BOOSTING FOR PREDICTIVE SUFFICIENCY

Anonymous authors
Paper under double-blind review

ABSTRACT

Out-of-distribution (OOD) generalization is a defining hallmark of truly robust and reliable machine learning systems. Recently, it has been empirically observed that existing OOD generalization methods often underperform on real-world tabular data, where hidden confounding shifts drive distribution changes that boosting models handle more effectively. Part of boosting's success is attributed to variance reduction, handling missing variables, feature selection, and connections to multicalibration. This paper uncovers a crucial reason behind its success in OOD generalization: boosting's ability to infer stable environments robust to hidden confounding shifts and maximize predictive performance within those environments. This paper introduces an information-theoretic notion called α-predictive sufficiency and formalizes its link to OOD generalization under hidden confounding. We show that boosting implicitly identifies suitable environments and produces an α-predictive sufficient predictor. We validate our theoretical results through synthetic and real-world experiments and show that boosting achieves robust performance by identifying these environments and maximizing the association between predictions and true outcomes.

1 INTRODUCTION

The ability to generalize beyond the training distribution is a defining hallmark of trustworthy machine learning. Numerous methods have been proposed to enhance out-of-distribution (OOD) performance on data that differs from the in-distribution (ID) training data (Muandet et al., 2013; Arjovsky et al., 2019; Sagawa et al., 2019; Liu et al., 2021c; Zhou et al., 2022; Singh et al., 2024; Yang et al., 2024). These methods typically rely on assumptions such as invariance to ensure generalization beyond training environments. In practice, however, factors such as differing data-generating processes, selection bias, measurement error, and shifts in unobserved confounding variables often undermine the validity of these assumptions (Fan et al., 2014; Alabdulmohsin et al., 2023; Tsai et al., 2024; Liu & Cui, 2025; Prashant et al., 2025; Gowtham Reddy et al., 2025). Consequently, sophisticated methods for OOD generalization often underperform more traditional methods such as boosting, mixture-of-experts (MoE), and multi-layer perceptrons (MLP) (Gulrajani & Lopez-Paz, 2021; Vedantam et al., 2021; Rosenfeld et al., 2022; Gardner et al., 2023; Liu et al., 2023; Nastl & Hardt, 2024). It is therefore crucial to understand the underlying mechanisms that enable traditional methods to generalize better under real-world distribution shifts (Fan et al., 2014; Liu & Cui, 2025; Gowtham Reddy et al., 2025).

The nature of underlying distribution shifts guides the development of generalizable methods. Traditionally, it is assumed that distribution shifts are due to either label shift (Tachet des Combes et al., 2020; Garg et al., 2020; Alexandari et al., 2020; Wu et al., 2021) or covariate shift (Gretton et al., 2009; Sugiyama & Kawanabe, 2012; Schneider et al., 2020). Recent studies however reveal that *hidden confounding shift* is also prevalent in real-world data (Landeiro & Culotta, 2018; Reddy et al., 2022; Alabdulmohsin et al., 2023; Liu et al., 2023; Tsai et al., 2024; Reddy & N Balasubramanian, 2024; Prashant et al., 2025; Gowtham Reddy et al., 2025). Based on these assumptions, existing approaches are typically framed by partitioning the data to capture the inherent heterogeneity of the underlying distribution. Such partitioning—whether specified a priori or defined by researchers—serves as the basis for different notions of generalization such as invariance (Arjovsky et al., 2019; Krueger et al., 2021; Creager et al., 2021), robustness (Sagawa et al., 2019), multicalibration (Kim et al., 2019; Wald et al., 2021; Gopalan et al., 2022; Wu et al., 2024a), and predictive information (Gowtham Reddy et al., 2025).

A popular approach of defining such partitioning is to assign environment labels using metadata from the data-collection process. For instance, in housing price prediction, region identifiers such as zip codes or states are commonly used as environment labels (Gardner et al., 2023). Similarly, in medical diagnostics, hospital IDs—reflecting differences in equipment, protocols, and patient populations—often serve as environment labels when training models to predict disease outcomes from lab-test data. Data categorization into different subpopulations or environments may lead to different performances (Liu et al., 2021a; Liu & Cui, 2025). When the underlying environment labels are not available or the readily available environment labels do not accurately represent the underlying data heterogeneity, recent methods focus on identifying *correct* subpopulations so that the invariance relationships between covariates and labels can be learned effectively (Liu et al., 2021a;b; Lin et al., 2022; Liu et al., 2024).

Due to its reliance on the partitioning, OOD generalization is an algorithmic manifestation of the reference class problem: *Given a single case (an individual, an event, a situation), which group or "reference class" should we use to assign its probability?* For example, to predict the price of a house in New York City, one may consider the reference class to be the set of all houses in the New York City and attribute their average price as a prediction for the current house. Another possible reference class is the set of houses within a radius of 10 kilometers. Hence, the prediction depends heavily on the partition we choose. Philosophers have argued there is no purely objective way to pick a unique reference class (Hájek, 2007; Hu, 2025). Likewise, in OOD generalization, one must decide which features/relations are stable (e.g., causal mechanisms) and which are domain-specific artifacts. If the model partitions the data "wrong", e.g., grouping patients by hospital ID rather than disease mechanism, predictions fail OOD. This is particularly challenging when the distribution shifts are due to shifts in hidden confounders, as we discuss in § 3 (Landeiro & Culotta, 2018; Alabdulmohsin et al., 2023; Tsai et al., 2024; Prashant et al., 2025; Gowtham Reddy et al., 2025). Recent work addresses this challenge through automatic environment inference (Liu et al., 2021a; Creager et al., 2021) in the presence of hidden confounding shifts (Wu et al., 2024a). Unfortunately, there is no way to guarantee that the "correct" environments will be recovered.

This connection highlights an epistemic root of the OOD generalization problem: *it hinges on selecting the "right" partition of data into environments, factors, or causal classes*. This choice determines which distributional features are "stable" and transferable across domains, and which are merely spurious. Choosing the wrong partition, by contrast, leads to brittle predictors that exploit spurious correlations. However, since there is no way to resolve this ambiguity objectively, we argue that—instead of focusing on identifying the right partition—one should focus on designing methods that acknowledge this uncertainty explicitly and faithfully, for instance by working with sets of reference classes rather than a single one. To this end, recent theoretical analyses show that boosting models are the key to developing OOD generalization methods (Kim et al., 2019; Gopalan et al., 2022; Globus-Harris et al., 2023). These methods show a strong connection between multicalibration and boosting-based algorithms for regression (Globus-Harris et al., 2023; Wu et al., 2024a) and classification.

We conjecture that ensemble methods, such as boosting, owe their competitive performance in OOD settings to an inherent form of epistemic humility regarding the choice of reference classes. Specifically, the final predictions are aggregated from a diverse collection of weak learners, each of which captures the data from a distinct perspective, thereby mitigating overreliance on any single partition of the world. To formally investigate this, we define an information-theoretic notion called *α-predictive sufficiency*. We first show the connection between α-predictive sufficiency and generalization under a hidden confounding shift. We then show that boosting can be viewed as an algorithm that learns an α-predictive sufficient predictor. Unlike existing methods that identify ideal environment partitioning of data, we show that boosting implicitly learns environment labels corresponding to hidden confounding shifts. Since boosting returns α-predictive sufficient predictors, boosting can solve the OOD generalization problem under a hidden confounding shift. Unlike the traditional explanations for the success of boosting based methods, our explanations focus on the aspect of implicit identification of environment variables that lead to generalization under hidden confounding shift. Our contributions are as follows.

- We define an information-theoretic notion of α-predictive sufficiency. We then present its equivalence with the notion of generalization under hidden confounding shift, expressed in terms of predictive information between ground truth labels and predictions (§ 4).

- We show that the boosting algorithm returns a predictor that is α-predictive sufficient and, in doing so, boosting implicitly identifies environments corresponding to hidden confounding shifts (§ 5).
- Our experiments on synthetic and real-world data validate our claims that boosting implicitly captures hidden confounding shifts for generalization (§ 6).

2 Related work

OOD Generalization Under Hidden Confounding Shift. In recent years, out-of-distribution (OOD) generalization under hidden confounding has attracted considerable attention due to its prevalence in real-world data (Landeiro & Culotta, 2018; Alabdulmohsin et al., 2023; Tsai et al., 2024; Prashant et al., 2025). Solutions to this problem often include either adjusting for hidden confounder value (Alabdulmohsin et al., 2023; Tsai et al., 2024) or inferring hidden confounder value (Prashant et al., 2025) under proxy variable assumptions. Because the true confounder (parent of the outcome) is latent, achieving full invariance under hidden confounding is challenging. A practical alternative is to identify regions of the input distribution corresponding to different confounder values and deploy specialized predictors per region (Gowtham Reddy et al., 2025).

Model architecture integrally affects OOD behavior. When an architecture aligns with the underlying invariant structure, generalization improves (Li et al., 2023; Wu et al., 2024b). Motivated by the `if-then-else` like conditional structure of classifying visual attributes, Li et al. (2023) propose a sparse MoE model for learning generalizable models. More generally, models of them form: $\mathbb{P}(Y \mid \mathbf{X}) = \sum_U \mathbb{P}(U \mid \mathbf{X})\mathbb{P}(Y \mid \mathbf{X}, U)$ can equivalently be interpreted as routing weights $\mathbb{P}(U \mid \mathbf{X})$ with per-region invariant predictions $\mathbb{P}(Y \mid \mathbf{X}, U)$. Consequently, recent backbone designs for OOD robustness draw inspiration from MoE architectures (Wu et al., 2024b; Prashant et al., 2025). Since boosting can also be viewed as a model architecture aligning with the above-mentioned generative model, in this paper, we study how boosting methods provide structural inductive biases required for generalization under the hidden confounding shift. Even if boosting is acknowledged to perform well under hidden confounding shifts, understanding why boosting performs well is underexplored and is the main focus of this work. Our insights might also add another explanation why boosting methods perform successfully on typical tabular benchmark data suffering from relatively high amounts of noise with high risk of distribution shift between test and training data: They implicitly detect and partially control for hidden confounding.

Boosting, Multicalibration, and OOD Generalization. Multicalibration, originally proposed as a fairness tool (Hébert-Johnson et al., 2018), has recently been adapted for invariance learning (Wu et al., 2024a). Recent methods show the connection between multicalibration and boosting for regression (Globus-Harris et al., 2023; Wu et al., 2024a) and classification (Kim et al., 2019; Gopalan et al., 2022). Crucially, Bayes-optimality guarantees for these approaches depend on structural assumptions about the grouping (reference) functions used for multicalibration. Those assumptions can be restrictive in OOD settings (e.g., when hidden confounders misalign with the chosen groups), limiting the applicability of standard multicalibration/boosting guarantees to OOD generalization.

For instance, Wu et al. (2024a) enforce multicalibration across environments by grouping data according to the entire family of density-ratio functions between a target and a source distribution. Each ratio (or its thresholded variants) serves as a soft grouping function over the joint distribution of inputs and outputs. This approach automatically isolates the subpopulations that are most susceptible to distributional shift and thereby imparts invariance without requiring any explicit group annotations. However, note that this framework addresses covariate shifts in $\mathbb{P}(\mathbf{X})$ and concept shifts in $\mathbb{P}(Y \mid \mathbf{X})$, but does not explicitly accommodate shifts in $\mathbb{P}(\mathbf{X} \mid Y)$, which often arise under hidden confounder shifts and are studied in this paper.

Learning invariant relationships between covariates and labels across training environments is enough for inference at test time. Moreover, a grouping-function class must be rich enough that, for every allowed target, the joint density-ratio itself lies in the grouping function class. This "closure under density ratios" ensures that calibration on P_S carries over to every reweighted P_T. This process creates environment labels based on density ratios and then uses those environment labels for downstream invariant learning. Similar techniques for inferring environment labels can be found in (Liu et al., 2021a; Creager et al., 2021). Accordingly, learnability implicitly requires optimal labeling of data points into different environments. Under hidden confounding shift, this is particularly hard, since the corresponding environments are not directly observed or a priori known, but are critical for OOD performance. How boosting addresses this challenge is the focus of this paper.

Reference Class and Predictive Information. Density-ratio–based grouping acts as a stand-in for data partitioning by an unobserved confounder. While it may not recover the true latent assignments exactly, it reliably highlights the same-risk regions. Recently, (Gowtham Reddy et al., 2025) has shown that under hidden confounding shift, the objective of generalization can be viewed as maximizing predictive information between ground truth labels and predictions for each environment, where environments precisely encode shifts in hidden confounding values. In the ideal case, the environments should correspond to the reference classes with respect to the hidden confounder. In this work, we show how boosting achieves predictive information using the notion of predictive sufficiency, and also how boosting implicitly learns to cluster data with respect to hidden confounding.

3 OOD GENERALIZATION UNDER HIDDEN CONFOUNDING SHIFT

In this section, we provide the necessary background on hidden confounding and reference class to understand the rest of the paper.

Notations and preliminaries. Following Alabdulmohsin et al. (2023); Tsai et al. (2024); Prashant et al. (2025); Gowtham Reddy et al. (2025), we model hidden confounding shift using the causal graph shown in Figure 1. The causal graph contains covariates $\mathbf{X}$, label Y, environment variable E, hidden confounding variable U, feature extractor ϕ, and the prediction $\hat{Y}$. E encodes the shifts in the distribution of hidden confounding variable U across environments. A model prediction can be obtained as $\hat{Y} = (f \circ \phi)(\mathbf{X})$. Throughout the paper, we denote entropy and mutual information by $H(X) = -\mathbb{E}_X[\log(\mathbb{P}(X))]$ and $I(X;Y) = \mathbb{E}_{X,Y}[\log \frac{\mathbb{P}(X,Y)}{\mathbb{P}(X)\mathbb{P}(Y)}]$, respectively. Conditional entropy and conditional mutual information are defined similarly. We measure the model performance in an information-theoretic way using the mutual information $I(Y;\hat{Y})$ between the true label Y and predictive counterpart $\hat{Y}$. Following Federici et al. (2021) and Gowtham Reddy et al. (2025), we measure the concept shift by $I(Y; E \mid \phi(\mathbf{X}))$, which can also be viewed as a measure of invariance.

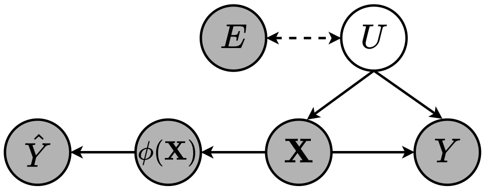

Figure 1: Causal graph for modeling hidden confounding shifts across environments.

Hidden Confounding Shift. Distribution shifts are usually modeled through a shift in the distribution of a hidden variable U and how that variable influences other observed variables $\mathbf{X}, Y$. For instance, when $U \to Y$ and $U \not\to \mathbf{X}$, we observe lable shift (Tachet des Combes et al., 2020; Garg et al., 2020; Alexandari et al., 2020; Wu et al., 2021). When $U \not\to Y$ and $U \to \mathbf{X}$, we observe covariate shift (Gretton et al., 2009; Sugiyama & Kawanabe, 2012; Schneider et al., 2020). In this work, we consider the case where $U \to Y$ and $U \to \mathbf{X}$, which is more prevalent in real-world data and is referred to as a hidden confounding shift(Alabdulmohsin et al., 2023; Liu et al., 2023; Tsai et al., 2024; Prashant et al., 2025; Gowtham Reddy et al., 2025).

We briefly review key definitions from causal graphical models (Pearl, 2009). A causal graph $\mathcal{G}$ is a directed graph whose vertices correspond to random variables and directed edges represent direct causal relationships. A path is a sequence of distinct nodes connected by edges, and it is said to be directed if every edge aligns with the direction of the path. Along such a path, one may refer to parents, children, ancestors, and descendants in the usual sense. Three elementary substructures are distinguished in a causal graph: (i) a *chain* $X_i \to X_j \to X_k$, (ii) a *fork* $X_i \leftarrow X_j \to X_k$, and (iii) a *collider* $X_i \to X_j \leftarrow X_k$. In both chains and forks, X_i and X_k are marginally dependent but become independent when conditioning on X_j. In a collider, X_i and X_k are marginally independent yet conditioning on the collider X_j (or its descendants) renders them dependent. A path is said to be *blocked* by a conditioning set $\mathcal{S}$ if either (a) it contains a chain or fork with its middle node in $\mathcal{S}$, or (b) it contains a collider such that neither the collider nor any of its descendants belongs to $\mathcal{S}$. Two nodes are conditionally independent given $\mathcal{S}$ precisely when every path between them is blocked.

3.1 Reference class for generalization under hidden confounding shift

As discussed in § 1 and § 2, the notion of reference class is crucial for OOD generalization if the generalization is achieved via multicalibration or predictive information. Here, we first formally state the definition of reference class, and environments induced by those reference classes (Definition 3.1), and then describe the crucial assumption of common confounder support.

Definition 3.1 (Reference classes and Environments)**.** *Let $\mathcal{X} \subset \mathbb{R}^d$ be the feature space and fix a subset of feature indices $S \subset \{1, \ldots, d\}$. For any feature vector $x \in \mathcal{X}$, we write x_S for the projection of x onto the coordinates in S. The* reference class *of x with respect to S is the set of all vectors that agree with x on those coordinates, i.e. $[x]_S := \{x' \in \mathcal{X} : x'_S = x_S\}$. Equivalently, on a finite dataset $D = \{x^{(1)}, \ldots, x^{(n)}\}$, we define an equivalence relation $x \sim_S x' \iff x_S = x'_S$, and the equivalence classes under $\sim_S$ are precisely the reference classes. The collection of all reference classes $\mathcal{E}_S := \{[x]_S : x \in \mathcal{X}\}$ (or, when working with a dataset, $\{[x]_S : x \in D\}$) forms environment partition of the data: each point belongs to exactly one reference class and different reference classes are disjoint.*

Assumption 3.1 (Common confounder support (Prashant et al., 2025))**.** *Let U denote the unobserved confounder taking values in some space $\mathcal{U}$. We assume that every confounder value that can occur in the target environment can also occur in the source (training) environment i.e., supp$(\mathbb{P}(U|\mathcal{E}_{te})) \subseteq$ supp$(\mathbb{P}(U|\mathcal{E}_{tr}))$. Equivalently, in finite-sample terms, the set of confounder levels observed in the target is contained in the set observed in the source.*

Assumption 3.1 rules out target-only confounder values and ensures that the training data provide examples from all confounder regions that may be encountered at test time. Common confounder support is necessary for generalization because of the decomposition $\mathbb{P}_{te}(y \mid x) = \sum_u \mathbb{P}_{te}(u \mid x)\mathbb{P}_{tr}(y \mid u, x)$. From this expression, if the $\mathbb{P}_{te}(u|x) > 0$ and $\mathbb{P}_{tr}(u|x) = 0$, then the model cannot learn $\mathbb{P}_{tr}(y \mid u, x)$ for those u vales during training. This leads to challenges in generalization because the model encounters previously unseen confounding patterns at test time.

4 α-Predictive sufficiency for OOD generalization

In this section, we use the notion of maximizing predictive information as the target metric for OOD generalization under hidden confounding shift. We then define the notion of α-predictive sufficiency (Definition 4.2) as a proof concept that relates to this target metric, and using the decomposition results recently proposed by Gowtham Reddy et al. (2025), we prove that achieving α-predictive sufficiency is the key to achieve generalization under hidden confounding shift (Proposition 4.1). We will leverage these insights to explain the success of boosting methods for OOD generalization under the hidden confounding shift. We begin by formally defining predictive information.

Definition 4.1 (Predictive Information)**.** *The predictive information between true outputs Y and model predictions $\hat{Y}$ is defined as the mutual information $I(Y; \hat{Y})$.*

In a recent work, Gowtham Reddy et al. (2025) show that predictive information can be decomposed into a combination of various terms such as *variation* $(I(\phi(\mathbf{X}); E \mid Y))$, *feature shift* $(I(\phi(\mathbf{X}); E))$, *label shift* $(I(Y; E))$, *concept shift* $(I(Y; E \mid \phi(\mathbf{X})))$, *conditional informativeness* $(I(\phi(\mathbf{X}); Y \mid E))$, and *residual* $(I(\phi(\mathbf{X}); Y \mid \hat{Y}))$. Furthermore, under the hidden confounding shift, the decomposition simplifies further and the predictive information can be represented in terms of conditional informativeness and residual:

$$I(Y; \hat{Y}) = I(Y; \phi(\mathbf{X}) \mid E) - I(Y; \phi(\mathbf{X}) \mid \hat{Y}), \tag{1}$$

Minimizing the residual term implies that all the information contained in $\phi(\mathbf{X})$ is utilized by the function f so that no residual information is left in $\phi(\mathbf{X})$. Conditional informativeness motivates maximizing the mutual information between representations and labels within each environment corresponding to the reference class induced by hidden confounder values. Model architectures such as boosting trees and MoE are especially good at modeling such environment-specific requirements (Wu et al., 2024b; Prashant et al., 2025; Li et al., 2023). Our goal in this work is to explain this puzzling empirical phenomenon and provide a better theoretical understanding of the mechanisms behind the success of boosting methods. A key technical quantity we use towards this goal is the information-theoretic notion of α-predictive sufficiency, formally defined below.

Definition 4.2 (α-Predictive Sufficiency)**.** *For $\alpha \geq 0$, a prediction $\hat{Y}$ is α-predictive sufficient for Y across environments E if the mutual information between prediction error $Y - \hat{Y}$ and E given prediction $\hat{Y}$ is less than or equal to α i.e., $I(Y - \hat{Y}; E \mid \hat{Y}) \leq \alpha$.*

Intuitively, 0-predictive sufficiency implies that the prediction error $Y - \hat{Y}$ is independent of E for each outcome value $\hat{Y}$. Since E encodes the information about U (a direct parent of Y), achieving predictive sufficiency implies that the predictor $\hat{Y}$ relies on $\mathbf{X}$ and Y through model training to implicitly account for the impact of U. This helps in learning a robust predictor capable of OOD generalization.

As a special case where f is an identity function, i.e., $\hat{Y} = (f \circ \phi)(\mathbf{X}) = \phi(\mathbf{X})$ and $\alpha = 0$, predictive sufficiency is equivalent to the concept shift $I(Y; E \mid \phi(\mathbf{X}))$. That is, under this special case, predictive sufficiency implies concept shift and hence invariance. Minimizing concept shift is known to be a primary factor for performance improvements (Liu et al., 2023; Gowtham Reddy et al., 2025). While α-predictive sufficiency is inspired by the notion of α-approximate multicalibration, we note that α-predictive sufficiency is a stronger condition than α-approximate multicalibration (Globus-Harris et al., 2023; Wu et al., 2024a). That is, predictive sufficiency implies multicalibration, but not vice versa. As a first technical result, we prove that α-predictive sufficiency naturally relates to the predictive information in the following proposition.

Proposition 4.1. *For a covariate vector $\mathbf{X}$, label Y, with causal structure $\mathbf{X} \to Y$, an environment variable E, a feature extractor ϕ, and prediction $\hat{Y}$, α-predictive sufficiency of $\hat{Y}$ for Y across environments E can be expressed as follows:*

$$I(Y - \hat{Y}; E \mid \hat{Y}) = -I(Y; \phi(\mathbf{X}) \mid E, \hat{Y}) + I(Y; \phi(\mathbf{X}) \mid \hat{Y}) + I(Y; E \mid \phi(\mathbf{X})). \tag{2}$$

Proofs of theoretical results are presented in Appendix § A. It follows from the causal graph in Figure 1 that $I(Y; \phi(\mathbf{X}) \mid E, \hat{Y}) \leq I(Y; \phi(\mathbf{X}) \mid E)$. This is because conditioning reduces mutual information unless the conditioned variable opens any spurious path between Y and $\phi(\mathbf{X})$. Hence $I(Y; \phi(\mathbf{X}) \mid E, \hat{Y})$ is a lower bound on the conditional informativeness. Thus, we note the crucial observation: *achieving α-predictive sufficiency can therefore be viewed as maximizing predictive information under the hidden confounding shift.* Building on this observation, we will next show that boosting can return a predictor that is α-sufficient for Y—thereby explaining the OOD generalization behavior of the boosting methods.

5 Boosting returns α-predictive sufficient predictor

In this section, we leverage the insights from the previous section to argue that boosting returns α-predictive sufficient predictor. On a mechanistic level, boosting has weak learning as its primitive. Crucially, boosting iteratively combines several weak learners to form a strong learner. There are several notions of weak learning (Natekin & Knoll, 2013; Schapire & Freund, 2013; Mayr et al., 2014; Bentéjac et al., 2021; Globus-Harris et al., 2023; Wu et al., 2024a), but on an intuitive level, a weak learner is a learner that performs slightly better than random guessing or a constant predictor. In the same spirit, we begin by defining an information-theoretic weak learner below. Let $\mathcal{H}$ be a hypothesis space with a set of hypothesis functions of the form $h : \mathbf{X} \to \mathbb{R}; h \in \mathcal{H}$.

Assumption 5.1 (γ-approximate weak learner)**.** *For any environment $e \in \mathcal{E}$, we assume that the hypothesis class $\mathcal{H}$ satisfies the γ-approximate weak learning condition in the sense that whenever the Bayes predictor Y^*, defined as $Y^*(\mathbf{X}) = \mathbb{P}(Y \mid \mathbf{X})$, yields strictly more predictive information than a baseline predictor c_e by a margin γ, i.e.,*

$$I(Y; Y^* \mid \mathbf{X} \in e) \geq \kappa(c_e) + \gamma.$$

where $\kappa(c_e) := I(Y; c_e \mid \mathbf{X} \in e)$. Then, there exists an $h \in \mathcal{H}$ that yields strictly more predictive information than a baseline predictor c_E by the same margin γ, i.e.,

$$I(Y; h(\mathbf{X}) \mid \mathbf{X} \in e) \geq \kappa(c_e) + \gamma.$$

In Assumption 5.1, $\kappa(c_e) = 0$ for a constant predictor c_e. In what follows, we consider the constant predictor c_e as a baseline predictor. Intuitively, a hypothesis class $\mathcal{H}$ satisfies γ-approximate weak

Algorithm 1 Standard boosting algorithm

```
Require: Step size $\eta$, base predictor $\hat{Y}_0$, hypothesis class $\mathcal{H}$, reweighting rule to obtain $\mathcal{D}_t$
1: Initialize $\hat{Y}_0 \leftarrow h_0, t \leftarrow 0$                    ▷ $h_0$ is often a constant predictor
2: while training error decreases do
3:     Find weak learner $h_{t+1} \in \mathcal{H}$ that maximizes $I(Y; h_{t+1}(\mathbf{X}))$ under distribution $\mathcal{D}_t$
4:     Update predictor: $\hat{Y}_{t+1} \leftarrow \hat{Y}_t + \eta h_{t+1}(\mathbf{X})$
5:     Update distribution $\mathcal{D}_{t+1}$ using the reweighting rule and increment $t$ by 1.
6: end while
```

learning condition if one can find a predictor $h \in \mathcal{H}$ that weakly improves the predictive information compared to the lower bound of γ on the irreducible predictive information of the Bayes predictor. We next state some standard assumptions before stating our main result.

Assumption 5.2 (Existence of reweighted distributions)**.** *At each round t, the exists a $p \in (0, 1)$ such that the boosting algorithm chooses reweighted distributions $\mathcal{D}_t(v)$ (e.g., based on model outputs/level sets) such that $\mathbb{P}(\hat{Y}_t = v) \geq p$ for all t.*

Assumption 5.2 avoids degenerated reweighting at each step and ensures non-trivial information is gained at each round t when combined with Assumption 5.3 as described below. Following Globus-Harris et al. (2023); Wu et al. (2024a), given any environment E, we assume access to a local oracle as described below.

Assumption 5.3 (Local Oracle)**.** *For any environment $e \in \mathcal{E}$, there exists an oracle $\mathcal{A}_{\mathcal{H}}$ that returns a hypothesis function $h' \in \mathcal{H}$ such that the following holds:*

$$h' \in \arg\max_{h \in \mathcal{H}} I(Y; h(\mathbf{X}) \mid \mathbf{X} \in e)$$

Assumption 5.4 (Strong learner is a deterministic function of weak learners)**.** *At any round t, $\hat{Y}_t$ is a deterministic function of $(h_0, \dots, h_t)$.*

Assumption 5.4 is required to decompose the predictive information $I(Y; \hat{Y})$ into contributions from each weak individual learner. In practice, this assumption holds as the models are usually deterministic once they are trained.

For our analysis, we consider the standard boosting algorithm described in Algorithm 1. We now show that this boosting algorithm can return an α-predictive sufficient predictor after training for a certain number of time steps.

Theorem 5.1. *Under Assumptions 5.2-5.4, there exists a finite $T < \infty$ such that the predictor $\hat{Y}_t$ learned by Algorithm 1 after $t \geq T$ rounds is α-predictive sufficient. The lower bound is given by*

$$T = \frac{H(Y) - H(Y \mid \mathbf{X}, E) - \alpha - I(Y; \hat{Y}_0)}{p \cdot \gamma}.$$

When the environment E is unknown, the same result holds by setting $H(Y \mid \mathbf{X}, E) = 0$.

In particular, Theorem 5.1 guarantees that the iterative boosting algorithm converges in finite number of iterations, and lead to the α-sufficient predictor.

We now prove the correspondence between a boosting model's leaf embeddings and the hidden confounder variable U. Since leaf embeddings in boosting model correspond to both outcome and input representations, we show how boosting achieves $H(U \mid \hat{Y}_T) \leq \delta$ for some $\delta > 0$ such that after the time step T, the uncertainty in U given the predictions $\hat{Y}_T$ is less than or equal to δ. We start with the assumption below.

Assumption 5.5. *We assume that $I(U; \hat{Y}) \geq c \cdot I(Y; \hat{Y})$ for some $c > 0$.*

From the causal graph shown in Figure 1, there exists a causal path from U to $\hat{Y}$ and hence there always exists a $c > 0$ that satisfies the Assumption 5.5. Another way of interpreting the Assumption 5.5 is that partial information about Y must come from U, which is true from the causal graph 1.

Corollary 5.1. *Under the Assumptions 5.2-5.5, there exists a finite $T < \infty$ such that the predictor $\hat{Y}_t$ learned by Algorithm 1 after $t \geq T$ rounds satisfies $H(U \mid \hat{Y}_t) \leq \delta$ for a small δ. The lower bound on T is given by*

$$T = \frac{H(U) - \delta - c \cdot I(Y; \hat{Y}_0)}{c \cdot p \cdot \gamma}$$

6 Experimental results

We perform experiments on both synthetic and real-world datasets to empirically explain how boosting excels at OOD generalization under hidden confounding shifts. Specifically, we compare the performance of boosting methods in terms of performance, predictive information, and predictive sufficiency. Code to reproduce the results is presented in the supplementary material. Additional results are presented in Appendix § B.

Methods: We experiment on two standard boosting algorithms: CatBoost (Dorogush et al., 2018) and XGBoost (Chen & Guestrin, 2016). We use t-SNE (Maaten & Hinton, 2008) and PCA (Abdi & Williams, 2010) as dimensionality reduction methods for visualizations. We perform hyperparameter tuning to choose the best model when comparing the performance of models.

Evaluation Metrics: We consider the test accuracy to evaluate the performance of models in a classification setting, and test mean squared error (MSE) in a regression setting. We use the Adjusted Rand Index (ARI) and the Normalized Mutual Information (NMI) to evaluate the goodness of clustering (usually with respect to the hidden confounding variables in synthetic data). We evaluate predictive sufficiency and predictive information and compare the performance of models related to these measures. We use the nonparametric entropy estimation toolbox (Kraskov et al., 2004; Steeg & Galstyan, 2011; 2013) to evaluate corresponding mutual information terms.

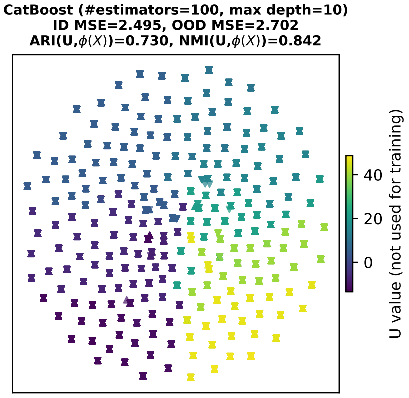

Figure 2: CatBoost representations with respect to hidden confounder value.

Synthetic Experiment 1: We use linear structural equations to generate synthetic data following the causal graph: $U \rightarrow X, U \rightarrow Y, X \rightarrow Y$, $U \sim \mathcal{N}(\mu_e, \sigma_e)$ where $e \in \{1, 2, \ldots, 10\}$, μ_e is sampled randomly between -50 and $+50$, and σ_e is set to 0.5. Each environment has 500 samples. We shift μ_e by a small fraction to induce a distribution shift where the test data lives. We train CatBoost and XGBoost on this dataset. To get the representations from these boosting methods, we obtain leaf embeddings for each data point as a vector of length d, where d is the number of trees in the model. We observe that *models that achieve low MSE are those whose representations are aligned with hidden confounder values*. As shown in Figure 2, the clusters of representations of a trained CatBoost model align with the hidden confounder value. For similar visualizations for various combinations of dimensionality reduction techniques, values of the number of estimators (trees), and the maximum depth hyperparameters of CatBoost and XGBoost models, see the results in Figures B1-B4 in Appendix B.

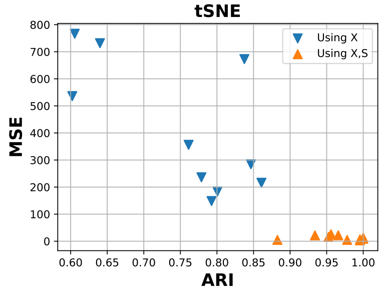

Figure 3: XGBoost ARI vs. MSE. Markers indicate results for different random seeds.

Synthetic Experiment 2: Next, we consider a causal graph of $U_1 \rightarrow X, U_1 \rightarrow Y, X \rightarrow Y, U_2 \rightarrow S, U_2 \rightarrow Y$, $U_1 \sim \mathcal{N}(\mu_e, \sigma_e), U_2 \sim \mathcal{N}(\mu_f, \sigma_f)$. We generate 10 environments, each containing 50 samples. Note that S does not causally influence the outcome Y. When XGBoost is only trained with X as input, it fails to capture the underlying shifts in hidden confounder values because neither

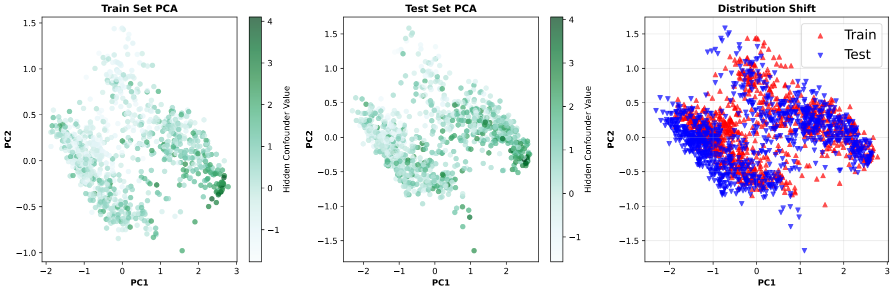

Figure 4: California housing dataset. XGBoost model representations are clustered according to hidden confounder values. There is a common confounder support between the train and test data.

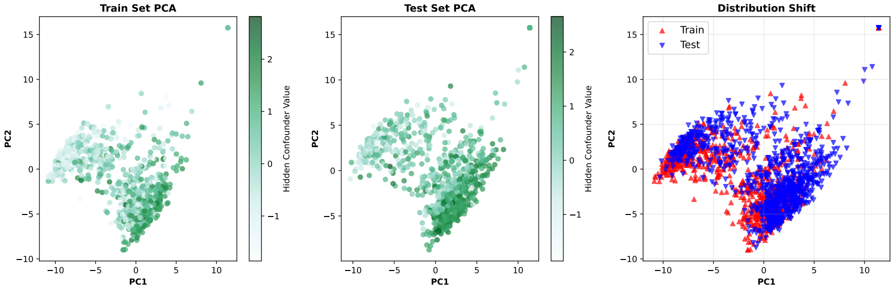

Figure 5: 20 Newsgroups dataset. XGBoost model representations are clustered according to the hidden confounder values. There is a common confounder support between the train and test data.

U_2 nor S are observed during training; see Figure 3 where ARI is low and MSE is high. However, when XGBoost is trained using both X and S as inputs, we observe low MSE and high ARI. This result explains the observed phenomena that *adding additional covariates helps in generalization performance* as observed in (Nastl & Hardt, 2024). Any covariate that acts as a proxy for an unobserved confounder helps improve generalization performance.

Real-world Data - California Housing Dataset: In this dataset, the goal is to predict the *median house price* based on the features *median income, house age, rooms, bedrooms, population, occupancy, latitude, longitude*. We simulate an artificial hidden confounding shift using observed covariates and the outcome. Specifically, we use a combination of the values of *median income, house age, and median house price* to induce a distribution shift. Notably, we ensure that the values of hidden confounders at test time belong to the set of hidden confounder values at train time (Assumption 3.1). The results in Figure 4 clearly show clustering of PCA representations of XGBoost leaf embeddings with respect to hidden confounder values.

Real-world Data - 20 Newsgroups: The goal is to predict the news category of a document from its raw text. Documents are vectorized using TF-IDF and subsequently reduced with Truncated SVD to have 200 features. We simulate an artificial hidden confounding shift between train and test data using a combination of features such as document length, keyword indicator, and the outcome (class index). Similar to the previous experiment, we ensure that the common confounding support (Assumption 3.1) holds. Figure 5 shows a clear association between clustering of PCA representations of XGBoost leaf embeddings and the clusters associated with hidden confounder values.

Table 1: Comparison of XGBoost and CatBoost on real-world datasets.

	California Housing			20 Newsgroups		
Method	**MSE**	**Pred. Info.**	**Pred. Suffi.**	**Accuracy**	**Pred. Info.**	**Pred. Suffi.**
XGBoost	0.31 ± 0.00	0.47 ± 0.03	$\mathbf{0.00 \pm 0.00}$	62.35 ± 0.40	0.27 ± 0.10	0.03 ± 0.00
CatBoost	$\mathbf{0.29 \pm 0.00}$	$\mathbf{0.56 \pm 0.10}$	$\mathbf{0.00 \pm 0.00}$	$\mathbf{62.61 \pm 0.00}$	$\mathbf{0.68 \pm 0.08}$	$\mathbf{0.00 \pm 0.00}$

Comparison of Performance and Predictive Information: Finally, we compare the performance of XGBoost and CatBoost and observe the underlying predictive information and predictive sufficiency values. It is evident from the results, shown in Table 1, that higher predictive information implies better performance, and a lower predictive sufficiency value means better performance. These results corroborate our theoretical claims.

7 Limitations, conclusions, and future work

We reframed OOD generalization under hidden confounding as a reference-class inference problem and introduced α-predictive sufficiency as an information-theoretic target that characterizes when predictors transfer across environments. We prove that standard boosting algorithms return α-predictive sufficient predictors in finitely many rounds, thereby implicitly inferring environments and maximizing predictive information, explaining their strong OOD behavior beyond variance reduction or feature selection alone. Empirically, across synthetic and real-world tabular tasks, boosting's learned representations cluster by hidden confounders and achieve high predictive information with low predictive-sufficiency residuals, aligning with the theory and yielding robust OOD performance. These results provide a principled account of why boosting often outperforms specialized OOD methods. We see this work as a foundation for new OOD algorithms that estimate or regularize α, relax common-support assumptions, and extend predictive-sufficiency guarantees beyond tabular settings.

Ethics and reproducibility statement

All authors have read and agree to adhere to the ICLR Code of Ethics. This work complies with all ethical guidelines outlined therein. Proofs of the theoretical results are presented in the appendix. The code and instructions to reproduce the results are provided in the supplementary material.

LLM statement

LLMs were used to aid or polish writing and to produce parts of the code.

REFERENCES

Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2010.

Ibrahim Alabdulmohsin, Nicole Chiou, Alexander D'Amour, Arthur Gretton, Sanmi Koyejo, Matt J. Kusner, Stephen R. Pfohl, Olawale Salaudeen, Jessica Schrouff, and Katherine Tsai. Adapting to latent subgroup shifts via concepts and proxies. In *International Conference on Artificial Intelligence and Statistics*, 2023.

Amr Alexandari, Anshul Kundaje, and Avanti Shrikumar. Maximum likelihood with bias-corrected calibration is hard-to-beat at label shift adaptation. In *International Conference on Machine Learning*. PMLR, 2020.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Candice Bentéjac, Anna Csörgő, and Gonzalo Martínez-Muñoz. A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review*, 2021.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *ACM SIGKDD international conference on knowledge discovery and data mining*, 2016.

Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *International Conference on Machine Learning*, 2021.

Anna Veronika Dorogush, Vasily Ershov, and Andrey Gulin. Catboost: gradient boosting with categorical features support. *arXiv preprint arXiv:1810.11363*, 2018.

Jianqing Fan, Fang Han, and Han Liu. Challenges of big data analysis. *National Science Review*, 2014.

Marco Federici, Ryota Tomioka, and Patrick Forré. An information-theoretic approach to distribution shifts. In *Advances in Neural Information Processing Systems*, 2021.

Joshua P Gardner, Zoran Popovi, and Ludwig Schmidt. Benchmarking distribution shift in tabular data with tableshift. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

Saurabh Garg, Yifan Wu, Sivaraman Balakrishnan, and Zachary Lipton. A unified view of label shift estimation. *Advances in Neural Information Processing Systems*, 2020.

Ira Globus-Harris, Declan Harrison, Michael Kearns, Aaron Roth, and Jessica Sorrell. Multicalibration as boosting for regression. In *International Conference on Machine Learning*, 2023.

Parikshit Gopalan, Adam Tauman Kalai, Omer Reingold, Vatsal Sharan, and Udi Wieder. Omnipredictors. In *nnovations in Theoretical Computer Science Conference*, 2022.

Abbavaram Gowtham Reddy, Celia Rubio-Madrigal, Rebekka Burkholz, and Krikamol Muandet. When shift happens-confounding is to blame. *arXiv preprint arXiv:2505.21422*, 2025.

Arthur Gretton, Alex Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, Bernhard Schölkopf, et al. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 2009.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Alan Hájek. The reference class problem is your problem too. *Synthese*, 2007.

Ursula Hébert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*, 2018.

Lily Hu. Does calibration mean what they say it means; or, the reference class problem rises again. *Philosophical Studies*, 2025.

Michael P Kim, Amirata Ghorbani, and James Zou. Multiaccuracy: Black-box post-processing for fairness in classification. In *AAAI/ACM Conference on AI, Ethics, and Society*, 2019.

Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical Review E*, 2004.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International conference on machine learning*, 2021.

Virgile Landeiro and Aron Culotta. Robust text classification under confounding shift. *Journal of Artificial Intelligence Research*, 2018.

Bo Li, Yifei Shen, Jingkang Yang, Yezhen Wang, Jiawei Ren, Tong Che, Jun Zhang, and Ziwei Liu. Sparse mixture-of-experts are domain generalizable learners. In *International Conference on Learning Representations*, 2023.

Yong Lin, Shengyu Zhu, Lu Tan, and Peng Cui. Zin: When and how to learn invariance without environment partition? *Advances in Neural Information Processing Systems*, 2022.

Jiashuo Liu and Peng Cui. Data heterogeneity modeling for trustworthy machine learning. *arXiv preprint arXiv:2506.00969*, 2025.

Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Heterogeneous risk minimization. In *International Conference on Machine Learning*, 2021a.

Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Kernelized heterogeneous risk minimization. In *Advances in Neural Information Processing Systems*, 2021b.

Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021c.

Jiashuo Liu, Tianyu Wang, Peng Cui, and Hongseok Namkoong. On the need for a language describing distribution shifts: Illustrations on tabular datasets. In *Advances in Neural Information Processing Systems*, 2023.

Jiashuo Liu, Jiayun Wu, Jie Peng, Xiaoyu Wu, Yang Zheng, Bo Li, and Peng Cui. Enhancing distributional stability among sub-populations. In *International Conference on Artificial Intelligence and Statistics*, 2024.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 2008.

Andreas Mayr, Harald Binder, Olaf Gefeller, and Matthias Schmid. The evolution of boosting algorithms. *Methods of information in medicine*, 2014.

Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, 2013.

Vivian Yvonne Nastl and Moritz Hardt. Do causal predictors generalize better to new domains? In *Advances in Neural Information Processing Systems*, 2024.

Alexey Natekin and Alois Knoll. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 2013.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Parjanya Prajakta Prashant, Seyedeh Baharan Khatami, Bruno Ribeiro, and Babak Salimi. Scalable out-of-distribution robustness in the presence of unobserved confounders. In *International Conference on Artificial Intelligence and Statistics*, 2025.

Abbavaram Gowtham Reddy and Vineeth N Balasubramanian. Detecting and measuring confounding using causal mechanism shifts. *Advances in Neural Information Processing Systems*, 2024.

Abbavaram Gowtham Reddy, Benin L Godfrey, and Vineeth N Balasubramanian. On causally disentangled representations. In *AAAI Conference on Artificial Intelligence*, 2022.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. Domain-adjusted regression or: Erm may already learn features sufficient for out-of-distribution generalization. *arXiv preprint arXiv:2202.06856*, 2022.

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

Robert E Schapire and Yoav Freund. Boosting: Foundations and algorithms. *Kybernetes*, 2013.

Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *Advances in Neural Information Processing Systems*, 2020.

Anurag Singh, Siu Lun Chau, Shahine Bouabid, and Krikamol Muandet. Domain generalisation via imprecise learning. In *International Conference on Machine Learning*, 2024.

Greg Ver Steeg and Aram Galstyan. Information transfer in social media, 2011.

Greg Ver Steeg and Aram Galstyan. Information-theoretic measures of influence based on content dynamics, 2013.

Masashi Sugiyama and Motoaki Kawanabe. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press, 2012.

Remi Tachet des Combes, Han Zhao, Yu-Xiang Wang, and Geoffrey J Gordon. Domain adaptation with conditional distribution matching and generalized label shift. *Advances in Neural Information Processing Systems*, 2020.

Katherine Tsai, Stephen R Pfohl, Olawale Salaudeen, Nicole Chiou, Matt Kusner, Alexander D'Amour, Sanmi Koyejo, and Arthur Gretton. Proxy methods for domain adaptation. In *International Conference on Artificial Intelligence and Statistics*, 2024.

Shanmukha Ramakrishna Vedantam, David Lopez-Paz, and David J. Schwab. An empirical investigation of domain generalization with empirical risk minimizers. In *Advances in Neural Information Processing Systems*, 2021.

Yoav Wald, Amir Feder, Daniel Greenfeld, and Uri Shalit. On calibration and out-of-domain generalization. *Advances in Neural Information Processing Systems*, 2021.

Jiayun Wu, Jiashuo Liu, Peng Cui, and Steven Z Wu. Bridging multicalibration and out-of-distribution generalization beyond covariate shift. *Advances in Neural Information Processing Systems*, 2024a.

Qitian Wu, Fan Nie, Chenxiao Yang, Tianyi Bao, and Junchi Yan. GraphSHINE: Training shift-robust graph neural networks with environment inference. In *The Web Conference 2024*, 2024b.

Ruihan Wu, Chuan Guo, Yi Su, and Kilian Q Weinberger. Online adaptation to label distribution shift. *Advances in Neural Information Processing Systems*, 2021.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *International Journal of Computer Vision*, 2024.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

APPENDIX

A PROOFS OF THEORETICAL RESULTS

Proposition 4.1. *For a covariate vector* $\mathbf{X}$, *label* Y, *with causal structure* $\mathbf{X} \to Y$, *an environment variable* E, *a feature extractor* ϕ, *and prediction* $\hat{Y}$, α*-predictive sufficiency of* $\hat{Y}$ *for* Y *across environments* E *can be expressed as follows:*

$$I(Y - \hat{Y}; E \mid \hat{Y}) = -I(Y; \phi(\mathbf{X}) \mid E, \hat{Y}) + I(Y; \phi(\mathbf{X}) \mid \hat{Y}) + I(Y; E \mid \phi(\mathbf{X})). \tag{2}$$

Proof. It follows directly that

$$\begin{aligned} I(Y; E \mid \hat{Y}) &= I(Y; E, \phi(\mathbf{X}) \mid \hat{Y}) - I(Y; \phi(\mathbf{X}) \mid E, \hat{Y}) \\ &= I(Y; \phi(\mathbf{X}) \mid \hat{Y}) + I(Y; E \mid \hat{Y}, \phi(\mathbf{X})) - I(Y; \phi(\mathbf{X}) \mid E, \hat{Y}) \\ &= I(Y; \phi(\mathbf{X}) \mid \hat{Y}) + I(Y, \hat{Y}; E \mid \phi(\mathbf{X})) - I(\hat{Y}; E \mid \phi(\mathbf{X})) - I(Y; \phi(\mathbf{X}) \mid E, \hat{Y}) \\ &= I(Y; \phi(\mathbf{X}) \mid \hat{Y}) + I(Y; E \mid \phi(\mathbf{X})) + I(\hat{Y}; E \mid \phi(\mathbf{X}), Y) - I(Y; \phi(\mathbf{X}) \mid E, \hat{Y}) \\ &= I(Y; \phi(\mathbf{X}) \mid \hat{Y}) + I(Y; E \mid \phi(\mathbf{X})) - I(Y; \phi(\mathbf{X}) \mid E, \hat{Y}). \end{aligned}$$

From the causal graph in Figure 1, we have $\hat{Y} \perp\!\!\!\perp E \mid \phi(\mathbf{X})$. Hence $I(\hat{Y}; E \mid \phi(\mathbf{X})) = I(\hat{Y}; E \mid \phi(\mathbf{X}), Y) = 0$ in the proof above. □

Theorem 5.1. *Under Assumptions 5.2-5.4, there exists a finite* $T < \infty$ *such that the predictor* $\hat{Y}_t$ *learned by Algorithm 1 after* $t \geq T$ *rounds is* α*-predictive sufficient. The lower bound is given by*

$$T = \frac{H(Y) - H(Y \mid \mathbf{X}, E) - \alpha - I(Y; \hat{Y}_0)}{p \cdot \gamma}.$$

When the environment E *is unknown, the same result holds by setting* $H(Y \mid \mathbf{X}, E) = 0$.

Proof. From Assumption 5.4, $\hat{Y}_T$ is a deterministic function of $(h_0, h_1, h_2, \ldots, h_T)$, and $I(Y; \hat{Y}_0) = I(Y; h_0)$ because $\hat{Y}_0$ is initialized to h_0 before boosting training loop. Now we have the following:

$$I(Y; \hat{Y}_T) \leq I(Y; h_0, \ldots, h_T) = I(Y; \hat{Y}_0) + \sum_{t=1}^{T} I(Y; h_t \mid h_1, \ldots, h_{t-1}) \tag{3}$$

Consider any round $t \geq 1$, and let $S := (h_1, \ldots, h_{t-1})$. Then, we have

$$I(Y; h_t \mid S) = \mathbb{E}_s[I(Y; h_t \mid S = s)] = \mathbb{E}_s[I(Y; h_t \mid S = s)\mathbb{1}_{s \in S_t}] \geq p \cdot \gamma. \tag{4}$$

From Equations (3) and (4) we have the following.

$$I(Y; \hat{Y}_T) \leq I(Y; \hat{Y}_0) + T \cdot p \cdot \gamma \tag{5}$$

Given $\hat{Y}$, $Y - \hat{Y}$ and Y have one-to-one correspondence. Hence, it follows that: $I(Y - \hat{Y}; E \mid \hat{Y}) = I(Y; E \mid \hat{Y})$. We next use the following simple upper bound on $I(Y; E \mid \hat{Y}_T)$:

$$I(Y; E \mid \hat{Y}_T) = H(Y \mid \hat{Y}_T) - H(Y \mid \hat{Y}_T, E) \leq H(Y \mid \hat{Y}_T), \tag{6}$$

and hence $I(Y; E \mid \hat{Y}_T) \leq H(Y \mid \hat{Y}_T)$. Furthermore, since $H(Y \mid \hat{Y}_T) = H(Y) - I(Y; \hat{Y}_T)$, we have: $I(Y; E \mid \hat{Y}_T) \leq H(Y) - I(Y; \hat{Y}_T)$. Now to get $I(Y; E \mid \hat{Y}_T) \leq \alpha$, it is sufficient to have $H(Y) - \alpha \leq I(Y; \hat{Y}_T)$, and following from Equation 5, it follows that

$$I(Y; \hat{Y}_0) + T \cdot p \cdot \gamma \geq H(Y) - \alpha$$

$$T \geq \frac{H(Y) - \alpha - I(Y; \hat{Y}_0)}{p \cdot \gamma}.$$

We argue that the our claim will hold if we can show that T is some finite integer. However, it is easy as the right hand side in the above expression is bounded from above by $\frac{\log K - \alpha}{p \cdot \gamma} \in \mathbb{R}_+$, and hence from the Archimedean property of the real numbers, we can always find a $T \in \mathbb{N}$ for which the property holds, thereby arguing the existence of the iteration step T for which the boosting algorithm achieves the argued α-predictive sufficiency. Furthermore, we recall from 6, $I(Y; E \mid \hat{Y}_T) = H(Y \mid \hat{Y}_T) - H(Y \mid \hat{Y}_T, E)$ Without removing the term with E, we have $I(Y; E \mid \hat{Y}_T) = H(Y \mid \hat{Y}_T) - H(Y \mid \hat{Y}_T, E) = H(Y) - I(Y; \hat{Y}_T) - H(Y \mid \hat{Y}_T, E)$.

Now to get $I(Y; E \mid \hat{Y}_T) \leq \alpha$, it is sufficient to have $H(Y) - H(Y \mid \hat{Y}_T, E) - \alpha \leq I(Y; \hat{Y}_T)$

$$I(Y; \hat{Y}_0) + Tp\gamma \geq H(Y) - H(Y \mid \hat{Y}_T, E) - \alpha$$
$$T \geq \frac{H(Y) - H(Y \mid \hat{Y}_T, E) - \alpha - I(Y; \hat{Y}_0)}{p \cdot \gamma}$$

To avoid the dependence on $\hat{Y}_T$, we can replace $H(Y \mid \hat{Y}_T, E)$ with its lower bound $H(Y \mid \mathbf{X}, E)$ and still have a valid bound as below.

$$T \geq \frac{H(Y) - H(Y \mid \mathbf{X}, E) - \alpha - I(Y; \hat{Y}_0)}{p \cdot \gamma}$$

□

Corollary 5.1. *Under the Assumptions 5.2-5.5, there exists a finite $T < \infty$ such that the predictor $\hat{Y}_t$ learned by Algorithm 1 after $t \geq T$ rounds satisfies $H(U \mid \hat{Y}_t) \leq \delta$ for a small δ. The lower bound on T is given by*

$$T = \frac{H(U) - \delta - c \cdot I(Y; \hat{Y}_0)}{c \cdot p \cdot \gamma}$$

Proof. From the proof of the Theorem 5.1, we have the following:

$$I(Y; \hat{Y}_T) \geq I(Y; \hat{Y}_0) + T \cdot p \cdot \gamma \tag{7}$$

From Assumption 5.5, we have the following:

$$I(U; \hat{Y}_T) \geq c \cdot (I(Y; \hat{Y}_0) + T \cdot p \cdot \gamma) \tag{8}$$
$$H(U \mid \hat{Y}_T) = H(U) - I(U; \hat{Y}_T) \leq H(U) - c \cdot (I(Y; \hat{Y}_0) + T \cdot p \cdot \gamma) \tag{9}$$
$$\tag{10}$$

To ensure $H(U \mid \hat{Y}_T) \leq \delta$, it is enough to ensure $H(U) - c \cdot (I(Y; \hat{Y}_0) + T \cdot p \cdot \gamma) \leq \delta$. Solving this for T implies the desired inequality below:

$$T \geq \frac{H(U) - \delta - c \cdot I(Y; \hat{Y}_0)}{c \cdot p \cdot \gamma} \tag{11}$$

□

B Additional Experimental Results

Figures B1- B4 show the results with respect to various choices of hyperparameters, dimensionality reduction methods, and metrics. A key takeaway from these results is that better clustering with respect to hidden confounders is consistently associated with better performance. This explains the reason behind the success of boosting methods. Figure B5 shows ID and OOD MSE for different values of distribution shifts. Since high distribution shifts cannot satisfy the common confounder support assumption, the generalization performance drops significantly due to large shifts in data distribution. Figure B6 shows performance comparison of XGBoost and CatBoost with invariant

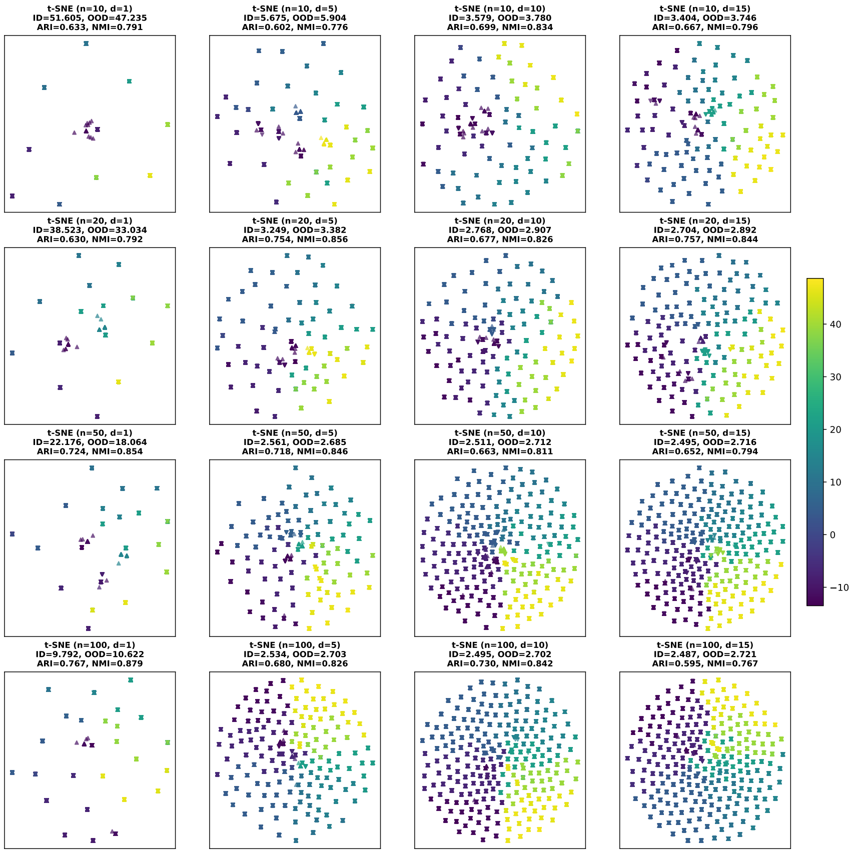

Figure B1: Results on CatBoost method. Dimensionality reduction is done using t-SNE. Results show the goodness of clustering with respect to the hidden confounder value. Above each subplot, clustering metrics: ARI, NMI are reported along with hyperparameter choices, ID, and OOD MSE values.

risk minimization (IRM) (Arjovsky et al., 2019) and group DRO (Sagawa et al., 2019). We consider the same setup as the synthetic experiment 2 presented in the main paper, where the causal graph is: $U_1 \rightarrow X, U_1 \rightarrow Y, X \rightarrow Y, U_2 \rightarrow S, U_2 \rightarrow Y$, $U_1 \sim \mathcal{N}(\mu_e, \sigma_e), U_2 \sim \mathcal{N}(\mu_f, \sigma_f)$. When only X is used as input, we observe that XGBoost and CatBoost perform better than IRM and GroupDRO, indicating that boosting methods are effective under a hidden confounding shift. When we use both X, S as inputs, only GroupDRO performs on par with boosting, while IRM still performs worse. This also shows that invariance learning is insufficient for generalization under a hidden confounding shift.

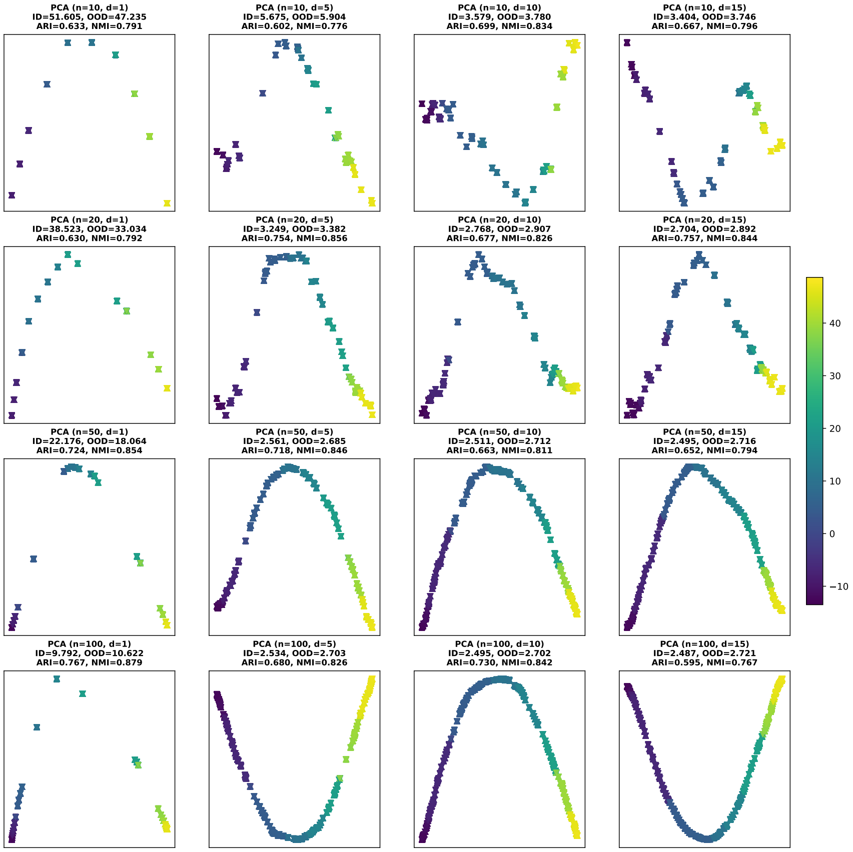

Figure B2: Results on CatBoost method. Dimensionality reduction is done using PCA. Results show the goodness of clustering with respect to the hidden confounder value. Above each subplot, clustering metrics: ARI, NMI are reported along with hyperparameter choices, ID, and OOD MSE values.

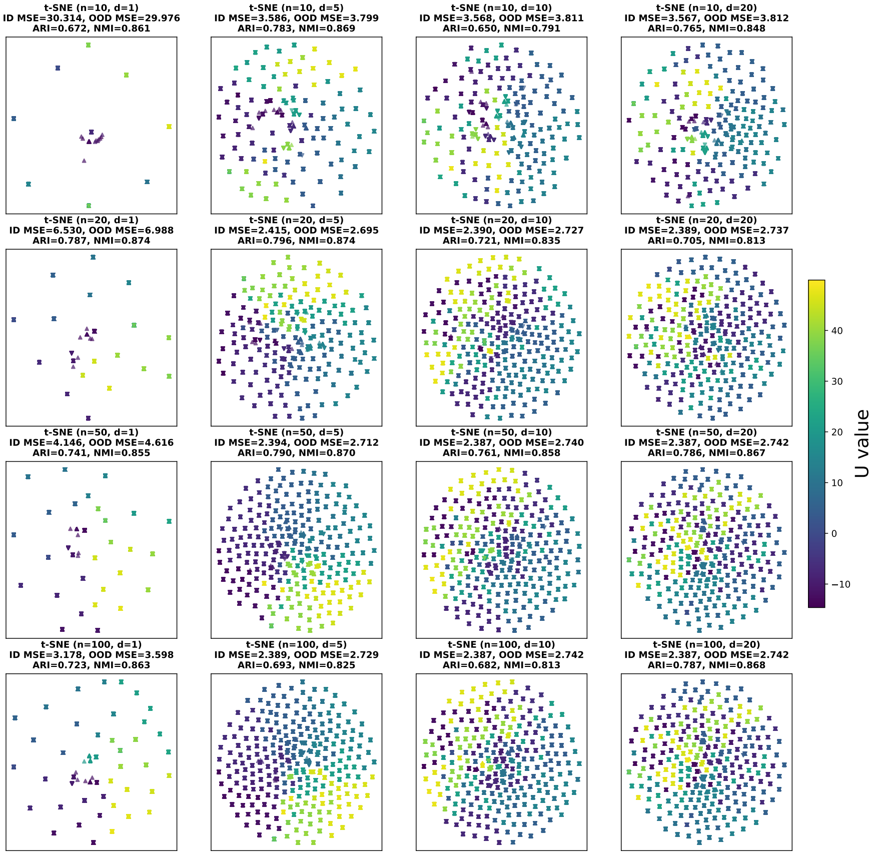

Figure B3: Results on XGBoost method. Dimensionality reduction is done using t-SNE. Results show the goodness of clustering with respect to the hidden confounder value. Above each subplot, clustering metrics: ARI, NMI are reported along with hyperparameter choices, ID, and OOD MSE values.

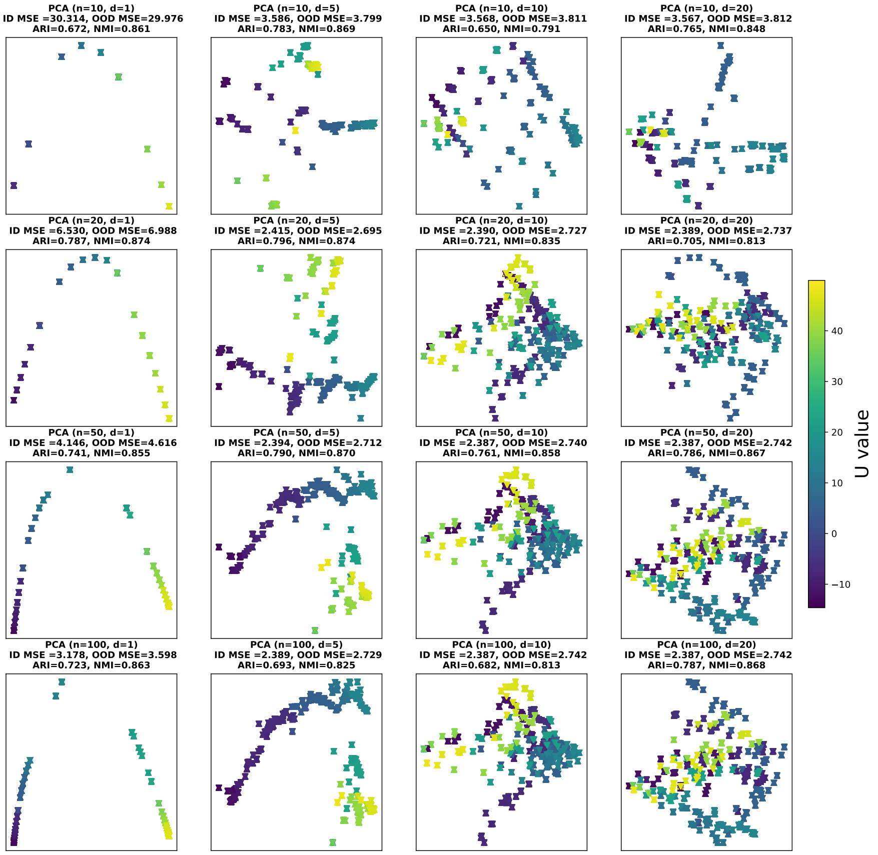

Figure B4: Results on XGBoost method. Dimensionality reduction is done using PCA. Results show the goodness of clustering with respect to the hidden confounder value. Above each subplot, clustering metrics: ARI, NMI are reported along with hyperparameter choices, ID, and OOD MSE values.

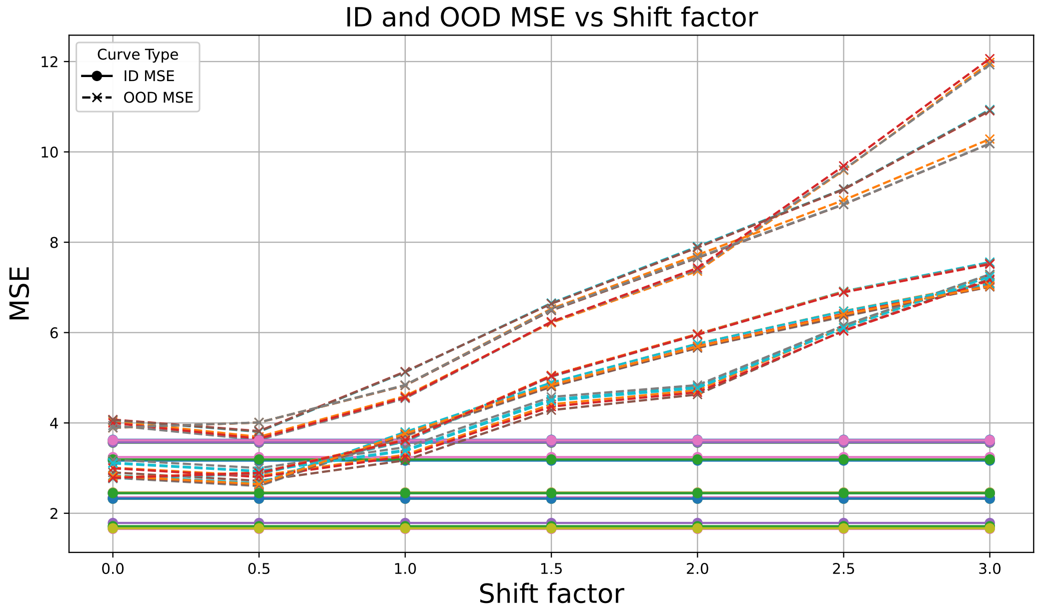

Figure B5: Performance of XGBoost for different shift factor values. Colors indicate different combinations of *number of trees, number of samples in each domain, and depths of trees*.

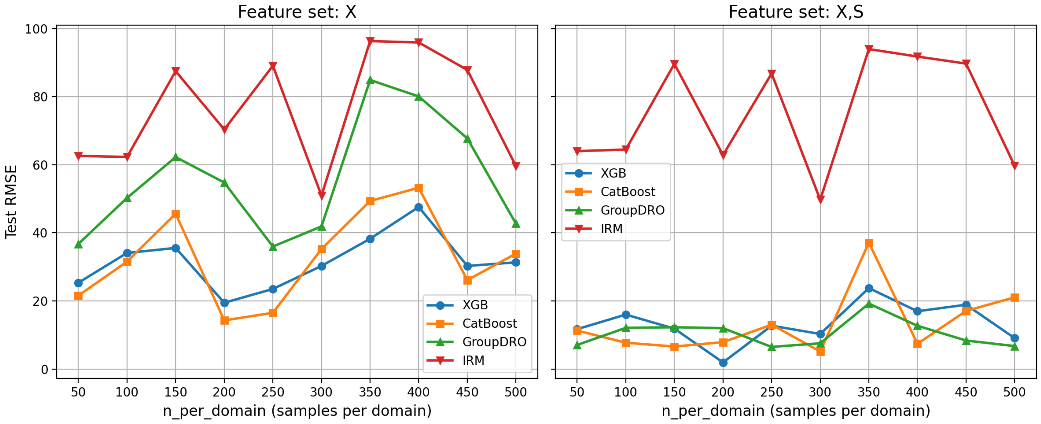

Figure B6: Comparison with OOD generalization methods.