
MODELING THE DATA-GENERATING PROCESS IS NECESSARY FOR OUT-OF-DISTRIBUTION GENERALIZATION

Jivat Neet Kaur
Microsoft Research
t-kaurjivat@microsoft.com

Emre Kıcıman
Microsoft Research
emrek@microsoft.com

Amit Sharma
Microsoft Research
amshar@microsoft.com

ABSTRACT

Real-world data collected from multiple domains can have multiple, distinct distribution shifts over multiple attributes. However, state-of-the-art advances in domain generalization (DG) algorithms focus only on specific shifts over a single attribute. We introduce datasets with *multi*-attribute distribution shifts and find that existing DG algorithms fail to generalize. To explain this, we use causal graphs to characterize the different types of shifts based on the relationship between spurious attributes and the classification label. Each multi-attribute causal graph entails different constraints over observed variables, and therefore any algorithm based on a single, fixed independence constraint cannot work well across all shifts. We present *Causally Adaptive Constraint Minimization (CACM)*, a new algorithm for identifying the correct independence constraints for regularization. Results on fully synthetic, MNIST and small NORB datasets, covering binary and multi-valued attributes and labels, confirm our theoretical claim: correct independence constraints lead to the highest accuracy on unseen domains whereas incorrect constraints fail to do so. Our results demonstrate the importance of modeling the causal relationships inherent in the data-generating process: in many cases, it is impossible to know the correct regularization constraints without this information.

1 Introduction

To perform reliably in real world settings, machine learning models must be robust to distribution shifts – where the training distribution differs from the test distribution. The *domain generalization (DG)* task [1, 2] encapsulates this challenge by evaluating accuracy on an unseen domain given data from multiple domains that share a common optimal predictor. Recent state-of-the-art advances in representation learning for DG [3, 4, 5, 6, 7] focus on a limited setting where the domains exhibit a single kind of distribution shift over one attribute (where an attribute refers to a spurious high-level variable). Using MNIST as an example, domains are created either by adding new values of a spurious attribute like rotation (e.g., Rotated-MNIST dataset [8, 9]) or by changing the correlation between the class label and a spurious attribute like color (e.g., Colored-MNIST [4]), but not both simultaneously. Recent work [10, 11] shows that the accuracy of state-of-the-art DG algorithms are not consistent over these different datasets, indicating the importance of the kind of shift in a dataset.

In real-world data, however, different sources of distribution shift can *co-exist*. Differences across domains may involve multiple attributes with different kinds of shifts. For example, in our Col+Rot-MNIST dataset (see Figure 1), the color and rotation angle of digits can shift independently across data distributions. In satellite imagery [12], the appearance of land cover such as vegetation (trees and grasses) changes seasonally and independently of regional variations in vegetation. To capture such data, we provide a characterization of *multi-attribute* distribution shifts based on the relationship between each attribute and the class label. Using causal graphs and the principle of d-separation, we show that each type of shift leads to a different set of independence constraints on the observed variables. As a consequence,

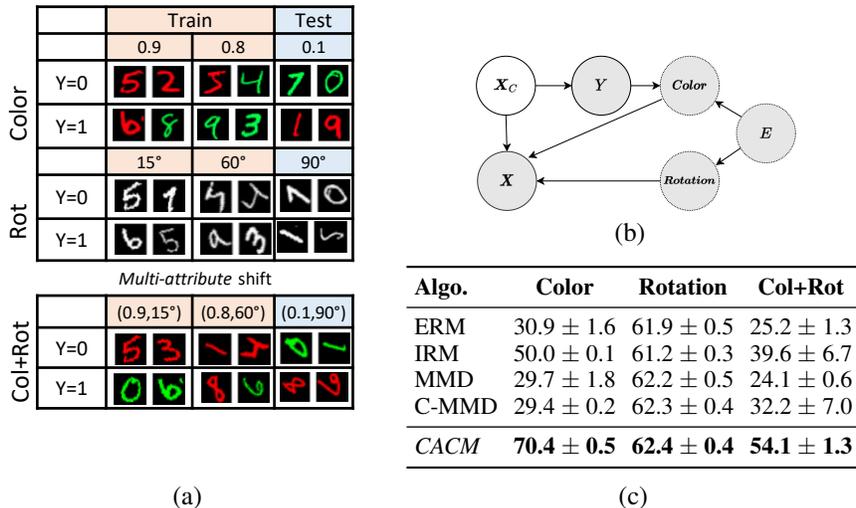


Figure 1: (a) Our *multi-attribute* distribution shift dataset Col+Rot-MNIST. We combine Colored MNIST [4] and Rotated MNIST [8] to introduce distinct shifts over *Color* and *Rotation* attributes. (b) The causal graph representing the data generating process for Col+Rot-MNIST – *Color* has a correlation with Y which changes across environments while *Rotation* varies independently. (c) Comparison with state-of-the-art DG algorithms optimizing for different constraints shows the superiority of our method *Causally Adaptive Constraint Minimization (CACM)* (full table in Section 4).

for datasets like Col+Rot-MNIST, we find that existing DG algorithms that are often targeted for a specific shift fail to generalize.

Beyond existing algorithms, our theoretical analysis shows that any representation learning algorithm based on a single, fixed independence constraint will fail to generalize under multi-attribute shifts. Therefore, we propose to leverage the information provided by multiple independent shifts across attributes, assuming structural knowledge of the shifts. As we discuss in Section 2.2, the type of shift for an attribute is often available or can be inferred for real-world datasets. Then, given a dataset and the canonical causal graph for multi-attribute shifts (Figure 2), our proposed method identifies the correct constraints and applies them as a regularizer in the learning algorithm’s loss function. Note that the type of shift cannot be learned from data even with multiple domains: the same observed training data distribution can be generated irrespective of whether the relationship between attribute and class label is causal, confounded or reflects a selection bias.

To evaluate, we create novel multi-attribute shift datasets based on synthetic data, and MNIST and small NORB images. Across all datasets, applying the incorrect constraint, often through an existing DG algorithm, leads to substantially lower accuracy than the correct constraint. Further, the correct constraint performs achieves better accuracy than existing algorithms. Our contributions include:

- Multi-attribute shifts-based benchmarks for domain generalization where existing algorithms fail.
- Theoretical analysis showing that algorithms using a fixed independence constraint cannot yield an optimal classifier on datasets with multi-attribute shifts.
- A new algorithm, *Causally Adaptive Constraint Minimization (CACM)*, to derive the correct regularization constraint(s) based on the causal graph that outperforms existing DG algorithms.

2 Problem Formulation: Generalization under multi-attribute shifts

We focus on representation learning-based [1] DG algorithms, typically characterized by a regularization term that constrains the standard ERM loss such as cross-entropy (see Table 8 in Suppl.). *Which regularization constraint is the correct one?* This question has attracted much discussion [13, 6, 10, 11, 14] without resolution. Class-conditional independence of learnt representation and domain is a popular constraint but it was recently shown [6] to be inadequate when the distribution of causal or stable features changes across domains. Many papers [15, 13, 14] show that unconditional independence between the representation and domain is an incorrect constraint, yet it performs better than recent methods such as Invariant Risk Minimization [4] on benchmark datasets when the spurious attribute is varied independently of the class label [6, 11]. Moreover, recent empirical work [10, 11] shows that different algorithms perform better under different shifts, but none performs across all shifts. As a result, [10] suggests that instead of a universal algorithm for any shift, adaptable algorithms that use auxiliary attribute information can be more useful. To

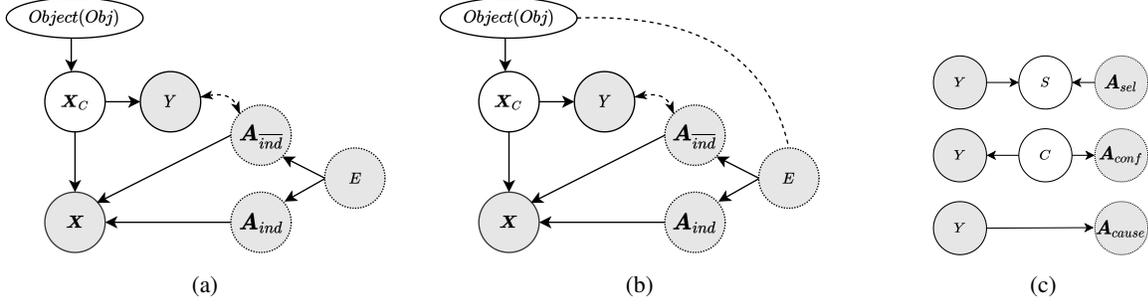


Figure 2: (a), (b) Causal graphs used for specifying *multi-attribute* distribution shifts. Shaded nodes denote observed variables; since not all attributes may be observed, we use dotted boundary. Dashed lines denote correlation. Anti-causal graph shown in Suppl. E. (c) represents different mechanisms which can introduce $Y - \mathbf{A}_{ind}^{-}$ relationship leading to *Causal*, *Confounded* and *Selected* shifts (bottom to top).

explore this question further, while the above efforts considered domains with a single distribution shift, we consider the generalization problem over a more realistic setup where each domain can have multiple shifts over different attributes.

2.1 Risk-invariant predictor over a set of distributions

We consider the supervised learning setup from [10] where each row of train data $(\mathbf{x}_i, \mathbf{a}_i, y_i)_{i=1}^n$ contains input features \mathbf{x}_i (e.g., X-ray pixels), a set of nuisance or spurious attributes \mathbf{a}_i (e.g., vertical shift, hospital) and class label y_i (e.g., disease diagnosis). The attributes represent variables that are often recorded during data collection or can be inferred. Some attributes represent a property of the input (e.g., vertical shift) while others represent the domain from which the input was collected (e.g., hospital). Attributes and class labels are assumed to be discrete.

We assume that the attributes have a correlation with the label that can change between train and test data. Since the nuisance attribute’s distribution or its correlation with the label can change, we obtain different data distributions. Given a set of domains sampled from \mathcal{P} , the train data is sampled from domains, $\mathcal{P}_{Etr} = \{P_{E1}, P_{E2}, \dots\} \subset \mathcal{P}$ while the test data is assumed to be sampled from a single unseen domain, $\mathcal{P}_{Ete} = \{P_{Ete}\} \subset \mathcal{P}$. The goal is to learn a classifier $g(\mathbf{x})$ using train domains such that it generalizes and achieves a similar, small risk on test data from P_{Ete} as it achieves on the train data. Formally, given a set of distributions \mathcal{P} , we define a risk-invariant predictor [16] as,

Definition 2.1. Optimal Risk Invariant Predictor for \mathcal{P} (from [16]) Define the risk of predictor g on distribution $P \in \mathcal{P}$ as $R_P(g) = \mathbb{E}_{\mathbf{x}, y \sim P} \ell(g(\mathbf{x}), y)$ where ℓ is cross-entropy or another classification loss. Then, the set of risk-invariant predictors obtain the same risk across all distributions $P \in \mathcal{P}$, and set of the optimal risk-invariant predictors is defined as the risk-invariant predictors that obtain minimum risk on all distributions.

$$g_{rinv} \in \arg \min_{g \in G_{rinv}} R_P(g) \forall P \in \mathcal{P} \text{ where } G_{rinv} = \{g : R_P(g) = R_{P'}(g) \forall P, P' \in \mathcal{P}\} \quad (1)$$

2.2 Using causal graphs to specify the set of distributions under multi-attribute shifts

To specify the set of distributions \mathcal{P} to generalize over, using causal graphs, we characterize the different data-generating processes that can lead to a multi-attribute shift dataset. Figure 2 shows the corresponding causal directed acyclic graph (DAG). Shaded nodes represent observed variables \mathbf{X} , Y ; and the sets of attributes \mathbf{A}_{ind}^{-} , \mathbf{A}_{ind} , and E such that $\mathbf{A}_{ind}^{-} \cup \mathbf{A}_{ind} \cup \{E\} = \mathbf{A}$. \mathbf{A}_{ind}^{-} represents the attributes correlated with label, \mathbf{A}_{ind} the attributes that are independent of label, while E is a special attribute for the domain. Not all attributes need to be observed. For example, in some cases, only E and a subset of \mathbf{A}_{ind}^{-} , \mathbf{A}_{ind} may be observed. In other cases, only \mathbf{A}_{ind}^{-} and \mathbf{A}_{ind} may be observed while E is not available. Regardless, all attributes, along with the stable/causal features \mathbf{X}_c , determine the observed features \mathbf{X} . And the stable features, \mathbf{X}_c are the only features that cause Y . In the simplest case, we assume no label shift across environments i.e. marginal distribution of Y is constant across train domains and test, $P_{Etr}(y) = P_{Ete}(y)$ (see Figure 2a). More generally, different domains may have different distribution of objects and hence there may be a correlation between E and Obj , as represented by the right subfigure (Figure 2b).

We characterize different kinds of shifts based on the relationship between nuisance attributes \mathbf{A} and the classification label Y . Specifically, \mathbf{A}_{ind} has varying distribution across environments but is *Independent* of the class label. The dashed bidirectional arrow represents the correlation between \mathbf{A}_{ind}^{-} and Y . There are different mechanisms which can introduce the dashed-line relationship (Figure 2c) – direct-causal relationship (Y causing \mathbf{A}_{ind}^{-}), confounding between Y and \mathbf{A}_{ind}^{-} due to a common cause, or selection during the data-generating process. Thus, we define four kinds of shifts based on the causal graph: *Independent*, *Causal*, *Confounded*, and *Selected*. As we shall see, these shifts

correspond to different independence constraints between observed variables. Thus, in addition to the dataset, to fully specify the problem of multi-attribute shift generalization for a learning algorithm, we require knowledge of the kind of shift for each observed attribute.

Definition 2.2. Generalization under Multi-attribute shifts. Given training data $(\mathbf{x}_i, \mathbf{a}_i, y_i)_{i=1}^n$ and the type of causal relationship of each attribute A with the label Y , construct a realized causal graph \mathcal{G} based on the canonical graph in Figure 2 and define $\mathcal{P}_{\mathcal{G}}$ as the set of all distributions obtained by changing the relationship between Y and each attribute while keeping the same graph (type of shift). The generalization goal is to learn an optimal risk-invariant predictor over $\mathcal{P}_{\mathcal{G}}$.

Availability of multiple attributes. Unlike the full causal graph, the type of relationship between label and an attribute is often known. For example, in text toxicity classification, toxicity labels are found to be spuriously correlated with certain demographics ($\mathbf{A}_{\overline{ind}}$) [17, 12, 18]; while in medical applications where data is collected from small number of hospitals, shifts arise due to different methods of slide staining and image acquisition (\mathbf{A}_{ind}) [12, 19, 20]. Suppl. A contains real-world examples where these relationships as well as attribute values are known. Note that while it can be learned from data whether an attribute belongs to $\mathbf{A}_{\overline{ind}}$ or \mathbf{A}_{ind} (since $\mathbf{A}_{ind} \perp\!\!\!\perp Y$), it is not possible to differentiate between the \mathbf{A}_{cause} , \mathbf{A}_{conf} and \mathbf{A}_{sel} using observed data.

3 Identifying the correct regularizer for ERM under multi-attribute shift

3.1 Deriving conditional independence constraints for a risk-invariant representation

We assume that the predictor can be represented as $g(\mathbf{x}) = g_1(\phi(\mathbf{x}))$ where ϕ is the representation. To derive the constraints that should be satisfied by a risk-invariant $g_1(\phi)$, we utilize a strategy from past work [6, 21]. Specifically, we identify the conditional independence constraints satisfied by \mathbf{X}_c in the causal graph and enforce that learnt representation ϕ should follow the same constraint. If ϕ satisfies the constraints, then any function $g_1(\phi)$ will also satisfy them. Below we justify the strategy theoretically showing that the constraints are necessary.

Proposition 3.1. Given a dataset $(\mathbf{x}_i, \mathbf{a}_i, y_i)_{i=1}^n$ and a causal DAG \mathcal{G} over $\langle \mathbf{X}_c, \mathbf{X}, \mathbf{A}, Y \rangle$ such that \mathbf{X}_c is the only variable (or set of variables) that causes Y and is not independent of \mathbf{X} , then the conditional independence constraints satisfied by \mathbf{X}_c are necessary for a risk-invariant predictor over $\mathcal{P}_{\mathcal{G}}$. That is, if a predictor does not satisfy any of these constraints, then there exists a data distribution $P' \in \mathcal{P}_{\mathcal{G}}$ such that predictor’s risk will be higher than its risk in other distributions.

Using the strategy and applying d-separation on \mathbf{X}_c and observed variables, let us examine two common constraints on independence between ϕ and a nuisance attribute: either unconditional [22, 23, 24] or conditional on the label Y [25, 26, 27, 28] (see Suppl. for details on these baseline methods). Under the canonical graph from Figure 2b, none of these constraints are valid because there could be a correlation path between \mathbf{X}_c and E (under the X-ray example, this can be because more women visit one hospital compared to the other). When we simplify the graph by removing the correlation between object and E (Figure 2a), the unconditional constraint is true when $A \perp\!\!\!\perp Y$ ($A \in \mathbf{A}_{ind}$) but not always for $\mathbf{A}_{\overline{ind}}$. For any attribute $A \in \mathbf{A}_{\overline{ind}}$, if the relationship between Y and A is *Confounded*, then the unconditional constraint is correct. If the relationship is *Causal* or *Selected*, then the conditional constraint is correct. Critically, as [21] shows for a single-attribute graph, the conditional constraint is not always better under Graph 2a; it is an incorrect constraint (not satisfied by \mathbf{X}_c) under *Confounded* setting.

In general, for multi-attribute shifts, the constraints need not be restricted to the simple ones above. Given the graphs from Figure 2, or any other graph representing a dataset, we propose the following algorithm. Let \mathcal{V} be the set of observed variables in the graph except Y .

1. For every observed variable $V \in \mathcal{V}$ in the graph, check whether (\mathbf{X}_c, V) are d-separated.
2. If not, check whether (\mathbf{X}_c, V) are d-separated conditioned on any subset of the remaining observed variables in $\mathcal{V} \setminus \{V\}$.

Since the graphs are expected to be contain a few variables, the above brute-force algorithm is appropriate as it will give a complete list of constraints. Below, we apply the algorithm to Figure 2 using $\mathcal{V} = \mathbf{A}$, the set of observed attributes.

Theorem 3.1. Given a causal DAG with the structure as shown in Figure 2a, the correct constraint depends on the relationship of label Y with the nuisance attributes \mathbf{A} . As shown, \mathbf{A} can be split into $\mathbf{A}_{\overline{ind}}$, \mathbf{A}_{ind} and E , where $\mathbf{A}_{\overline{ind}}$ can be further split into subsets that have a causal (\mathbf{A}_{cause}), confounded (\mathbf{A}_{conf}), selected (\mathbf{A}_{sel}) relationship with Y ($\mathbf{A}_{\overline{ind}} = \mathbf{A}_{cause} \cup \mathbf{A}_{conf} \cup \mathbf{A}_{sel}$). Then, the (conditional) independence constraints that \mathbf{X}_c should satisfy are,

1. Independent: $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind}$; $\mathbf{X}_c \perp\!\!\!\perp E$; $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y$; $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | E$; $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y, E$

2. *Causal*: $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y, E$
3. *Confounded*: $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{conf}; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{conf} | E$
4. *Selected*: $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{sel} | Y; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{sel} | Y, E$

Corollary 3.1.1. *All the above derived constraints are valid for Graph 2a. However, in the presence of a correlation between E and Obj (Graph 2b), only the constraints conditioned on E hold true.*

Hence, if information on Obj - E correlation is not available, it is advisable to use E -conditioned constraints. While we list all constraints, if one of the attributes is unobserved (say $\mathbf{A}_{\overline{ind}}$ or \mathbf{A}_{ind} is not available), then we use the subset of constraints derived for the observed features. By considering independence constraints over *attributes* that may represent any observed variable, our graph-based characterization unites the single-domain (group-wise) and multi-domain generalization tasks. Attributes may represent group indicators from a single dataset or domain indicators denoting different data sources; in both cases, the same algorithm provides the correct regularization constraint.

3.2 An algorithm for generalizing under multi-attribution shifts

We now describe the proposed *CACM* algorithm. Given a causal graph, it first utilizes the steps highlighted above to identify the correct independence constraints. Then it applies those constraints as a regularizer to the standard ERM loss, $g_1, \phi = \arg \min_{g_1, \phi} \ell(g_1(\phi(\mathbf{x})), y) + \lambda^*(RegPenalty)$, where λ is a hyperparameter and ℓ is cross-entropy loss. We design the regularizer such that it optimizes for valid constraints over all observed variables $V \in \mathcal{V}$ ($\mathcal{V} = \mathbf{A}$ for our causal graph in Figure 2). If there is a choice between multiple constraints, we choose the constraint that will be valid over both Figure 2a and 2b.

Since \mathbf{A} includes multiple attributes, the regularizer penalty depends on the type of distribution shift for each attribute. For instance, for $A \in \mathbf{A}_{ind}$ (*Independent*), to enforce $\phi(\mathbf{x}) \perp\!\!\!\perp A$, we aim to minimize the distributional discrepancy between $P(g(\mathbf{x})|A = a_i)$ and $P(g(\mathbf{x})|A = a_j)$, for all i, j values of A . However, the same constraint is applicable on E . So if domain variable E is available, it is statistically efficient to apply the constraint on E since there would typically be multiple closely related values of A in a domain (e.g., slide stains collected from one hospital may be spread over similar colors, but not exactly the same). Hence, we apply the constraint on distributions $P(g_1(\phi(\mathbf{x}))|E = E_i)$ and $P(g_1(\phi(\mathbf{x}))|E = E_j)$ if E is observed (and A may/may not be unobserved), otherwise we apply the constraint over A .

$$RegPenalty_{\mathbf{A}_{ind}} = \sum_{i=1}^{|E|} \sum_{j>i} MMD(P(g_1(\phi(\mathbf{x}))|E = E_i), P(g_1(\phi(\mathbf{x}))|E = E_j)) \quad (2)$$

For $A \in \mathbf{A}_{cause}$ (*Causal*), following Theorem 3.1, we consider distributions $P(g_1(\phi(\mathbf{x}))|A = a_i, Y = y)$ and $P(g_1(\phi(\mathbf{x}))|A = a_j, Y = y)$. We additionally condition on domain E as there may be a correlation between E and Obj (Figure 2b), which renders other constraints incorrect (Corollary 3.1.1). Regularization terms for *Confounded* and *Selected* are obtained in a similar way (see Suppl. C.3 for the full algorithm).

$$RegPenalty_{\mathbf{A}_{cause}} = \sum_{|E|} \sum_{y \in Y} \sum_{i=1}^{|\mathbf{A}_{cause}|} \sum_{j>i} MMD(P(g_1(\phi(\mathbf{x}))|a_{i,cause}, y), P(g_1(\phi(\mathbf{x}))|a_{j,cause}, y)) \quad (3)$$

The final *RegPenalty* is a sum of penalties over all attributes, $RegPenalty = \sum_{A \in \mathbf{A}} Penalty_A$. While we presented the algorithm for the canonical graph in Figure 2, *CACM* algorithm can provide the correct constraints for any graph. We choose the Maximum Mean Discrepancy (MMD) [29] metric to implement our penalty (although, in principle, any estimable metric for enforcing conditional independence would work). Unlike prior work [16, 21], we do not restrict ourselves to binary-valued attributes and classes.

That said, we want to emphasize that the constraints from the *CACM* algorithm are necessary but not sufficient. While regularizers like *CACM* restrict the set of possible solutions to a smaller subset that contains \mathbf{X}_c [6], they are not guaranteed to return \mathbf{X}_c . Formally, \mathbf{X}_c is not identified under the current graph. In practical settings, one of the nuisance attributes may be unobserved and thus the resultant constraints would not be able to remove its influence.

3.3 A fixed conditional independence constraint cannot work for all datasets

Since the observed data distribution can be identical for all three types of relationship between Y and $\mathbf{A}_{\overline{ind}}$, the type of relationship cannot be learned from observed data. Since the constraints are different for different relationship types, it implies that any algorithm relying on a single (conditional) independence constraint [29, 4, 30, 7] cannot work for all datasets.

Theorem 3.2. *Under the canonical causal graph in Figure 2, there exists no (conditional) independence constraint such that it is valid for all realizations of the graph as the type of multi-attribute shifts vary. Thus, for any predictor algorithm for Y that uses a single type of (conditional) independence constraint, there exists a realized graph \mathcal{G} and a corresponding training dataset such that the learned predictor cannot be a risk-invariant predictor across distributions in $\mathcal{P}_{\mathcal{G}}$.*

4 Empirical Evaluation

We perform experiments on fully-synthetic, semi-synthetic (MNIST) and natural (small NORB) datasets to demonstrate our main claims: existing DG algorithms perform worse on multi-attribute shifts, *CACM* with the correct graph-based constraints significantly outperforms these algorithms, and incorrect constraints cannot match the above accuracy. While we provided constraints for all shifts in Theorem 3.1 for completeness, our experiments focus on commonly occurring *Causal* and *Independent* shifts. All experiments are performed in PyTorch 1.10 with NVIDIA Tesla P40 and P100 GPUs. We build upon the code from DomainBed [31] and OoD-Bench [11]. Regularizing on $g_1(\phi(\mathbf{x}))$ provided better accuracy than $\phi(\mathbf{x})$; hence we adopt it for all our experiments.

4.1 Datasets

We introduce three new datasets for the multi-attribute shift problem. For all datasets, details of environments, architectures, visualizations, causal graphs, and setup generation are in Suppl. C.2.

Synthetic. Our synthetic dataset is based on the slab dataset [6, 32]. Causal feature X_c has a non-linear “slab” relationship with Y while A_{ind} has a linear, *Causal* relationship with Y . With probability p , $A_{cause} = y$ and with probability $1 - p$, $A_{cause} = abs(y - 1)$. Following [6], the two training domains have p as 0.9 and 1.0, and the test domain has $p = 0.0$.

MNIST. Colored and Rotated [8] MNIST [4] present A_{cause} and A_{ind} distribution shifts, respectively. We combine these to obtain a multi-attribute dataset with $A_{cause} = \{C\}$ and $A_{ind} = \{R\}$.

small NORB. We use small NORB [33, 10], an object recognition dataset, to create a challenging task with multi-valued classes and attributes over realistic 3D objects with varying lighting and azimuths. We create multi-attribution shifts, wherein there is a correlation between lighting condition $A_{cause} = \{l\}$ and object category y ; and $A_{ind} = \{azi\}$ that varies independently across domains.

To compare the effect of shifts over two attributes, for each dataset, we also create single-attribute shift datasets involving a change in only one of the attributes. Thus, we have three evaluation setups for each dataset: A_{cause} , A_{ind} and $A_{cause} \cup A_{ind}$.

4.2 Baseline DG algorithms & implementation

We consider baseline algorithms optimizing for different constraints and statistics (see Suppl.) to compare to causal adaptive regularization: IRM [4], VREx [5], MMD [30], CORAL [7], DANN [29], Conditional-MMD (C-MMD) [30], and conditional-DANN (CDANN) [28]. Following DomainBed [31], a random search is performed 20 times over the hyperparameter distribution for 3 seeds. The best models obtained across the three seeds are used to compute the mean and standard error. We use a validation set that follows the test domain distribution consistent with previous work on these datasets [4, 10, 11]. Refer to Suppl. C for further experimental details.

4.3 Results

Correct constraint derived from the causal graph matters. Table 2 shows the accuracy on test domain for the MNIST dataset. Comparing the three prediction tasks, for all algorithms, accuracy on unseen test domain is highest under A_{ind} shift and lowest under two-attribute shift ($A_{ind} \cup A_{cause}$), reflecting the difficulty of a distribution shift over multiple attributes. On the two-attribute shift task, all DG algorithms obtain less than 40% accuracy whereas *CACM* obtains a 14.5% absolute improvement. Results on the Synthetic (Table 1) and small NORB (Table 3) datasets are similar. In particular, on small-NORB, *CACM* obtains 69.6% accuracy on the two-attribute task while the nearest baseline is ERM at 64%.

Across all datasets, *CACM* also obtains highest accuracy on the A_{cause} task. On MNIST, even though IRM and VREx are designed for the Color-only (A_{cause}) task, under an extensive hyperparameter sweep as recommended in

Table 1: Synthetic dataset. Accuracy on unseen domain for single-attribute (\mathcal{A}_{cause} , \mathcal{A}_{ind}), and multi-attribute ($\mathcal{A}_{cause} \cup \mathcal{A}_{ind}$) distribution shifts.

Algo.	Accuracy		
	\mathcal{A}_{cause}	\mathcal{A}_{ind}	$\mathcal{A}_{cause} \cup \mathcal{A}_{ind}$
ERM	32.2 ± 2.9	86.3 ± 0.7	26.4 ± 1.3
IRM	68.4 ± 3.4	84.7 ± 1.0	51.0 ± 3.9
VREx	66.0 ± 2.2	84.1 ± 1.4	62.4 ± 5.6
MMD	23.3 ± 1.7	86.0 ± 1.0	23.8 ± 2.1
CORAL	28.6 ± 3.0	87.6 ± 0.4	21.7 ± 1.1
DANN	44.6 ± 3.6	84.0 ± 0.6	46.4 ± 4.3
C-MMD	36.7 ± 4.1	85.3 ± 1.3	27.6 ± 1.8
CDANN	40.0 ± 7.2	84.9 ± 1.1	40.5 ± 2.1
<i>CACM</i>	94.1 ± 0.5	86.4 ± 0.7	84.3 ± 3.5

Table 2: Colored + Rotated MNIST. Accuracy on unseen domain for single-attribute (color, rotation) and multi-attribute (col+rot) distribution shifts.

Algo.	Accuracy		
	color	rotation	col+rot
ERM	30.9 ± 1.6	61.9 ± 0.5	25.2 ± 1.3
IRM	50.0 ± 0.1	61.2 ± 0.3	39.6 ± 6.7
VREx	30.3 ± 1.6	62.1 ± 0.4	23.3 ± 0.4
MMD	29.7 ± 1.8	62.2 ± 0.5	24.1 ± 0.6
CORAL	28.5 ± 0.8	62.5 ± 0.7	23.5 ± 1.1
DANN	20.7 ± 0.8	61.9 ± 0.7	32.0 ± 7.8
C-MMD	29.4 ± 0.2	62.3 ± 0.4	32.2 ± 7.0
CDANN	30.8 ± 8.0	61.8 ± 0.2	32.2 ± 7.0
<i>CACM</i>	70.4 ± 0.5	62.4 ± 0.4	54.1 ± 1.3

Table 3: small NORB. Accuracy on unseen domain for single- and multi-attribute shifts.

Algo.	Accuracy		
	lighting (\mathcal{A}_{cause})	azimuth (\mathcal{A}_{ind})	lighting+azimuth ($\mathcal{A}_{cause} \cup \mathcal{A}_{ind}$)
ERM	65.5 ± 0.7	78.6 ± 0.7	64.0 ± 1.2
IRM	66.7 ± 1.5	75.7 ± 0.4	61.7 ± 0.5
VREx	64.7 ± 1.0	77.6 ± 0.5	62.5 ± 1.6
MMD	66.6 ± 1.6	76.7 ± 1.1	62.5 ± 0.3
CORAL	64.7 ± 0.5	77.2 ± 0.7	62.9 ± 0.3
DANN	64.6 ± 1.4	78.6 ± 0.7	60.8 ± 0.7
C-MMD	65.8 ± 0.8	76.9 ± 1.0	61.0 ± 0.9
CDANN	64.9 ± 0.5	77.3 ± 0.3	60.8 ± 0.9
<i>CACM</i>	85.4 ± 0.5	80.5 ± 0.6	69.6 ± 1.6

past work [31, 5, 11], we find that *CACM* achieves a substantially higher accuracy (70%) than these methods, just 5 units lower than the optimal 75%. While the \mathcal{A}_{ind} task is relatively easier, algorithms optimizing for the correct constraint achieve highest accuracy. Note that MMD, CORAL, DANN, and *CACM* are based on the same independence constraint (see Table 8 in Suppl.). For full comparability, we present results where *CACM* uses the domain attribute E like the other algorithms since the constraint is equally valid on E or \mathcal{A}_{ind} (see Section 3.2). These results indicate the importance of regularization based on data-specific correct constraints for generalization.

Incorrect constraints hurt generalization. We now directly compare the effect of using correct versus *incorrect* (but commonly used) constraints for a dataset. To isolate the effect of a single constraint, we consider the single-attribute shift on \mathcal{A}_{cause} and compare the application of different regularizer constraints. Theorem 3.1 provides the correct constraint for \mathcal{A}_{cause} : $\mathbf{X}_c \perp\!\!\!\perp \mathcal{A}_{cause} \mid Y, E$. In addition, using d-separation on Figure 2, we see the following invalid constraints, $\mathbf{X}_c \perp\!\!\!\perp \mathcal{A}_{cause} \mid E$, $\mathbf{X}_c \perp\!\!\!\perp \mathcal{A}_{cause}$. Without knowing that the shift corresponds to a *Causal* shift, one may apply these constraints that do not condition on the class. Results on both Synthetic (Table 4) and small NORB (Table 6) datasets show that using the incorrect constraint has an adverse effect on model performance. For the Synthetic data, accuracy using the incorrect unconditional constraint (29.7%) is lower than ERM (32.2%, Table 1). On small NORB, the correct constraint yields 85% accuracy while the best incorrect constraint achieves 79.7%. Moreover, application of the incorrect constraint is sensitive to the λ (regularization weight) parameter (Figure 3): as λ increases, accuracy drops to less than 40%. However, accuracy with the correct constraint stays invariant across different values of λ .

Comparing small NORB and MNIST (Table 5) reveals the importance of making the right structural assumptions. Typically, DG algorithms assume that distribution of causal features \mathbf{X}_c does not change across domains. Then, both $\mathbf{X}_c \perp\!\!\!\perp \mathcal{A}_{cause} \mid Y, E$ and $\mathbf{X}_c \perp\!\!\!\perp \mathcal{A}_{cause} \mid Y$ should be correct constraints. However, conditioning on both Y and E provides a 5% point gain over conditioning on Y in NORB while the accuracy is comparable for MNIST. Auxiliary information about the data-generating process explains the result: Different domains in MNIST include samples from the same distribution whereas small NORB domains are sampled from a different set of toy objects, thus creating a correlation between Obj and E . Without such auxiliary information, such gains will be difficult.

Table 4: Synthetic dataset. Comparison of $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y, E$ (correct) and $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | E$ (incorrect) constraints for *Causal* shift.

Constraint	Accuracy
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} E$	29.7 ± 3.8
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} Y, E$	94.1 ± 0.5

Table 6: small NORB *Causal* shift. Comparing $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y, E$ with possible incorrect constraints.

Constraint	Accuracy
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause}$	72.7 ± 1.1
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} E$	76.2 ± 0.9
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} Y$	79.7 ± 0.9
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} Y, E$	85.4 ± 0.5

Table 5: Comparing constraints $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y, E$ and $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y$ for *Causal* shift in MNIST and small NORB. We show that the constraint derived in the presence of *Obj - E* correlation (Fig. 2b, Theorem 3.1) affects accuracy.

Constraint	MNIST Acc.	small NORB Acc.
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} Y$	69.7 ± 0.2	79.7 ± 0.9
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} Y, E$	70.4 ± 0.5	85.4 ± 0.5

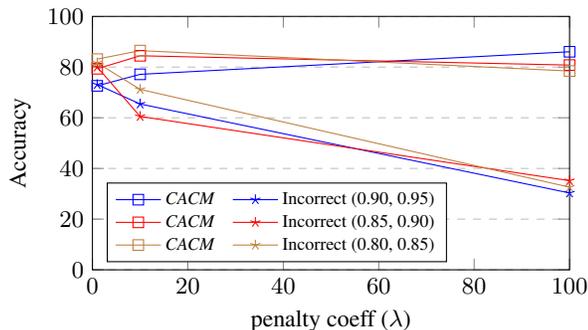


Figure 3: Accuracy of CACM and incorrect constraint on small NORB *Causal* shift with varying $\lambda \{1, 10, 100\}$ and spurious correlation in train envs (in parantheses in legend).

Finally, we replicate the above experiment for the multi-attribute shift setting for small NORB. In addition to applying the correct CACM constraints, we consider a case where we interchange the variables before inputting to CACM algorithm (\mathbf{A}_{ind} gets used as \mathbf{A}_{cause} and vice-versa) and then apply the resultant (incorrect) constraints. Accuracy with interchanged variables (65.1 ± 1.6) is lower than that of correct CACM (69.6 ± 1.6). More ablations are in Suppl. D.

5 Related Work

Improving the robustness of models in the face of distribution shifts is a key challenge. Several works have attempted to tackle the domain generalization problem [1, 2] using different approaches – data augmentation [34, 35, 36], and representation learning [4, 37, 38] being popular ones. Trying to gauge the progress made by these approaches, Gulrajani and Lopez-Paz [31] find that existing state-of-the-art DG algorithms do not improve over ERM. More recent work [10, 11] empirically shows that different algorithms perform well over different distribution shifts, but no single algorithm performs consistently across all. While they evaluate on single-attribute shift datasets, [10] discuss the importance of having auxiliary knowledge of and evaluating methods under different underlying shifts. To this end, we provide (1) multi-attribute shift benchmark datasets; (2) a causal interpretation of different kinds of shifts; and (3) an adaptive algorithm to identify the correct regularizer.

Causally-motivated learning. There has been recent work focused on *causal representation learning* [4, 5, 39, 40] for OoD generalization. While these works attempt to learn the constraints for causal features from input features, we show that it is necessary to model the data-generating process and have access to auxiliary attributes to obtain a risk-invariant predictor, especially in *multi-attribute* distribution shift setups. Recent research has shown how causal graphs can be used to characterize and analyze the different kinds of distribution shifts that occur in real-world settings [16, 21]. Our approach is similar in motivation but we extend from single-domain, single-attribute setups in past work to formally introduce *multi-attribute* distribution shifts in more complex and real-world settings. Additionally, we do not restrict ourselves to binary-valued classes and attributes.

6 Discussion

We introduced CACM, an adaptive OoD generalization algorithm to characterize *multi-attribute* shifts and apply the correct independence constraints. Through empirical experiments and theoretical analysis, we show the importance of modeling the causal relationships in the data-generating process. The main limitation is that CACM does not address data sparsity – applying the constraints might be statistically inefficient if an attribute value is undersampled compared

to others. Future work includes statistical improvements in the regularization penalty (e.g., multiple regularization coefficients λ).

References

- [1] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Wenjun Zeng, and Tao Qin. Generalizing to unseen domains: A survey on domain generalization. *arXiv preprint arXiv:2103.03097*, 2021.
- [2] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *arXiv e-prints*, pages arXiv–2103, 2021.
- [3] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018.
- [4] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2019.
- [5] 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 Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5815–5826. PMLR, 18–24 Jul 2021.
- [6] Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 7313–7324. PMLR, 18–24 Jul 2021.
- [7] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, pages 443–450, Cham, 2016. Springer International Publishing.
- [8] M. Ghifary, W. Kleijn, M. Zhang, and D. Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2551–2559, Los Alamitos, CA, USA, dec 2015. IEEE Computer Society.
- [9] Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. Efficient domain generalization via common-specific low-rank decomposition. *Proceedings of the International Conference of Machine Learning (ICML) 2020*, 2020.
- [10] Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre-Alvise Rebuffi, Ira Ktena, Krishnamurthy Dj Dvijotham, and Ali Taylan Cemgil. A fine-grained analysis on distribution shift. In *International Conference on Learning Representations*, 2022.
- [11] Nanyang Ye, Kaican Li, Haoyue Bai, Runpeng Yu, Lanqing Hong, Fengwei Zhou, Zhenguo Li, and Jun Zhu. Ood-bench: Quantifying and understanding two dimensions of out-of-distribution generalization. In *CVPR*, 2022.
- [12] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard L. Phillips, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts. In *ICML*, 2021.
- [13] Fredrik D Johansson, David Sontag, and Rajesh Ranganath. Support and invertibility in domain-invariant representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 527–536, 2019.
- [14] Han Zhao, Remi Tachet des Combes, Kun Zhang, and Geoffrey J Gordon. On learning invariant representation for domain adaptation. *arXiv preprint arXiv:1901.09453*, 2019.
- [15] Kei Akuzawa, Yusuke Iwasawa, and Yutaka Matsuo. Adversarial invariant feature learning with accuracy constraint for domain generalization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 315–331. Springer, 2019.
- [16] Maggie Makar, Ben Packer, Dan Moldovan, Davis Blalock, Yoni Halpern, and Alexander D’Amour. Causally motivated shortcut removal using auxiliary labels. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 739–766. PMLR, 28–30 Mar 2022.
- [17] Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES ’18, page 67–73, New York, NY, USA, 2018. Association for Computing Machinery.

- [18] Ji Ho Park, Jamin Shin, and Pascale Fung. Reducing gender bias in abusive language detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2799–2804, Brussels, Belgium, October–November 2018. Association for Computational Linguistics.
- [19] Daisuke Komura and Shumpei Ishikawa. Machine learning methods for histopathological image analysis. *Computational and Structural Biotechnology Journal*, 16:34–42, 2018.
- [20] David Tellez, Geert Litjens, Péter Bándi, Wouter Bulten, John-Melle Bokhorst, Francesco Ciompi, and Jeroen van der Laak. Quantifying the effects of data augmentation and stain color normalization in convolutional neural networks for computational pathology. *Medical Image Analysis*, 58:101544, 2019.
- [21] Victor Veitch, Alexander D’Amour, Steve Yadlowsky, and Jacob Eisenstein. Counterfactual invariance to spurious correlations in text classification. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.
- [22] Isabela Albuquerque, João Monteiro, Mohammad Darvishi, Tiago H. Falk, and Ioannis Mitliagkas. Generalizing to unseen domains via distribution matching, 2020.
- [23] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.
- [24] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18, 2013.
- [25] Muhammad Ghifary, David Balduzzi, W Bastiaan Kleijn, and Mengjie Zhang. Scatter component analysis: A unified framework for domain adaptation and domain generalization. *IEEE transactions on pattern analysis and machine intelligence*, 39(7):1414–1430, 2016.
- [26] Shoubo Hu, Kun Zhang, Zhitang Chen, and Laiwan Chan. Domain generalization via multidomain discriminant analysis. In *Uncertainty in artificial intelligence: proceedings of the... conference. Conference on Uncertainty in Artificial Intelligence*, volume 35. NIH Public Access, 2019.
- [27] Ya Li, Mingming Gong, Xinmei Tian, Tongliang Liu, and Dacheng Tao. Domain generalization via conditional invariant representations. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [28] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639, 2018.
- [29] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012.
- [30] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C. Kot. Domain generalization with adversarial feature learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2018.
- [31] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *ArXiv*, abs/2007.01434, 2021.
- [32] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.
- [33] Y. LeCun, Fu Jie Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–104 Vol.2, 2004.
- [34] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18613–18624. Curran Associates, Inc., 2020.
- [35] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
- [36] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. pages 2242–2251, 10 2017.
- [37] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.
- [38] Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

- [39] Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. ICML’20. JMLR.org, 2020.
- [40] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.
- [41] Shiori Sagawa*, Pang Wei Koh*, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020.
- [42] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016.

A Presence of auxiliary attribute information in datasets

Unlike the full causal graph, attribute values as well as the relationships between class labels and attributes is often known. CACM assumes access to attribute labels \mathbf{A} only during training time, which are collected as part of the data collection process (e.g., as metadata with training data [16]). We start by discussing the availability of attributes in WILDS [12], a set of real-world datasets adapted for the domain generalization setting. Attribute labels available in the datasets include, the *time (year)* and *region* associated with satellite images in FMoW dataset (Christie et al. 2018) for predicting land use category, *hospital* from where the tissue patch was collected for tumor detection in Camelyon17 dataset (Bandi et al., 2018) and the *demographic* information for CivilComments dataset (Borkan et al., 2019). [12] create different domains in WILDS using this metadata, consistent with our definition of $E \in \mathbf{A}$ as a special domain attribute.

In addition, CACM requires the type of relationship between label Y and attributes. This is often known, either based on how the dataset was collected or inferred based on domain knowledge or observation. While the distinction between \mathbf{A}_{ind} and $\mathbf{A}_{\overline{ind}}$ can be established using a statistical test of independence on a given dataset, the distinction between \mathbf{A}_{cause} , \mathbf{A}_{sel} and \mathbf{A}_{conf} within $\mathbf{A}_{\overline{ind}}$ must be provided by the user. In the above datasets, for FMoW, *time* can be considered an *Independent* attribute (\mathbf{A}_{ind}) since it reflects the time at which images are captured which is not correlated with Y ; whereas *region* is a *Confounded* attribute since certain regions associated with certain Y labels are over-represented due to ease of data collection. Note that region cannot lead to *Causal* shift since the decision to take images in a region was not determined by the final label nor *Selected* for the same reason that the decision was not taken based on values of Y . Similarly, for the Camelyon17 dataset, it is known that differences in slide staining or image acquisition leads to variation in tissue slides across *hospitals*, thus implying that *hospital* is an *Independent* attribute (\mathbf{A}_{ind}) [12, 19, 20]; As another example from healthcare, a study in MIT Technology Review¹ discusses biased data where a person’s *position* (\mathbf{A}_{conf}) was spuriously correlated with disease prediction as patients lying down were more likely to be ill. As another example, [41] adapt MultiNLI dataset for OoD generalization due to the presence of spurious correlation between *negation words* (attribute) and the contradiction label (*Causal* shift), however this relationship between negation words and label may not always hold. Finally, for the CivilComments dataset, we expect the *demographic* features to be *Confounded* attributes as there could be biases which result in spurious correlation between comment toxicity and demographic information.

To provide examples showing the availability of attributes and their type of relationship with the label, Table 7 lists some popular datasets used for DG and the associated auxiliary information present as metadata. In addition to above discussed datasets, we include the popularly used Waterbirds dataset [41] where the type of *background* (land/water) is assigned to bird images based on bird label; hence, being a *Causal* attribute.

Datasets cited in this section

G. Christie, N. Fendley, J. Wilson, and R. Mukherjee. Functional map of the world. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

P. Bandi, O. Geessink, Q. Manson, M. V. Dijk, M. Balkenhol, M. Hermsen, B. E. Bejnordi, B. Lee, K. Paeng, A. Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the CAMELYON17 challenge. *IEEE Transactions on Medical Imaging*, 38(2):550–560, 2018.

D. Borkan, L. Dixon, J. Sorensen, N. Thain, and L. Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *WWW*, pages 491–500, 2019.

¹<https://www.technologyreview.com/2021/07/30/1030329/machine-learning-ai-failed-covid-hospital-diagnosis-pandemic/>

Table 7: Commonly used DG datasets include auxiliary information.

Dataset	Attribute(s)	$Y - \mathbf{A}$ relationship
FMoW-WILDS [12]	time region	\mathbf{A}_{ind} \mathbf{A}_{conf}
Camelyon17-WILDS [12]	hospital	\mathbf{A}_{ind}
Waterbirds [41]	background (land/water)	\mathbf{A}_{cause}
MultiNLI [41]	negation word	\mathbf{A}_{cause}
CivilComments-WILDS [12]	demographic	\mathbf{A}_{conf}

B Proofs

B.1 Proof of Proposition 3.1

Proposition 3.1. *Given a dataset $(\mathbf{x}_i, \mathbf{a}_i, y_i)_{i=1}^n$ and a causal DAG \mathcal{G} over $\langle \mathbf{X}_c, \mathbf{X}, \mathbf{A}, Y \rangle$ such that \mathbf{X}_c is the only variable (or set of variables) that causes Y and is not independent of \mathbf{X} , then the conditional independence constraints satisfied by \mathbf{X}_c are necessary for a risk-invariant predictor over $\mathcal{P}_{\mathcal{G}}$. That is, if a predictor does not satisfy any of these constraints, then there exists a data distribution $P' \in \mathcal{P}_{\mathcal{G}}$ such that predictor’s risk will be higher than its risk in other distributions.*

Proof. Let $\mathbf{X}, Y, \mathbf{X}_c$ be random variables where \mathbf{X}_c causes Y . \mathbf{X}_c also causes the observed features \mathbf{X} but \mathbf{X} may be additionally affected by the attributes \mathbf{A} . Let $\hat{y} = g(\mathbf{x})$ be a candidate predictor. Then $g(\mathbf{X})$ represents a random vector based on a deterministic function g of \mathbf{X} . Suppose there is an independence constraint ψ that is satisfied by \mathbf{X}_c but not $g(\mathbf{X})$.² Since \mathbf{A} refers to the set of all other variables (attributes) that also cause \mathbf{X} , \mathbf{A} cannot be empty otherwise \mathbf{X} is only caused by \mathbf{X}_c and hence would satisfy all independence constraints that \mathbf{X}_c satisfies. Below we show that such a predictor g is not risk-invariant: there exist two data distributions generated according to Definition 2.2 such that the risk of g is different for them.

Without loss of generality, we can write $g(\mathbf{x})$ as,

$$g(\mathbf{x}) = (g(\mathbf{x})/h(\mathbf{x}_c)) * h(\mathbf{x}_c) = g'(\mathbf{x}, \mathbf{x}_c)h(\mathbf{x}_c) \quad \forall \mathbf{x} \sim P(\mathbf{X}) \quad (4)$$

where h is an arbitrary, non-zero, deterministic function of the random variable \mathbf{X}_c . Since \mathbf{X}_c satisfies the (conditional) independence constraint ψ and h is a deterministic function, $h(\mathbf{X}_c)$ also satisfies ψ . Also since the predictor $g(\mathbf{X})$ does not satisfy the constraint ψ , it implies that the random vector $g'(\mathbf{X}, \mathbf{X}_c)$ cannot satisfy the constraint ψ . Thus, $g'(\mathbf{X}, \mathbf{X}_c)$ cannot be a function of \mathbf{X}_c only. Since \mathbf{X} has two parents, \mathbf{X}_c and \mathbf{A} , this implies that $g'(\mathbf{X}, \mathbf{X}_c)$ and \mathbf{A} are not independent.

Now, let us construct two data distributions P_1 and P_2 such that $P(\mathbf{X}_c, Y)$ stays invariant, i.e., $P_1(\mathbf{X}_c, Y) = P_2(\mathbf{X}_c, Y)$. But $P(\mathbf{A})$ can change or $P(\mathbf{A}|Y)$ can change. Since \mathbf{A} causes \mathbf{X} , the conditional distribution $P(Y|\mathbf{X})$ will also change. Further, since $g'(\mathbf{X}, \mathbf{X}_c)$ and \mathbf{A} are not independent, $P(Y|g'(\mathbf{X}, \mathbf{X}_c))$ will change, i.e., $P_1(Y|g'(\mathbf{X}, \mathbf{X}_c)) \neq P_2(Y|g'(\mathbf{X}, \mathbf{X}_c))$.

The risk over any distribution P can be written as (using the cross-entropy loss),

$$\begin{aligned} R_P(g) &= \mathbb{E}_P[\ell(Y, g'(\mathbf{X}, \mathbf{X}_c)h(\mathbf{X}_c))] \\ &= -\mathbb{E}_P\left[\sum_y y \log g'(\mathbf{X}, \mathbf{X}_c)h(\mathbf{X}_c)\right] \\ &= -\mathbb{E}_P\left[\sum_y y \log g'(\mathbf{X}, \mathbf{X}_c)\right] - \mathbb{E}_P\left[\sum_y y \log h(\mathbf{X}_c)\right] \end{aligned} \quad (5)$$

The risk difference is,

$$\begin{aligned} R_{P_2}(g) - R_{P_1}(g) &= \mathbb{E}_{P_1}\left[\sum_y y \log g'(\mathbf{X}, \mathbf{X}_c)\right] - \mathbb{E}_{P_2}\left[\sum_y y \log g'(\mathbf{X}, \mathbf{X}_c)\right] + \mathbb{E}_{P_1}\left[\sum_y y \log h(\mathbf{X}_c)\right] - \mathbb{E}_{P_2}\left[\sum_y y \log h(\mathbf{X}_c)\right] \\ &= \mathbb{E}_{P_1}\left[\sum_y y \log g'(\mathbf{X}, \mathbf{X}_c)\right] - \mathbb{E}_{P_2}\left[\sum_y y \log g'(\mathbf{X}, \mathbf{X}_c)\right] \end{aligned}$$

²In practice, the constraint may be evaluated on an intermediate representation of g , such that g can be written as, $g(\mathbf{X}) = g_1(\phi(\mathbf{X}))$ where ϕ denotes the representation function. However, for simplicity, we assume it is applied on $g(\mathbf{X})$.

where the second equality is because $P_1(\mathbf{X}_c, Y) = P_2(\mathbf{X}_c, Y)$. The risk of $h(\mathbf{X}_c)$ would be the same across P_1 and P_2 but not for g' since $g'(\mathbf{X}, \mathbf{X}_c)$ changes across the two distributions. Thus the absolute risk difference is non-zero,

$$|R_{P_2}(g) - R_{P_1}(g)| > 0 \quad (6)$$

and g is not a risk-invariant predictor. Hence, satisfying conditional independencies that \mathbf{X}_c satisfies is necessary for a risk-invariant predictor. \square

B.2 Proof of Theorem 3.1

Theorem 3.1. *Given a causal DAG with the structure as shown in Figure 2a, the correct constraint depends on the relationship of label Y with the nuisance attributes \mathbf{A} . As shown, \mathbf{A} can be split into $\mathbf{A}_{\overline{ind}}$, \mathbf{A}_{ind} and E , where $\mathbf{A}_{\overline{ind}}$ can be further split into subsets that have a causal (\mathbf{A}_{cause}), confounded (\mathbf{A}_{conf}), selected (\mathbf{A}_{sel}) relationship with Y ($\mathbf{A}_{\overline{ind}} = \mathbf{A}_{cause} \cup \mathbf{A}_{conf} \cup \mathbf{A}_{sel}$). Then, the (conditional) independence constraints that \mathbf{X}_c should satisfy are,*

1. *Independent:* $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind}; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y, E$
2. *Causal:* $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y, E$
3. *Confounded:* $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{conf}; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{conf} | E$
4. *Selected:* $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{sel} | Y; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{sel} | Y, E$

Proof. The proof follows from d-separation (Pearl, 2009) on the causal graphs realized from Figure 2a. For each condition, *Independent*, *Causal*, *Confounded* and *Selected*, we provide the realized causal graphs below and derive the constraints.

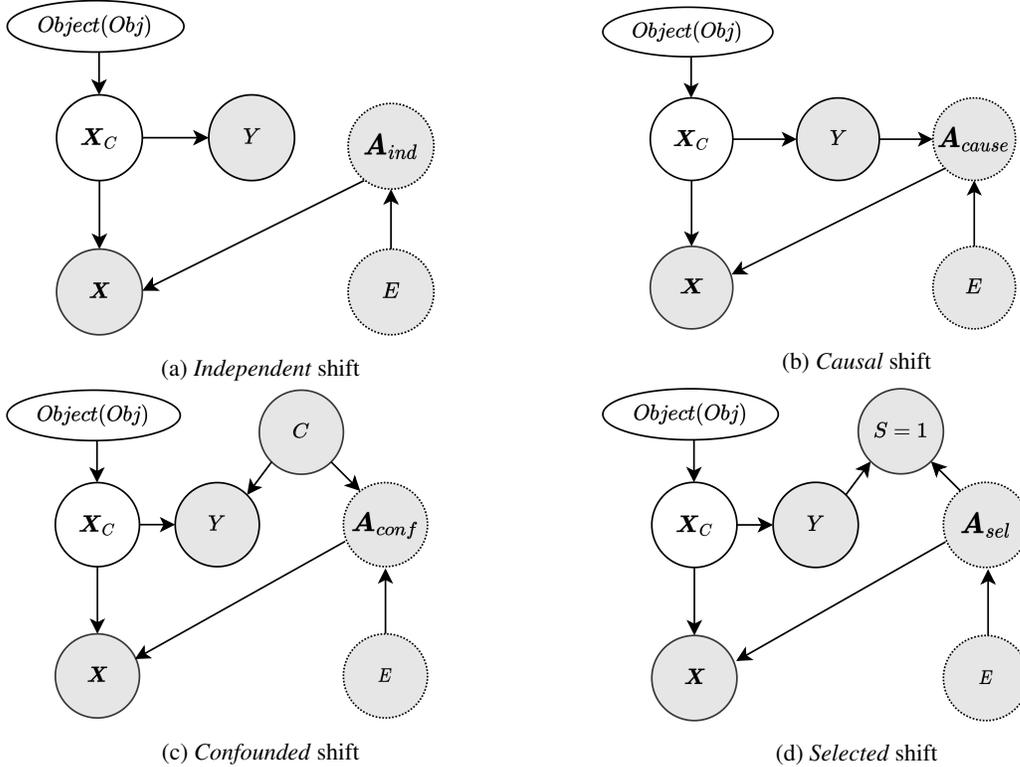


Figure 4: Causal graphs for distinct distribution shifts based on $Y - \mathbf{A}$ relationship.

Independent: As we can see in Figure 4a, we have a collider X on the path from \mathbf{X}_c to \mathbf{A}_{ind} and \mathbf{X}_c to E . Since there is a single path here, we obtain the independence constraints $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind}$ and $\mathbf{X}_c \perp\!\!\!\perp E$. Additionally, we see that conditioning on Y or E would not block the path from \mathbf{X}_c to \mathbf{A}_{ind} , which results in the remaining constraints: $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | E$ and $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y, E$. Hence, we obtain,

$$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind}; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y, E$$

Causal: From Figure 4b, we see that while the path $X_c \rightarrow X \rightarrow A_{cause}$ from X_c to A_{cause} contains a collider X , $X_c \not\perp\!\!\!\perp A_{cause}$ due to the presence of node Y as a chain. By the d-separation criteria, X_c and A_{cause} are conditionally independent given $Y \implies X_c \perp\!\!\!\perp A_{cause} | Y$. Additionally, conditioning on E is valid since E does not appear as a collider on any paths between X_c and $A_{cause} \implies X_c \perp\!\!\!\perp A_{cause} | Y, E$. We get the constraint $X_c \perp\!\!\!\perp E$ since all paths connecting X_c to E contain a collider (collider X in $X_c \rightarrow X \rightarrow A_{cause} \rightarrow E$, collider A_{cause} in $X_c \rightarrow Y \rightarrow A_{cause} \rightarrow E$). Hence, we obtain,

$$X_c \perp\!\!\!\perp A_{cause} | Y; X_c \perp\!\!\!\perp E; X_c \perp\!\!\!\perp A_{cause} | Y, E$$

Confounded: From Figure 4c, we see that all paths connecting X_c and A_{conf} contain a collider (collider X in $X_c \rightarrow X \rightarrow A_{conf}$, collider Y in $X_c \rightarrow Y \rightarrow C \rightarrow A_{conf}$). Hence, $X_c \perp\!\!\!\perp A_{conf}$. Additionally, conditioning on E is valid since E does not appear as a collider on any paths between X_c and $A_{conf} \implies X_c \perp\!\!\!\perp A_{conf} | E$. We get the constraint $X_c \perp\!\!\!\perp E$ since all paths connecting X_c and E also contain a collider (collider X in $X_c \rightarrow X \rightarrow A_{conf} \rightarrow E$, collider Y in $X_c \rightarrow Y \rightarrow C \rightarrow A_{conf} \rightarrow E$). Hence, we obtain,

$$X_c \perp\!\!\!\perp A_{conf}; X_c \perp\!\!\!\perp E; X_c \perp\!\!\!\perp A_{conf} | E$$

Selected: For the observed data, the selection variable is always conditioned on, with $S = 1$ indicating inclusion of sample in data. The selection variable S is a collider in Figure 4d and we condition on it. Hence, $X_c \not\perp\!\!\!\perp A_{sel}$. Conditioning on Y breaks the edge $X_c \rightarrow Y$, and hence all paths between X_c and A_{sel} now contain a collider (collider X in $X_c \rightarrow X \rightarrow A_{sel}$) $\implies X_c \perp\!\!\!\perp A_{sel} | Y$. Additionally, conditioning on E is valid since E does not appear as a collider on any paths between X_c and $A_{sel} \implies X_c \perp\!\!\!\perp A_{sel} | Y, E$. Hence, we obtain,

$$X_c \perp\!\!\!\perp A_{sel} | Y; X_c \perp\!\!\!\perp A_{sel} | Y, E$$

B.2.1 Proof of Corollary 3.1.1

Corollary 3.1.1. *All the above derived constraints are valid for Graph 2a. However, in the presence of a correlation between E and Obj (Graph 2b), only the constraints conditioned on E hold true.*

If there is a correlation between Obj and E , $X_c \not\perp\!\!\!\perp E$. We can see from Figure 4 that in the presence of $Obj - E$ correlation, $X_c \not\perp\!\!\!\perp A_{ind}; X_c \not\perp\!\!\!\perp A_{ind} | Y$ (4a), $X_c \not\perp\!\!\!\perp A_{cause} | Y$ (4b), $X_c \not\perp\!\!\!\perp A_{conf}$ (4c) and $X_c \not\perp\!\!\!\perp A_{sel} | Y$ (4d). Hence, conditioning on environment E is required for the valid independence constraints.

□

B.3 Proof of Theorem 3.2

Theorem 3.2. *Under the canonical causal graph in Figure 2, there exists no (conditional) independence constraint such that it is valid for all realizations of the graph as the type of multi-attribute shifts vary. Thus, for any predictor algorithm for Y that uses a single type of (conditional) independence constraint, there exists a realized graph \mathcal{G} and a corresponding training dataset such that the learned predictor cannot be a risk-invariant predictor across distributions in $\mathcal{P}_{\mathcal{G}}$.*

Proof. The proof follows from an application of Theorem 3.1 and Proposition 3.1. Under the canonical graph from Figure 2(a or b), the four types of attribute shifts possible are *Independent*, *Causal*, *Confounded* and *Selected*. From the constraints provided for these four types of attribute shifts in Theorem 3.1, it is easy to observe that there is no single constraint that is satisfied across all four shifts. Thus, given a data distribution (and hence, dataset) with specific types of multi-attribute shifts such that X_c satisfies certain (conditional) independence constraints, it is always possible to change the type of at least one of the those shifts to create a new data distribution (dataset) where the same constraints will not hold.

To prove the second claim, suppose that there exists a predictor for Y based on a single type of conditional independence constraint. Since the same constraint is not valid across all attribute shifts, we can always construct a data distribution (corresponding to a realized graph \mathcal{G}) where X_c would not satisfy the same constraint, by changing the type of at least one attribute shift. From Proposition 3.1, all conditional independence constraints satisfied by X_c under \mathcal{G} are necessary to be satisfied for a risk-invariant predictor. Hence, for the class of distributions $\mathcal{P}_{\mathcal{G}}$, a single constraint-based predictor cannot be a risk-invariant predictor. □

C Experimental Details

C.1 Additional details about baseline methods

Table 8 lists the baseline methods we compare to, the independence constraints imposed and the statistics matched/optimized by each method across environments E .

Table 8: Statistic matched/optimized by different DG algorithms. match operation matches the statistic value across E . h is a learnable domain classifier on top of shared representation ϕ . ℓ represents the main classifier loss while ℓ_d is domain classifier loss.

Constraint	Statistic	DG Algorithm
$\phi \perp\!\!\!\perp E$	match $\mathbb{E}[\phi(x) E] \forall E$ $\max_E \mathbb{E}[\ell_d(h(\phi(x)), E)]$ match $\text{Cov}[\phi(x) E] \forall E$	MMD [29] DANN [42] CORAL [7]
$Y \perp\!\!\!\perp E \phi$	match $\mathbb{E}[Y \phi(x), E] \forall E$ match $\text{Var}[\ell(f(x), y) E] \forall E$	IRM [4] VREx [5]
$\phi \perp\!\!\!\perp E Y$	match $\mathbb{E}[\phi(x) E, Y = y] \forall E$ $\max_E \mathbb{E}[\ell_d(h(\phi(x)), E) Y = y]$	C-MMD [30] CDANN [28]

C.2 Datasets

Synthetic. Our synthetic dataset is constructed based on the data-generating processes of the slab dataset [6, 32]. The original slab dataset was introduced by [32] to demonstrate the simplicity bias in neural networks as they learn the linear feature which is easier to learn in comparison to the slab feature. Our extended slab dataset, adds to the setting from [6] by using non-binary attributes and class labels to create a more challenging task and allows us to study DG algorithms in the presence of linear spurious features.

Our dataset consists of label Y ($|Y| = 5$) and 3-dimensional input \mathbf{X} consisting of features \mathbf{X}_c , \mathbf{A}_{ind} and $\mathbf{A}_{\overline{ind}}$. This is consistent with the graph in Figure 2 where attributes and causal features together determine observed features \mathbf{X} ; we concatenate \mathbf{X}_c , \mathbf{A}_{ind} and $\mathbf{A}_{\overline{ind}}$ to generate \mathbf{X} in our synthetic setup. Causal feature \mathbf{X}_c has a non-linear ‘‘slab’’ relationship with Y while $\mathbf{A}_{\overline{ind}}$ has a linear, *Causal* relationship with Y . \mathbf{A}_{ind} is independent of Y and has varying uniform distribution p_{ind} across environments. We have three environments, $E_1, E_2 \in \mathcal{E}_{tr}$ (training) and $E_3 \in \mathcal{E}_{te}$ (test). \mathbf{X}_c has a uniform distribution $\text{Uniform}[0, 1]$ across all environments.

$$y = \begin{cases} 0 & \text{if } \mathbf{X}_c \in [0, 0.2) \\ 1 & \text{if } \mathbf{X}_c \in [0.2, 0.4) \\ 2 & \text{if } \mathbf{X}_c \in [0.4, 0.6) \\ 3 & \text{if } \mathbf{X}_c \in [0.6, 0.8) \\ 4 & \text{if } \mathbf{X}_c \in [0.8, 1.0] \end{cases}$$

$$\mathbf{A}_{cause} = \begin{cases} y & \text{with prob.} = p \\ \text{abs}(y - 1) & \text{with prob.} = 1 - p \end{cases}$$

$$p_{ind}(\mathbf{A}_{ind} | E_i) = \begin{cases} \text{Uniform}[-0.4, 0.4] & \text{if } i = 1 \\ \text{Uniform}[-0.5, 0.5] & \text{if } i = 2 \\ \text{Uniform}[-0.8, 0.8] & \text{if } i = 3 \end{cases}$$

Hence, we have a five-way classification setup with multi-valued attributes and *multi-attribute* distribution shifts. Following [6], the two training domains have p as 0.9 and 1.0, and the test domain has $p = 0.0$. We add 10% noise to Y in all environments. We use the default 3-layer MLP architecture from DomainBed and use mean difference (L2) instead of MMD as the regularization penalty given the simplicity of the data.

MNIST. Rotated [8] and Colored MNIST [4] present distinct distribution shifts. While Rotated MNIST only has \mathcal{A}_{ind} wrt. rotation attribute (R), Colored MNIST only has \mathcal{A}_{cause} wrt. color attribute (C). We combine these datasets to obtain a multi-attribute dataset with $\mathcal{A}_{cause} = \{C\}$ and $\mathcal{A}_{ind} = \{R\}$. Each domain E_i has a specific rotation angle r_i and a specific correlation $corr_i$ between color C and label Y . Our setup consists of 3 domains: $E_1, E_2 \in \mathcal{E}_{tr}$ (training), $E_3 \in \mathcal{E}_{te}$ (test). We define $corr_i = P(Y = 1|C = 1) = P(Y = 0|C = 0)$ in E_i . In our setup, $r_1 = 15^\circ, r_2 = 60^\circ, r_3 = 90^\circ$ and $corr_1 = 0.9, corr_2 = 0.8, corr_3 = 0.1$. All environments have 25% label noise, as in [4]. For all experiments on MNIST, we use a two-layer perceptron consistent with previous works [4, 5].

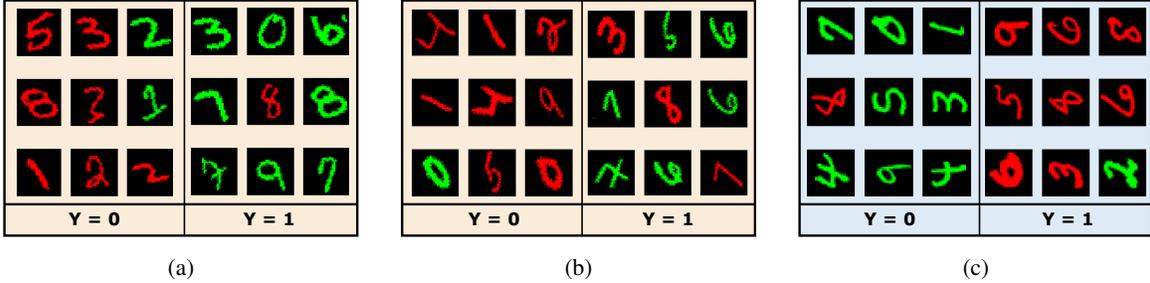


Figure 5: (a), (b) Train and (c) Test domains for MNIST.

small NORB. Moving beyond simple binary classification, we use small NORB [33], an object recognition dataset, to create a challenging setup with multi-valued classes and attributes over realistic 3D objects. It consists of images of toys of five categories with varying lighting (l), elevation (ele) and azimuths (azi). The objective is to classify unseen samples of the five categories. [10] introduced single-attribute shifts for this dataset. We combine them to yield $\mathcal{A}_{cause} = \{l\}$ wherein there is a correlation between lighting condition l and toy category y ; and $\mathcal{A}_{ind} = \{azi\}$ that varies independently across domains. Training domains have 0.9 and 0.95 spurious correlation with l whereas there is no correlation in test domain. We add 5% label noise in all environments. We use ResNet-18 (pre-trained on ImageNet) for all settings and fine tune for our task.

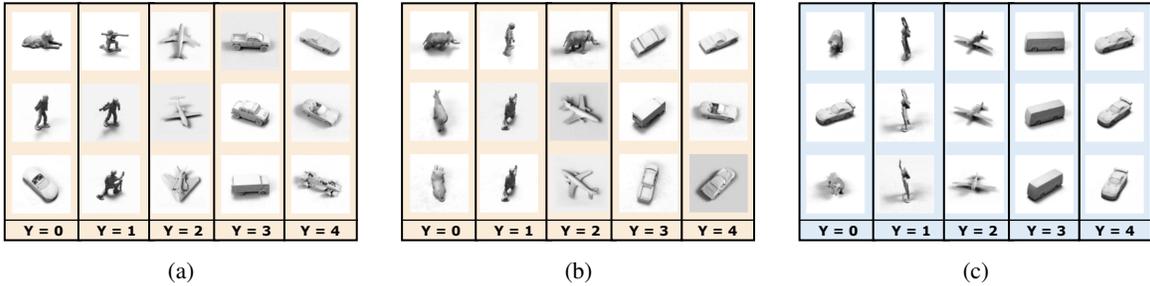


Figure 6: (a), (b) Train and (c) Test domains for small NORB.

C.3 Implementation details

All methods are trained using Adam optimizer. Synthetic and MNIST datasets are trained for 5000 steps (default in DomainBed [31]) while small NORB is trained for 2000 steps. Consistent with the default value in DomainBed, we use a batch size 64 for MNIST and 128 for Synthetic and small NORB datasets.

Regularization Penalty. We provide the *CACM* algorithm for a general graph \mathcal{G} below.

Remark. If E is observed, we always condition on E because of Corollary 3.1.1.

We provide the regularization penalty (*RegPenalty*) for *Independent*, *Causal*, *Confounded* and *Selected* shifts for our causal graph in Figure 2.

$$RegPenalty_{\mathcal{A}_{ind}} = \sum_{i=1}^{|E|} \sum_{j>i} MMD(P(g_1(\phi(\mathbf{x}))|E = E_i), P(g_1(\phi(\mathbf{x}))|E = E_j))$$

Algorithm 1 *CACM*

Input: Dataset $(\mathbf{x}_i, \mathbf{a}_i, y_i)_{i=1}^n$, causal DAG \mathcal{G} **Output:** Function $g(\mathbf{x}) = g_1(\phi(\mathbf{x})) : \mathbf{X} \rightarrow Y$ $\mathcal{A} \leftarrow$ set of observed variables in \mathcal{G} except Y, E (special domain attribute) $C \leftarrow \{\}$ \triangleright mapping of A to \mathbf{A}_s **Phase I:** Derive correct independence constraints**for** $A \in \mathcal{A}$ **do** **if** (\mathbf{X}_c, A) are d-separated **then** $\mathbf{X}_c \perp\!\!\!\perp A$ is a valid independence constraint **else if** (\mathbf{X}_c, A) are d-separated conditioned on any subset \mathbf{A}_s of the remaining observed variables in $\mathcal{A} \setminus \{A\}$ **then** $\mathbf{X}_c \perp\!\!\!\perp A | \mathbf{A}_s$ is a valid independence constraint $C[A] = \mathbf{A}_s$ **end if****end for****Phase II:** Apply regularization penalty using constraints derived**for** $A \in \mathcal{A}$ **do** **if** $\mathbf{X}_c \perp\!\!\!\perp A$ **then** $RegPenalty_A = \sum_{|E|} \sum_{i=1}^{|A|} \sum_{j>i} \text{MMD}(P(g_1(\phi(\mathbf{x}))|A_i), P(g_1(\phi(\mathbf{x}))|A_j))$ **else if** A is in C **then** $\mathbf{A}_s = C[A]$ $RegPenalty_A = \sum_{|E|} \sum_{a \in \mathbf{A}_s} \sum_{i=1}^{|A|} \sum_{j>i} \text{MMD}(P(g_1(\phi(\mathbf{x}))|A_i, a), P(g_1(\phi(\mathbf{x}))|A_j, a))$ **end if****end for** $RegPenalty = \sum_{A \in \mathcal{A}} RegPenalty_A$ $g_1, \phi = \arg \min_{g_1, \phi} \ell(g_1(\phi(\mathbf{x})), y) + \lambda^*(RegPenalty)$

$$RegPenalty_{\mathbf{A}_{cause}} = \sum_{|E|} \sum_{y \in Y} \sum_{i=1}^{|\mathbf{A}_{cause}|} \sum_{j>i} \text{MMD}(P(g_1(\phi(\mathbf{x}))|a_{i,cause}, y), P(g_1(\phi(\mathbf{x}))|a_{j,cause}, y))$$

$$RegPenalty_{\mathbf{A}_{conf}} = \sum_{|E|} \sum_{i=1}^{|\mathbf{A}_{conf}|} \sum_{j>i} \text{MMD}(P(g_1(\phi(\mathbf{x}))|a_{i,conf}), P(g_1(\phi(\mathbf{x}))|a_{j,conf}))$$

$$RegPenalty_{\mathbf{A}_{sel}} = \sum_{|E|} \sum_{y \in Y} \sum_{i=1}^{|\mathbf{A}_{sel}|} \sum_{j>i} \text{MMD}(P(g_1(\phi(\mathbf{x}))|a_{i,sel}, y), P(g_1(\phi(\mathbf{x}))|a_{j,sel}, y))$$

Model Selection. We create 90% and 10% splits from each domain to be used for training and model selection (as needed) respectively. For our main results, we use a validation set that follows the test domain distribution consistent with previous work on these datasets [4, 11, 10]. Specifically, we adopt the *test-domain validation* from DomainBed where early stopping is not allowed and all models are trained for the same fixed number of steps to limit test domain access. We additionally report results using *test-domain validation* with early stopping as well as *train-domain validation* in Suppl. D. *Train-domain validation* uses a validation set that follows the distribution of the training domains.

C.4 Hyperparameter search

Following DomainBed [31], we perform a random search 20 times over the hyperparameter distribution and this process is repeated for total 3 seeds. The best models are obtained across the three seeds over which we compute the mean and standard error. The hyperparameter search space for all datasets and algorithms is given in Table 20.

Table 9: Comparison of constraints $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}|Y, E$ and $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}|E$ in *Causal* and *Confounded* shifts. $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}|Y, E$ is a correct constraint for *Causal* shift but invalid for *Confounded* shift; $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}|E$ is a correct constraint for *Confounded* shift but invalid for *Causal* shift.

Constraint	Accuracy	
	\mathbf{A}_{cause}	\mathbf{A}_{conf}
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A} E$	29.7 ± 3.8	62.4 ± 1.9
$\mathbf{X}_c \perp\!\!\!\perp \mathbf{A} Y, E$	94.1 ± 0.5	56.0 ± 1.0

Table 10: Synthetic dataset. Accuracy on unseen domain for *Causal* distribution shift when \mathbf{A}_{cause} is provided in input (column 2) and when \mathbf{A}_{cause} is additionally used to create domains (column 3).

Algo.	Accuracy	
	\mathbf{A}_{cause} (input)	\mathbf{A}_{cause} (input+domains)
ERM	32.2 ± 2.9	29.1 ± 4.6
IRM	68.4 ± 3.4	36.4 ± 1.7
VREx	66.0 ± 2.2	24.9 ± 1.2
MMD	23.3 ± 1.7	39.7 ± 7.3
CORAL	28.6 ± 3.0	37.7 ± 4.8
DANN	44.6 ± 3.6	58.0 ± 11.6
C-MMD	36.7 ± 4.1	33.9 ± 5.6
CDANN	40.0 ± 7.2	49.8 ± 5.0
<i>CACM</i>	94.1 ± 0.5	

D Results

D.1 Comparing constraints for *Confounded* vs *Causal* shift

Here, we extend our experiments from the main paper to consider a *Confounded* shift setting. In Theorem 3.1, we see that for the *Causal* shift, $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause}|Y$; $\mathbf{X}_c \not\perp\!\!\!\perp \mathbf{A}_{cause}$ (also, $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause}|Y, E$; $\mathbf{X}_c \not\perp\!\!\!\perp \mathbf{A}_{cause}|E$) whereas for the *Confounded* shift, $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{conf}$; $\mathbf{X}_c \not\perp\!\!\!\perp \mathbf{A}_{conf}|Y$ (also, $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{conf}|E$; $\mathbf{X}_c \not\perp\!\!\!\perp \mathbf{A}_{conf}|Y, E$). We construct a synthetic setup with *Confounded* shift to demonstrate the importance of using the valid independence constraints for different kinds of shifts.

We have three environments, $E_1, E_2 \in \mathcal{E}_{tr}$ (training) and $E_3 \in \mathcal{E}_{te}$ (test). \mathbf{X}_c has a uniform distribution $\text{Uniform}[0, 1]$ across all environments. Our confounding variable c has different functional relationships with Y and \mathbf{A}_{conf} which vary across environments. Our observed input \mathbf{X} is 2-dimensional and formed by concatenating \mathbf{X}_c and \mathbf{A}_{conf} .

$$c_{E_1, E_2} = \begin{cases} 1 & \text{with prob.} = 0.25 \\ 0 & \text{with prob.} = 0.75 \end{cases} \quad c_{E_3} = \begin{cases} 1 & \text{with prob.} = 0.75 \\ 0 & \text{with prob.} = 0.25 \end{cases}$$

$$y_{true} = \begin{cases} 0 & \text{if } \mathbf{X}_c \in [0, 0.25) \\ 1 & \text{if } \mathbf{X}_c \in [0.25, 0.5) \\ 2 & \text{if } \mathbf{X}_c \in [0.5, 0.75) \\ 3 & \text{if } \mathbf{X}_c \in [0.75, 1.0] \end{cases}$$

$$y_{E_1, E_2} = \begin{cases} y_{true} + c & \text{with prob.} = 0.9 \\ y_{true} & \text{with prob.} = 0.1 \end{cases} \quad y_{E_3} = y_{true}$$

$$\mathbf{A}_{conf} = \begin{cases} 2 * c & \text{with prob.} = p \\ 0 & \text{with prob.} = 1 - p \end{cases} \quad ; p_{E_1} = 1.0, p_{E_2} = 0.9, p_{E_3} = 0.8$$

Table 9 compares the performance of these constraints in synthetic *Confounded* and *Causal* setups (Section C.2). We can see that the valid constraints according to the graph significantly outperform the incorrect constraints in both shifts. Hence, the information on the specific relationship between Y and \mathbf{A} is necessary for obtaining an optimal predictor.

D.2 Providing attribute information to DG algorithms for a fairer comparison

CACM leverages attribute labels to apply the correct independence constraints derived from the causal graph. However, existing DG algorithms only use the input features \mathbf{X} and the domain attribute. Here we provide this attribute

information to existing DG algorithms to create a more favorable setting for their application. We show that even in a relatively fairer setup, these algorithms are not able to close the performance gap with *CACM*, showing the importance of the causal information through graphs.

We consider our Synthetic dataset with *Causal* distribution shift where our observed features $\mathbf{X} = (\mathbf{X}_c, \mathbf{A}_{cause})$. Note that by construction of \mathbf{X} , since one of our input dimensions already consists of \mathbf{A}_{cause} , we explicitly make \mathbf{A}_{cause} available to all DG algorithms for applying their respective constraints. Thus, in the synthetic setup, all baselines do receive information about \mathbf{A}_{cause} in addition to the domain attribute E .

As a more informative way of providing the attribute information (\mathbf{A}_{cause}) for existing DG algorithms, we run a separate experiment where the attribute is provided as the domain indicator. Using the same underlying data distribution, we group the data (i.e., create environments/domains) based on \mathbf{A}_{cause} i.e, each environment E has samples with same value of \mathbf{A}_{cause} . In this setup (Table 10, third column), we see MMD, CORAL, DANN and CDANN show significant improvement in accuracy but the best performance is still 36% lower than *CACM* while showing high estimate variance. This reinforces our motivation to use the causal graph of the data-generating process to derive the constraint, as the attribute values alone are not sufficient. We also see IRM and VREx perform much worse than earlier, highlighting the sensitivity of DG algorithms to domain definition. In contrast, *CACM* uses the causal graph to study the structural relationships and derive the regularization penalty, which remains the same in this new dataset too.

D.3 Complete results

We provide complete results here for experiments in Section 4.

Tables 11, 12 and 13 show results on *Causal* (\mathbf{A}_{cause}), *Independent* (\mathbf{A}_{ind}) and *multi-attribute* ($\mathbf{A}_{cause} \cup \mathbf{A}_{ind}$) shifts respectively for Synthetic dataset. Tables 14, 15 and 16 show results on *Causal* (\mathbf{A}_{cause}), *Independent* (\mathbf{A}_{ind}) and *multi-attribute* ($\mathbf{A}_{cause} \cup \mathbf{A}_{ind}$) shifts respectively for MNIST. Finally, Tables 17, 18 and 19 show results on *Causal* (\mathbf{A}_{cause}), *Independent* (\mathbf{A}_{ind}) and *multi-attribute* ($\mathbf{A}_{cause} \cup \mathbf{A}_{ind}$) shifts respectively for small NORB.

While we report results using *test-domain validation* without early stopping in Section 4.3, we present additional results here using early stopping. Overall, early stopping improves accuracy across datasets and shifts for all methods. *CACM* outperforms all methods using model selection with as well as without early stopping, with the exception of Tables 12 and 15. Tables 12 and 15 show results for the *Independent* shift which is a relatively easier task and hence all methods perform similarly. For *Independent* shift in Synthetic dataset (Table 12) and MNIST (Table 15), CORAL achieves the highest accuracy. It is important to note that CORAL uses the same valid independence constraint derived by *CACM* for *Independent* shift (Theorem 3.1).

For completeness, we also include results using *train-domain validation*. However, as noted by previous work [11], using a validation set based on training domain distribution may not be suitable in the presence of spurious correlations as achieving high accuracy in training domains often leads to low accuracy in sufficiently different, novel test domains.

E Anti-Causal Graph

Figure 7 shows causal graphs used for specifying *multi-attribute* distribution shifts in an anti-causal setting. These graphs are identical to Figure 2, with the exception of change in direction of causal arrow from $\mathbf{X}_c \longrightarrow Y$ to $Y \longrightarrow \mathbf{X}_c$.

We derive the (conditional) independence constraints for the anti-causal DAG for *Independent*, *Causal*, *Confounded* and *Selected* shifts.

Theorem E.1. *Given a causal DAG with the structure as shown in Figure 7a, the correct constraint depends on the relationship of label Y with the nuisance attributes \mathbf{A} . As shown, \mathbf{A} can be split into \mathbf{A}_{ind} , \mathbf{A}_{cause} and E , where \mathbf{A}_{ind} can be further split into subsets that have a causal (\mathbf{A}_{cause}), confounded (\mathbf{A}_{conf}), selected (\mathbf{A}_{sel}) relationship with Y ($\mathbf{A}_{ind} = \mathbf{A}_{cause} \cup \mathbf{A}_{conf} \cup \mathbf{A}_{sel}$). Then, the (conditional) independence constraints that \mathbf{X}_c should satisfy are,*

1. *Independent:* $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind}; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{ind} | Y, E$
2. *Causal:* $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{cause} | Y, E$
3. *Confounded:* $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{conf} | Y; \mathbf{X}_c \perp\!\!\!\perp E; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{conf} | Y, E$
4. *Selected:* $\mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{sel} | Y; \mathbf{X}_c \perp\!\!\!\perp \mathbf{A}_{sel} | Y, E$

Proof. The proof follows from d-separation using the same logic as earlier proof in Section B.2. We observe that for all attributes $A \in \mathbf{A}_{ind}$ (\mathbf{A}_{cause} , \mathbf{A}_{conf} , \mathbf{A}_{sel}), it is required to condition on Y to obtain valid constraints as Y node appears as a chain or fork in the causal graph but never as a collider due to the $Y \longrightarrow \mathbf{X}_c$ causal arrow. \square

Corollary E.1.1. All the above derived constraints are valid for Graph 7a. However, in the presence of a correlation between E and Obj (Graph 7b), only the constraints conditioned on E hold true.

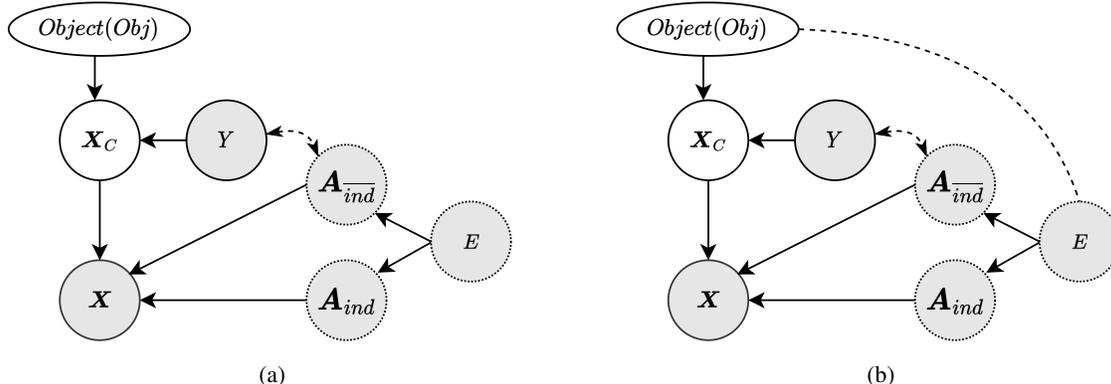


Figure 7: Corresponding anti-causal graphs for Figure 2. Note the graphs are identical to Figure 2 with the exception of the causal arrow pointing from $Y \rightarrow X_c$ instead of from $X_c \rightarrow Y$.

F Broader impact and ethical considerations

Our work on modeling the data-generating process for improved out-of-distribution generalization is an important advance in building robust predictors for practical settings. Such prediction algorithms, including methods building on representation learning, are increasingly a key element of decision-support and decision-making systems. We expect our approach to creating a robust predictor to be particularly valuable in real world setups where *spurious* attributes and real-world multi-attribute settings lead to biases in data. While not the focus of this paper, *CACM* may be applied to mitigate social biases (e.g., in language and vision datasets) whose structures can be approximated by the graphs in Figure 2. Risks of using methods such as *CACM*, include excessive reliance or a false sense of confidence. While methods such as *CACM* ease the process of building robust models, there remain many ways that an application may still fail (e.g., incorrect structural assumptions). AI applications must still be designed appropriately with support of all stakeholders and potentially affected parties, tested in a variety of settings, etc.

Table 11: Synthetic dataset. Complete results for *Causal* (A_{cause}) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	32.2 ± 2.9	37.2 ± 1.1	10.6 ± 3.5
IRM	68.4 ± 3.4	68.4 ± 3.4	19.1 ± 2.0
VREx	66.0 ± 2.2	78.9 ± 4.8	6.9 ± 2.3
MMD	23.3 ± 1.7	51.7 ± 5.3	12.1 ± 0.9
CORAL	28.6 ± 3.0	41.5 ± 2.3	15.4 ± 7.4
DANN	44.6 ± 3.6	54.3 ± 1.7	10.4 ± 6.3
C-MMD	36.7 ± 4.1	37.3 ± 1.0	10.6 ± 5.3
CDANN	40.0 ± 7.2	57.4 ± 1.7	7.7 ± 6.3
<i>CACM</i>	94.1 ± 0.5	95.6 ± 0.3	20.4 ± 2.6

Table 12: Synthetic dataset. Complete results for *Independent* (A_{ind}) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	86.3 ± 0.7	86.2 ± 1.3	88.3 ± 0.5
IRM	84.7 ± 1.0	84.2 ± 0.5	85.3 ± 0.7
VREx	84.1 ± 1.4	84.4 ± 0.9	84.5 ± 0.9
MMD	86.0 ± 1.0	86.9 ± 0.7	80.1 ± 2.3
CORAL	87.6 ± 0.4	87.6 ± 1.1	86.4 ± 1.4
DANN	84.0 ± 0.6	83.5 ± 1.4	84.0 ± 1.2
C-MMD	85.3 ± 1.3	86.0 ± 0.6	86.2 ± 1.2
CDANN	84.9 ± 1.1	85.0 ± 1.4	86.0 ± 0.4
<i>CACM</i>	86.4 ± 0.7	86.0 ± 1.3	83.3 ± 1.5

Table 13: Synthetic dataset. Complete results for *multi-attribute* ($A_{cause} \cup A_{ind}$) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	26.4 ± 1.3	36.3 ± 1.8	7.6 ± 6.2
IRM	51.0 ± 3.9	67.5 ± 1.6	3.2 ± 2.6
VREx	23.3 ± 0.4	67.6 ± 3.1	5.6 ± 4.3
MMD	23.8 ± 2.1	40.6 ± 0.3	12.9 ± 1.6
CORAL	21.7 ± 1.1	31.3 ± 2.9	11.7 ± 4.9
DANN	46.4 ± 4.3	58.6 ± 1.6	1.9 ± 1.6
C-MMD	27.6 ± 1.8	42.9 ± 5.2	12.5 ± 5.4
CDANN	40.5 ± 2.1	52.9 ± 1.1	5.0 ± 2.7
<i>CACM</i>	84.3 ± 3.5	85.2 ± 3.2	5.7 ± 2.4

Table 14: Colored + Rotated MNIST. Complete results for *Causal* (A_{cause}) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	30.9 ± 1.6	63.2 ± 2.7	10.1 ± 0.1
IRM	50.0 ± 0.1	66.1 ± 1.5	10.9 ± 0.7
VREx	30.3 ± 1.6	62.1 ± 2.6	10.2 ± 0.4
MMD	29.7 ± 1.8	57.8 ± 4.5	10.1 ± 0.1
CORAL	28.5 ± 0.8	63.3 ± 4.8	10.2 ± 0.1
DANN	20.7 ± 0.8	64.1 ± 2.4	9.6 ± 0.0
C-MMD	29.4 ± 0.2	68.3 ± 1.3	10.1 ± 0.4
CDANN	30.3 ± 9.1	63.3 ± 3.4	10.2 ± 0.2
<i>CACM</i>	70.4 ± 0.5	71.7 ± 0.7	10.1 ± 0.2

Table 15: Colored + Rotated MNIST. Complete results for *Independent* (A_{ind}) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	61.9 ± 0.5	63.4 ± 0.8	61.1 ± 0.4
IRM	61.2 ± 0.3	63.1 ± 1.0	60.5 ± 0.6
VREx	62.1 ± 0.4	62.2 ± 0.5	61.5 ± 0.2
MMD	62.2 ± 0.5	61.6 ± 0.2	60.7 ± 0.6
CORAL	62.5 ± 0.7	62.0 ± 0.4	60.3 ± 0.6
DANN	61.9 ± 0.7	62.8 ± 0.5	61.7 ± 0.7
C-MMD	62.3 ± 0.4	62.4 ± 0.3	62.3 ± 0.1
CDANN	61.8 ± 0.2	63.5 ± 0.5	62.6 ± 0.4
<i>CACM</i>	62.4 ± 0.4	63.0 ± 0.1	61.6 ± 0.3

Table 16: Colored + Rotated MNIST. Complete results for *multi-attribute* ($A_{cause} \cup A_{ind}$) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	25.2 ± 1.3	64.2 ± 5.3	10.3 ± 0.1
IRM	39.6 ± 6.7	66.2 ± 3.1	10.5 ± 0.0
VREx	23.3 ± 0.4	65.2 ± 4.4	10.0 ± 0.1
MMD	24.1 ± 0.6	62.6 ± 3.4	10.6 ± 0.3
CORAL	23.5 ± 1.1	65.9 ± 5.5	10.2 ± 0.3
DANN	32.0 ± 7.8	62.1 ± 2.4	10.9 ± 0.5
C-MMD	32.2 ± 7.0	60.0 ± 2.4	10.4 ± 0.4
CDANN	30.8 ± 8.0	67.6 ± 2.8	10.3 ± 0.2
<i>CACM</i>	54.1 ± 1.3	69.7 ± 2.6	10.2 ± 0.1

Table 17: Small NORB. Complete results for *Causal* (A_{cause}) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	65.5 ± 0.7	67.6 ± 1.3	60.0 ± 1.4
IRM	66.7 ± 1.5	68.4 ± 1.2	62.3 ± 2.1
VREx	64.7 ± 1.0	67.5 ± 0.3	58.1 ± 0.9
MMD	66.6 ± 1.6	67.5 ± 1.2	60.7 ± 0.1
CORAL	64.7 ± 0.5	67.4 ± 0.2	61.5 ± 1.7
DANN	64.6 ± 1.4	69.6 ± 0.5	61.5 ± 1.1
C-MMD	65.8 ± 0.8	68.5 ± 0.1	62.1 ± 2.4
CDANN	64.9 ± 0.5	70.9 ± 1.1	64.6 ± 1.2
<i>CACM</i>	85.4 ± 0.5	87.2 ± 0.4	75.7 ± 4.7

Table 18: Small NORB. Complete results for *Independent* (\mathcal{A}_{ind}) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	78.6 ± 0.7	79.2 ± 1.1	74.2 ± 1.5
IRM	75.7 ± 0.4	79.4 ± 0.4	72.0 ± 0.9
VREx	77.6 ± 0.5	79.6 ± 0.1	75.2 ± 0.7
MMD	76.7 ± 1.1	79.9 ± 0.7	74.7 ± 0.9
CORAL	77.2 ± 0.7	79.5 ± 0.9	75.3 ± 0.8
DANN	78.6 ± 0.7	80.0 ± 0.3	74.4 ± 0.8
C-MMD	76.9 ± 1.0	79.4 ± 0.3	75.5 ± 1.5
CDANN	77.3 ± 0.3	78.6 ± 0.9	72.5 ± 1.4
<i>CACM</i>	80.5 ± 0.6	81.3 ± 0.7	77.4 ± 1.5

Table 19: Small NORB. Complete results for *multi-attribute* ($\mathcal{A}_{cause} \cup \mathcal{A}_{ind}$) shift.

Algorithm	Test-domain validation		Train-domain validation
	no early stopping	early stopping	
ERM	64.0 ± 1.2	64.2 ± 1.1	55.6 ± 0.7
IRM	61.7 ± 0.5	64.1 ± 1.3	57.4 ± 1.0
VREx	62.5 ± 1.6	63.1 ± 1.5	48.1 ± 6.7
MMD	62.5 ± 0.3	63.1 ± 0.2	60.1 ± 1.9
CORAL	62.9 ± 0.3	63.9 ± 1.6	42.4 ± 5.0
DANN	60.8 ± 0.7	65.1 ± 1.0	57.9 ± 1.4
C-MMD	61.0 ± 0.9	62.9 ± 1.2	58.7 ± 3.0
CDANN	60.8 ± 0.9	65.6 ± 1.1	60.5 ± 1.8
<i>CACM</i>	69.6 ± 1.6	69.5 ± 1.6	55.4 ± 6.5

Table 20: Search space for random hyperparameter sweeps.

Method	Sweeps
MLP	learning rate: [1e-2, 1e-3, 1e-4, 1e-5] dropout: 0
ResNet	learning rate: [1e-2, 1e-3, 1e-4, 1e-5] dropout: [0, 0.1, 0.5]
MNIST	weight decay: 0 generator weight decay: 0
not MNIST	weight decay: $10^{\text{Uniform}(-6, -2)}$ generator weight decay: $10^{\text{Uniform}(-6, -2)}$
IRM	learning rate: [1e-2, 1e-3, 1e-4, 1e-5] λ : [0.01, 0.1, 1, 10, 100] iterations annealing: [10, 100, 1000]
VREx	learning rate: [1e-2, 1e-3, 1e-4, 1e-5] λ : [0.01, 0.1, 1, 10, 100] iterations annealing: [10, 100, 1000]
MMD	learning rate: [1e-2, 1e-3, 1e-4, 1e-5] λ : [0.1, 1, 10, 100] γ : [0.01, 0.0001, 0.000001]
CORAL	learning rate: [1e-2, 1e-3, 1e-4, 1e-5] λ : [0.1, 1, 10, 100]
DANN, CDANN	generator learning rate: [1e-2, 1e-3, 1e-4, 1e-5] discriminator learning rate: [1e-2, 1e-3, 1e-4, 1e-5] discriminator weight decay: $10^{\text{Uniform}(-6, -2)}$ λ : [0.1, 1, 10, 100] discriminator steps: [1, 2, 4, 8] gradient penalty: [0.01, 0.1, 1, 10] adam β_1 : [0, 0.5]
C-MMD	learning rate: [1e-2, 1e-3, 1e-4, 1e-5] λ : [0.1, 1, 10, 100] γ : [0.01, 0.0001, 0.000001]
CACM	learning rate: [1e-2, 1e-3, 1e-4, 1e-5] λ : [0.1, 1, 10, 100] γ : [0.01, 0.0001, 0.000001]