# Model Transferability with Responsive Decision Subjects

**Yatong Chen** [1] **Zeyu Tang** [2] **Kun Zhang** [2 3] **Yang Liu** [1 4]

## Abstract

Given an algorithmic predictor that is accurate on some source population consisting of strategic human decision subjects, will it remain accurate if the population *respond* to it? 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 $(X, Y)$. Our formulation is motivated by applications where the deployed machine learning models are subjected to 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 performance of the model trained on the available source distribution (data) would translate to the performance on its *induced* domain. We provide both upper bounds for the performance gap due to the induced domain shift, as well as lower bounds 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, including *covariate shift* and *target shift*.

## 1. Introduction

Decision-makers are increasingly required to be transparent on their decision-making rules to offer the "right to explanation" (Goodman & Flaxman, 2017; Selbst & Powles, 2018; Ustun et al., 2019). 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, and will ultimately face data distributions that reflect human agents' responses 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 new shifted distribution caused by its own deployment (e.g., see Figure 1 for a demonstration). Similar observations can be articulated for application in the insurance sector, e.g., insurance companies may develop policy such that customers' behaviors might adapt to lower premium (Haghtalab et al., 2020), the education sector, e.g., teachers may want to design courses in a way that students are less incentivized to cheat (Kleinberg & Raghavan, 2020), and so on.

| Feature | Weight | Original Value | | Adapted Value |
|---|---|---|---|---|
| Income | 2 | \$ 6,000 | $\longrightarrow$ | \$ 6,000 |
| Education Level | 3 | College | $\longrightarrow$ | College |
| Debt | **-10** | \$40,000 | $\longrightarrow$ | **\$20,000** |
| Savings | **5** | \$20,000 | $\longrightarrow$ | **\$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 debt.

In this paper, we provide a general framework for quantifying the transferability of a decision rule when facing responsive decision subjects. What we would like to achieve is some characterizations of the *performance guarantee* of a classifier — that is, given a model primarily trained on the source distribution $\mathcal{D}_S$, how good or bad will it perform on the distribution it induces $\mathcal{D}(h)$, which depends on the model $h$ itself. A key concept in our setting is the *induced risk*, defined as the error a model incurs on the distribution induced by itself:

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

Most relevant to the above formulation are the works of literature on *strategic classification* (Hardt et al., 2016a), and *performative prediction* (Perdomo et al., 2020). In strategic classification, agents are modeled as rational utility maximizers, and under a specific agent's response model, game

[1]Department of Computer Science and Engineering, University of California, Santa Cruz, California, United States. [2]Department of Philosophy, Carnegie Mellon University, Pennsylvania, United States. [3]Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, United Arab Emirates. [4]ByteDance Research. Correspondence to: Yang Liu <yangliu@ucsc.edu>.

*Proceedings of the $40^{th}$ International Conference on Machine Learning*, Honolulu, Hawaii, USA. PMLR 202, 2023. 

theoretical solutions were proposed to model the interactions between the agents and the decision-maker. In performative prediction, a similar notion of risk called the *performative prediction risk* is introduced to measure a given model's performance on the distribution itself induces. Different from ours, one of their main focus is to find the optimal classifier that achieves minimum induced risk after a sequence of model deployments and observing the corresponding response datasets, which might be computationally expensive.

In particular, 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 accurately characterize the agents' utilities. Furthermore, agents might not be fully rational when they respond. 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 $Y$ (Hardt et al., 2016a; Chen et al., 2020a; Dong et al., 2018b).
- **Lack of access to response data** During training, machine learning practitioners may only have access to data from the source distribution, and even when they can anticipate changes in the population due to human agents' responses, they cannot observe the newly shifted 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, and will result in another round of agents' response at its deployment.

The above observations motivate us to focus on understanding the transferability of a model before diving into finding the optimal solutions that achieve the minimum induced risk – the latter problem often requires more specific knowledge on the mapping between the model and its induced distribution, which might not be available during the training process. Another related research problem is to find models that will perform well on both the source and the induced distribution. This question might be solved using techniques from *domain generalization* (Zhou et al., 2021; Sheth et al., 2022).

We add detailed discussions on how our work relates to and differs from these fields in related works (Section 1.2), as well as on how to use existing techniques to solve these two questions in Appendix A.2. and Appendix F. We leave the full discussions of these topics to future work.

### 1.1. Our Contributions

In this paper, we aim to provide answers to the following 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?
- **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 $h_S^* := \arg\min_h \text{Err}_{\mathcal{D}_S}(h)$, compare to $h_T^* := \arg\min_h \text{Err}_{\mathcal{D}(h)}(h)$?
- **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* (Shimodaira, 2000; Zadrozny, 2004; Sugiyama et al., 2007; 2008; Zhang et al., 2013) and *target shift* (Lipton et al., 2018; Guo et al., 2020; Zhang et al., 2013).

### 1.2. Related Work

Our work most closely relates to the fields of strategic classification, domain adaptation, and performative prediction. In particular, our work considers a setting similar to the studies of strategic classification (Hardt et al., 2016a; Chen et al., 2020a; Dong et al., 2018a; Chen et al., 2020b; Miller et al., 2020), which primarily focus on developing robust classifiers in the presence of strategic agents, rather than characterizing the transferability of a given model's performance on the distribution itself induces. Our work also builds on efforts in domain adaptation (Jiang, 2008; Ben-David et al., 2010; Sugiyama et al., 2008; Zhang et al., 2019; Kang et al., 2019; Zhang et al., 2020). The major difference between our setting and those from previous works is that the changes in distribution are not passively provided by the environment, but rather an active consequence of model deployment. We reference specific prior work in these two domains in Appendix A.2, and here provide more detailed discussions on the existing work in performative prediction.

**Performative Prediction** Performative prediction is a new type of supervised learning problem in which the underlying data distribution shifts in response to the deployed model (Perdomo et al., 2020; Mendler-Dünner et al., 2020; Brown

et al., 2020; Drusvyatskiy & Xiao, 2020; Izzo et al., 2021; Li & Wai, 2022; Maheshwari et al., 2021). In particular, Perdomo et al. (2020) first propose the notion of the *performative risk* defined as $\text{PR}(\theta) := \mathbb{E}_{z\sim\mathcal{D}(\theta)}[\ell(\theta; z)]$, where $\theta$ is the model parameter, and $\mathcal{D}(\theta)$ is the induced distribution as a result of the deployment of $\theta$. Similar to our definition of induced risk, performative risk also measures a given model's performance on the distribution itself induces.

The major difference between our work and performative prediction is that we focus on different aspects of the induced domain adaptation problem. One of the primary focuses of performative prediction is to find the optimal model $\theta_{\mathsf{OPT}}$ which achieves the minimum performative prediction risk, or performative stable model $\theta_{\mathsf{ST}}$, which is optimal under its own induced distribution. In particular, one way to find a performative stable model $\theta_{\mathsf{ST}}$ is to perform repeated retraining (Perdomo et al., 2020). In order to get meaningful theoretical guarantees on any proposed algorithms, works in this field generally require particular assumptions on the mapping between the model parameter and its induced distribution (e.g., the smoothness of the mapping), or requires multiple rounds of deployments and observing the corresponding induced distributions, which can be costly in practice (Jagadeesan et al., 2022; Mendler-Dünner et al., 2020). On the contrary, our work's primary focus is to study the *transferability* of a particular model trained primarily on the source distribution and provide theoretical bounds on its performance on its induced distribution, which is useful for estimating the effect of a given classifier when repeated retraining is unavailable. As a result, our work does not assume the knowledge of the supervision/label information on the transferred domain. Also related are the recently developed lines of work on the multiplayer version of the performative prediction problem (Piliouras & Yu, 2022; Narang et al., 2022), and the economic aspects of performative prediction (Hardt et al., 2022; Mendler-Dünner et al., 2022). The details for reproducing our experimental results can be found at `https://github.com/UCSC-REAL/Model_Transferability`.

## 2. Notation and Formulation

*All proofs of our results can be found in the Appendix.*

Suppose we are given a parametric model $h \in \mathcal{H}$ primarily trained on the training data set $S := \{x_i, y_i\}_{i=1}^N$, which 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 finding a classifier that minimizes the prediction error on the source distribution $\text{Err}_{\mathcal{D}_S}(h) := \mathbb{P}_{\mathcal{D}_S}(h(X) \neq Y)$, ideally the decision maker would like 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 its dependency of $\mathcal{D}_S$. For now, we do not specify the dependency of $\mathcal{D}(h)$ as a function 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:[1]

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

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 $\ell$-risk* as

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

The induced risks will be the primary quantities we are interested in quantifying. The following additional notation will also help present our theoretical results in the following few sections:

- Distributions of $Y$ on a distribution $\mathcal{D}$: $\mathcal{D}_Y := \mathbb{P}_{\mathcal{D}}(Y = y)$, 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 $\mathcal{D}_X(h) := \mathbb{P}_{\mathcal{D}(h)}(X = x)$, $\mathcal{D}_{X|S} := \mathbb{P}_{\mathcal{D}_S}(X = x)$.[2]
- Total variation distance (Ali & Silvey, 1966): $d_{\text{TV}}(\mathcal{D}, \mathcal{D}') := \sup_{\mathcal{O}} |\mathbb{P}_{\mathcal{D}}(\mathcal{O}) - \mathbb{P}_{\mathcal{D}'}(\mathcal{O})|$.

### 2.1. Example Induced Domain Adaptation Settings

We provide two example models to demonstrate the use cases of the distribution shift models described in our paper. We provide more detailed descriptions of both settings

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

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

and instantiate our bounds in Section 4.3 and Section 5.3, respectively.

**Strategic Response** As mentioned before, one example of induced distribution shift is when human agents perform *strategic response* to a decision rule. In particular, it is natural to assume that the mapping between feature vector $X$ and the qualification $Y$ before and after the human agents' best response satisfies *covariate shift*: the feature distribution $\mathbb{P}(X)$ will change, but $\mathbb{P}(Y|X)$, the mapping between $Y$ and $X$, remain unchanged. Notice that this is different from the assumption made in the classic strategic classification setting (Hardt et al., 2016a), where *any* adaptations are considered malicious, which means any changes in the feature vector $X$ *do not* change the underlying true qualification $Y$. In this example, we assume that changes in feature $X$ could potentially lead to changes in the true qualification $Y$ and that the mapping between $Y$ and $X$ remains the same before and after the adaptation. This is a common assumption made in a recent line of work on incentivizing improvement behaviors from human agents (see, e.g., Chen et al., 2020b; Shavit et al., 2020). We use Figure 2 (top) as a demonstration of how distribution might shift for the strategic response setting. In Section 4.3, we will use the strategic classification setup to verify our obtained results.

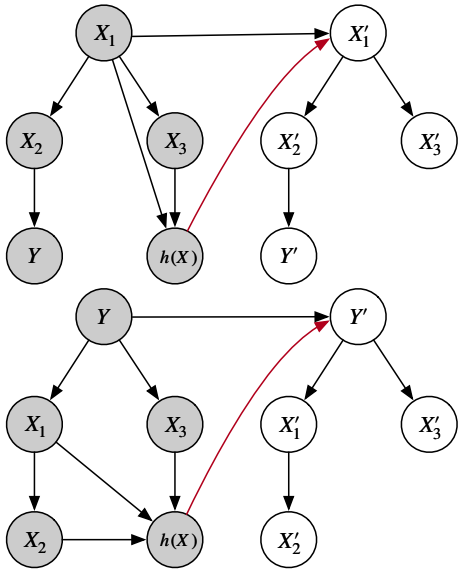



*Figure 2.* Example causal graph annotated to demonstrate covariate shift (the top panel) and target shift (the bottom panel) as a result of the deployment of $h$. Grey nodes indicate observable variables and transparent nodes are not observed at the training stage. Red arrow emphasizes $h$ induces changes in certain variables.

**Replicator Dynamics** Replicator dynamics is a commonly used model to study the evolution of an adopted "strategy" in evolutionary game theory (Tuyls et al., 2006; Friedman & Sinervo, 2016; Taylor & Jonker, 1978; Raab & Liu, 2021). 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:

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

The intuition behind the above equation is that 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 $+1$, namely $\textbf{Fitness}(Y = +1) := \mathbb{P}_{\mathcal{D}_S}(h(X) = +1|Y = +1)$. Intuitively speaking, a higher "fitness" describes more success of agents who adopted a certain strategy ($Y = -1$ or $Y = +1$). Therefore, agents will imitate or replicate their successful peers by adopting the same strategy, resulting in an increase in the population ($\mathbb{P}_{\mathcal{D}(h)}(Y)$).

With the assumption that $\mathbb{P}(X|Y)$ stays unchanged, this instantiates one example of a specific induced *target shift*. We will provide detailed conditions for target shift in Section 5. We also use Figure 2 (bottom) as a demonstration 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.

## 3. General Bounds

In this section, we first provide upper and lower bounds for *any* induced domain without specifying the particular type of distribution shift. In particular, we first provide upper bounds for the transfer error of any classifier $h$ (that is, the difference between $\text{Err}_{\mathcal{D}(h)}(h)$ and $\text{Err}_{\mathcal{D}_S}(h)$), as well as between $\text{Err}_{\mathcal{D}(h)}(h)$ and the minimum induced risk $\text{Err}_{\mathcal{D}(h^*_T)}(h^*_T)$. We then provide lower bounds for $\max\{\text{Err}_{\mathcal{D}_S}(h), \text{Err}_{\mathcal{D}(h)}(h)\}$, 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 the upper bounds for the transfer errors. We begin by showing generic bounds and further instantiate the bound for specific domain adaptation settings in Section 4 and 5. We begin by answering the following question:

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

To that end, we define the minimum and $h$-dependent combined error of any two distributions $\mathcal{D}$ and $\mathcal{D}'$ as:

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

and their corresponding $\mathcal{H}$-divergence 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 a celebrated measure proposed in the domain adaptation literature (Ben-David et al., 2010) which will be useful for bounding the difference in errors of any two classifiers. Following the classical arguments from Ben-David et al. (2010), 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: $Err_{\mathcal{D}(h)}(h) \le Err_{\mathcal{D}_S}(h) + \lambda_{\mathcal{D}_S\to\mathcal{D}(h)} + \frac{1}{2}d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}_S,\mathcal{D}(h))$.*

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 (Ben-David et al., 2010).

An arguably more interesting quantity in our setting to understand is the difference between the induced error of any given model $h$ and the error induced by the optimal model $h^*_T$: $\mathrm{Err}_{\mathcal{D}(h)}(h) - \mathrm{Err}_{\mathcal{D}(h^*_T)}(h^*_T)$. We get the following bound, which 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: $Err_{\mathcal{D}(h)}(h) - Err_{\mathcal{D}(h^*_T)}(h^*_T) \le \frac{\lambda_{\mathcal{D}(h)\to\mathcal{D}(h^*_T)}+\Lambda_{\mathcal{D}(h)\to\mathcal{D}(h^*_T)}(h)}{2} + \frac{1}{2}\cdot d_{\mathcal{H}\times\mathcal{H}}(\mathcal{D}(h^*_T),\mathcal{D}(h))$.*

The above theorem informs us that the induced transfer error is generally bounded by the "average" achievable error on both distributions $\mathcal{D}(h)$ and $\mathcal{D}(h^*_T)$, as well as the $\mathcal{H}$ divergence between the two distributions.

The major benefit of the results in Theorem 3.2 is that it provides the decision maker a way to estimate the minimum induced risk $\mathrm{Err}_{\mathcal{D}(h^*_T)}(h^*_T)$ even when she only has access to the induced risk of some available classifier $h$, as long as she can characterize the statistical difference between the two induced distribution. The latter, however, might not seem to be a trivial task itself. Later in Section 3.3, we briefly discuss how our bounds can still be useful even when we do not have the exact characterizations of this quantity.

### 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 $\mathrm{Err}_{\mathcal{D}_S}(h)$, and $\mathrm{Err}_{\mathcal{D}(h)}(h)$, 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: $\max\{Err_{\mathcal{D}_S}(h), Err_{\mathcal{D}(h)}(h)\} \ge \frac{d_{TV}(\mathcal{D}_{Y|S},\mathcal{D}_Y(h)) - d_{TV}(\mathcal{D}_{h|S},\mathcal{D}_h(h))}{2}$.*

The proof leverages the triangle inequality of $d_{\mathrm{TV}}$. This bound is dependent on $h$; however, by the data processing inequality of $d_{\mathrm{TV}}$ (and $f$-divergence functions in general) (Liese & Vajda, 2006), we have $d_{\mathrm{TV}}(\mathcal{D}_{h|S},\mathcal{D}_h(h)) \le d_{\mathrm{TV}}(\mathcal{D}_{X|S},\mathcal{D}_X(h))$. Applying this to Theorem 3.3 yields:

**Corollary 3.4.** *For any model $h$,*

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

The benefit of Corollary 3.4 is that the bound does not contain any quantities that are functions of the induced distribution; as a result, for any classifier $h$, we can estimate the learning tradeoffs between its source risk and its induced risk using values that are computable without actually deploying the classifier at the first place.

### 3.3. How to Use Our Bounds

The upper and lower bounds we derived in the previous sections (Theorem 3.2 and Theorem 3.3) depend on the following two quantities either explicitly or implicitly: (1) the distribution $\mathcal{D}(h)$ induced by the deployment of the model $h$ in question, and (2) the optimal target classifier $h^*_T$ as well as the distribution $\mathcal{D}(h^*_T)$ it induces. The bounds may therefore seem to be of only theoretical interest since in reality we generally cannot compute $\mathcal{D}(h)$ without actual deployment, let alone compute $h^*_T$. Thus in general it is unclear how to compute the value of these bounds.

Nevertheless, our bounds can still be useful and informative in the following ways:

**General modeling framework with flexible hypothetical shifting models** The bounds can be evaluated if the decision maker has a particular shift model in mind, which specifies how the population would adapt to a model. A common special case is when the decision maker posits an individual-level agent response model (e.g. the strategic agent (Hardt et al., 2016a) - we demonstrate how to evaluate in Section 4.3). In these cases, the $\mathcal{H}$-divergence can be consistently estimated from finite samples of the population (Wang et al., 2005), allowing the decision maker to estimate the performance gap of a given $h$ without deploying it. The general bounds provided can thus be viewed as a framework by which specialized, computationally tractable bounds can be derived.

**Estimate the optimal target classifier $h^*_T$ from a set of imperfect models** Secondly, when the decision maker

has access to a set of imperfect models $\tilde{h}_1, \tilde{h}_2 \cdots \tilde{h}_t \in H^T$ that will predict a range of possible shifted distribution $\mathcal{D}(\tilde{h}_1), \cdots \mathcal{D}(\tilde{h}_t) \in \mathcal{D}^T$ and a range of possibly optimal target distribution $h_T \in \mathcal{H}^T$, the bounds on $h_T^*$ can be further instantiated by calculating the worst case in this predicted set :[3]

$$\begin{aligned}&\text{Err}_{\mathcal{D}(h)}(h) - \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)\\ &\qquad\lesssim \max_{\mathcal{D}'\in\mathcal{D}^T, h'\in\mathcal{H}^T} \text{UpperBound}(\mathcal{D}', h'),\\ &\max\{\text{Err}_{\mathcal{D}_S}(h), \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)\}\\ &\qquad\gtrsim \min_{\mathcal{D}'\in\mathcal{D}^T, h'\in\mathcal{H}^T} \text{LowerBound}(\mathcal{D}', h').\end{aligned}$$

We provide discussions on the tightness of our bounds in Appendix H.

## 4. Covariate Shift

In this section, we focus on a particular distribution shift model known as *covariate shift*, in which the distribution of features changes, but the distribution of labels conditioned on features remains the same:

$$\mathbb{P}_{\mathcal{D}(h)}(Y = y|X = x) = \mathbb{P}_{\mathcal{D}_S}(Y = y|X = x) \quad (3)$$
$$\mathbb{P}_{\mathcal{D}(h)}(X = x) \neq \mathbb{P}_{\mathcal{D}_S}(X = x) \quad (4)$$

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 $\ell$ we have:

**Proposition 4.1** (Expected Loss on $\mathcal{D}(h)$ Under Covariate Shift)**.** $\mathbb{E}_{\mathcal{D}(h)}[\ell(h; X, Y)] = \mathbb{E}_{\mathcal{D}_S}[\omega_x(h) \cdot \ell(h; x, y)]$.

The above derivation is a classic trick and offers the basis for performing importance reweighting when learning under covariate shift (Sugiyama et al., 2008). The particular form informs us that $\omega_x(h)$ controls the generation of $\mathcal{D}(h)$ and encodes its dependency on 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 particularly focus 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:*

$$\begin{aligned}&Err_{\mathcal{D}(h_S^*)}(h_S^*) - Err_{\mathcal{D}(h_T^*)}(h_T^*)\\ \leq&\sqrt{Err_{\mathcal{D}_S}(h_T^*)} \cdot \left(\sqrt{Var(\omega_X(h_S^*))} + \sqrt{Var(\omega_X(h_T^*))}\right).\end{aligned}$$

This result 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 induced risks 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 that in 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. Under several assumptions, our previously provided lower bound in Theorem 3.3 is strictly positive with covariate shift.

**Assumption 4.3.** $|\mathbb{E}_{X\in X_+(h), Y=+1}[1 - \omega_X(h)]| \geq |\mathbb{E}_{X\in X_-(h), Y=+1}[1 - \omega_X(h)]|$ .

where $X_+(h) = \{x : \omega_x(h) \geq 1\}$ and $X_-(h) = \{x : \omega_x(h) < 1\}$.

This assumption states that increased $\omega_x(h)$ value points are more likely to have positive labels.

**Assumption 4.4.** $|\mathbb{E}_{X\in X_+(h), h(X)=+1}[1 - \omega_X(h)]| \geq |\mathbb{E}_{X\in X_-(h), h(X)=+1}[1 - \omega_X(h)]|$.

This assumption states that increased $\omega_x(h)$ value points are more likely to be classified as positive.

**Assumption 4.5.** $\text{Cov}\big(\mathbb{P}_{\mathcal{D}_S}(Y = +1|X = x) - \mathbb{P}_{\mathcal{D}_S}(h(x) = +1|X = x), \omega_x(h)\big) > 0$.

This assumption is stating that 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 $\omega_x(h)$.

**Theorem 4.6.** *Under Assumption 4.3 - Assumption 4.5, the following lower bound is strictly positive under covariate shift:*

$$\begin{aligned}&\max\{Err_{\mathcal{D}_S}(h), Err_{\mathcal{D}(h)}(h)\}\\ \geq&\frac{d_{TV}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{TV}(\mathcal{D}_{h|S}, \mathcal{D}_h(h))}{2} > 0.\end{aligned}$$

### 4.3. Covariate Shift via Strategic Response

As introduced in Section 2.1, we consider a setting caused by *strategic response* in which agents are classified by and

[3] UpperBound and LowerBound are the RHS expressions in Theorem 3.2 and Theorem 3.3.

adapt to a binary threshold classifier. In particular, 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 (Hardt et al., 2016a), 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 $x'$. We 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 the agent will not change their feature if they are already accepted (i.e. $h(x) = +1$). For a given threshold $\tau_h$ and manipulation budget $B$, the theoretical best response of an agent with original feature $x$ is:

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

To make the problem tractable and meaningful, we further specify the following setups:

*Setup* 1. (Initial Feature) Agents' initial features are uniformly distributed between $[0, 1] \in \mathbb{R}^1$.

*Setup* 2. (Agent's Cost Function) The cost of changing from $x$ to $x'$ is proportional to the distance between them: $c(x, x') = \|x - x'\|$.

Setup 2 implies that 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 $\tau_h$ have a greater *chance* of updating their feature and each updated feature $x'$ is sampled from a uniform distribution depending on $\tau_h$, $B$, and $x$ (see Setup 3 & 4):

*Setup* 3. (Agent's Success Manipulation Probability) For agents who *attempt* to update their features, the probability of a successful feature update is $\mathbb{P}(X' \neq X) = 1 - \frac{|x - \tau_h|}{B}$.

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

*Setup* 4 (Adapted Feature's Distribution). An agent's updated feature $x'$, given original $x$, manipulation budget $B$, and classification boundary $\tau_h$, is sampled as $X' \sim \text{Unif}(\tau_h, \tau_h + |B - x|)$.

Setup 4 aims to capture the fact that even though agent targets to change their feature to the decision boundary $\tau_h$ (i.e. the least cost action to get a favorable prediction outcome), they might end up reaching a feature that is beyond the decision boundary.

With the above setups, we can specify the bound in Theorem 4.2 for the strategic response setting as follows:

**Proposition 4.7.** *For our assumed setting of strategic response described above, Theorem 4.2 implies*

$$Err_{\mathcal{D}(h_S^*)}(h_S^*) - Err_{\mathcal{D}(h_T^*)}(h_T^*) \leq \sqrt{\tfrac{2B}{3} Err_{\mathcal{D}_S}(h_T^*)}.$$

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 $\text{Err}_{D_S}(h_T^*)$. This aligns with our intuition that the smaller the manipulation budget is, the fewer 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 expression also allows us to provide bounds 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 the distribution of features conditioned on the label remains the same:

$$\mathbb{P}_{\mathcal{D}(h)}(X = x|Y = y) = \mathbb{P}_{\mathcal{D}_S}(X = x|Y = y) \quad (5)$$

$$\mathbb{P}_{\mathcal{D}(h)}(Y = y) \neq \mathbb{P}_{\mathcal{D}_S}(Y = y) \quad (6)$$

For binary classification, let $p(h) := \mathbb{P}_{\mathcal{D}(h)}(Y = +1)$, and $\mathbb{P}_{\mathcal{D}(h)}(Y = -1) = 1 - p(h)$. Notice that $p(h)$ encodes the full adaptation information from $\mathcal{D}_S$ to $\mathcal{D}(h)$, since the mapping between $Y$ and $X$, $\mathbb{P}(X = x|Y = y)$, is known and remains unchanged during target shift. We have for any proper loss function $\ell$:

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

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 first provide characterizations of the upper bound on 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:*

$$Err_{\mathcal{D}(h_S^*)}(h_S^*) - Err_{\mathcal{D}(h_T^*)}(h_T^*) \leq |\omega(h_S^*) - \omega(h_T^*)| + (1+p) \cdot (d_{TV}(\mathcal{D}_+(h_S^*), \mathcal{D}_+(h_T^*)) + d_{TV}(\mathcal{D}_-(h_S^*), \mathcal{D}_-(h_T^*)) .$$

The above 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 $\mathcal{D}(h)$:*

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

The proof extends the bound of Theorem 3.3 by further explicating each of $d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h))$, $d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h))$ under the assumption of target shift. Since $|\text{TPR}_S(h) - \text{FPR}_S(h)| < 1$ 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: $p - p(h)$. 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. Target Shift via Replicator Dynamics

We now further instantiate our theoretical bound for target shift (Theorem 5.1) using a particular replicator dynamics model previously used in (Raab & Liu, 2021). In particular, the fitness function is specified as the prediction accuracy of $h$ for class $y$:

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

Then we have $\mathbb{E}[\textbf{Fitness}(Y)] = 1 - \text{Err}_{\mathcal{D}_S}(h)$, and $\frac{p(h)}{\mathbb{P}_{\mathcal{D}_S}(Y=+1)} = \frac{\Pr_{\mathcal{D}_S}(h(X)=+1|Y=+1)}{1-\text{Err}_{\mathcal{D}_S}(h)}$. Plugging the result back into Theorem 5.1 we get the following bound for the above replicator dynamic setting:

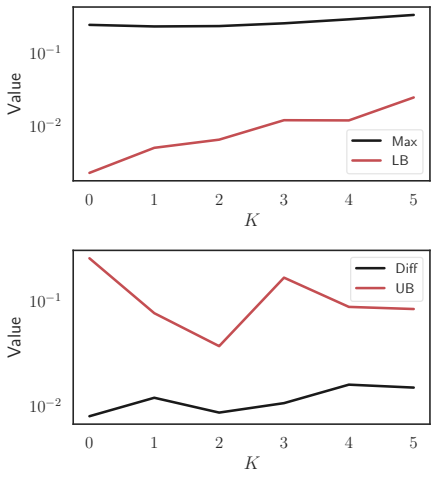



*Figure 3.* Results for synthetic experiments on real-world data. Diff := $\text{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$, Max := $\max\{\text{Err}_{\mathcal{D}_S}(h_T^*), \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)\}$, UB := upper bound specified in Theorem 4.2, and LB := lower bound specified in Theorem 4.6. For each time step $K = k$, we compute and deploy the source optimal classifier $h_S^*$ 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 are deferred to Appendix D.

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

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

The above result shows that the difference between the induced risks $\text{Err}_{\mathcal{D}(h_S^*)}(h_S^*)$ and $\text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$ only depends on the difference between the two classifiers' performances on the source data $\mathcal{D}_S$. This offers the decision maker a great opportunity to evaluate the performance gap by using their corresponding evaluations on the source data only without observing their corresponding induced distributions.

## 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 the 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 D.

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 $\tau$ and take the $h_\tau$ with the lowest prediction error. Then, we posit a specific adaptation function $\Delta(x, h_\tau)$. Finally, to compute $h_T^*$, we vary $\tau$ from 0 to 1 and find the classifier $h_\tau$ 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 4.

For all four datasets, we do observe positive gaps $\text{Err}_{D(h_S^*)}(h_S^*) - \text{Err}_{D(h_T^*)}(h_T^*)$, indicating the suboptimality of training on $\mathcal{D}_S$. The gaps are well bounded by the theoretical results. For the lower bound, the empirical observation and the theoretical bounds are roughly within the same magnitude except for one target shift dataset, indicating the effectiveness of our theoretical result. Regarding the upper bound, for target shift, the empirical observations are well within the same magnitude of the theoretical bounds while the results for the covariate shift are relatively loose.

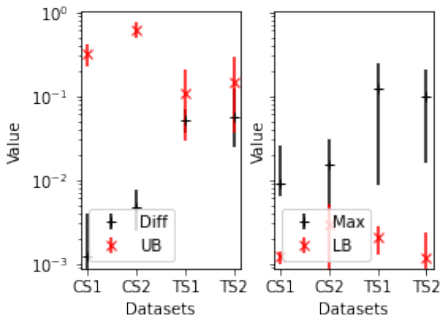



*Figure 4.* Results for synthetic experiments on simulated and real-world data. Diff $:= \text{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$, Max $:= \max\{\text{Err}_{\mathcal{D}_S}(h_T^*), \text{Err}_{\mathcal{D}(h_T^*)}(h_T^*)\}$, UB := upper bound specified in Theorem 4.2, and LB := lower bound specified in Theorem 4.6.

**Synthetic experiments using real-world data** We also perform synthetic experiments using real-world data to demonstrate our bounds. In particular, we use the FICO credit score data set (Board of Governors of the Federal Reserve System (US), 2007) which contains more than 300k records of TransUnion credit scores of clients from different demographic groups. For our experiment on the preprocessed FICO data set (Hardt et al., 2016b), 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 are again deferred to Appendix D. We report our results in Figure 3. 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 (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. We also provide additional experimental results using synthetic datasets generated according to causal graphs defined in Figure 2. Due to page limits, we defer the detailed discussions of these results to Appendix D.2.2.

## 7. Conclusions and Future Directions

Unawareness of the potential distribution shift might lead to unintended consequences when training a machine learning model. One goal of our paper is to raise awareness of this issue for the safe deployment of machine learning methods in high-stake scenarios. We also provide a general framework for characterizing the performance difference for a fixed-trained classifier when the decision subjects respond to it.

Our contributions are mostly theoretical. A natural extension of our work is to collect real human experiment data to verify the usefulness and tightness of our bounds. Another potential future direction is to develop algorithms to find an optimal model that achieves minimum induced risk, which has been an exciting ongoing research problem in the field of performative prediction. Furthermore, using techniques from general domain adaptation to find robust classifiers that perform well in both the source and induced distribution is another promising direction.

**Ackowledgement** Y. Chen is partially supported by the National Science Foundation (NSF) under grants IIS-2143895 and IIS-2040800. The work is also supported in part by the NSF-Convergence Accelerator Track-D award #2134901, by the National Institutes of Health (NIH) under Contract R01HL159805, by grants from Apple Inc., KDDI Research, Quris AI, and IBT, and by generous gifts from Amazon, Microsoft Research, and Salesforce.

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

## References


Ali, S. M. and Silvey, S. D. 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.

Azizzadenesheli, K., Liu, A., Yang, F., and Anandkumar, A. Regularized learning for domain adaptation under label shifts. *arXiv preprint arXiv:1903.09734*, 2019.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. A theory of learning from different domains. *Machine Learning*, 2010.

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.

Brown, G., Hod, S., and Kalemaj, I. Performative prediction in a stateful world, 2020.

Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., and Mukhopadhyay, D. Adversarial attacks and defences: A survey, 2018.

Chen, Y., Liu, Y., and Podimata, C. Learning strategy-aware linear classifiers, 2020a.

Chen, Y., Wang, J., and Liu, Y. Strategic recourse in linear classification. *arXiv preprint arXiv:2011.00355*, 2020b.

Crammer, K., Kearns, M., and Wortman, J. Learning from multiple sources. *Journal of Machine Learning Research*, 9(8), 2008.

Dong, J., Roth, A., Schutzman, Z., Waggoner, B., and Wu, Z. S. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, EC '18, New York, NY, USA, 2018a. Association for Computing Machinery.

Dong, J., Roth, A., Schutzman, Z., Waggoner, B., and Wu, Z. S. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, 2018b.

Drusvyatskiy, D. and Xiao, L. Stochastic optimization with decision-dependent distributions. *arXiv preprint arXiv:2011.11173*, 2020.

Flaxman, A. D., Kalai, A. T., and McMahan, H. B. Online convex optimization in the bandit setting: Gradient descent without a gradient. In *The Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 385–394, 2005.

Friedman, D. and Sinervo, B. *Evolutionary games in natural, social, and virtual worlds*. Oxford University Press, 2016.

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

Goodman, B. and Flaxman, S. European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, 38(3):50–57, Oct 2017.

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., and Schölkopf, B. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4): 5, 2009.

Guo, J., Gong, M., Liu, T., Zhang, K., and Tao, D. Ltf: A label transformation framework for correcting label shift. In *International Conference on Machine Learning*, pp. 3843–3853. PMLR, 2020.

Haghtalab, N., Immorlica, N., Lucier, B., and Wang, J. Z. Maximizing welfare with incentive-aware evaluation mechanisms. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2020.

Hardt, M., Megiddo, N., Papadimitriou, C., and Wootters, M. Strategic classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, New York, NY, USA, 2016a. Association for Computing Machinery.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, pp. 3315–3323, 2016b.

Hardt, M., Jagadeesan, M., and Mendler-Dünner, C. Performative power. *arXiv preprint arXiv:2203.17232*, 2022.

Huang, L., Joseph, A. D., Nelson, B., Rubinstein, B. I., and Tygar, J. D. Adversarial machine learning. In *ACM Workshop on Security and Artificial Intelligence*, 2011.

Izzo, Z., Ying, L., and Zou, J. How to learn when data reacts to your model: performative gradient descent. In *International Conference on Machine Learning*, pp. 4641–4650. PMLR, 2021.

Jagadeesan, M., Zrnic, T., and Mendler-Dünner, C. Regret minimization with performative feedback. In *International Conference on Machine Learning*, pp. 9760–9785. PMLR, 2022.

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

Kang, G., Jiang, L., Yang, Y., and Hauptmann, A. G. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4893–4902, 2019.

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

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. Learning to generalize: Meta-learning for domain generalization, 2017.

Li, Q. and Wai, H.-T. State dependent performative prediction with stochastic approximation. In *International Conference on Artificial Intelligence and Statistics*, pp. 3164–3186. PMLR, 2022.

Liese, F. and Vajda, I. On divergences and informations in statistics and information theory. *IEEE Transactions on Information Theory*, 52(10):4394–4412, 2006.

Lipton, Z., Wang, Y.-X., and Smola, A. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pp. 3122–3130. PMLR, 2018.

Liu, Y. and Liu, M. An online learning approach to improving the quality of crowd-sourcing. *ACM SIGMETRICS Performance Evaluation Review*, 2015.

Long, M., Zhu, H., Wang, J., and Jordan, M. I. Unsupervised domain adaptation with residual transfer networks. *arXiv preprint arXiv:1602.04433*, 2016.

Lowd, D. and Meek, C. Adversarial learning. In *ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, 2005.

Maheshwari, C., Chiu, C.-Y., Mazumdar, E., Sastry, S. S., and Ratliff, L. J. Zeroth-order methods for convex-concave minmax problems: Applications to decision-dependent risk minimization, 2021.

Mendler-Dünner, C., Perdomo, J., Zrnic, T., and Hardt, M. Stochastic optimization for performative prediction. In *Advances in Neural Information Processing Systems*, pp. 4929–4939. Curran Associates, Inc., 2020.

Mendler-Dünner, C., Ding, F., and Wang, Y. Anticipating performativity by predicting from predictions. In *Advances in Neural Information Processing Systems*, 2022.

Miller, J., Milli, S., and Hardt, M. Strategic classification is causal modeling in disguise. In *International Conference on Machine Learning*, pp. 6917–6926. PMLR, 2020.

Muandet, K., Balduzzi, D., and Schölkopf, B. Domain generalization via invariant feature representation, 2013.

Nado, Z., Padhy, S., Sculley, D., D'Amour, A., Lakshminarayanan, B., and Snoek, J. Evaluating prediction-time batch normalization for robustness under covariate shift, 2021.

Narang, A., Faulkner, E., Drusvyatskiy, D., Fazel, M., and Ratliff, L. J. Multiplayer performative prediction: Learning in decision-dependent games. *arXiv preprint arXiv:2201.03398*, 2022.

Papernot, N., McDaniel, P., and Goodfellow, I. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, 2016.

Perdomo, J., Zrnic, T., Mendler-Dünner, C., and Hardt, M. Performative prediction. In *International Conference on Machine Learning*, pp. 7599–7609. PMLR, 2020.

Piliouras, G. and Yu, F.-Y. Multi-agent performative prediction: From global stability and optimality to chaos. *arXiv preprint arXiv:2201.10483*, 2022.

Raab, R. and Liu, Y. Unintended selection: Persistent qualification rate disparities and interventions. *Advances in Neural Information Processing Systems*, 2021.

Selbst, A. and Powles, J. "meaningful information" and the right to explanation. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, Proceedings of Machine Learning Research. PMLR, 2018.

Shavit, Y., Edelman, B., and Axelrod, B. Causal strategic linear regression. *International Conference on Machine Learning*, pp. 8676–8686, 2020.

Sheth, P., Moraffah, R., Candan, K. S., Raglin, A., and Liu, H. Domain generalization–a causal perspective. *arXiv preprint arXiv:2209.15177*, 2022.

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

Song, C., He, K., Wang, L., and Hopcroft, J. E. Improving the generalization of adversarial training with domain adaptation, 2019.

Sugiyama, M., Krauledat, M., and Müller, K.-R. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(5), 2007.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., and Kawanabe, M. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.

Taylor, P. D. and Jonker, L. B. Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, 1978.

Tuyls, K., Hoen, P. J., and Vanschoenwinkel, B. An evolutionary dynamical analysis of multi-agent learning in iterated games. *Autonomous Agents and Multi-Agent Systems*, 2006.

Ustun, B., Spangher, A., and Liu, Y. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 10–19, 2019.

Varsavsky, T., Orbes-Arteaga, M., Sudre, C. H., Graham, M. S., Nachev, P., and Cardoso, M. J. Test-time unsupervised domain adaptation, 2020.

Vorobeychik, Y. and Kantarcioglu, M. *Adversarial Machine Learning*. Morgan & Claypool Publishers, 2018.

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., and Darrell, T. Tent: Fully test-time adaptation by entropy minimization, 2021a.

Wang, J., Lan, C., Liu, C., Ouyang, Y., Qin, T., Lu, W., Chen, Y., Zeng, W., and Yu, P. S. Generalizing to unseen domains: A survey on domain generalization, 2021b.

Wang, Q., Kulkarni, S., and Verdu, S. Divergence estimation of continuous distributions based on data-dependent partitions. *IEEE Transactions on Information Theory*, 51 (9):3064–3074, 2005. doi: 10.1109/TIT.2005.853314.

Xie, R., Wei, H., Feng, L., and An, B. Gearnet: Stepwise dual learning for weakly supervised domain adaptation. *AAAI Conference on Artificial Intelligence*, 2022.

Zadrozny, B. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 114, 2004.

Zhang, K., Schölkopf, B., Muandet, K., and Wang, Z. Domain adaptation under target and conditional shift. In *International Conference on Machine Learning*, pp. 819–827. PMLR, 2013.

Zhang, K., Gong, M., Stojanov, P., Huang, B., LIU, Q., and Glymour, C. Domain adaptation as a problem of inference on graphical models. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 4965–4976. Curran Associates, Inc., 2020.

Zhang, Y., Liu, T., Long, M., and Jordan, M. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pp. 7404–7413. PMLR, 2019.

Zhou, K., Liu, Z., Qiao, Y., Xiang, T., and Loy, C. C. Domain generalization in vision: A survey. *arXiv preprint arXiv:2103.02503*, 2021.

# Supplement to "Model Transferability with Responsive Decision Subjects"

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 additional related work on strategic classification and domain adaptation, as well as a detailed comparison of our setting and other sub-areas in domain adaptation.
- Appendix B.1 provides proof for Theorem 3.1.
- Appendix B.2 provides proof for Theorem 3.2.
- Appendix B.3 provides proof of Theorem 3.3.
- Appendix B.4 provides proof for Proposition 4.1.
- Appendix B.5 provides proof for Theorem 4.2.
- Appendix B.6 provides proof for Theorem 4.6.
- Appendix B.7 provides omitted assumptions and proof for Section 4.3.
- Appendix B.8 provides proof for Theorem 5.1.
- Appendix B.9 provides proof for Theorem 5.2.
- Appendix B.10 provides proof for Proposition 5.3.
- Appendix C provides additional lower bound and examples for the target shift setting.
- Appendix D provides missing experimental details.
- Appendix E discusses challenges in minimizing induced risk.
- Appendix F provides discussions on how to directly minimize the induced risk.
- Appendix G provides discussions on adding regularization to the objective function.
- Appendix H provides discussions on the tightness of our theoretical bounds.

## A. Additional Discussions

### 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. Additional Related Work

**Strategic Classification** Strategic Classification focuses on the problem of how to make predictions in the presence of agents who behave strategically in order to obtain desirable outcomes (Hardt et al., 2016a; Chen et al., 2020a; Dong et al., 2018a; Chen et al., 2020b; Miller et al., 2020). In particular, (Hardt et al., 2016a) first formalizes strategic classification tasks as a two-player sequential game (i.e., a Stackelberg game) between a model designer and strategic agents. Agent best response behavior is typically viewed as malicious in the traditional setting; as a result, the model designer seeks to disincentivize this behavior or limit its impact by publishing classifiers that are robust to any agent's adaptations. In our work, the agents' strategic behaviors are not necessarily malicious; instead, we aim to provide a general framework that works for any distribution shift resulting from the human agency.

Most existing work in strategic classification assumes that human agents are fully rational and will always perform *best response* to any given classifier. As a result, their behaviors can be fully characterized based on pre-specified human response models (Hardt et al., 2016a; Chen et al., 2020a). While we are also interested in settings where agents respond to a decision rule, we focus on the distribution shift of human agents at a population level and characterize the induced distribution as a function of the deployed model. Instead of specifying a particular individual-level agent's response model, we only require the knowledge of the source data $\mathcal{D}_S$, as well as some characterizations of the relationship between the source and the induced distribution, e.g., they satisfy some particular distribution shift models, like covariate shift (see Section 4), or target shift (see Section 5), or we have access to some data points from the induced distribution so we can estimate their statistical differences like H-divergence (see Section 3). In addition, the focus of our work is different from strategic classification. Instead of designing models robust to strategic behavior, we primarily study the *transferability* of a given model's performance on the distribution itself induces.

**Domain Adaptation** There has been tremendous work in domain adaptation studying different distribution shifts and learning from shifting distributions (Jiang, 2008; Ben-David et al., 2010; Sugiyama et al., 2008; Zhang et al., 2019; Kang et al., 2019; Zhang et al., 2020; Xie et al., 2022). Our results differ from these previous works since in our setting, changes in distribution are not passively provided by the environment, but rather an active consequence of model deployment. Part of our technical contributions is inspired by the transferability results in domain adaptations (Ben-David et al., 2010; Zadrozny, 2004; Gretton et al., 2009; Sugiyama et al., 2008; Lipton et al., 2018; Azizzadenesheli et al., 2019).

Our work, at first sight, looks similar to several sub-areas within the literature of domain adaptation, e.g., domain generalization, adversarial attack, and test-time adaptation, to name a few. For instance, the notion of observing an "induced distribution" resembles similarity of the adversarial machine learning literature (Lowd & Meek, 2005; Huang et al., 2011; Vorobeychik & Kantarcioglu, 2018). One of the major differences between ours and adversarial machine learning is that in adversarial machine learning, the true label $Y$ stays the same for the attacked feature, while in our paper, both $X$ and $Y$ might change in the induced distribution $\mathcal{D}(h)$. In Appendix A.2, we provide detailed comparisons between our setting and the other subfields in domain adaptation mentioned above.

**Comparisons 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 (Chakraborty et al., 2018; Papernot et al., 2016; Song et al., 2019): 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 into how standard training results transfer to the attack setting.
- Domain generalization (Wang et al., 2021b; Li et al., 2017; Muandet et al., 2013): the goal of domain generalization is to learn a model that can be generalized to any unseen distribution; Similar to our setting, one of the biggest challenges in domain generalization also the lack of target distribution during training. The major difference, however, is that 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 (Varsavsky et al., 2020; Wang et al., 2021a; Nado et al., 2021): 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.

**Remark** We believe that techniques from Domain Adaptation can potentially be applied to our setting when the decision-maker is interested in producing a classifier that performs well on the induced distribution. In general, we suspect that it will require the decision maker to know certain information about how the induced distribution and the source distribution differ, e.g. some potential characterizations of their statistical differences. Here we provide two possible directions:

- Data Augmentation: The basic idea of data augmentation is to augment the original data point $(x, y)$ pairs with new pairs $(A(x), B(x))$where $A(\cdot)$ and $B(\cdot)$ denote a pair of transformations. Then we add the new pairs to the training dataset. $A(\cdot)$ and $B(\cdot)$ are usually seen as a way of simulating domain shift, and the design of $A(\cdot)$ and $B(\cdot)$ is key to performance. In our case, A and B are functions of the classifier $h$, and they capture how the classifier influences the potential response from the decision subjects. This requires the decision maker to have a specific response model in mind when designing the augmented data points.
- Learning Disentangled Representations: Instead of forcing the entire model or features to be domain invariant, which is challenging, one can relax this constraint by allowing some parts to be domain-specific, essentially learning disentangled representations. In our induced domain adaptation setting, this means separating features into two sets, one set contains features that are invariant to the deployment of a classifier, and the other set contains features that will be potentially affected by. Then we can decompose a classifier into domain-specific biases and domain-agnostic weights, and only keep the latter when dealing with the unseen induced domain.

## B. Proof of Results

### B.1. 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 (Ben-David et al., 2010).

**Lemma B.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 $\text{Err}_{\mathcal{D}}(h, h') := \mathbb{P}_{\mathcal{D}}(h(X) \neq h'(X))$. By the definition of the $\mathcal{H}-$divergence,

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

□

Another helpful lemma for us is the well-known fact that the 0-1 error obeys the triangle inequality (see, e.g., (Crammer et al., 2008)):

**Lemma B.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:

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

We have:

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

□

### B.2. 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

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

Now observe that

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

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

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

Dividing both sides by 2 completes the proof. □

### B.3. Proof of Theorem 3.3

*Proof.* Using the triangle inequality of $d_{\mathrm{TV}}$, we have

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

and by the definition of $d_{\text{TV}}$, 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

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

□

### B.4. Proof of Proposition 4.1

*Proof.*

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

□

### B.5. 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:

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

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

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

Moving the error terms to one side, we have

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

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

### B.6. Omitted Assumptions and Proof of Theorem 4.6

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

*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 $Y = -1$:

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

Similarly we have

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

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

$$\begin{aligned}
&d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) \\
=&\left| \int \mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(1-\omega_x(h))dx \right| \\
=&\Big| \underbrace{\int_{X_+(h)} \mathbb{P}_{\mathcal{D}}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(1-\omega_x(h))dx}_{\leq 0} \\
&+ \underbrace{\int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(1-\omega_x(h))dx}_{>0} \Big| \\
=&-\int_{X_+(h)} \mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(1-\omega_x(h))dx \\
&-\int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(1-\omega_x(h))dx && \text{(by Assumption 4.3)} \\
=&\int_{X_+(h)} \mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(\omega_x(h)-1)dx \\
&+\int_{X_-(h)} \mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(\omega_x(h)-1)dx && \text{(by equation 10)} \\
=&\int \mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(\omega_x(h)-1)dx
\end{aligned}$$

Similarly, by assumption 4.4 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:

$$\begin{aligned}
&d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) \\
=&\int \mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(\omega_x(h)-1)dx \\
&-\int \mathbb{P}_{\mathcal{D}}(h(x)=+1|X=x)\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(\omega_x(h)-1)dx \\
=&\int [\mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x) - \mathbb{P}_{\mathcal{D}_S}(h(x)=+1|X=x)]\mathbb{P}_{\mathcal{D}_S}(X=x)\cdot(\omega_x(h)-1)dx \\
=&\mathbb{E}_{\mathcal{D}_S}[(\mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x) - \mathbb{P}_{\mathcal{D}_S}(h(x)=+1|X=x))(\omega_x(h)-1)] && \text{(by Assumption 4.5)} \\
>&\mathbb{E}_{\mathcal{D}_S}[\mathbb{P}_{\mathcal{D}_S}(Y=+1|X=x) - \mathbb{P}_{\mathcal{D}_S}(h(x)=+1|X=x)]\mathbb{E}_{\mathcal{D}_S}[\omega_x(h)-1] \\
=&0
\end{aligned}$$

Combining the above with Theorem 3.3, we have

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

□

### B.7. Omitted details for Section 4.3

With Setup 2 - Setup 4, we can further specify the important weight $w_x(h)$ for the strategic response setting:

**Lemma B.3.** *Recall the definition for the covariate shift important weight coefficient $\omega_x(h) := \frac{\mathbb{P}_{D(h)}(X=x)}{\mathbb{P}_{D_S}(X=x)}$, for our strategic response setting, we have,*

$$w_x(h) = \begin{cases} 1, & x \in [0, \tau_h - B) \\ \frac{\tau_h - x}{B}, & x \in [\tau_h - B, \tau_h) \\ \frac{1}{B}(-x + \tau_h + 2B), & x \in [\tau_h, \tau_h + B) \\ 1, & x \in [\tau_h + B, 1] \end{cases} \tag{12}$$

Proof for Lemma B.3:

*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 2, 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 3.
- For distribution between $[\tau_h, \tau_h + B]$, consider a particular feature $x^\star \in [\tau_h, \tau_h + B]$, under Setup 4, we know its new distribution becomes:

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

- For the target distribution between $[\tau_h + B, 1]$: under assumption 2 and 4, 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:

$$\omega_x(h) = \begin{cases} 1, & x \in [0, \tau_h - B) \text{ and } x \in [\tau_h + B, 1] \\ \frac{\tau_h - x}{B}, & x \in [\tau_h - B, \tau_h) \\ \frac{1}{B}(-x + \tau_h + 2B), & x \in [\tau_h, \tau_h + B) \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

□

Proof for Proposition 4.7:

*Proof.* According to Lemma B.3, 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 plugging it into the general bound for Theorem 4.2 gives us the desired result. □

### B.8. Proof of Theorem 5.1

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

$$\begin{aligned} \text{Err}_{\mathcal{D}(h^*_S)}(h^*_S) &= p(h^*_S) \cdot \text{Err}_+(h^*_S) + (1 - p(h^*_S)) \cdot \text{Err}_-(h^*_S) \qquad \text{(by definitions of } p(h^*_S), \text{Err}_+(h^*_S), \text{ and Err}_-(h^*_S)) \\ &= \underbrace{p \cdot \text{Err}_+(h^*_S) + (1 - p) \cdot \text{Err}_-(h^*_S)}_{\text{(I)}} + (p(h^*_S) - p)[\text{Err}_+(h^*_S) - \text{Err}_-(h^*_S)] \end{aligned} \tag{14}$$

We can expand (I) as follows:

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

Plugging this back into equation 14, we have

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

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 a uniform prior. Then

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

Adding together these two equations yields

$$\begin{aligned}
&(p(h_S^*) - p)[\text{Err}_+(h_S^*) - \text{Err}_-(h_S^*)] + (p - p(h_T^*)) \cdot [\text{Err}_+(h_T^*) - \text{Err}_-(h_T^*)] \\
&= 2(p(h_S^*) - p) \cdot (0.5 - \mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1)) + 2(p - p(h_T^*)) \cdot (0.5 - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)) \\
&= (p(h_S^*) - p(h_T^*)) - 2\left(p(h_S^*)\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - p(h_T^*)\mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)\right) \\
&\quad + 2p \cdot (\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)) \\
&\leq |p(h_S^*) - p(h_T^*)| \cdot (1 + 2|\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)|) \\
&\quad + 2p \cdot |\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)|
\end{aligned} \tag{15}$$

Meanwhile,

$$\begin{aligned}
&|\mathbb{P}_{\mathcal{D}_u}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}_u}(h_T^*(X) = +1)| \\
&\leq 0.5 \cdot |\mathbb{P}_{\mathcal{D}|Y=+1}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}|Y=+1}(h_T^*(X) = +1)| \\
&\quad + 0.5 \cdot |\mathbb{P}_{\mathcal{D}|Y=-1}(h_S^*(X) = +1) - \mathbb{P}_{\mathcal{D}|Y=-1}(h_T^*(X) = +1)| \\
&= 0.5\left(d_{\text{TV}}(\mathcal{D}_+(h_S^*), \mathcal{D}_+(h_T^*)) + d_{\text{TV}}(\mathcal{D}_-(h_S^*), \mathcal{D}_-(h_T^*))\right)
\end{aligned} \tag{16}$$

Combining equation 15 and equation 16 gives

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

□

### B.9. Proof of Theorem 5.2

We will make use of the following fact:

**Lemma B.4.** *Under label shift, $TPR_S(h) = TPR_h(h)$ and $FPR_S(h) = FPR_h(h)$.*

*Proof.* We have

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

The argument for $\text{TPR}_h(h) = \text{TPR}_S(h)$ 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 $\text{Err}_{\mathcal{D}_S}(h)$ and $\text{Err}_{\mathcal{D}(h)}(h)$:

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

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

$$d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) = |\mathbb{P}_{\mathcal{D}_S}(Y = +1) - \mathbb{P}_{\mathcal{D}(h)}(Y = +1)| = |p - p(h)| \tag{17}$$

In addition, we have

$$\mathcal{D}_{h|S} = \mathbb{P}_S(h(X) = +1) = p \cdot \text{TPR}_S(h) + (1 - p) \cdot \text{FPR}_S(h) \tag{18}$$

Similarly

$$\begin{aligned}
\mathcal{D}_h(h) &= \mathbb{P}_{\mathcal{D}(h)}(h(X) = +1) \\
&= p(h) \cdot \text{TPR}_h(h) + (1 - p(h)) \cdot \text{FPR}_h(h) \\
&= p(h) \cdot \text{TPR}_S(h) + (1 - p(h)) \cdot \text{FPR}_S(h) && \text{(by Lemma B.4)}
\end{aligned} \tag{19}$$

Therefore

$$\begin{aligned}
d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) =& |\mathbb{P}_{\mathcal{D}_S}(h(X) = +1) - \mathbb{P}_{\mathcal{D}(h)}(h(X) = +1)| \\
=& |(p - p(h)) \cdot \text{TPR}_S(h) + (p(h) - p) \cdot \text{FPR}_S(h)| && \text{(By equation 19 and equation 18)} \\
=& |p - p(h)| \cdot |\text{TPR}_S(h) - \text{FPR}_S(h)| && (20)
\end{aligned}$$

which yields:

$$d_{\text{TV}}(\mathcal{D}_{Y|S}, \mathcal{D}_Y(h)) - d_{\text{TV}}(\mathcal{D}_{h|S}, \mathcal{D}_h(h)) = |p - p(h)|(1 - |\text{TPR}_S(h) - \text{FPR}_S(h)|) \quad \text{(By equation 17 and equation 20)}$$

completing the proof. □

### B.10. Proof of Proposition 5.3

*Proof.*

$$\begin{aligned}
&|p(h_S^*) - p(h_T^*)| \cdot \frac{1}{\mathbb{P}_{\mathcal{D}_S}(Y=+1)} \\
=&\frac{|(1-\text{Err}_{\mathcal{D}_S}(h_S^*))\text{TPR}_S(h_S^*) - (1-\text{Err}_{\mathcal{D}_S}(h_T^*))\text{TPR}_S(h_T^*)|}{(1-\text{Err}_{\mathcal{D}_S}(h_S^*))\cdot(1-\text{Err}_{\mathcal{D}_S}(h_T^*))} \\
\leq&\frac{|\text{Err}_{\mathcal{D}_S}(h_S^*) - \text{Err}_{\mathcal{D}_S}(h_T^*)|\cdot|\text{TPR}_S(h_S^*) - \text{TPR}_S(h_T^*)|}{(1-\text{Err}_{\mathcal{D}_S}(h_S^*))\cdot(1-\text{Err}_{\mathcal{D}_S}(h_T^*))}
\end{aligned} \tag{21}$$

The inequality above is due to Lemma 7 of (Liu & Liu, 2015). □

## C. Lower Bound and Example for Target Shift

### C.1. 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 C.1.** *Under target shift, any model $h$ must incur the following error on either the $\mathcal{D}_S$ or $\mathcal{D}(h)$:*

$$\begin{aligned}
&\max\{\mathit{Err}_{\mathcal{D}_S}(h), \mathit{Err}_{\mathcal{D}(h)}(h)\} \\
\geq&\frac{|p-p(h)|\cdot(1-|\mathit{TPR}_S(h)-\mathit{FPR}_S(h)|)}{2}.
\end{aligned}$$

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: $p - p(h)$. 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.

### C.2. 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 $+1$:

$$[\textbf{Fitness of } Y=+1] := \mathbb{P}_{\mathcal{D}_S}(h(X)=+1|Y=+1) \tag{22}$$

Then we have $\mathbb{E}[\textbf{Fitness of } Y] = \text{Err}_{\mathcal{D}_S}(h)$, and

$$\frac{p(h)}{\mathbb{P}_{\mathcal{D}_S}(Y=+1)} = \frac{\mathbb{P}_{\mathcal{D}_S}(h(X)=+1|Y=+1)}{\text{Err}_{\mathcal{D}_S}(h)}$$

Plugging the result back to our Theorem 5.1 we have

**Proposition C.2.** *Under the replicator dynamics model in Eqn. (22), $|p(h_S^*) - p(h_T^*)|$ further bounds as:*

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

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.

# D. Missing Experimental Details

## D.1. Synthetic Experiments Using DAG

Here we provide details in terms of the data-generating process for the simulated dataset.

**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 $\sigma_2^2$ and $\sigma_3^2$ 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. $\alpha$, $\mu_y$, $\sigma^2$,$\sigma_2^2$ and $\sigma_3^2$ 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 ($Y' = +1$).

## D.2. Synthetic Experiments Using Real-world Data

On the preprocessed FICO credit score data set (Board of Governors of the Federal Reserve System (US), 2007; Hardt et al., 2016b), 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 (detailed in Appendix D.2.1) to model the real-world scenario of repeated resource allocation (with decision denoted as $D$).

#### D.2.1. Parameters for Dynamics

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 choices. In our experiments, we set $\epsilon_1 = \epsilon_2 = \sigma = 0.1$.

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 $(X_{t,1}, X_{t,2}, X_{t,3}, D_t, Y_t)$. At each time step $T = t$, the institution first estimates the credit score $Q_t$ (which is not directly visible to the institution, but is reflected in the visible outcome label $Y_t$) based on $(A_t, X_{t,1}, X_{t,2}, X_{t,3})$, then produces the binary decision $D_t$ according to the optimal threshold (in terms of the accuracy).

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. In our experiments, we set $\alpha_D = 0.01$ and $\alpha_Y = 0.005$ to capture the scenario where one-step influence of the decision on the credit score is stronger than that for ground truth label.

#### D.2.2. Additional Experimental Results

In this section, we present additional experimental results on the real-world FICO credit score data set. With the initialization of the distribution of credit score $Q$ and the specified dynamics, we present results comparing the influence of vanilla regularization terms in decision-making (when estimating the credit score $Q$) on the calculation of bounds for induced risks.[7] In particular, we consider $L1$ norm (Figure 5) and $L2$ norm (Figure 6) regularization terms when optimizing decision-making policies on the source domain. As we can see from the results, applying vanilla regularization terms (e.g., $L1$ norm and $L2$ norm) on source domain without specific considerations of the inducing-risk mechanism does not provide significant performance improvement in terms of smaller induced risk. For example, there is no significant decrease of the term Diff as the regularization strength increases, for both $L1$ norm (Figure 5) and $L2$ norm (Figure 6) regularization terms.

## E. Challenges in Minimizing Induced Risk

In this section, we provide discussion on the challenges in performing induced domain adaptation.

### E.1. Computational Challenges

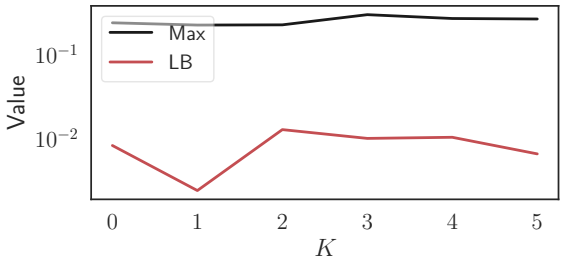
The literature of domain adaptation has provided us solutions to minimize the risk on the target distribution via a nicely developed set of results (Sugiyama et al., 2008; 2007; Shimodaira, 2000). This allows us to extend the solutions to minimize the induced risk too. Nonetheless we will highlight additional computational challenges.

We focus on the covariate shift setting. The scenario for target shift is similar. For covariate shift, recall that earlier we derived the following fact:

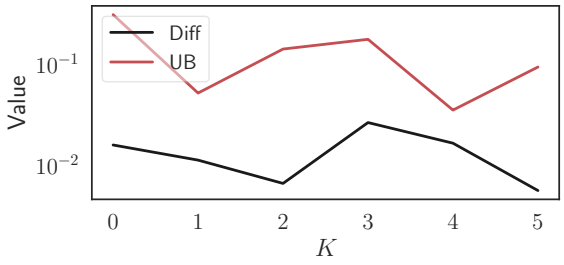
$$\mathbb{E}_{\mathcal{D}(h)}[\ell(h; X, Y)] = \mathbb{E}_{\mathcal{D}}[\omega_x(h) \cdot \ell(h; x, y)]$$

This formula informs us that a promising solution that uses $\omega_x(h)$ to perform reweighted ERM. Of course, the primary challenge that stands in the way is how do we know $\omega_x(h)$. There are different methods proposed in the literature to estimate $\omega_x(h)$ when one has access to $\mathcal{D}(h)$ (Zhang et al., 2013; Long et al., 2016; Gong et al., 2016). How any of the specific techniques work in our induced domain adaptation setting will be left for a more thorough future study. In this section, we

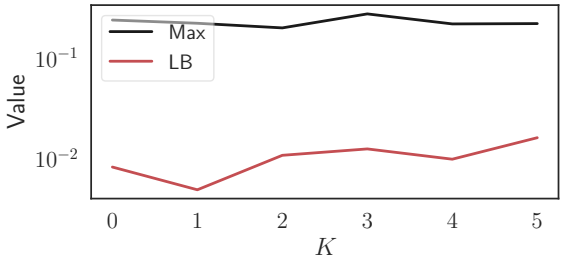
[7] The regularization that involves induced risk considerations will be discussed in Appendix G.



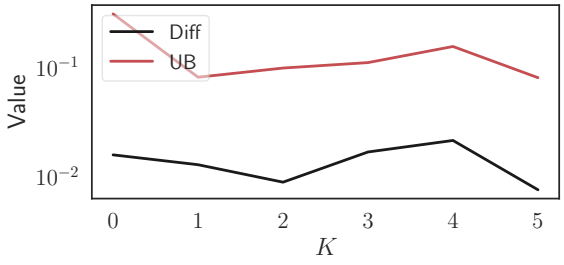

(a) $L1$ penalty, strong regularization strength.



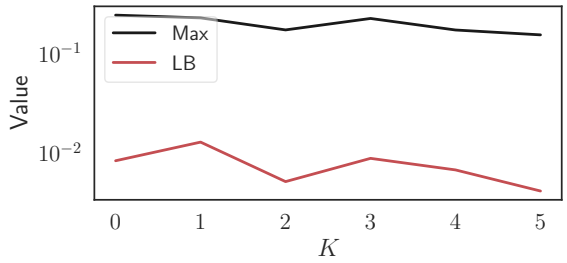

(b) $L1$ penalty, strong regularization strength.



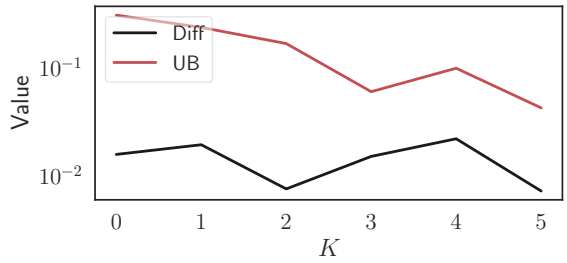

(c) $L1$ penalty, medium regularization strength.




(d) $L1$ penalty, medium regularization strength.




(e) $L1$ penalty, weak regularization strength.




(f) $L1$ penalty, weak regularization strength.

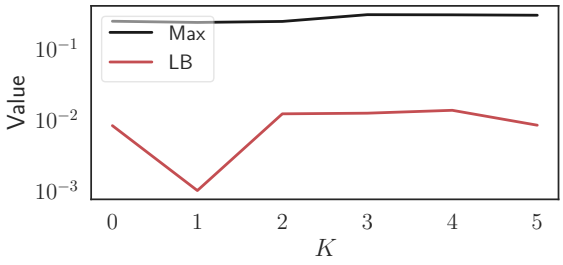
*Figure 5.* Results of applying $L1$ penalty with different strength when constructing $h_S^*$. The left column consisting of panels (a), (c), and (e) compares $\mathsf{Max} := \max\{\mathrm{Err}_{\mathcal{D}_S}(h_T^*), \mathrm{Err}_{\mathcal{D}(h_T^*)}(h_T^*)\}$ and $\mathsf{LB}$ := lower bound specified in Theorem 4.6. The right column consisting of panels (b), (d), and (f) compares $\mathsf{Diff} := \mathrm{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \mathrm{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$ and $\mathsf{UB}$ := upper bound specified in Theorem 4.2. For each time step $K = k$, we compute and deploy the source optimal classifier $h_S^*$ and update the credit score for each individual according to the received decision as the new reality for time step $K = k + 1$.

focus on explaining the computational challenges even when such knowledge of $\omega_x(h)$ can be obtained for each model $h$ being considered during training.

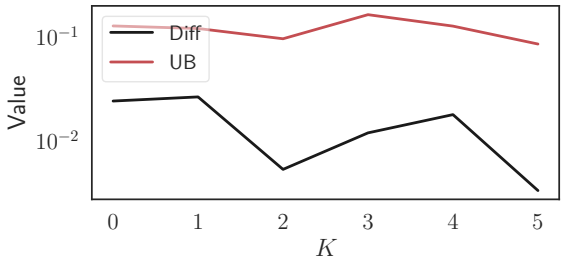
Though $\omega_x(h), \ell(h; x, y)$ might both be convex with respect to (the output of) the classifier $h$, their product is not necessarily convex. Consider the following example:

*Example* 1 ($\omega_x(h) \cdot \ell(h; x, y)$ is generally non-convex). Let $\mathcal{X} = (0, 1]$. Let the true label of each $x \in \mathcal{X}$ be $y(x) = \mathbb{1}\left(x \geq \frac{1}{2}\right)$. Let $\ell(h; x, y) = \frac{1}{2}(h(x) - y)^2$, and let $h(x) = x$ (simple linear model). Notice that $\ell$ is convex in $h$. Let $\mathcal{D}$ be the uniform distribution, whose density function is $f_{\mathcal{D}} = \begin{cases} 1, & 0 < x \leq 1 \\ 0, & \text{otherwise} \end{cases}$. Notice that if the training data is drawn from $\mathcal{D}$, then $h$ is the linear classifier that minimizes the expected loss. Suppose that, since $h$ rewards large values of $x$, it induces decision subjects to shift towards higher feature values. In particular, let $\mathcal{D}(h)$ have density function

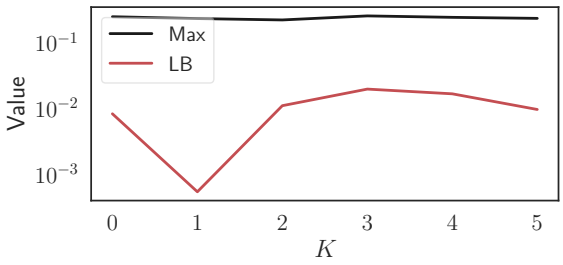
$$f_{\mathcal{D}(h)} = \begin{cases} 2x, & 0 < x \leq 1 \\ 0, & \text{otherwise} \end{cases}$$



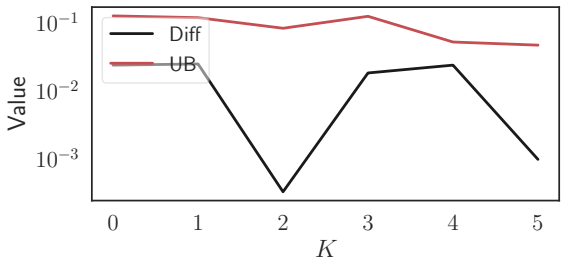

(a) $L2$ penalty, strong regularization strength.



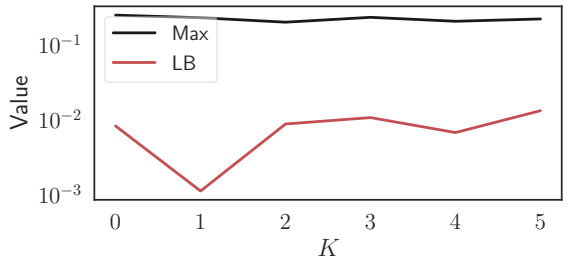

(b) $L2$ penalty, strong regularization strength.



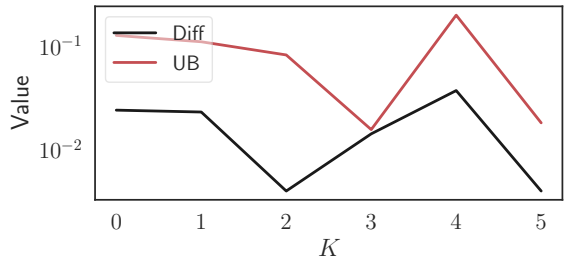

(c) $L2$ penalty, medium regularization strength.




(d) $L2$ penalty, medium regularization strength.




(e) $L2$ penalty, weak regularization strength.




(f) $L2$ penalty, weak regularization strength.

*Figure 6.* Results of applying $L2$ penalty with different strength when constructing $h_S^*$. The left column consisting of panels (a), (c), and (e) compares $\mathsf{Max} := \max\{\mathrm{Err}_{\mathcal{D}_S}(h_T^*), \mathrm{Err}_{\mathcal{D}(h_T^*)}(h_T^*)\}$ and $\mathsf{LB}$ := lower bound specified in Theorem 4.6. The right column consisting of panels (b), (d), and (f) compares $\mathsf{Diff} := \mathrm{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \mathrm{Err}_{\mathcal{D}(h_T^*)}(h_T^*)$ and $\mathsf{UB}$ := upper bound specified in Theorem 4.2. For each time step $K = k$, we compute and deploy the source optimal classifier $h_S^*$ and update the credit score for each individual according to the received decision as the new reality for time step $K = k + 1$.

Then for all $x \in \mathcal{X}$, $\omega_x(h) = \frac{f_{\mathcal{D}(h)}(x)}{f_{\mathcal{D}}(x)} = 2x$. Notice that $\omega_x(h) = 2x$ is convex in $h(x) = x$. Then

$$\begin{aligned}\omega_x(h) \cdot \ell(h; x, y) &= 2x \cdot \frac{1}{2}(h(x) - y)^2 \\ &= x(x-y)^2 = \begin{cases} x^3, & 0 < x < \frac{1}{2} \\ x(x-1)^2, & \frac{1}{2} \le x \le 1 \end{cases}\end{aligned}$$

which is clearly non-convex.

Nonetheless, we provide sufficient conditions under which $\omega_x(h) \cdot \ell(h; x, y)$ is in fact convex:

**Proposition E.1.** *Suppose $\omega_x(h)$ and $\ell(h; x, y)$ are both convex in $h$, and $\omega_x(h)$ and $\ell(h; x, y)$ satisfy $\forall h, h', x, y$: $(\omega_x(h) - \omega_x(h')) \cdot (\ell(h; x, y) - \ell(h'; x, y)) \ge 0$. Then $\omega_x(h) \cdot \ell(h; x, y)$ is convex.*

*Proof.* Let us use the shorthand $\omega(h) := \omega_x(h)$ and $\ell(h) := \ell(h; x, y)$. To show that $\omega(h) \cdot \ell(h)$ is convex, it suffices to show that for any $\alpha \in [0, 1]$ and any two hypotheses $h, h'$ we have

$$\omega(\alpha \cdot h + (1-\alpha) \cdot h') \cdot \ell(\alpha \cdot h + (1-\alpha) \cdot h') \le \alpha \cdot \omega(h) \cdot \ell(h) + (1-\alpha) \cdot \omega(h') \cdot \ell(h')$$

By the convexity of $\omega$,

$$\omega(\alpha \cdot h + (1-\alpha) \cdot h') \leq \alpha \cdot \omega(h) + (1-\alpha) \cdot \omega(h')$$

and by the convexity of $\ell$,

$$\ell(\alpha \cdot h + (1-\alpha) \cdot h') \leq \alpha \cdot \ell(h) + (1-\alpha) \cdot \ell(h')$$

Therefore it suffices to show that

$$\begin{aligned}
&[\alpha \cdot \omega(h) + (1-\alpha) \cdot \omega(h')] \cdot [\alpha \cdot \ell(h) + (1-\alpha) \cdot \ell(h')] - \alpha \cdot \omega(h) \cdot \ell(h) + (1-\alpha) \cdot \omega(h') \cdot \ell(h') \leq 0 \\
&\Leftrightarrow \alpha(\alpha-1) \cdot \omega(h)\ell(h) - \alpha(\alpha-1) \cdot [\omega(h)\ell(h') + \omega(h')\ell(h)] + \alpha(\alpha-1) \cdot \omega(h')\ell(h') \leq 0 \\
&\Leftrightarrow \alpha(\alpha-1) \cdot [\omega(h) - \omega(h')] \cdot [\ell(h) - \ell(h')] \leq 0 \\
&\Leftrightarrow [\omega(h) - \omega(h')] \cdot [\ell(h) - \ell(h')] \geq 0
\end{aligned}$$

By the assumed condition, the left-hand side is indeed non-negative, which proves the claim. □

This condition is intuitive when each $x$ belongs to a rational agent who responds to a classifier $h$ to maximize her chance of being classified as $+1$: For $y = +1$, the higher loss point corresponds to the ones that are close to decision boundary, therefore, more $-1$ negative label points might shift to it, resulting to a larger $\omega_x(h)$. For $y = -1$, the higher loss point corresponds to the ones that are likely mis-classified as +1, which "attracts" instances to deviate to.

### E.2. Challenges due to the lack of access to data

In the standard domain adaptation settings, one often assumes the access to a sample set of $X$, which already poses challenges when there is no access to label $Y$ after the adaptation. Nonetheless, the literature has observed a fruitful development of solutions (Sugiyama et al., 2008; Zhang et al., 2013; Gong et al., 2016).

One might think the above idea can be applied to our IDA setting rather straightforwardly by assuming observing samples from $\mathcal{D}(h)$, the induced distribution under each model $h$ during the training. However, we often do not know precisely how the distribution would shift under a model $h$ until we deploy it. This is particularly true when the distribution shifts are caused by human responding to a model. Therefore, the ability to "predict" accurately how samples "react" to $h$ plays a very important role (Ustun et al., 2019). Indeed, the strategic classification literature enables this capability by assuming full rational human agents. For a more general setting, building robust domain adaptation tools that are resistant to the above "prediction error" is also going to be a crucial criterion.

## F. Discussions On Performing Direct Induced Risk Minimization

In this section, we provide discussions on how to directly perform induced risk minimization for our induced domain adaptation setting. We first provide a gradient descent based method for a particular label shift setting where the underlying dynamic is replicator dynamic described in Section 5.3. Then we propose a solution for a more general induced domain adaptation setting where we do not make any particular assumptions on the undelying distribution shift model.

### F.1. Gradient descent based method

Here we provide a toy example of performing direct induced risk minimization under the assumption of label shift with underlying dynamics as the replicator dynamics described in Section 5.3.

**Setting** Consider a simple setting in which each decision subject is associated with a 1-dimensional continuous feature $x \in \mathbb{R}$ and a binary true qualification $y \in \{-1, +1\}$. We assume label shift setting, and the underlying population dynamic evolves the replicator dynamic setting described in Section 5.3. We consider a simple threshold classifier, where $\hat{Y} = h(x) = 1[X \geq \theta]$, meaning that the classifier is completely characterized by the threshold parameter $\theta$. Below we will use $\hat{Y}$ and $h(X)$ interchangeably to represent the classification outcome. Recall that the replicator dynamics is specified as follows:

$$\frac{\mathbb{P}_{\mathcal{D}(h)}(Y=y)}{\mathbb{P}_{\mathcal{D}_S}(Y=y)} = \frac{\textbf{Fitness}(Y=y)}{\mathbb{E}_{\mathcal{D}_S}[\textbf{Fitness}(Y)]} \tag{23}$$

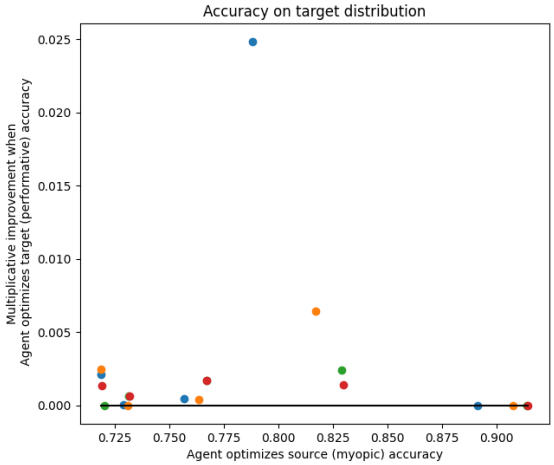



*Figure 7.* Experimental results of directly optimizing for the induced risk under the assumption of replicator dynamic. The X-axis denotes the prediction accuracy of $\text{Err}_{D(h_S^*)}(h_S^*)$, where $h_S^*$ is the source optimal classifier under each settings. The Y-axis is the percent of performance improvement using the classifier that optimize for $h_T^* = \arg\min \text{Err}_{D(h)}(h)$, which the decision maker considers the underlying response dynamics (according to replicator dynamics in Equation (23)) of the decision subjects. Different color represents different utility function, which is reflected by the specifications of values in $U_{y,\hat{y}}$; within each color, different dots represent different initial qualification rate.

where $\mathbb{E}_{\mathcal{D}_S}[\textbf{Fitness}(Y)] = \textbf{Fitness}(Y=y)\mathbb{P}_{\mathcal{D}_S}(Y=y) + \textbf{Fitness}(Y=-y)(1-\mathbb{P}_{\mathcal{D}_S}(Y=y))$. $\textbf{Fitness}(Y=y)$ is the fitness of strategy $Y=y$, which is further defined in terms of the expected utility $U_{y,\hat{y}}$ of each qualification-classification outcome pair $(y,\hat{y})$:

$$\textbf{Fitness}(Y=y) := \sum_{\hat{y}} \mathbb{P}[\hat{Y}=\hat{y}|Y=y]\cdot U_{y,\hat{y}}$$

where $U_{y,\hat{y}}$ is the utility (or reward) for each qualification-classification outcome combination.$\mathbb{P}(X|Y=y)$ is sampled according to a Gaussian distribution, and will be unchanged since we consider a label shift setting.

We initialize the distributions we specify the initial qualification rate $\mathbb{P}_{\mathcal{D}_S}(Y=+1)$. To test different settings, we vary the specification of the utility matrix $U_{y,\hat{y}}$ and generate different dynamics.

**Formulate the induced risk as a function of $h$** To minimize the induced risk, we first formulate the induced risk as a function of the classifier $h$'s parameter $\theta$ taking into account of the underlying dynamic, and then perform gradient descent to solve for locally optimal classifier $h_T^*$.

Recall from Section 5, under label shift, we can rewrite the induced risk as the following form:

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

where $p(h) = \mathbb{P}_{\mathcal{D}(h)}(Y=+1)$.

Since $\mathbb{E}_{\mathcal{D}_S}[\ell(h;X,Y)|Y=+1]$ and $\mathbb{E}_{\mathcal{D}_S}[\ell(h;X,Y)|Y=-1]$ are already functions of both $h$ and $\mathcal{D}_S$, it suffices to show that the accuracy on $\mathcal{D}(h)$, $p(h) = \mathbb{P}_{\mathcal{D}(h)}(Y=+1)$, can also be expressed as a function of $\theta$ and $\mathcal{D}_S$.

To see this, recall that for a threshold classifier $\hat{Y} = 1[X > \theta]$, it means that the prediction accuracy can be written as a

function of the threshold $\theta$ and target distribution $\mathcal{D}(h)$:

$$\begin{aligned}
&\mathbb{P}_{\mathcal{D}(h)}(Y=+1)\\
&=\mathbb{P}_{\mathcal{D}(h)}(\hat{Y}=+1,Y=+1)+\mathbb{P}_{\mathcal{D}(h)}(\hat{Y}=-1,Y=-1)\\
&=\mathbb{P}_{\mathcal{D}(h)}(X\geq\theta,Y=+1)+\mathbb{P}_{\mathcal{D}(h)}(X\leq\theta,Y=-1)\\
&=\int_{\theta}^{\infty}\mathbb{P}_{\mathcal{D}(h)}(Y=+1)\quad\underbrace{\mathbb{P}(X=x|Y=1)}_{\text{unchanged because of label shift}}\quad dx\\
&\quad+\int_{-\infty}^{\theta}\mathbb{P}_{\mathcal{D}(h)}(Y=-1)\quad\underbrace{\mathbb{P}(X=x|Y=-1)}_{\text{unchanged because of label shift}}\quad dx
\end{aligned}\tag{24}$$

where $\mathbb{P}(X|Y=y)$ remains unchanged over time, and $\mathbb{P}_{\mathcal{D}(h)}(Y=y)$ evolves over time according to Equation (23), namely

$$\begin{aligned}
&\mathbb{P}_{\mathcal{D}(h)}(Y=y)\\
=&\mathbb{P}_{\mathcal{D}_S}(Y=y)\times\frac{\textbf{Fitness}_g(Y=y)}{\mathbb{E}_{\mathcal{D}_S}[\textbf{Fitness}_g(Y)]}\\
=&\mathbb{P}_{\mathcal{D}_S}(Y=y)\times\frac{\sum_{\hat{y}}\mathbb{P}_{\mathcal{D}_S}[\hat{Y}=\hat{y}|Y=y,G=g]\cdot U_{\hat{y},y}}{\sum_y(\sum_{\hat{y}}\mathbb{P}_{\mathcal{D}_S}[\hat{Y}=\hat{y}|Y=y,G=g]\cdot U_{\hat{y},y})\mathbb{P}_{\mathcal{D}_S}[Y=y]}
\end{aligned}\tag{25}$$

Notice that $\hat{Y}$ is only a function of $\theta$, and $U_{y,\hat{y}}$ are fixed quantities, the above derivation indicates that we can express $\mathbb{P}_{\mathcal{D}(h)}(Y=y)$ as a function of $\theta$ and $\mathcal{D}_S$. Plugging it back to Equation (24), we can see that the accuracy can also be expressed as a function of the classifier's parameter $\theta$, indicating that the induced risk can be expressed as a function of $\theta$. Thus we can use gradient descent using automatic differentiation w.r.t $\theta$ to find a optimal classifier $h_T^*$ that minimize the induced risk.

**Experimental Results** Figure 7 shows the experimental results for this toy example. We can see that for each setting, compared to the baseline classifier $h_S^*$, the proposed gradient based optimization procedure returns us a classifier that achieves a better prediction accuracy (thus lower induced risk) compared to the accuracy of the source optimal classifier.

### F.2. General Setting: Induced Risk Minimization with Bandit Feedback

In general, finding the optimal classifier that achieves the optimal induced risk $h_T^*$ is a hard problem due to the interactive nature of the problem (see, e.g. the literature of performative prediction (Perdomo et al., 2020) for more detailed discussions). Without making any assumptions on the mapping between $h$ and $\mathcal{D}(h)$, one can only potentially rely on the *bandit feedbacks* from the decision subjects to estimate the influence of $h$ on $\mathcal{D}(h)$: when the induced risk is a convex function of the classifier $h$'s parameter $\theta$, one possible approach is to use the standard techniques from bandit optimization (Flaxman et al., 2005) to iteratively find induced optimal classifier $h_T^*$. The basic idea is: at each step $t=1,\cdots,T$, the decision maker deploy a classifier $h_t$, then observe data points sampled from $\mathcal{D}(h_t)$ and their losses, and use them to construct an approximate gradient for the induced risk as a function of the model parameter $\theta_t$. When the induced risk is a convex function in the model parameter $\theta$, the above approach guarantees to converge to $h_T^*$, and have sublinear regret in the total number of steps $T$.

The detailed description of the algorithm for finding $h_T^*$ is as follows:

**Algorithm 1** One-point bandit gradient descent for performative prediction

**Result:** return $\theta_T$ after $T$ rounds
$\theta_1 \leftarrow 0$
**foreach** *time step* $t \leftarrow 1, \ldots, T$ **do**
  Sample a unit vector $u_t \sim \mathrm{Unif}(\mathbf{S})$
  $\theta_t^+ \leftarrow \theta_t + \delta u_t$
  Observe data points $z_1, \ldots, z_{n_t} \sim \mathcal{D}(\theta_t^+)$
  $\widetilde{\mathsf{IR}}(\theta_t^+) \leftarrow \frac{1}{n_t}\sum_{i=1}^{n_t} \ell(z_i; \theta_t^+)$
  $\tilde{g}_t(\theta_t) \leftarrow \frac{d}{\delta}\widetilde{\mathsf{IR}}(\theta_t^+) \cdot u_t$ ▷ $\tilde{g}_t(\theta_t)$ *is an approximation of* $\nabla_\theta \widehat{\mathsf{IR}}(\theta_t)$
  $\theta_{t+1} \leftarrow \Pi_{(1-\delta)\Theta}(\theta_t - \eta \tilde{g}_t(\theta_t))$ ▷ *Take gradient step; project onto* $(1-\delta)\Theta := \{(1-\delta)\theta \mid \theta \in \Theta\}$
**end**

## G. Regularized Training

In this section, we discuss the possibility that indeed minimizing regularized risk will lead to a tighter upper bound. Consider the target shift setting. Recall that $p(h) := \mathbb{P}_{\mathcal{D}(h)}(Y = +1)$ and we have for any proper loss function $\ell$:

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

Suppose $p < p(h_T^*)$, now we claim that minimizing the following regularized/penalized risk leads to a smaller upper bound.

$$\mathbb{E}_{\mathcal{D}_S}[\ell(h; X, Y)] + \alpha \cdot \mathbb{E}_{\mathcal{D}_{\text{uniform}}}||\frac{h(X) + 1}{2}||$$

where in above $\mathcal{D}_{\text{uniform}}$ is a distribution with uniform prior for $Y$.

We impose the following assumption:

- The number of predicted $+1$ for examples with $Y = +1$ and for examples with $Y = -1$ are monotonic with respect to $\alpha$.

Consider the easier setting with $\ell = 0$-1 loss. Then

$$\begin{aligned}\mathbb{E}_{\mathcal{D}_{\text{uniform}}}||h(X)|| &= 0.5 \cdot (\mathbb{P}_{X|Y=+1}[h(X) = +1] + \mathbb{P}_{X|Y=-1}[h(X) = +1]) - 0.5 \\ &= 0.5 \cdot (\mathbb{E}_{X|Y=+1}[\ell(h(X), +1)] - \mathbb{E}_{X|Y=-1}[\ell(h(X), -1])\end{aligned}$$

The above regularized risk minimization problem is equivalent to

$$(p + 0.5 \cdot \alpha) \cdot \mathbb{E}_{X|Y=+1}[\ell(h(X), +1)] + (p - 0.5 \cdot \alpha) \cdot \mathbb{E}_{X|Y=-1}[\ell(h(X), -1]$$

Recall the upper bound in Theorem 5.1:

$$\begin{aligned}\mathrm{Err}_{\mathcal{D}(h_S^*)}(h_S^*) - \mathrm{Err}_{\mathcal{D}(h_T^*)}(h_T^*) \leq & \underbrace{|p(h_S^*) - p(h_T^*)|}_{\text{Term 1}} \\ & + (1 + p) \cdot \underbrace{(d_{\text{TV}}(\mathcal{D}_+(h_S^*), \mathcal{D}_+(h_T^*)) + d_{\text{TV}}(\mathcal{D}_-(h_S^*), \mathcal{D}_-(h_T^*))}_{\text{Term 2}}.\end{aligned}$$

With a properly specified $\alpha > 0$, this leads to a distribution with a smaller gap of $|p(\tilde{h}_S) - p(h_T^*)|$, where $\tilde{h}_S$ denotes the optimal classifier of the penalized risk minimization - this leads to a smaller Term 1 in the bound of Theorem 5.1. Furthermore, the induced risk minimization problem will correspond to an $\alpha$ s.t. $\alpha^* = \frac{p(h_T^*) - p}{0.5}$, and the original $h_S^*$ corresponds to a distribution of $\alpha = 0$. Using the monotonicity assumption, we will establish that the second term in Theorem 5.1 will also smaller when we tune a proper $\alpha$.

## H. Discussion on the tightness of our theoretical bounds

**General Bounds in Section 3** For the general bounds reported in Section 3, it is not trivial to fully quantify the tightness without further quantifying the specific quantities of the terms, e.g. the H divergence of the source and the induced distribution, and the average error a classifier have to incur for both distribution. This part of our results adapted from the classical literature in learning from multiple domains (Ben-David et al., 2010). The tightness of using $\mathcal{H}$-divergence and other terms seem to be partially validated therein.

**Bounds in Section 4 and Section 5** For more specific bounds provided in Section 4 (for covariate shift) and Section 5 (target shift), however, it is relatively easier to argue about the tightness: the proofs there are more transparent and are easier to back out the conditions where the inequalities are relaxed. For example, in Theorem 5.1, the inequalities of our bound are introduced primarily in the following two places: 1) one is using the optimiality of $h_S^*$ on the source distribution. 2) the other is bounding the statistical difference in $h_S^*$ and $h_T^*$'s predictions on the positive and negative examples. Both are saying that if the differences in the two classifiers' predictions are bounded in a range, then the result in Theorem 5.1 is relatively tight.