [7].
- **Lack of access to response data** Another relevant literature to our work is performative prediction [33]. In performative prediction, one would often require knowing $\mathcal{D}(h)$ or having samples observed from $\mathcal{D}(h)$ through repeated experiments. We posit that machine learning practitioners may only have access to data from the source distribution during training, and although they anticipate changes in the population due to human agents' responses, they cannot observe this new distribution until the model is actually deployed.
- **Retraining being costly** Even when samples from the induced data distribution are available, retraining the model from scratch may be impractical due to computational constraints.

The above observations motivate us to understand the transferability of a model trained on the source data to the domain induced by the deployment of itself. We study several fundamental questions:

- **Source risk $\Rightarrow$ Induced risk** For a given model h, how different is $\text{Err}_{\mathcal{D}(h)}(h)$, the error on the distribution induced by h, from $\text{Err}_{\mathcal{D}_S}(h) := \mathbb{P}_{\mathcal{D}_S}(h(X) \neq Y)$, the error on the source distribution?
- **Induced risk $\Rightarrow$ Minimum induced risk** How much higher is $\text{Err}_{\mathcal{D}(h)}(h)$, the error on the induced distribution, than $\min_{h'} \text{Err}_{\mathcal{D}(h')}(h')$, the minimum achievable induced error?
- **Induced risk of *source optimal* $\Rightarrow$ Minimum induced risk** Of particular interest, and as a special case of the above, how does $\text{Err}_{\mathcal{D}(h^*_S)}(h^*_S)$, the induced error of the optimal model trained on the source distribution $h^*_S := \min_h \text{Err}_{\mathcal{D}_S}(h)$, compare to $\min_h \text{Err}_{\mathcal{D}(h)}(h)$?
- **Lower bound for learning tradeoffs** What is the minimum error a model must incur on either the source distribution $\text{Err}_{\mathcal{D}_S}(h)$ or its induced distribution $\text{Err}_{\mathcal{D}(h)}(h)$?

For the first three questions, we prove upper bounds on the additional error incurred when a model trained on a source distribution is transferred over to its induced domain. We also provide lower bounds for the trade-offs a classifier has to suffer on either the source training distribution or the induced target distribution. We then show how to specialize our results to two popular domain adaptation settings: *covariate shift* [36, 47, 38, 39, 50] and *target shift* [25, 15, 50].

All omitted proofs can be found in the Appendix (supplementary materials).

1.1 Related works

Most relevant to us are three topics: strategic classification [17, 7, 10, 11, 6, 29, 22], a recently proposed notion of *performative prediction* [33, 28], and domain adaptation [20, 3, 39, 51, 21, 49].

Hardt et al. [17] pioneered the formalization of strategic behavior in classification based on a sequential two-player game between agents and classifiers. Subsequently, Chen et al. [7] addressed the question of repeatedly learning linear classifiers against agents who are strategically trying to game the deployed classifiers. Most of the existing literature focuses on finding the optimal classifier by assuming fully rational agents (and by characterizing the equilibrium response). In contrast, we do not make these assumptions and primarily study the transferability when only having knowledge of source data.

Perdomo et al. [33] advocate minimizing the error $\mathbb{E}_{z\sim\mathcal{D}(\theta)}[\ell(\theta;z)]$, which they call the *performative prediction risk*: in above θ is the model parameter to be optimized. This falls into the same category as induced risk, but the two concepts differ due to different requirements of knowing the distribution of a $\mathcal{D}(\theta)$. Again, our focus is on the transferability when learning on the source training data. In addition, we study specific domain adaptation setting which generally do not assume the knowledge of $\mathcal{D}(\theta)$ (and particularly we will not assume the knowledge of the supervision/label information on the transferred domain).

Our result was inspired by the transferability results in domain adaptations [3, 8, 9]. Later works examined specific domain adaptation models, such as covariate shift [36, 47, 14, 39, 50, 48] and target/label shift [25, 2]. A commonly established solution is to perform reweighted training on the source data, and robust and efficient solutions have been developed to estimate the weights accurately [39, 50, 48, 25, 15].

Our work, at the first sight, looks similar to several other area of studies. For instance, the notion of observing an "induced distribution" resembles similarity to the adversarial machine learning literature [27, 19, 44]. One of the major differences between us and adversarial machine learning is the true label Y stays the same for the attacked feature while in our paper, both X and Y might change in the adapted distribution $\mathcal{D}(h)$. In Appendix A.2, we provide detailed comparisons with some areas in domain adaptations, including domain generalization, adversarial attack and test-time adaptation.

2 Formulation

Suppose we are learning a parametric model $h \in \mathcal{H}$ for a binary classification problem. Its training data set $S := \{x_i, y_i\}_{i=1}^N$ is drawn from a *source* distribution $\mathcal{D}_S$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$. However, h will then be deployed in a setting where the samples come from a *test* or *target* distribution $\mathcal{D}_T$ that can differ substantially from $\mathcal{D}_S$. Therefore instead of minimizing the prediction error on the source distribution $\text{Err}_{\mathcal{D}_S}(h) := \mathbb{P}_{\mathcal{D}_S}(h(X) \neq Y)$, the goal is to find h^* that minimizes $\text{Err}_{\mathcal{D}_T}(h) := \mathbb{P}_{\mathcal{D}_T}(h(X) \neq Y)$. This is often referred to as the *domain adaptation problem*, where typically, the transition from $\mathcal{D}_S$ to $\mathcal{D}_T$ is assumed to be independent of the model h being deployed.

We consider a setting in which the distribution shift depends on h, or is thought of as being *induced* by h. We will use $\mathcal{D}(h)$ to denote the *induced domain* by h:

$$\mathcal{D}_S \;\rightarrow\; \textit{encounters model } h \;\rightarrow\; \mathcal{D}(h)$$

Strictly speaking, the induced distribution is a function of both $\mathcal{D}_S$ and h and should be better denoted by $\mathcal{D}_S(h)$. To ease the notation, we will stick with $\mathcal{D}(h)$, but we shall keep in mind of its dependency of $\mathcal{D}_S$. For now, we do not restrict the dependency of $\mathcal{D}(h)$ of $\mathcal{D}$ and h, but later in Section 4 and 5 we will further instantiate $\mathcal{D}(h)$ under specific domain adaptation settings.

The challenge in the above setting is that when training h, the learner needs to carry the thoughts that $\mathcal{D}(h)$ should be the distribution it will be evaluated on and that the training cares about. Formally, we define the *induced risk* of a classifier h as the 0-1 error on the distribution h induces:

$$\text{Induced risk}: \quad \text{Err}_{\mathcal{D}(h)}(h) := \mathbb{P}_{\mathcal{D}(h)}(h(X) \neq Y) \tag{2}$$

Denote by $h_T^* := \arg\min_{h\in\mathcal{H}} \text{Err}_{\mathcal{D}(h)}(h)$ the classifier with minimum induced risk. More generally, when the loss may not be the 0-1 loss, we define the *induced ℓ-risk* as

$$\text{Induced } \ell\text{-risk}: \quad \text{Err}_{\ell,\mathcal{D}(h)}(h) := \underset{h\in\mathcal{H}}{\arg\min}\; \mathbb{E}_{z\sim\mathcal{D}(h)}[\ell(h;z)]$$

The induced risks will be the primary quantities that we are interested in minimizing. The following additional notation will also be helpful:

- Distributions of Y on a distribution $\mathcal{D}$: $\mathcal{D}_Y := \mathbb{P}_{\mathcal{D}}(Y = y)$[2], and in particular $\mathcal{D}_Y(h) := \mathbb{P}_{\mathcal{D}(h)}(Y = y)$, $\mathcal{D}_{Y|S} := \mathbb{P}_{\mathcal{D}_S}(Y = y)$.
- Distribution of h on a distribution $\mathcal{D}$: $\mathcal{D}_h := \mathbb{P}_{\mathcal{D}}(h(X) = y)$, and in particular $\mathcal{D}_h(h) := \mathbb{P}_{\mathcal{D}(h)}(h(X) = y)$, $\mathcal{D}_{h|S} := \mathbb{P}_{\mathcal{D}_S}(h(X) = y)$.
- Marginal distribution of X for a distribution $\mathcal{D}$: $\mathcal{D}_X := \mathbb{P}_{\mathcal{D}}(X = x)$, and in particular $\mathcal{D}_X(h) := \mathbb{P}_{\mathcal{D}(h)}(X = x)$, $\mathcal{D}_{X|S} := \mathbb{P}_{\mathcal{D}_S}(X = x)$[3].
- Total variation distance defined between $\mathcal{D}$ and $\mathcal{D}'$ [1]: $d_{\text{TV}}(\mathcal{D}, \mathcal{D}') := \sup_{\mathcal{O}} |\mathbb{P}_{\mathcal{D}}(\mathcal{O}) - \mathbb{P}_{\mathcal{D}'}(\mathcal{O})|$.

2.1 Examples of Distribution Shifts Induced by Model Deployment

We provide two exemplery models to demonstrate the use cases for the distribution shift models described in our paper. We provide more details in Section 4.3 and Section 5.3.

Strategic Classification An example of distribution shift is the setting where decision subjects perform *strategic response* to a decision rule. It is well-known that when human agents are subject to a decision rule, they will adapt their feature so as to get a favorable prediction outcome. In the literature of strategic classification, we say the human agents perform strategic adaptation [17].

It is natural to assume that the feature distribution before and after the human agents' best response satisfies *covariate shift*: namely the feature distribution $\mathbb{P}(X)$ will change, but $\mathbb{P}(Y|X)$, the mapping between Y and X, remain unchanged. We use Figure 2 (Left) as a demonstrating of how distribution might shift for strategic response setting. In Section 4.3, we will use the strategic classification setup to verify our obtained results.

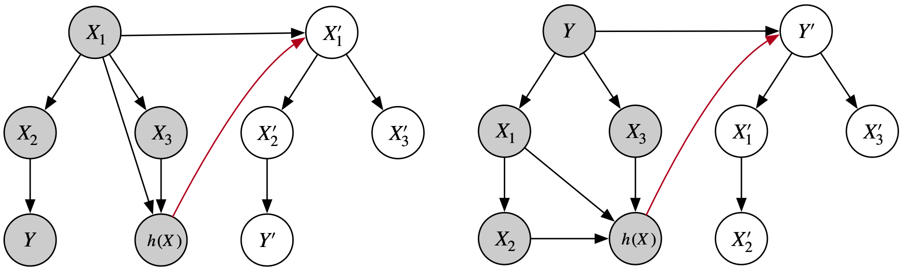


Figure 2: Example causal graph annotated to demonstrate covariate shift (**Left**) / target shift (**Right**) as a result of the deployment of h. Grey nodes indicate observable variables and transparent nodes are not observed at the training stage. Red arrow emphasises h induces changes of certain variables.

Replicator Dynamics Replicator dynamics is a commonly used model to study the evolution of an adopted "strategy" in evolutionary game theory [41, 12, 40, 34]. The core notion of it is the growth or decline of the population of each strategy depends on its "fitness". Consider the label $Y = \{-1, +1\}$ as the strategy, and the following behavioral response model to capture the induced target shift:

$$\frac{\mathbb{P}_{\mathcal{D}(h)}(Y = +1)}{\mathbb{P}_{\mathcal{D}_S}(Y = +1)} = \frac{\textbf{Fitness}(Y = +1)}{\mathbb{E}[\textbf{Fitness}(Y)]}$$

In short, the change of the $Y = +1$ population depends on how predicting $Y = +1$ "fits" a certain utility function. For instance, the "fitness" can take the form of the prediction accuracy of h for class $+1$. With assuming $\mathbb{P}(X|Y)$ stays unchanged, this instantiates one example of a specific induced *target shift*. We will specify the condition for target shift in Section 5. We use Figure 2 (Right) as a demonstrating of how distribution might shift for the replicator dynamic setting. In Section 5.3, we will use a detailed replicator dynamics model to further instantiate our results.

[2] The ":=" defines the RHS as the probability measure function for the LHS.

[3] For continuous X, the probability measure shall be read as the density function.

3 Transferability of Learning to Induced Domains

In this section, we first provide upper bounds for the transfer error of a classifier h (that is, the difference between $\mathrm{Err}_{\mathcal{D}(h)}(h)$ and $\mathrm{Err}_{\mathcal{D}_S}(h)$), as well as between $\mathrm{Err}_{\mathcal{D}(h)}(h)$ and $\mathrm{Err}_{\mathcal{D}(h^*_T)}(h^*_T)$. We then provide lower bounds for $\max\{\mathrm{Err}_{\mathcal{D}_S}(h), \mathrm{Err}_{\mathcal{D}(h)}(h)\}$; that is, the minimum error a model h must incur on either the source distribution $\mathcal{D}_S$ or the induced distribution $\mathcal{D}(h)$.

3.1 Upper Bound

We first investigate upper bounds for the transfer errors. We begin by showing generic upper bounds, and further strengthen the bound for specific domain adaptation settings in Section 4 and 5 . We begin with answering a central question in domain adaptation:

How does a model h trained on its training data set fare on the induced distribution $\mathcal{D}(h)$?

To that end, define the minimum and maximum combined error of two distributions $\mathcal{D}$ and $\mathcal{D}'$ as:

$$\lambda_{\mathcal{D}\to\mathcal{D}'} := \min_{h'\in\mathcal{H}} \mathrm{Err}_{\mathcal{D}'}(h') + \mathrm{Err}_{\mathcal{D}}(h'),\ \Lambda_{\mathcal{D}\to\mathcal{D}'} := \max_{h'\in\mathcal{H}} \mathrm{Err}_{\mathcal{D}'}(h') + \mathrm{Err}_{\mathcal{D}}(h')$$

and the $\mathcal{H}$-divergence [3] as

$$d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D},\mathcal{D}') = 2 \sup_{h,h'\in\mathcal{H}} |\mathbb{P}_{\mathcal{D}}(h(X)\neq h'(X)) - \mathbb{P}_{\mathcal{D}'}(h(X)\neq h'(X))|$$

The $\mathcal{H}$-divergence is celebrated measure proposed in the domain adaptation literature [3] which will be useful for bounding the difference in errors of two classifiers. Repeating classical arguments from [3], we can easily prove the following:

Theorem 3.1 (Source risk ⇒ Induced risk). *The difference between $Err_{\mathcal{D}(h)}(h)$ and $Err_{\mathcal{D}_S}(h)$ is upper bounded by: $Err_{\mathcal{D}(h)}(h) \leq Err_{\mathcal{D}_S}(h) + \lambda_{\mathcal{D}_S\to\mathcal{D}(h)} + \frac{1}{2}d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}_S,\mathcal{D}(h))$.*

The transferability of a model h between $\mathrm{Err}_{\mathcal{D}(h)}(h)$ and $\mathrm{Err}_{\mathcal{D}_S}(h)$ looks precisely the same as in the classical domain adaptation setting [3]. The above practice informs us that the classical transferability bounds under domain adaptation still hold when the adaptation is induced by the model too. Nonetheless, an arguably more interesting quantity in our setting to understand is the difference between the induced error of a given model h and the error induced by a globally optimal model:

$$\mathrm{Err}_{\mathcal{D}(h)}(h) - \mathrm{Err}_{\mathcal{D}(h^*_T)}(h^*_T) \tag{3}$$

The proof is slightly more involved, and the bound differs from the one in Theorem 3.1:

Theorem 3.2 (Induced risk ⇒ Minimum induced risk). *The difference between $Err_{\mathcal{D}(h)}(h)$ and $Err_{\mathcal{D}(h^*_T)}(h^*_T)$ is upper bounded by:*

$$Err_{\mathcal{D}(h)}(h) - Err_{\mathcal{D}(h^*_T)}(h^*_T) \leq \frac{\lambda_{\mathcal{D}(h)\to\mathcal{D}(h^*_T)} + \Lambda_{\mathcal{D}(h)\to\mathcal{D}(h^*_T)}}{2} + \frac{1}{2}\cdot d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h^*_T),\mathcal{D}(h)).$$

The above theorem informs us that the induced transfer error is bounded by the "average" achievable error on both distributions $\mathcal{D}(h)$ and $\mathcal{D}(h^*_T)$, as well as the $\mathcal{H}\times\mathcal{H}$ divergence between the two distributions. Reflecting on the difference between the bounds of Theorem 3.1 and Theorem 3.2, we see that the primary change is replacing the minimum achievable error λ with the average of λ and Λ.

3.2 Lower Bound

Now we provide a lower bound on the induced transfer error. We particularly want to show that at least one of the two errors $\mathrm{Err}_{\mathcal{D}_S}(h)$, $\mathrm{Err}_{\mathcal{D}(h)}(h)$ must be lower-bounded by a certain quantity.

Theorem 3.3 (Lower bound for learning tradeoffs). *Any model h must incur the following error on either the source or induced distribution:*

$$\max\{Err_{\mathcal{D}_S}(h), Err_{\mathcal{D}(h)}(h)\} \geq \frac{d_{TV}(\mathcal{D}_{Y|S},\mathcal{D}_Y(h)) - d_{TV}(\mathcal{D}_{h|S},\mathcal{D}_h(h))}{2}.$$

The proof leverages the triangle inequality of d_{TV}. This bound is dependent on h; however, by the data processing inequality of d_{TV} (and f-divergence functions in general) [24], we have

$$d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) \le d_{\text{TV}}(\mathcal{D}_{X|S}, \mathcal{D}_X(h))$$

Applying this to Theorem 3.3 gives the following model-independent bound:

Corollary 3.4. *For any model h,*

$$\max\{Err_{\mathcal{D}_S}(h), Err_{\mathcal{D}(h)}(h)\} \ge \frac{d_{TV}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{TV}(\mathcal{D}_{X|S}, \mathcal{D}_X(h))}{2}.$$

A couple of remarks:

- Without further assumptions, it is unclear if $d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) \ge 0$. After introducing specific domain adaption settings, we will revisit this bound.
- When $d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{X|S}, \mathcal{D}_X(h)) > 0$, we know there is a positive tradeoff between a model's achievable training error on the source distribution and the induced distribution.

4 Covariate Shift

In this section, we focus on a particular domain adaptation setting known as *covariate shift*, in which the distribution of features changes, but the distribution of labels conditioned on features does not:

$$\mathbb{P}_{\mathcal{D}(h)}(Y = y|X = x) = \mathbb{P}_{\mathcal{D}_S}(Y = y|X = x), \;\; \mathbb{P}_{\mathcal{D}(h)}(X = x) \ne \mathbb{P}_{\mathcal{D}_S}(X = x) \tag{4}$$

Thus with covariate shift, we have

$$\begin{aligned}\mathbb{P}_{\mathcal{D}(h)}(X = x, Y = y) =& \mathbb{P}_{\mathcal{D}(h)}(Y = y|X = x) \cdot \mathbb{P}_{\mathcal{D}(h)}(X = x)\\ =& \mathbb{P}_{\mathcal{D}_S}(Y = y|X = x) \cdot \mathbb{P}_{\mathcal{D}(h)}(X = x)\end{aligned}$$

Let $\omega_x(h) := \frac{\mathbb{P}_{\mathcal{D}(h)}(X=x)}{\mathbb{P}_{\mathcal{D}_S}(X=x)}$ be the *importance weight* at x, which characterizes the amount of adaptation induced by h at instance x. Then for any loss function ℓ we have

Proposition 4.1 (Expected Loss on the New Distribution)**.**

$$\mathbb{E}_{\mathcal{D}(h)}[\ell(h; X, Y)] = \mathbb{E}_{\mathcal{D}_S}[\omega_x(h) \cdot \ell(h; x, y)].$$

The above derivation was not new and offered the basis for performing importance reweighting when learning under coviarate shift [39]. The particular form informs us that $\omega_x(h)$ controls the generation of $\mathcal{D}(h)$ and encodes its dependency of both $\mathcal{D}_S$ and h, and is critical for deriving our results below.

4.1 Upper Bound

We now derive an upper bound for transferability under covariate shift. We will focus particularly on the optimal model trained on the source data $\mathcal{D}_S$, which we denote as $h^*_S := \arg\min_{h\in\mathcal{H}} \text{Err}_S(h)$. Recall that the classifier with minimum induced risk is denoted as $h^*_T := \arg\min_{h\in\mathcal{H}} \text{Err}_{\mathcal{D}(h)}(h)$. We can upper bound the difference between h^*_S and h^*_T as follows:

Theorem 4.2 (Suboptimality of h^*_S)**.** *Let X be distributed according to $\mathcal{D}_S$. We have:*

$$Err_{\mathcal{D}(h^*_S)}(h^*_S) - Err_{\mathcal{D}(h^*_T)}(h^*_T) \le \sqrt{Err_{\mathcal{D}_S}(h^*_T)} \cdot \left(\sqrt{Var(\omega_X(h^*_S))} + \sqrt{Var(\omega_X(h^*_T))}\right).$$

This result can can be interpreted as follows: h^*_T incurs an irreducible amount of error on the source data set, represented by $\sqrt{\text{Err}_{\mathcal{D}_S}(h^*_T)}$. Moreover, the difference in error between h^*_S and h^*_T is at its maximum when the two classifiers induce adaptations in "opposite" directions; this is represented by the sum of the standard deviations of their importance weights, $\sqrt{\text{Var}(\omega_X(h^*_S))} + \sqrt{\text{Var}(\omega_X(h^*_T))}$.

4.2 Lower Bound

Recall from Theorem 3.3, for the general setting, it is unclear whether the lower bound is strictly positive or not. In this section, we provide further understanding for when the lower bound $\frac{d_{\text{TV}}(\mathcal{D}_{Y|S},\mathcal{D}_Y(h))-d_{\text{TV}}(\mathcal{D}_{h|S},\mathcal{D}_h(h))}{2}$ is indeed positive under covariate shift.

We show under several assumptions, our previously provided lower bound in Theorem 3.3 is indeed strictly positive in the covarite shift setting. Details of the required conditions are specified in the Appendix, but the intuitions of the conditions are:

- Increased $\omega_x(h)$ value points are more likely to have positive labels. (Assumption A.3)
- Increased $\omega_x(h)$ value points are more likely to be classified as positive instances. (Assumption A.4)
- For a classifier h, within all $h(X) = +1$ or $h(X) = -1$, a higher $\mathbb{P}_{\mathcal{D}}(Y = +1|X = x)$ associates with a higher $\omega_x(h)$. (Assumption A.5)

Theorem 4.3. *With A.3 - A.5, the following lower bound is strictly positive for covariate shift:*

$$\max\{\mathit{Err}_{\mathcal{D}_S}(h), \mathit{Err}_{\mathcal{D}(h)}(h)\} \geq \frac{d_{TV}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{TV}(\mathcal{D}_{h|S}, \mathcal{D}_h(h))}{2} > 0.$$

4.3 Example Using Strategic Classification

As introduced in Section 2.1, we consider a setting caused by *strategic response* in which agents are classified by and adapt to a binary threshold classifier.

Consider a setup where each agent is associated with a d dimensional continuous feature $x \in \mathbb{R}^d$ and a binary true qualification $y(x) \in \{-1, +1\}$, where $y(x)$ is a function of the feature vector x.

Consistent with the literature in strategic classification [17], a simple case where after seeing the threshold binary decision rule $h(x) = 2 \cdot \mathbb{1}[x \geq \tau_h] - 1$, the agents will *best response* to it by maximizing the following utility function:

$$u(x, x') = h(x') - h(x) - c(x, x')$$

where $c(x, x')$ is the *cost function* for decision subjects to modify their feature from x to x'.

Assume all agents are rational utility maximizers: they will only *attempt* to change their features when the benefit of manipulation is greater than the cost (i.e. when $c(x, x') \leq 2$) and agent will not change their feature if they are already accepted (i.e. $h(x) = +1$). For a given threshold τ_h and manipulation budget B, the theoretical best response of an agent with original feature x is:

$$\Delta(x) = \arg\max_{x'} u(x, x') \;\; s.t. \; c(x, x') \leq B \tag{5}$$

We show that under some further characterizations of the agents' responsive behaviors (see Assumption A.6 - A.9 in Appendix A.9), we can specify the bound in Theorem 4.2 for the strategic response setting as follows:

Proposition 4.4 (Upper bound for the Strategic Response Setting)**.** *Under assumption Assumption A.6 - A.9, we can bound the differences between $\mathit{Err}_{\mathcal{D}(h_S^*)}(h_S^*)$ and $\mathit{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$ by*

$$\mathit{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \mathit{Err}_{\mathcal{D}(h_T^*)}(h_T^*) \leq \sqrt{\frac{2B}{3}\mathit{Err}_{\mathcal{D}_S}(h_T^*)}.$$

To interpret this result, we can see that the upper bound for strategic response depends on the manipulation budget B, and the error the ideal classifier made on the source distribution $\text{Err}_{\mathcal{D}_S}(h_T^*)$. This aligns with our intuition that the smaller manipulation budget is, the less agents will change their features, thus leading to a tighter upper bound on the difference between $\text{Err}_{h_S^*}(h_S^*)$ and $\text{Err}_{h_T^*}(h_T^*)$. This bound also allows us to bound this quantity even without the knowledge of the mapping between $\mathcal{D}(h)$ and h, since we can directly compute $\text{Err}_{\mathcal{D}_S}(h_T^*)$ from the source distribution and an estimated optimal classifier h_T^*.

5 Target Shift

We consider another popular domain adaptation setting known as *target shift*, in which the distribution of labels changes, but not the distribution of features conditioned on the label:

$$\mathbb{P}_{\mathcal{D}(h)}(X = x|Y = y) = \mathbb{P}_{\mathcal{D}_S}(X = x|Y = y), \ \ \mathbb{P}_{\mathcal{D}(h)}(Y = y) \neq \mathbb{P}_{\mathcal{D}_S}(Y = y) \tag{6}$$

In the case of binary classification, let $\omega(h) := \mathbb{P}_{\mathcal{D}(h)}(Y = +1)$, and $\mathbb{P}_{\mathcal{D}(h)}(Y = -1) = 1 - \omega(h)$. Again, $\omega(h)$ encodes the induced adaptation from $\mathcal{D}_S$ and h. Then we have for any proper loss function ℓ:

$$\begin{aligned}\mathbb{E}_{\mathcal{D}(h)}[\ell(h; X, Y)] =&\omega(h) \cdot \mathbb{E}_{\mathcal{D}(h)}[\ell(h; X, Y)|Y = +1] + (1 - \omega(h)) \cdot \mathbb{E}_{\mathcal{D}(h)}[\ell(h; X, Y)|Y = -1] \\ =&\omega(h) \cdot \mathbb{E}_{\mathcal{D}_S}[\ell(h; X, Y)|Y = +1] + (1 - \omega(h)) \cdot \mathbb{E}_{\mathcal{D}_S}[\ell(h; X, Y)|Y = -1]\end{aligned}$$

We will adopt the following shorthands:

$$\text{Err}_+(h) := \mathbb{E}_{\mathcal{D}_S}[\ell(h; X, Y)|Y = +1], \ \ \text{Err}_-(h) := \mathbb{E}_{\mathcal{D}_S}[\ell(h; X, Y)|Y = -1]$$

Note that $\text{Err}_+(h), \text{Err}_-(h)$ are both defined on the conditional source distribution, which is invariant under the target shift assumption.

5.1 Upper bound

We again upper bound the transferability of h^*_S under target shift. Denote by $\mathcal{D}_+$ the positive label distribution on $\mathcal{D}_S$ ($\mathbb{P}_{\mathcal{D}_S}(X = x|Y = +1)$) and $\mathcal{D}_-$ the negative label distribution on $\mathcal{D}_S$ ($\mathbb{P}_{\mathcal{D}_S}(X = x|Y = -1)$). Let $p := \mathbb{P}_{\mathcal{D}_S}(Y = +1)$.

Theorem 5.1. *For target shift, the difference between $Err_{\mathcal{D}(h^*_S)}(h^*_S)$ and $Err_{\mathcal{D}(h^*_T)}(h^*_T)$ bounds as:*

$$\begin{aligned}Err_{\mathcal{D}(h^*_S)}(h^*_S) - Err_{\mathcal{D}(h^*_T)}(h^*_T) \leq& \ |\omega(h^*_S) - \omega(h^*_T)| \\ &+ (1 + p) \cdot (d_{TV}(\mathcal{D}_+(h^*_S), \mathcal{D}_+(h^*_T)) + d_{TV}(\mathcal{D}_-(h^*_S), \mathcal{D}_-(h^*_T))) \, .\end{aligned}$$

The above upper bound consists of two components. The first quantity captures the difference between the two induced distributions $\mathcal{D}(h^*_S)$ and $\mathcal{D}(h^*_T)$. The second quantity characterizes the difference between the two classifiers h^*_S, h^*_T on the source distribution.

5.2 Lower Bound

Now we discuss lower bounds. Denote by $\text{TPR}_S(h)$ and $\text{FPR}_S(h)$ the true positive and false positive rates of h on the source distribution $\mathcal{D}_S$. We prove the following:

Theorem 5.2. *For target shift, any model h must incur the following error on either $\mathcal{D}_S$ or $\mathcal{D}(h)$:*

$$\max\{Err_{\mathcal{D}_S}(h), Err_{\mathcal{D}(h)}(h)\} \geq \frac{|p - \omega(h)| \cdot (1 - |TPR_S(h) - FPR_S(h)|)}{2}.$$

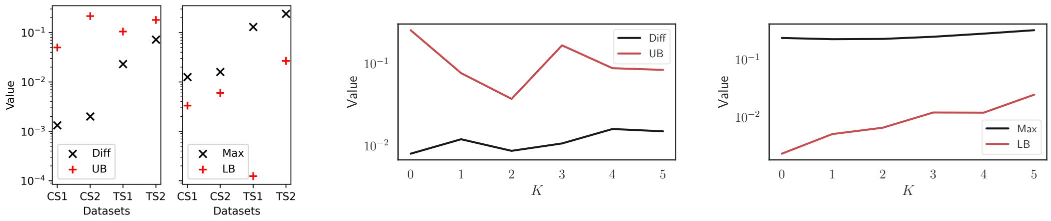
The proof extends the bound of Theorem 3.3 by further explicating each of $d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h))$, $d_{\text{TV}}(\mathcal{D}_{h|S}$, and $\mathcal{D}_h(h))$ under the assumption of target shift. Since $|\text{TPR}_S(h) - \text{FPR}_S(h)| < 0$ unless we have a trivial classifier that has either $\text{TPR}_S(h) = 1, \text{FPR}_S(h) = 0$ or $\text{TPR}_S(h) = 0, \text{FPR}_S(h) = 1$, the lower bound is strictly positive. Taking a closer look, the lower bound is determined linearly by how much the label distribution shifts: $p - \omega(h)$. The difference is further determined by the performance of h on the source distribution through $1 - |\text{TPR}_S(h) - \text{FPR}_S(h)|$. For instance, when $\text{TPR}_S(h) > \text{FPR}_S(h)$, the quality becomes $\text{FNR}_S(h) + \text{FPR}_S(h)$, that is the more error h makes, the larger the lower bound will be.

5.3 Example Using Replicator Dynamics

Let us instantiate the discussion using a specific fitness function for the replicator dynamics model (Section 2.1), which is the prediction accuracy of h for class $+1$:

$$\textbf{Fitness}(Y = y) := \mathbb{P}_{\mathcal{D}_S}(h(X) = y|Y = y) \tag{7}$$

Then we have $\mathbb{E}\left[\textbf{Fitness}(Y)\right] = 1 - \text{Err}_{\mathcal{D}_S}(h)$, and $\frac{\omega(h)}{\mathbb{P}_{\mathcal{D}_S}(Y=+1)} = \frac{\Pr_{\mathcal{D}_S}(h(X)=+1|Y=+1)}{1-\text{Err}_{\mathcal{D}_S}(h)}$. Plugging the result back to our Theorem 5.1 we have



(a) results on simulated data

(b) results of synthetic experiments on FICO credit score data set

Figure 3: Results for synthetic experiments on simulated and real-world data. Diff := $\text{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$, Max := $\max\{\text{Err}_{\mathcal{D}_S}(h_T^*), \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)\}$, UB := upper bound specified in Theorem 4.2 for covarite shift or in Theorem 5.1 for target shift, LB := lower bound specified in Theorem 4.3 for covariate shift or in Theorem 5.2 for target shift.

Proposition 5.3. *Under the replicator dynamics model in Eqn. (7), $|\omega(h_S^*) - \omega(h_T^*)|$ bounds as:*

$$|\omega(h_S^*) - \omega(h_T^*)| \leq \mathbb{P}_{\mathcal{D}_S}(Y = +1) \cdot \frac{|\mathit{Err}_{\mathcal{D}_S}(h_S^*) - \mathit{Err}_{\mathcal{D}_S}(h_T^*)| \cdot |\mathit{TPR}_S(h_S^*) - \mathit{TPR}_S(h_T^*)|}{(1 - \mathit{Err}_{\mathcal{D}_S}(h_S^*)) \cdot (1 - \mathit{Err}_{\mathcal{D}_S}(h_T^*))}.$$

That is, the difference between $\text{Err}_{\mathcal{D}(h_S^*)}(h_S^*)$ and $\text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$ is further dependent on the difference between the two classifiers' performances on the source data $\mathcal{D}_S$. This offers an opportunity to evaluate the possible error transferability using the source data only.

6 Experiments

We present synthetic experimental results on both simulated and real-world data sets.

Synthetic experiments using simulated data We generate synthetic data sets from structural equation models described on simple causal DAG in Figure 2 for covariate shift and target shift. To generate the induced distribution $\mathcal{D}(h)$, we posit a specific *adaptation function* $\Delta : \mathbb{R}^d \times \mathcal{H} \to \mathbb{R}^d$, so that when an input x encounters classifier $h \in \mathcal{H}$, its induced features are precisely $x' = \Delta(x, h)$. We provide details of the data generation processes and adaptation functions in Appendix B.

We take our training data set $\{x_1, \ldots, x_n\}$ and learn a "base" logistic regression model $h(x) = \sigma(w \cdot x)$[4]. We then consider the hypothesis class $\mathcal{H} := \{h_\tau \mid \tau \in [0, 1]\}$, where $h_\tau(x) := 2 \cdot \mathbb{1}[\sigma(w \cdot x) > \tau] - 1$. To compute h_S^*, the model that performs best on the source distribution, we simply vary τ and take the h_τ with lowest prediction error. Then, we posit a specific adaptation function $\Delta(x, h_\tau)$. Finally, to compute h_T^*, we vary τ from 0 to 1 and find the classifier h_τ that minimizes the prediction error on its induced data set $\{\Delta(x_1, h_\tau), \ldots, \Delta(x_n, h_\tau)\}$. We report our results in Figure 3 (a).

Synthetic experiments using real-world data The FICO credit score data set [4] contains more than 300k records of TransUnion credit score of clients from different demographic groups. For our experiment on the preprocessed FICO data set [18], we convert the cumulative distribution function (CDF) of TransRisk score among different groups into group-wise credit score densities, from which we generate a balanced sample to represent a population where groups have equal representations. We demonstrate the application of our results in a series of resource allocations.

Similar to the synthetic experiments on simulated data, we consider the hypothesis class of threshold classifiers and treat the classification outcome as the decision received by individuals. For each time step $K = k$, we compute h_S^*, the statistical optimal classifier on the source distribution (i.e., the current reality for step $K = k$), and update the credit score for each individual according to the received decision as the new reality for time step $K = k + 1$. Details of the data generation is again deferred to Appendix B. We report our results in Figure 3 (b).

For both sets of our results, we do observe positive gaps $\text{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$, indicating the suboptimality of training on $\mathcal{D}_S$. The gaps are well bounded by the theoretical upper bound

[4] $\sigma(\cdot)$ is the logistic function and $w \in \mathbb{R}^3$ denotes the weights.

(UB). Our lower bounds (LB) do return meaningful positive gaps, demonstrating the trade-offs that a classifier has to suffer on either the source distribution or the induced target distribution.

7 Concluding Remarks

We presented a sequence of model transferability results for settings where agents will respond to a deployed model. The response leads to an induced distribution that the learner would not know before deploying the model. Our results cover for both a general response setting and for specific ones (covariate shift and target shift). Our paper ends with empirical results that supported our claims. Unawareness of the potential distribution shift might lead to unintended consequence when training a machine learning model. One goal of this paper is to raise awareness of this issue for a safe deployment of machine learning methods in high-stake societal applications.

A subset of our results are developed under assumptions (e.g., Theorem 4.3). Therefore we want to caution readers of potential misinterpretation of applicability of the reported theoretical guarantees. Our contributions are mostly theoretical and our experiments use synthetic agent models to simulate distribution shift. A future direction is to collect real human experiment data to support the findings.

References

[1] Syed Mumtaz Ali and Samuel D Silvey. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society: Series B (Methodological)*, 28(1):131–142, 1966.

[2] Kamyar Azizzadenesheli, Anqi Liu, Fanny Yang, and Animashree Anandkumar. Regularized learning for domain adaptation under label shifts. *arXiv preprint arXiv:1903.09734*, 2019.

[3] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2010.

[4] Board of Governors of the Federal Reserve System (US). *Report to the congress on credit scoring and its effects on the availability and affordability of credit*. Board of Governors of the Federal Reserve System, 2007.

[5] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey, 2018.

[6] Yatong Chen, Jialu Wang, and Yang Liu. Strategic recourse in linear classification. *arXiv preprint arXiv:2011.00355*, 2020.

[7] Yiling Chen, Yang Liu, and Chara Podimata. Learning strategy-aware linear classifiers, 2020.

[8] Koby Crammer, Michael Kearns, and Jennifer Wortman. Learning from multiple sources. *Journal of Machine Learning Research*, 9(8), 2008.

[9] Shai Ben David, Tyler Lu, Teresa Luu, and Dávid Pál. Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 129–136. JMLR Workshop and Conference Proceedings, 2010.

[10] Ofer Dekel, Felix Fischer, and Ariel D. Procaccia. Incentive compatible regression learning. *J. Comput. Syst. Sci.*, 76(8):759–777, December 2010.

[11] Jinshuo Dong, Aaron Roth, Zachary Schutzman, Bo Waggoner, and Zhiwei Steven Wu. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, EC '18, page 55–70, New York, NY, USA, 2018. Association for Computing Machinery.

[12] Daniel Friedman and Barry Sinervo. *Evolutionary games in natural, social, and virtual worlds*. Oxford University Press, 2016.

[13] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a “right to explanation”. *AI Magazine*, 38(3):50–57, Oct 2017.

[14] Arthur Gretton, Alex Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, and Bernhard Schölkopf. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4):5, 2009.

[15] Jiaxian Guo, Mingming Gong, Tongliang Liu, Kun Zhang, and Dacheng Tao. Ltf: A label transformation framework for correcting label shift. In *International Conference on Machine Learning*, pages 3843–3853. PMLR, 2020.

[16] Nika Haghtalab, Nicole Immorlica, Brendan Lucier, and Jack Z. Wang. Maximizing welfare with incentive-aware evaluation mechanisms. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 160–166. International Joint Conferences on Artificial Intelligence Organization, 2020.

[17] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, page 111–122, New York, NY, USA, 2016. Association for Computing Machinery.

[18] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, pages 3315–3323, 2016.

[19] Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and J Doug Tygar. Adversarial machine learning. In *ACM Workshop on Security and Artificial Intelligence*, pages 43–58, 2011.

[20] Jing Jiang. A literature survey on domain adaptation of statistical classifiers. *URL: http://sifaka. cs. uiuc. edu/jiang4/domainadaptation/survey*, 3:1–12, 2008.

[21] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4893–4902, 2019.

[22] Jon Kleinberg and Manish Raghavan. How do classifiers induce agents to invest effort strategically? *ACM Transactions on Economics and Computation (TEAC)*, 8(4):1–23, 2020.

[23] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize: Meta-learning for domain generalization, 2017.

[24] Friedrich Liese and Igor Vajda. On divergences and informations in statistics and information theory. *IEEE Transactions on Information Theory*, 52(10):4394–4412, 2006.

[25] Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pages 3122–3130. PMLR, 2018.

[26] Yang Liu and Mingyan Liu. An online learning approach to improving the quality of crowdsourcing. *ACM SIGMETRICS Performance Evaluation Review*, 43(1):217–230, 2015.

[27] Daniel Lowd and Christopher Meek. Adversarial learning. In *ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, pages 641–647, 2005.

[28] Celestine Mendler-Dünner, Juan Perdomo, Tijana Zrnic, and Moritz Hardt. Stochastic optimization for performative prediction. In *Advances in Neural Information Processing Systems*, pages 4929–4939. Curran Associates, Inc., 2020.

[29] John Miller, Smitha Milli, and Moritz Hardt. Strategic classification is causal modeling in disguise. In *International Conference on Machine Learning*, pages 6917–6926. PMLR, 2020.

[30] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation, 2013.

[31] Zachary Nado, Shreyas Padhy, D. Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift, 2021.

[32] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, 2016.

[33] Juan Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In *International Conference on Machine Learning*, pages 7599–7609. PMLR, 2020.

[34] Reilly Raab and Yang Liu. Unintended selection: Persistent qualification rate disparities and interventions. *Advances in Neural Information Processing Systems*, 34, 2021.

[35] Andrew Selbst and Julia Powles. "meaningful information" and the right to explanation. In Sorelle A. Friedler and Christo Wilson, editors, *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pages 48–48. PMLR, 23–24 Feb 2018.

[36] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

[37] Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. Improving the generalization of adversarial training with domain adaptation, 2019.

[38] Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(5), 2007.

[39] Masashi Sugiyama, Taiji Suzuki, Shinichi Nakajima, Hisashi Kashima, Paul von Bünau, and Motoaki Kawanabe. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.

[40] Peter D. Taylor and Leo B. Jonker. Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, 40(1):145–156, 1978.

[41] Karl Tuyls, Pieter Jan'T Hoen, and Bram Vanschoenwinkel. An evolutionary dynamical analysis of multi-agent learning in iterated games. *Autonomous Agents and Multi-Agent Systems*, 12(1):115–153, 2006.

[42] Berk Ustun, Alexander Spangher, and Yang Liu. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 10–19, 2019.

[43] Thomas Varsavsky, Mauricio Orbes-Arteaga, Carole H. Sudre, Mark S. Graham, Parashkev Nachev, and M. Jorge Cardoso. Test-time unsupervised domain adaptation, 2020.

[44] Yevgeniy Vorobeychik and Murat Kantarcioglu. *Adversarial Machine Learning*. Morgan & Claypool Publishers, 2018.

[45] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization, 2021.

[46] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip S. Yu. Generalizing to unseen domains: A survey on domain generalization, 2021.

[47] Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the twenty-first international conference on Machine learning*, page 114, 2004.

[48] Kai Zhang, Vincent Zheng, Qiaojun Wang, James Kwok, Qiang Yang, and Ivan Marsic. Covariate shift in hilbert space: A solution via surrogate kernels. In *International Conference on Machine Learning*, pages 388–395. PMLR, 2013.

[49] Kun Zhang, Mingming Gong, Petar Stojanov, Biwei Huang, QINGSONG LIU, and Clark Glymour. Domain adaptation as a problem of inference on graphical models. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4965–4976. Curran Associates, Inc., 2020.

[50] Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domain adaptation under target and conditional shift. In *International Conference on Machine Learning*, pages 819–827. PMLR, 2013.

[51] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413. PMLR, 2019.

A Appendix

We arrange the appendix as follows:

- Appendix A.1 provides some real life scenarios where transparent models are useful or required.
- Appendix A.2 provides comparisons of our setting and other sub-areas in domain adaptation.
- Appendix A.3 provides proof for Theorem 3.1.
- Appendix A.4 provides proof for Theorem 3.2.
- Appendix A.5 provides proof of Theorem 3.3.
- Appendix A.6 provides proof for Proposition 4.1.
- Appendix A.7 provides proof for Theorem 4.2.
- Appendix A.8 provides proof for Theorem 4.3.
- Appendix A.9 provides omitted assumptions and proof for Section 4.3.
- Appendix A.10 provides proof for Theorem 5.1.
- Appendix A.11 provides proof for Theorem 5.2.
- Appendix A.12 provides proof for Proposition 5.3.
- Appendix B provides missing experimental details.

A.1 Example Usages of Transparent Models

As we mentioned in Section 1, there is an increasing requirement of making the decision rule to be transparent due to its potential consequences impacts to individual decision subject. Here we provide the following reasons for using transparent models:

- Government regulation may require the model to be transparent, especially in public services;
- In some cases, companies may want to disclose their models so users will have explanations and are incentivized to better use the provided services.
- Regardless of whether models are published voluntarily, model parameters can often be inferred via well-known query “attacks”.

In addition, we name some concrete examples of some real-life applications:

- Consider the *Medicaid health insurance program* in the United States, which serves low-income people. There is an obligation to provide transparency/disclose the rules (model to automate the decisions) that decide whether individuals qualify for the program — in fact, most public services have ”terms” that are usually set in stone and explained in the documentation. Agents can observe the rules and will adapt their profiles to be qualified if needed. For instance, an agent can decide to provide additional documentation they need to guarantee approval. For more applications along these lines, please refer to this report[5].
- Credit score companies directly publish their criteria for assessing credit risk scores. In loan application settings, companies actually have the incentive to release criteria to incentivize agents to meet their qualifications and use their services.Furthermore, making decision models transparent will gain the trust of users.
- It is also known that it is possible to steal model parameters, if agents have incentives to do so[6]. For instance, spammers frequently infer detection mechanisms by sending different email variants; they then adjust their spam content accordingly.

A.2 Comparison of our setting and Some Areas in Domain Adaptation

We compare our setting (We address it as IDA, representing “induced domain adaptation”) with the following areas:

[5] https://datasociety.net/library/poverty-lawgorithms/

[6] https://www.wired.com/2016/09/how-to-steal-an-ai/

- Adversarial attack [5, 32, 37]: in adversarial attack, the true label Y stays the same for the attacked feature, while in IDA, we allow the true label to change as well. One can think of adversarial attack as a specific form of IDA where the induced distribution has a specific target, that is to maximize the classifier's error by only perturbing/modifying. Our transferability bound does, however, provide insights for how standard training results transfer to the attack setting.
- Domain generalization [46, 23, 30]: the goal of domain generalization is to learn a more general model that can be generalized to any unseen distribution; On the contrary, our focus is to understand how the performance of a classifier trained on the source distribution degrades when evaluated on the induced distribution (which depends on how the population of decision subjects responds); this degradation depends on the classifier itself.
- Test-time adaptation [43, 45, 31]: the issue of test-time adaptation falls into the classical domain adaptation setting where the adaptation is independent of the model being deployed. Applying this technique to solve our problem requires accessing data (either unsupervised or supervised) drawn from $\mathcal{D}_S(h)$ for each h being evaluated during different training epochs.

A.3 Proof of Theorem 3.1

Proof. We first establish two lemmas that will be helpful for bounding the errors of a pair of classifiers. Both are standard results from the domain adaption literature [3].

Lemma A.1. *For any hypotheses $h, h' \in \mathcal{H}$ and distributions $\mathcal{D}, \mathcal{D}'$,*

$$|Err_{\mathcal{D}}(h, h') - Err_{\mathcal{D}'}(h, h')| \leq \frac{d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}, \mathcal{D}')}{2}.$$

Proof. Define the-cross prediction disagreement between two classifiers h, h' on a distribution $\mathcal{D}$ as $\mathrm{Err}_{\mathcal{D}}(h, h') := \mathbb{P}_{\mathcal{D}}(h(X) \neq h'(X))$. By the definition of the $\mathcal{H}-$divergence,

$$\begin{aligned} d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}, \mathcal{D}') &= 2 \sup_{h,h' \in \mathcal{H}} |\mathbb{P}_{\mathcal{D}}(h(X) \neq h'(X)) - \mathbb{P}_{\mathcal{D}'}(h(X) \neq h'(X))| \\ &= 2 \sup_{h,h' \in \mathcal{H}} |\mathrm{Err}_{\mathcal{D}}(h, h') - \mathrm{Err}_{\mathcal{D}'}(h, h')| \\ &\geq 2 \left|\mathrm{Err}_{\mathcal{D}}(h, h') - \mathrm{Err}_{\mathcal{D}'}(h, h')\right| . \end{aligned}$$

□

Another helpful lemma for us is the well-known fact that the 0-1 error obeys the triangle inequality (see, e.g., [8]):

Lemma A.2. *For any distribution $\mathcal{D}$ over instances and any labeling functions f_1, f_2, and f_3, we have $Err_{\mathcal{D}}(f_1, f_2) \leq Err_{\mathcal{D}}(f_1, f_3) + Err_{\mathcal{D}}(f_2, f_3)$.*

Denote by $\bar{h}^*$ the *ideal joint hypothesis*, which minimizes the combined error:

$$\bar{h}^* := \underset{h' \in \mathcal{H}}{\arg\min}\, \mathrm{Err}_{\mathcal{D}(h)}(h') + \mathrm{Err}_{\mathcal{D}_S}(h')$$

We have:

$$\begin{aligned} \mathrm{Err}_{\mathcal{D}(h)}(h) &\leq \mathrm{Err}_{\mathcal{D}(h)}(\bar{h}^*) + \mathrm{Err}_{\mathcal{D}(h)}(h, \bar{h}^*) && \text{(Lemma A.2)} \\ &\leq \mathrm{Err}_{\mathcal{D}(h)}(\bar{h}^*) + \mathrm{Err}_{\mathcal{D}_S}(h, \bar{h}^*) + \left|\mathrm{Err}_{\mathcal{D}(h)}(h, \bar{h}^*) - \mathrm{Err}_{\mathcal{D}_S}(h, \bar{h}^*)\right| \\ &\leq \mathrm{Err}_{\mathcal{D}(h)}(\bar{h}^*) + \mathrm{Err}_{\mathcal{D}_S}(h) + \mathrm{Err}_{\mathcal{D}_S}(\bar{h}^*) + \frac{1}{2} d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}_S, \mathcal{D}(h)) && \text{(Lemma A.1)} \\ &= \mathrm{Err}_{\mathcal{D}_S}(h) + \lambda_{\mathcal{D}_S \to \mathcal{D}(h)} + \frac{1}{2} d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}_S, \mathcal{D}(h)). && \text{(Definition of } \bar{h}^*\text{)} \end{aligned}$$

□

A.4 Proof of Theorem 3.2

Proof. Invoking Theorem 3.1, and replacing h with h_T^* and S with $\mathcal{D}(h_T^*)$, we have

$$\text{Err}_{\mathcal{D}(h)}(h_T^*) \leq \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*) + \lambda_{\mathcal{D}(h)\to\mathcal{D}(h_T^*)} + \frac{1}{2}d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h_T^*), \mathcal{D}(h)) \tag{8}$$

Now observe that

$$\begin{aligned}
\text{Err}_{\mathcal{D}(h)}(h) &\leq \text{Err}_{\mathcal{D}(h)}(h_T^*) + \text{Err}_{\mathcal{D}(h)}(h, h_T^*) \\
&\leq \text{Err}_{\mathcal{D}(h)}(h_T^*) + \text{Err}_{\mathcal{D}(h_T^*)}(h, h_T^*) + \left|\text{Err}_{\mathcal{D}(h)}(h, h_T^*) - \text{Err}_{\mathcal{D}(h_T^*)}(h, h_T^*)\right| \\
&\leq \text{Err}_{\mathcal{D}(h)}(h_T^*) + \text{Err}_{\mathcal{D}(h_T^*)}(h, h_T^*) + \frac{1}{2}d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h_T^*), \mathcal{D}(h)) \qquad \text{(by Lemma A.1)} \\
&\leq \text{Err}_{\mathcal{D}(h)}(h_T^*) + \text{Err}_{\mathcal{D}(h_T^*)}(h) + \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*) + \frac{1}{2}d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h_T^*), \mathcal{D}(h)) \\
&\qquad \text{(by Lemma A.2)} \\
&\leq \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*) + \lambda_{\mathcal{D}(h)\to\mathcal{D}(h_T^*)} + \frac{1}{2}d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h_T^*), \mathcal{D}(h)) \qquad \text{(by equation 8)} \\
&\quad + \text{Err}_{\mathcal{D}(h_T^*)}(h) + \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*) + \frac{1}{2}d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h_T^*), \mathcal{D}(h))
\end{aligned}$$

Adding $\text{Err}_{\mathcal{D}(h)}(h)$ to both sides and rearranging terms yields

$$\begin{aligned}
2\text{Err}_{\mathcal{D}(h)}(h) - 2\text{Err}_{\mathcal{D}(h_T^*)}(h_T^*) &\leq \text{Err}_{\mathcal{D}(h)}(h) + \text{Err}_{\mathcal{D}(h_T^*)}(h) + \lambda_{\mathcal{D}(h)\to\mathcal{D}(h_T^*)} + d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h_T^*), \mathcal{D}(h)) \\
&\leq \Lambda_{\mathcal{D}(h)\to\mathcal{D}(h_T^*)} + \lambda_{\mathcal{D}(h)\to\mathcal{D}(h_T^*)} + d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h_T^*), \mathcal{D}(h))
\end{aligned}$$

Dividing both sides by 2 completes the proof. □

A.5 Proof of Theorem 3.3

Proof. Using the triangle inequality of d_{TV}, we have

$$d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) \leq d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_{h|S}) + d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) + d_{\text{TV}}(\mathcal{D}_h(h), \mathcal{D}_Y(h)) \tag{9}$$

and by the definition of d_{TV}, the divergence term $d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h))$ becomes

$$\begin{aligned}
d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_{h|S}) &= |\mathbb{P}_{\mathcal{D}_S}(Y = +1) - \mathbb{P}_{\mathcal{D}_S}(h(x) = +1)| \\
&= \left|\frac{\mathbb{E}_{\mathcal{D}_S}[Y] + 1}{2} - \frac{\mathbb{E}_{\mathcal{D}_S}[h(X)] + 1}{2}\right| \\
&= \left|\frac{\mathbb{E}_{\mathcal{D}_S}[Y]}{2} - \frac{\mathbb{E}_{\mathcal{D}_S}[h(X)]}{2}\right| \\
&\leq \frac{1}{2}\cdot\mathbb{E}_{\mathcal{D}_S}\left[|Y - h(X)|\right] \\
&= \text{Err}_{\mathcal{D}_S}(h)
\end{aligned}$$

Similarly, we have

$$d_{\text{TV}}(\mathcal{D}_h(h), \mathcal{D}_Y(h)) \leq \text{Err}_{\mathcal{D}(h)}(h)$$

As a result, we have

$$\begin{aligned}
\text{Err}_{\mathcal{D}_S}(h) + \text{Err}_{\mathcal{D}(h)}(h) &\geq d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_{h|S}) + d_{\text{TV}}(\mathcal{D}_h(h), \mathcal{D}_Y(h)) \\
&\geq d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) \qquad \text{(by equation 9)}
\end{aligned}$$

which implies

$$\max\{\text{Err}_{\mathcal{D}_S}(h), \text{Err}_{\mathcal{D}(h)}(h)\} \geq \frac{d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h))}{2}\ .$$

□

A.6 Proof of Proposition 4.1

Proof.

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}(h)}[\ell(h;X,Y)] \\
=&\int \mathbb{P}_{\mathcal{D}(h)}(X=x,Y=y)\ell(h;x,y)\;dxdy \\
=&\int \mathbb{P}_{\mathcal{D}_S}(Y=y|X=x)\cdot\mathbb{P}_{\mathcal{D}(h)}(X=x)\ell(h;x,y)\;dxdy \\
=&\int \mathbb{P}_{\mathcal{D}_S}(Y=y|X=x)\cdot\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot\frac{\mathbb{P}_{\mathcal{D}(h)}(X=x)}{\mathbb{P}_{\mathcal{D}_S}(X=x)}\cdot\ell(h;x,y)\;dxdy \\
=&\int \mathbb{P}_{\mathcal{D}_S}(Y=y|X=x)\cdot\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot\omega_x(h)\cdot\ell(h;x,y)\;dxdy \\
=&\mathbb{E}_{\mathcal{D}_S}[\omega_x(h)\cdot\ell(h;x,y)]
\end{aligned}$$

□

A.7 Proof of Theorem 4.2

Proof. We start from the error induced by h_S^*. Let the *average importance weight induced by* h_S^* be $\bar{\omega}(h_S^*)=\mathbb{E}_{\mathcal{D}_S}[\omega_x(h_S^*)]$; we add and subtract this from the error:

$$\begin{aligned}
\mathrm{Err}_{\mathcal{D}(h_S^*)}(h_S^*) &= \mathbb{E}_{\mathcal{D}_S}\left[\omega_x(h_S^*)\cdot\mathbb{1}(h_S^*(x)\neq y)\right] \\
&= \mathbb{E}_{\mathcal{D}_S}\left[\bar{\omega}(h_S^*)\cdot\mathbb{1}(h_S^*(x)\neq y)\right]+\mathbb{E}_{\mathcal{D}_S}\left[(\omega_x(h_S^*)-\bar{\omega}(h_S^*))\cdot\mathbb{1}(h_S^*(x)\neq y)\right]
\end{aligned}$$

In fact, $\bar{\omega}(h_S^*)=1$, since

$$\begin{aligned}
\bar{\omega}(h_S^*)=&\mathbb{E}_{\mathcal{D}_S}[\omega_x(h_S^*)]=\int\omega_x(h_S^*)\mathbb{P}_{\mathcal{D}_S}(X=x)dx \\
&=\int\frac{\mathbb{P}_{\mathcal{D}(h)}(X=x)}{\mathbb{P}_{\mathcal{D}_S}(X=x)}\mathbb{P}_{\mathcal{D}_S}(X=x)dx=\int\mathbb{P}_{\mathcal{D}(h)}(X=x)dx=1
\end{aligned}$$

Now consider any other classifier h. We have

$$\begin{aligned}
&\mathrm{Err}_{\mathcal{D}(h_S^*)}(h_S^*) \\
&=\mathbb{E}_{\mathcal{D}_S}\left[\mathbb{1}(h_S^*(x)\neq y)\right]+\mathbb{E}_{\mathcal{D}_S}\left[(\omega_x(h_S^*)-\bar{\omega}(h_S^*))\cdot\mathbb{1}(h_S^*(x)\neq y)\right] \\
&\leq\mathbb{E}_{\mathcal{D}_S}\left[\mathbb{1}(h(x)\neq y)\right]+\mathbb{E}_{\mathcal{D}_S}\left[(\omega_x(h_S^*)-\bar{\omega}(h_S^*))\cdot\mathbb{1}(h_S^*(x)\neq y)\right] \\
&\hspace{20em}\text{(by optimality of } h_S^* \text{ on } \mathcal{D}_S) \\
&=\mathbb{E}_{\mathcal{D}_S}\left[\bar{\omega}(h)\cdot\mathbb{1}(h(x)\neq y)\right]+\mathbb{E}_{\mathcal{D}_S}\left[(\omega_x(h_S^*)-\bar{\omega}(h_S^*))\cdot\mathbb{1}(h_S^*(x)\neq y)\right] \\
&\hspace{20em}\text{(multiply by } \bar{\omega}(h_S^*)=1) \\
&=\mathbb{E}_{\mathcal{D}_S}\left[\omega_x(h)\cdot\mathbb{1}(h(x)\neq y)\right]+\mathbb{E}_{\mathcal{D}_S}\left[(\bar{\omega}(h)-\omega_x(h))\cdot\mathbb{1}(h(x)\neq y)\right] \\
&\hspace{20em}\text{(add and subtract } \bar{\omega}(h_S^*)) \\
&\quad+\mathbb{E}_{\mathcal{D}_S}\left[(\omega_x(h_S^*)-\bar{\omega}(h_S^*))\cdot\mathbb{1}(h_S^*(x)\neq y)\right] \\
&=\mathrm{Err}_{\mathcal{D}(h)}(h)+\mathrm{Cov}(\omega_x(h_S^*),\mathbb{1}(h_S^*(x)\neq y))-\mathrm{Cov}(\omega_x(h),\mathbb{1}(h(x)\neq y))
\end{aligned}$$

Moving the error terms to one side, we have

$$\begin{aligned}
&\mathrm{Err}_{\mathcal{D}(h_S^*)}(h_S^*)-\mathrm{Err}_{\mathcal{D}(h)}(h) \\
&\leq\mathrm{Cov}(\omega_x(h_S^*),\mathbb{1}(h_S^*(x)\neq y))-\mathrm{Cov}(\omega_x(h),\mathbb{1}(h(x)\neq y)) \\
&\leq\sqrt{\mathrm{Var}(\omega_x(h_S^*))\cdot\mathrm{Var}(\mathbb{1}(h_S^*(x)\neq y))}\qquad\qquad(|\mathrm{Cov}(X,Y)|\leq\sqrt{\mathrm{Var}(X)\cdot\mathrm{Var}(Y)}) \\
&\quad+\sqrt{\mathrm{Var}(\omega_x(h))\cdot\mathrm{Var}(\mathbb{1}(h(x)\neq y))} \\
&=\sqrt{\mathrm{Var}(\omega_x(h_S^*))\cdot\mathrm{Err}_S(h_S^*)(1-\mathrm{Err}_S(h_S^*))}+\sqrt{\mathrm{Var}(\omega_x(h))\cdot\mathrm{Err}_{\mathcal{D}_S}(h)(1-\mathrm{Err}_{\mathcal{D}_S}(h))} \\
&\leq\sqrt{\mathrm{Var}(\omega_x(h_S^*))\cdot\mathrm{Err}_S(h_S^*)}+\sqrt{\mathrm{Var}(\omega_x(h))\cdot\mathrm{Err}_{\mathcal{D}_S}(h)}\qquad\qquad(1-\mathrm{Err}_{\mathcal{D}_S}(h)\leq 1) \\
&\leq\sqrt{\mathrm{Err}_{\mathcal{D}_S}(h)}\cdot\left(\sqrt{\mathrm{Var}(\omega_x(h_S^*))}+\sqrt{\mathrm{Var}(\omega_x(h))}\right)
\end{aligned}$$

Since this holds for any h, it certainly holds for $h = h_T^*$.

□

A.8 Omitted Assumptions and Proof of Theorem 4.3

Denote $X_+(h) = \{x : \omega_x(h) \geq 1\}$ and $X_-(h) = \{x : \omega_x(h) < 1\}$. First we observe that

$$\begin{aligned}
&\int_{X_+(h)} \mathbb{P}_{\mathcal{D}_S}(X = x)(1 - \omega_x(h))dx \\
&+ \int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(X = x)(1 - \omega_x(h))dx = 0
\end{aligned}$$

This is simply because of $\int_x \mathbb{P}_{\mathcal{D}_S}(X = x) \cdot \omega_x(h)dx = \int_x \mathbb{P}_{\mathcal{D}(h)}(X = x)dx = 1$.

Now we provide detailed specifications of the assumptions for proving Theorem 4.3:

Assumption A.3 (increased $\omega_x(h)$ value points are more likely to have $Y = +1$)**.**

$$\left|\int_{X_+(h)} \mathbb{P}_{\mathcal{D}_S}(Y = +1, X = x)(1 - \omega_x(h))dx\right| \geq \left|\int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(Y = +1, X = x)(1 - \omega_x(h))dx\right|$$

Assumption A.4 (increased $\omega_x(h)$ value points are more likely to be classified as $+1$)**.**

$$\left|\int_{X_+(h)} \mathbb{P}_{\mathcal{D}_S}(h(x) = +1, X = x)(1 - \omega_x(h))dx\right| \geq \left|\int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(h(x) = +1, X = x)(1 - \omega_x(h))dx\right|$$

Assumption A.5. $\mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x) - \mathbb{P}_{\mathcal{D}_S}(h(x) = +1|X = x)$ and $\omega_x(h)$ is positively correlated:

$$\text{Cov}\big(\mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x) - \mathbb{P}_{\mathcal{D}_S}(h(x) = +1|X = x), \omega_x(h)\big) > 0$$

The above assumption states that for a deterministic classifier h, within all $h(X) = +1$ or $h(X) = -1$, a higher $\mathbb{P}_{\mathcal{D}}(Y = +1|X = x)$ associates with a higher $\omega_x(h)$. With the help of Assumption A.3 - Assumption A.5, we proceed to proof for Theorem 4.3:

Proof. Notice that in the setting of binary classification, we can write the total variation distance between $\mathcal{D}_{Y|S}$ and $\mathcal{D}_Y(h)$ as the difference between the probability of $Y = +1$ and the probability of $Y = -1$:

$$\begin{aligned}
&d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) \\
=& \big|\mathbb{P}_{\mathcal{D}_S}(Y = +1) - \mathbb{P}_{\mathcal{D}(h)}(Y = +1)\big| \\
=& \left|\int \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x)dx - \int \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x)\omega_x(h)dx\right| \\
=& \left|\int \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (1 - \omega_x(h))dx\right| \qquad (10)
\end{aligned}$$

Similarly we have

$$d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) = \left|\int \mathbb{P}_{\mathcal{D}_S}(h(x) = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (1 - \omega_x(h))dx\right| \qquad (11)$$

We can further expand the total variation distance between $\mathcal{D}_{Y|S}$ and $\mathcal{D}_Y(h)$ as follows:

$$\begin{aligned}
&d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) \\
=&\left|\int \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (1 - \omega_x(h))dx\right| \\
=&\Big|\underbrace{\int_{X_+(h)} \mathbb{P}_{\mathcal{D}}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (1 - \omega_x(h))dx}_{\leq 0} \\
&\underbrace{+\int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (1 - \omega_x(h))dx\Big|}_{>0} \\
=&-\int_{X_+(h)} \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (1 - \omega_x(h))dx \\
&-\int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (1 - \omega_x(h))dx \qquad \text{(by Assumption A.3)} \\
=&\int_{X_+(h)} \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (\omega_x(h) - 1)dx \\
&+\int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (\omega_x(h) - 1)dx \qquad \text{(by equation 10)} \\
=&\int \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (\omega_x(h) - 1)dx
\end{aligned}$$

Similarly, by assumption A.4 and equation equation 11, we have

$$d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) = \int \mathbb{P}_{\mathcal{D}_S}(h(x) = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (\omega_x(h) - 1)dx$$

Thus we can bound the difference between $d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h))$ and $d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h))$ as follows:

$$\begin{aligned}
&d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) \\
=&\int \mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (\omega_x(h) - 1)dx \\
&-\int \mathbb{P}_{\mathcal{D}}(h(x) = +1|X = x)\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (\omega_x(h) - 1)dx \\
=&\int [\mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x) - \mathbb{P}_{\mathcal{D}_S}(h(x) = +1|X = x)]\mathbb{P}_{\mathcal{D}_S}(X = x) \cdot (\omega_x(h) - 1)dx \\
=&\,\mathbb{E}_{\mathcal{D}_S}[(\mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x) - \mathbb{P}_{\mathcal{D}_S}(h(x) = +1|X = x))\,(\omega_x(h) - 1)] \\
&\qquad \text{(by Assumption A.5)} \\
>&\,\mathbb{E}_{\mathcal{D}_S}[\mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x) - \mathbb{P}_{\mathcal{D}_S}(h(x) = +1|X = x)]\mathbb{E}_{\mathcal{D}_S}[\omega_x(h) - 1] \\
=&\,0
\end{aligned}$$

Combining the above with Theorem 3.3, we have

$$\max\{\text{Err}_{\mathcal{D}_S}(h), \text{Err}_{\mathcal{D}(h)}(h)\} \geq \frac{d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h))}{2} > 0$$

□

A.9 Omitted details for Section 4.3

To make the problem tractable and meaningful, we make the following assumptions:

Assumption A.6. (Agent's Initial Feature Distribution) Agents' initial features are uniformly distributed between $[0, 1] \in \mathbb{R}^1$:

$$\Pr_{\mathcal{D}_S}(x) = \begin{cases} 1, & \text{if } x \in [0,1] \\ 0, & \text{otherwise} \end{cases}$$

Assumption A.7. (Agent's Cost Function) The cost of changing from x to x' is proportional to the distance between them: $c(x, x') = \|x - x'\|$.

Under Assumption A.7, only agents whose features are in between $[\tau_h - B, \tau_h)$ will *attempt* to change their feature. We also assume that feature updates are *probabilistic*, such that agents with features closer to the decision boundary τ_h have a greater *chance* of updating their feature and each updated feature x' is sampled from a uniform distribution depending on τ_h, B, and x (see Assumption A.8 and Assumption A.9):

Assumption A.8 (Agent's Success Manipulation Probability)**.** For agents who *attempt* to update their features, the probability of a successful feature update is

$$\Pr(X' \neq X) = 1 - \frac{|x - \tau_h|}{B} \tag{12}$$

Intuitively this assumption means that the closer the agent's original feature x is to the decision boundary τ_h, the more likely they can successfully change their feature to cross the decision boundary.

Assumption A.9 (Newly Adapted Feature's Distribution)**.** An agent's updated feature x', given original feature x, manipulation budget B, and classification boundary τ_h, is sampled as

$$X' \sim \text{Unif}(\tau_h, \tau_h + B - x) \tag{13}$$

This assumption aims to capture the fact that even though agent targets to change their feature to the decision boundary τ_h (i.e. the least cost action to get a favorable prediction outcome), they might end up reaching to a feature that is beyond the decision boundary.

With Assumption A.7 - Assumption A.9, we can further specify the important weight $w_x(h)$ for the strategic response setting:

Lemma A.10. *Recall the definition for the covariate shift important weight coefficient* $\omega_x(h) := \frac{\mathbb{P}_{\mathcal{D}(h)}(X=x)}{\mathbb{P}_{\mathcal{D}_S}(X=x)}$*, for our strategic response setting, we have,*

$$w_x(h) = \begin{cases} 1, & x \in [0, \tau_h - B) \\ \frac{\tau_h - x}{B}, & x \in [\tau_h - B, \tau_h) \\ \frac{1}{B}(-x + \tau_h + 2B), & x \in [\tau_h, \tau_h + B) \\ 1, & x \in [\tau_h + B, 1] \end{cases} \tag{14}$$

Proof for Lemma A.10:

Proof. We discuss the induced distribution $\mathcal{D}(h)$ by cases:

- For the features distributed between $[0, \tau_h - B]$: since we assume the agents are rational, under assumption A.7, agents with feature that is smaller than $[0, \tau_h - B]$ will not perform any kinds of adaptations, and no other agents will adapt their features to this range of features either, so the distribution between $[0, \tau_h - B]$ will remain the same as before.
- For the target distribution between $[\tau_h - B, \tau_h]$ can be directly calculated from assumption A.8.
- For distribution between $[\tau_h, \tau_h + B]$, consider a particular feature $x^\star \in [\tau_h, \tau_h + B]$, under Assumption A.9, we know its new distribution becomes:

$$\begin{aligned} \Pr_{\mathcal{D}(h)}(x = x^\star) &= 1 + \int_{x^\star - B}^{\tau_h} \frac{1 - \frac{\tau_h - z}{B}}{B - \tau_h + z} dz \\ &= 1 + \int_{x^\star - B}^{\tau_h} \frac{1}{B} dz \\ &= \frac{1}{B}(-x^\star + \tau_h + 2B) \end{aligned}$$

- For the target distribution between $[\tau_h + B, 1]$: under assumption A.7 and A.9, we know that no agents will change their feature to this feature region. So the distribution between $[\tau_h + B, 1]$ remains the same as the source distribution.

Recall the definition for the covariate shift important weight coefficient $\omega_x(h) := \frac{\mathbb{P}_{D(h)}(X=x)}{\mathbb{P}_{D_S}(X=x)}$, the distribution of $\omega_x(h)$ after agents' strategic responding becomes:

$$\omega_x(h) = \begin{cases} 1, & x \in [0, \tau_h - B) \text{ and } x \in [\tau_h + B, 1] \\ \frac{\tau_h - x}{B}, & x \in [\tau_h - B, \tau_h) \\ \frac{1}{B}(-x + \tau_h + 2B), & x \in [\tau_h, \tau_h + B) \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

□

Proof for Proposition 4.4:

Proof. According to Lemma A.10, we can compute the variance of $w_x(h)$ as $\text{Var}(w_x(h)) = \mathbb{E}(w_x(h)^2) - \mathbb{E}(w_x(h)^2) = \frac{2}{3}B$. Then by plugging it to the general bound for Theorem 4.2 gives us the desirable result. □

A.10 Proof of Theorem 5.1

Proof. Defining $p := \mathbb{P}_{\mathcal{D}_S}(Y = +1)$, we have

$$\begin{aligned} \text{Err}_{\mathcal{D}(h_S^*)}(h_S^*) &= \omega(h_S^*) \cdot \text{Err}_+(h_S^*) + (1 - \omega(h_S^*)) \cdot \text{Err}_-(h_S^*) \\ &\qquad \text{(by definitions of } \omega(h_S^*), \text{Err}_+(h_S^*), \text{ and Err}_-(h_S^*)) \\ &= \underbrace{p \cdot \text{Err}_+(h_S^*) + (1-p) \cdot \text{Err}_-(h_S^*)}_{\text{(I)}} + (\omega(h_S^*) - p)[\text{Err}_+(h_S^*) - \text{Err}_-(h_S^*)] \end{aligned} \tag{16}$$

We can expand (I) as follows:

$$\begin{aligned} & p \cdot \text{Err}_+(h_S^*) + (1-p) \cdot \text{Err}_-(h_S^*) \\ & \leq p \cdot \text{Err}_+(h_T^*) + (1-p) \cdot \text{Err}_-(h_T^*) \qquad \text{(by optimality of } h_S^* \text{ on } \mathcal{D}_S) \\ & = \omega(h_T^*) \cdot \text{Err}_+(h_T^*) + (1 - \omega(h_T^*)) \cdot \text{Err}_-(h_T^*) + (p - \omega(h_T^*)) \cdot [\text{Err}_+(h_T^*) - \text{Err}_-(h_T^*)] \\ & = \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*) + (p - \omega(h_T^*)) \cdot [\text{Err}_+(h_T^*) - \text{Err}_-(h_T^*)] \, . \end{aligned}$$

Plugging this back into equation 16, we have

$$\text{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*) \leq (\omega(h_S^*) - p)[\text{Err}_+(h_S^*) - \text{Err}_-(h_S^*)] + (p - \omega(h_T^*)) \cdot [\text{Err}_+(h_T^*) - \text{Err}_-(h_T^*)]$$

Notice that

$$\begin{aligned} 0.5(\text{Err}_+(h) - \text{Err}_-(h)) &= 0.5 \cdot 1 - 0.5 \cdot \mathbb{P}(h(X) = +1 | Y = +1) - 0.5 \cdot \mathbb{P}(h(X) = +1 | Y = -1) \\ &= 0.5 - \mathbb{P}_{\mathcal{D}_u}(h(X) = +1) \end{aligned}$$

where $\mathcal{D}_u$ is a distribution with uniform prior. Then

$$\begin{aligned} (\omega(h_S^*) - p)[\text{Err}_+(h_S^*) - \text{Err}_-(h_S^*)] &= 2(\omega(h_S^*) - p) \cdot (0.5 - \mathbb{P}_{\mathcal{D}_u}(h(X) = +1)) \\ (p - \omega(h_T^*))[\text{Err}_+(h_T^*) - \text{Err}_-(h_T^*)] &= 2(p - \omega(h_T^*)) \cdot (0.5 - \mathbb{P}_{\mathcal{D}_u}(h(X) = +1)) \end{aligned}$$

Adding together these two equations yields

$$\begin{aligned} & (\omega(h_S^*) - p)[\text{Err}_+(h_S^*) - \text{Err}_-(h_S^*)] + (p - \omega(h_T^*)) \cdot [\text{Err}_+(h_T^*) - \text{Err}_-(h_T^*)] \\ & = 2(\omega(h_S^*) - p) \cdot (0.5 - \mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1)) + 2(p - \omega(h_T^*)) \cdot (0.5 - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)) \\ & = (\omega(h_S^*) - \omega(h_T^*)) - 2\left(\omega(h_S^*)\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \omega(h_T^*)\mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)\right) \\ & \quad + 2p \cdot (\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)) \\ & \leq |\omega(h_S^*) - \omega(h_T^*)| \cdot (1 + 2|\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)|) \\ & \quad + 2p \cdot |\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)| \end{aligned} \tag{17}$$

Meanwhile,

$$\begin{aligned} & |\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)| \\ & \leq 0.5 \cdot |\mathbb{P}_{\mathcal{D}|Y=+1}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}|Y=+1}(h_T^*(X) = +1)| \\ & \quad + 0.5 \cdot |\mathbb{P}_{\mathcal{D}|Y=-1}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}|Y=-1}(h_T^*(X) = +1)| \\ & = 0.5 \left(d_{\text{TV}}(\mathcal{D}_+(h_S^*), \mathcal{D}_+(h_T^*)) + d_{\text{TV}}(\mathcal{D}_-(h_S^*), \mathcal{D}_-(h_T^*))\right) \end{aligned} \tag{18}$$

Combining equation 17 and equation 18 gives

$$\begin{aligned}
&|\omega(h_S^*) - \omega(h_T^*)| \cdot (1 + 2 \cdot |\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)|) \\
&\quad + 2p \cdot |\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)| \\
\leq\ &|\omega(h_S^*) - \omega(h_T^*)| \cdot (1 + d_{\text{TV}}(\mathcal{D}_+(h_S^*), \mathcal{D}_+(h_T^*)) + d_{\text{TV}}(\mathcal{D}_-(h_S^*), \mathcal{D}_-(h_T^*)) \\
&\quad + p \cdot (d_{\text{TV}}(\mathcal{D}_+(h_S^*), \mathcal{D}_+(h_T^*)) + d_{\text{TV}}(\mathcal{D}_-(h_S^*), \mathcal{D}_-(h_T^*)) \\
\leq\ &|\omega(h_S^*) - \omega(h_T^*)| + (1+p) \cdot (d_{\text{TV}}(\mathcal{D}_+(h_S^*), \mathcal{D}_+(h_T^*)) + d_{\text{TV}}(\mathcal{D}_-(h_S^*), \mathcal{D}_-(h_T^*)) \ .
\end{aligned}$$

□

A.11 Proof of Theorem 5.2

We will make use of the following fact:

Lemma A.11. *Under label shift, TPR$_S(h)$ = TPR$_h(h)$ and FPR$_S(h)$ = FPR$_h(h)$.*

Proof. We have

$$\begin{aligned}
\text{TPR}_h(h) =&\mathbb{P}_{\mathcal{D}(h)}(h(X) = +1 | Y = +1) \\
=&\int \mathbb{P}_{\mathcal{D}(h)}(h(X) = +1, X = x | Y = +1) dx \\
=&\int \mathbb{P}_{\mathcal{D}(h)}(h(X) = +1 | X = x, Y = +1) \mathbb{P}_{\mathcal{D}(h)}(X = x | Y = +1) dx \\
=&\int \mathbb{1}(h(x) = +1) \mathbb{P}_{\mathcal{D}(h)}(X = x | Y = +1) dx \\
=&\int \mathbb{1}(h(x) = +1) \mathbb{P}_{\mathcal{D}_S}(X = x | Y = +1) dx \qquad \text{(by definition of label shift)} \\
=&\int \mathbb{P}_{\mathcal{D}_S}(h(X) = +1 | X = x, Y = +1) \mathbb{P}_{\mathcal{D}_S}(X = x | Y = +1) dx \\
=&\text{TPR}_S(h)
\end{aligned}$$

The argument for TPR$_h(h)$ = TPR$_S(h)$ is analogous. □

Now we proceed to prove the theorem.

Proof of Theorem 5.2. In section 3.2 we showed a general lower bound on the maximum of Err$_{\mathcal{D}_S}(h)$ and Err$_{\mathcal{D}(h)}(h)$:

$$\max\{\text{Err}_{\mathcal{D}_S}(h), \text{Err}_{\mathcal{D}(h)}(h)\} \geq \frac{d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h))}{2}$$

In the case of label shift, and by the definitions of p and $\omega(h)$,

$$d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) = |\mathbb{P}_{\mathcal{D}_S}(Y = +1) - \mathbb{P}_{\mathcal{D}(h)}(Y = +1)| = |p - \omega(h)| \tag{19}$$

In addition, we have

$$\mathcal{D}_{h|S} = \mathbb{P}_S(h(X) = +1) = p \cdot \text{TPR}_S(h) + (1-p) \cdot \text{FPR}_S(h) \tag{20}$$

Similarly

$$\begin{aligned}
\mathcal{D}_h(h) &= \mathbb{P}_{\mathcal{D}(h)}(h(X) = +1) \\
&= \omega(h) \cdot \text{TPR}_h(h) + (1 - \omega(h)) \cdot \text{FPR}_h(h) \\
&= \omega(h) \cdot \text{TPR}_S(h) + (1 - \omega(h)) \cdot \text{FPR}_S(h) \qquad \text{(by Lemma A.11)} \qquad (21)
\end{aligned}$$

Therefore

$$\begin{aligned}
d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) =&|\mathbb{P}_{\mathcal{D}_S}(h(X) = +1) - \mathbb{P}_{\mathcal{D}(h)}(h(X) = +1)| \\
=&|(p - \omega(h)) \cdot \text{TPR}_S(h) + (\omega(h) - p) \cdot \text{FPR}_S(h)| \\
&\qquad \text{(By equation 21 and equation 20)} \\
=&|p - \omega(h)| \cdot |\text{TPR}_S(h) - \text{FPR}_S(h)| \qquad (22)
\end{aligned}$$

which yields:

$$d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) = |p - \omega(h)|(1 - |\text{TPR}_S(h) - \text{FPR}_S(h)|)$$

(By equation 19 and equation 22)

completing the proof. □

A.12 Proof of Proposition 5.3

Proof.

$$\begin{aligned}
&|\omega(h_S^*) - \omega(h_T^*)| \cdot \frac{1}{\mathbb{P}_{\mathcal{D}_S}(Y = +1)} \\
=&\frac{|(1 - \text{Err}_{\mathcal{D}_S}(h_S^*))\text{TPR}_S(h_S^*) - (1 - \text{Err}_{\mathcal{D}_S}(h_T^*))\text{TPR}_S(h_T^*)|}{(1 - \text{Err}_{\mathcal{D}_S}(h_S^*)) \cdot (1 - \text{Err}_{\mathcal{D}_S}(h_T^*))} \\
\leq&\frac{|\text{Err}_{\mathcal{D}_S}(h_S^*) - \text{Err}_{\mathcal{D}_S}(h_T^*)| \cdot |\text{TPR}_S(h_S^*) - \text{TPR}_S(h_T^*)|}{(1 - \text{Err}_{\mathcal{D}_S}(h_S^*)) \cdot (1 - \text{Err}_{\mathcal{D}_S}(h_T^*))} \qquad (23)
\end{aligned}$$

The inequality above is due to Lemma 7 of [26]. □

B Missing Experimental Details

B.1 Synthetic Experiments Using DAG

Covariate Shift We specify the causal DAG for covariate shift setting in the following way:

$$\begin{aligned}
X_1 &\sim \text{Unif}(-1, 1) \\
X_2 &\sim 1.2X_1 + \mathcal{N}(0, \sigma_2^2) \\
X_3 &\sim -X_1^2 + \mathcal{N}(0, \sigma_3^2) \\
Y &:= 2\text{sign}(X_2 > 0) - 1
\end{aligned}$$

where σ_2^2 and σ_3^2 are parameters of our choices.
Adaptation function We assume the new distribution of feature X_1' will be generated in the following way:

$$X_1' = \Delta(X) = X_1 + c \cdot (h(X) - 1)$$

where $c \in \mathbb{R}^1 > 0$ is the parameter controlling how much the prediction $h(X)$ affect the generating of X_1', namely the magnitude of distribution shift. Intuitively, this adaptation function means that if a feature x is predicted to be positive ($h(x) = +1$), then decision subjects are more likely to adapt to that feature in the induced distribution; Otherwise, decision subjects are more likely to be moving away from x since they know it will lead to a negative prediction.

Target Shift We specify the causal DAG for target shift setting in the following way:

$$\begin{aligned}
(Y + 1)/2 &\sim \text{Bernoulli}(\alpha) \\
X_1|Y = y &\sim \mathcal{N}_{[0,1]}(\mu_y, \sigma^2) \\
X_2 &= -0.8X_1 + \mathcal{N}(0, \sigma_2^2) \\
X_3 &= 0.2Y + \mathcal{N}(0, \sigma_3^2)
\end{aligned}$$

where $\mathcal{N}_{[0,1]}$ represents a truncated Gaussian distribution taken value between 0 and 1. α, μ_y, σ^2,σ_2^2 and σ_3^2 are parameters of our choices.
Adaptation function We assume the new distribution of the qualification Y' will be updated in the following way:

$$\mathbb{P}(Y' = +1|h(X) = h, Y = y) = c_{hy}, \text{ where } \{h, y\} \in \{-1, +1\}$$

where $0 \leq c_{hy} \in \mathbb{R}^1 \leq 1$ represents the likelihood for a person with original qualification $Y = y$ and get predicted as $h(X) = h$ to be qualified in the next step ($Y' = +1$).

B.2 Synthetic Experiments Using Real-world Data

On the preprocessed FICO credit score data set [4, 18], we convert the cumulative distribution function (CDF) of TransRisk score among demographic groups (denoted as A, including Black, Asian, Hispanic, and White) into group-dependent densities of the credit score. We then generate a balanced sample where each group has equal representation, with credit scores (denoted as Q) initialized by sampling from the corresponding group-dependent density. The value of attributes for each data point is then updated under a specified dynamics (as detailed below) to model the real-world scenario of repeated resource allocation (with decision denoted as D). Since we are considering the dynamic setting, we further specify the data generating process in the following way (from time step $T = t$ to $T = t + 1$):

$$\begin{aligned}
X_{t,1} &\sim 1.5Q_t + U[-\epsilon_1, \epsilon_1] \\
X_{t,2} &\sim 0.8A_t + U[-\epsilon_2, \epsilon_2] \\
X_{t,3} &\sim A_t + \mathcal{N}(0, \sigma^2) \\
Y_t &\sim \text{Bernoulli}(q_t) \text{ for a given value of } Q_t = q_t \\
D_t &= f_t(A_t, X_{t,1}, X_{t,2}, X_{t,3}) \\
Q_{t+1} &= \{Q_t \cdot [1 + \alpha_D(D_t) + \alpha_Y(Y_t)]\}_{(0,1]} \\
A_{t+1} &= A_t \text{ (fixed population)}
\end{aligned}$$

where $\{\cdot\}_{(0,1]}$ represents truncated value between the interval $(0, 1]$, $f_t(\cdot)$ represents the decision policy from input features, and $\epsilon_1, \epsilon_2, \sigma$ are parameters of our choices.

Within the same time step, i.e., for variables that share the subscript t, Q_t and A_t are root causes for all other variables ($X_{t,1}, X_{t,2}, X_{t,3}, D_t, Y_t$). For different time steps, e.g., from $T = t$ to $T = t + 1$, the new distribution at $T = t + 1$ is induced by the deployment of the decision policy D_t. Such impact is modeled by a multiplicative update in Q_{t+1} from Q_t with parameters (or functions) $\alpha_D(\cdot)$ and $\alpha_Y(\cdot)$ that depend on D_t and Y_t, respectively.