Quantifying Uncertainty in Causal Graphs: A Key to Improved Domain Generalization Predictions

Anonymous Author(s)
Affiliation
Address
email

Abstract

Causal graphs play a crucial role in machine learning and computer vision, often uncovering the data generation process for various downstream tasks. They provide deep insights and enable more robust, interpretable models. However, limited data availability often leads to inaccurate causal graph estimation, compromising their transferability to unseen domains. To address this, we propose a Bayesian domain generalization framework that performs Bayesian inference on causal graphs. A key advantage of our framework is its ability to quantify the uncertainty of the causal graph and leverage it to boost domain generalization (DG) prediction. Specifically, we introduce three levels of uncertainty quantification. These uncertainties enhance DG prediction compared to traditional Bayesian frameworks and provide insights into the efficacy of our method on specific datasets and the confidence of our predictions. We evaluate our algorithm Uncertainty-guided Causal Discovery for Bayesian inference (UCD-Bayes) on multiple benchmark distribution shift datasets. The empirical results demonstrate the effectiveness of UCD-Bayes, in achieving state-of-the-art DG prediction performance and outperforming traditional Bayesian approaches.

1 Introduction

Although modern deep learning models have demonstrated impressive performance across a spectrum of tasks with large-scale annotated datasets, they still face significant limitations, including poor generalization, lack of explainability, and fairness issues due to their reliance on spurious correlations. These spurious correlations are easy to capture but prone to change across different domains. To address this, a natural approach is to employ the invariance of causality and model the joint distribution of variables within a causal framework. Under this framework, multiple studies [46, 5, 38, 27, 41, 42] aim to identify and eliminate features that exhibit domain-varying correlations with the label, distinguishing them from features with true causal relations that remain stable across domains. By constructing a predictive mechanism using domain-invariant features, these works propose to address the domain generalization (DG) issue and improve out-of-distribution (OOD) prediction performance.

The discovery of invariant features can be viewed as a causal structure learning problem involving features, labels, and domain variables. Directly learning the causal structure with existing algorithms is challenging when the features are latent. Given the difficulties in jointly learning the causal structure and latent variables using only observational data, existing works either specify a simplified causal structure [5, 42, 41, 43] to guide the learning of latent variables or use an identifiable VAE framework to learn a set of latent variables and then select the invariant ones [38, 46]. However, the accuracy of the causal graph is crucial and significantly impacts the performance of downstream tasks. A line of works [43, 41, 42] assumes there are two latent variables: content, and style and only the content variable has causal relations with the target. These methods fail to capture the complex causal

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

relations in real-world data, leading to limited DG performance improvement, poor explainability, and inadequate support for other machine-learning tasks. Efforts to relax assumptions on causal graphs, such as those by [38], involve first obtaining a set of identifiable and comprehensive latent variables and selecting invariant features via additional independence tests. Still, these lack accuracy guarantees as independence tests are unreliable with insufficient data or large variable sets. To address this issue, several works [49, 60] employ Bayesian approaches to capture the posterior distribution of invariant features using Bayesian Neural Networks (BNNs), achieving good empirical performance.

However, Bayesian DG methods have primarily used Bayesian inference to derive the posterior distribution of invariant features, without fully leveraging its advantages in quantifying and utilizing uncertainties. Uncertainty is important for OOD generalization [31, 47, 50]. The deterministic deep neural networks ignore model uncertainty and data uncertainty and are usually overconfident in their predictions, causing performance accuracy drops on OOD data [16, 18]. In our paper, we propose a novel approach to performing Bayesian inference for prediction tasks. Instead of inferring invariant features from its constructed posterior distribution, we infer the causal graph that describes the underlying data generation process and select the optimal set of invariant features according to the inferred causal graph. Bayesian inference of causal graphs not only enables the effective identification of theoretically invariant and maximally predictive features but also provides a framework to quantify uncertainty at different levels and leverage it to improve DG prediction performance. To the best of our knowledge, this is the first work to provide an explicit definition and quantification method for the uncertainty of causal graphs in the context of prediction tasks. These uncertainties can provide intuition such as the efficacy of our proposed framework on specific datasets, the fitness of our learned causal graph to unseen domain data, and the confidence of our predictions. Furthermore, since our causal graph describes the data generation process, its discovery allows for causal reasoning, intervention, and inference, thereby addressing the explainability issue of deep neural networks.

Contributions. In this paper, we propose a novel Bayesian framework for DG prediction tasks, making three key contributions: 1) We perform Bayesian inference on the causal graph that describes the underlying data generation process to improve OOD generalization prediction. 2) We introduce three different levels of uncertainty for causal graphs in the context of prediction tasks. 3) We leverage these quantified uncertainties to further enhance DG prediction performance. To demonstrate the effectiveness of our framework, we conduct experiments on benchmark datasets with distribution shifts. The empirical results show that our framework achieves state-of-the-art DG prediction performance, outperforming traditional Bayesian approaches.

2 Related Work

Domain Generalization. Domain generalization aims to learn a universally applicable prediction model from one or multiple observational training domains and leverage it for prediction to any unseen test domains. The popular approaches include disentanglement representation learning [1, 6, 7, 8], data augmentation [25, 35, 34, 50], meta-learning [29, 51], adversarial training strategies [55, 57] and “mix-up" kind of strategies [63, 62, 19]. Since we employ both causality and Bayesian inference in our DG method, we discuss the related works from these two perspectives.

Causal Domain Generalization. One major approach for causal DG is the invariant/stable features learning [11, 10, 21, 22, 5, 28, 3, 4, 48, 2, 43, 38]. These methods aim to identify the invariant causal/anti-causal representations by using the invariance properties of causality as learning constraints. One well-known approach within this realm is invariant risk minimization (IRM) and its subsequent works [3, 4], which identifies the causal parent of the target given multiple environments that correspond to different interventional distributions in a data generation process. However, these approaches have limitations in certain scenarios [48, 2], where it may fail to uncover such predictors. Content-style features disentanglement methods [43, 41] separate domain-invariant causal content features and domain-varying spurious style features with derived constraints and perform OOD prediction using only content features. [38] proposes to first obtain a set of comprehensive and identifiable latent variables using the variational auto-encoder framework and find the invariant causal features with independence tests. Interventional approaches encompass robust feature learning through data augmentation and transportable interventional inference-guided feature learning techniques. [41] performs intervention on input data by identifying a set of transformations that can be applied without compromising invariant features. However, the selection of admissible transformations necessitates domain-specific expertise. [32], [58], [42] and [42] estimate the invariant and transportable interven-

tional distribution between input and target through backdoor/frontdoor adjustments. Nevertheless, these approaches require the identification and estimation of all covariation sources between input and target for backdoor adjustment, limiting their applicability in real-world scenarios.

Bayesian Domain Generalization. Some recent works apply Bayesian inference frameworks to address DG tasks. For example, [60] estimates the distribution of domain-invariant features and classifiers via BNN using variational inference. [33] proposes a variational Bayesian inference framework for aligning conditional distribution and marginal label shift through distribution alignment. In particular, [49] focuses on obtaining the posterior distribution of domain-invariant features.

Bayesian Causal Discovery. Most of the existing algorithms for causal discovery return a single DAG. However, in the cases of limited data, those methods may lead to poorly calibrated predictions. The Bayesian causal discovery methods propose to estimate the posterior distribution of causal graph given data. In general, Bayesian causal discovery can be categorized into three classes: MCMC sampling [17, 54, 44], Variational inference [37, 12], and sequential decision-based methods [29, 13].

3 A Causal Framework for Prediction Tasks

We employ the causal framework, which models the data distribution with a causal graph and a series of causal generative mechanisms. Specifically, we outline the data generation process between variables of interest with a SCM $\mathcal{M}$, which comprises a causal graph $\mathcal{G}$ and mechanism parameters Θ. We first introduce the data generation process in Section 3.1.

3.1 Data Generation Process

We provide a general form of possible causal graphs outlining the data generation process in Figure 1. $\boldsymbol{X} \in \mathcal{X}$ represents the high-dimensional input data, such as images, videos, or texts. Y represents the target variable for prediction. We denote $\boldsymbol{Z} \in \mathcal{Z} \subseteq \mathbb{R}^n$ as the latent, high-level variables for generation input $\boldsymbol{X}$. Judged by their relations to the Y, $\boldsymbol{Z}$ can be further categorized into four types: parent variables $\boldsymbol{Z}_p \in \mathcal{Z}_p \subseteq \mathbb{R}^{n_p}$, child variables $\boldsymbol{Z}_c \in \mathcal{Z}_c \subseteq \mathbb{R}^{n_c}$, spouse variables $\boldsymbol{Z}_s \in \mathcal{Z}_s \subseteq \mathbb{R}^{n_s}$, and spurious variables $\boldsymbol{Z}_o \in \mathcal{Z}_o \subseteq \mathbb{R}^{n_o}$. $\boldsymbol{Z}_p$, $\boldsymbol{Z}_c$ and $\boldsymbol{Z}_s$ are the direct causes, direct effects, and the causes of the direct effects. $\boldsymbol{Z}_o$ are the variables that are spuriously correlated to target Y via other variables. To investigate how the data distributions shift in different domains, we introduce variable $U \in \mathcal{U}$ to encode the domain-specific information. U is constant for a specific domain. To generalize the SCM, we allow for arbitrary causal relations within $\boldsymbol{Z}$ as long as the causal graph satisfies the acyclicity constraint. The proposed general form of SCMs in Figure 1 is constructed based on human intuitions and standard settings from prior works [46, 38, 27, 42, 43]. It is practical and generally holds in real-world applications. Most importantly, it covers most scenarios from prior works [4, 38] and hence is a flexible model for performing causal analysis on prediction tasks. Please refer to Appendix B.1 for a detailed summary of assumptions for the proposed general form of SCM.

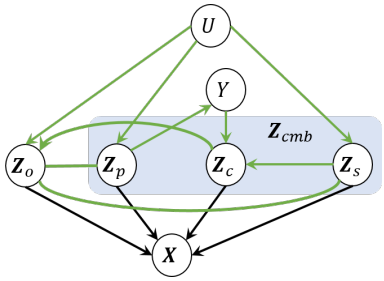

Figure 1: An illustration of the general form of causal graphs.

3.2 Prediction Tasks under Domain Generalization Settings

As illustrated in Figure 2, the data from distinct domains is generated from the same true causal graph $\mathcal{G}$ but with varying parameters Θ. The parameters consists of a series of causal mechanisms. Some causal mechanisms, such as $p(\boldsymbol{z}_o|\pi_o), p(\boldsymbol{z}_p|\pi_p)$, $p(\boldsymbol{z}_s|\pi_s)$, vary across domains since the parent sets π_o, π_p, π_s contain domain variable U. Other causal mechanisms, including $p(y|\boldsymbol{z}_p), p(\boldsymbol{z}_c|y, \boldsymbol{z}_s), p(\boldsymbol{x}|\boldsymbol{z})$, are under no influence of U and hence are considered to be domain-invariant. Ideally, under the domain generalization settings, we aim to recover the invariant[1] causal graph $\mathcal{G}$ from observational training domain data and leverage it to improve the prediction in an unseen test domain $U = u^t$.

[1]We refer to domain-invariant as invariant in this paper.

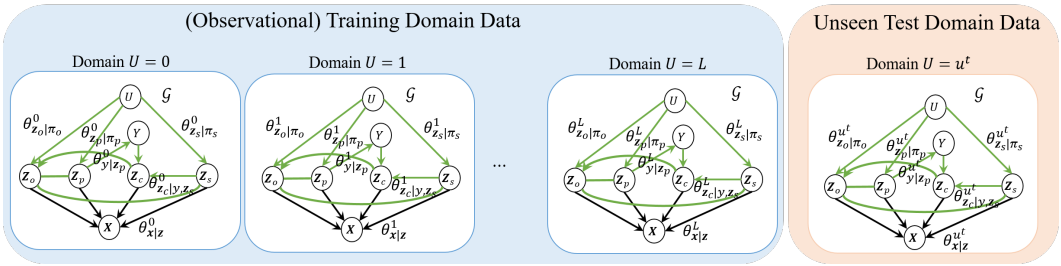

Figure 2: Illustration of data generation process for observational training domains and unseen test domain.

Let $\mathcal{D}$ be all the information provided in the observational training domain data , and $\boldsymbol{x}^t$ be an input sample drawn from an unseen test domain. The goal of the prediction tasks is to leverage the knowledge learned from the training domain and apply it for prediction on the test domain, i.e., construct $p(y|\boldsymbol{x}^t, \mathcal{D})$. In our framework, the knowledge we seek to learn from $\mathcal{D}$ is the causal graph $\mathcal{G}$ that encodes the underlying mechanisms for generating data from both training domains and all unseen test domains. In practice, learning $\mathcal{G}$ by employing causal discovery methods and maximizing $p(\mathcal{D}|\mathcal{G})$ is challenging. The severe lack of observations for most variables in the SCM renders learning $\mathcal{G}$ an ill-posed problem. Additionally, existing causal discovery methods typically offer accuracy guarantees only when a sufficient number of observations are available, a condition that is difficult to satisfy in numerous real-world applications characterized by scarce data observations. Empirical results on the benchmark Colored MNIST datasets, as outlined in Table 1, also show that by leveraging the optimal $\mathcal{G}$ learned using popular causal discovery methods merely achieves marginal improvements compared to ERM, worse than the state-of-the-art domain generalization approaches.

Thus, we propose to adopt a Bayesian approach for estimating the causal graph $\mathcal{G}$. Bayesian methods generally improve robustness to domain shift under limited observations through model averaging, and can provide explicit uncertainty estimates for model predictions.

3.3 Bayesian Inference

To address the inaccurate estimation of the causal graph $\mathcal{G}$ issue, we propose to perform Bayesian inference of the graph, i.e., sampling $\mathcal{G}$ from its constructed posterior $p(\mathcal{G}|\mathcal{D})$. The Bayesian causal discovery is usually employed when the data is limited and point-estimation causal discovery methods lead to poorly calibrated predictions [13]. Most importantly, the Bayesian approach renders the ability to quantify the uncertainty. In particular, we discover that combined with prediction tasks, we can quantify uncertainty at three levels, which we will further elaborate in Section 3.4. We introduce the integral of a causal graph $\mathcal{G}$ into the estimation of $p(y|\boldsymbol{x}^t, \mathcal{D})$, leading to the Bayesian inference outlined in Eq. (1).

$$p(y|\boldsymbol{x}^t, \mathcal{D}) = \int_{\mathcal{G}} p(y|\boldsymbol{x}^t, \mathcal{G}, \mathcal{D}) p(\mathcal{G}|\boldsymbol{x}^t, \mathcal{D}) \, \mathrm{d}\mathcal{G} \propto \mathbb{E}_{\mathcal{G}\sim p(\mathcal{G}|\mathcal{D})} \Big[p(y|\boldsymbol{x}^t, \mathcal{G}) p(\boldsymbol{x}^t|\mathcal{G}) \Big] \tag{1}$$

A detailed derivation is provided in Appendix A.1. Intuitively, one can derive potential causal graph candidates $\mathcal{G}$ from the posterior distribution $p(\mathcal{G}|\mathcal{D})$. These candidates can then be used to predict labels for inference samples from unseen domains using $p(y|\boldsymbol{x}^t, \mathcal{G})$. Notably, another term $p(\boldsymbol{x}^t|\mathcal{G})$ evaluates the compatibility between the estimated causal graph $\mathcal{G}$ and the inference sample $\boldsymbol{x}^t$. The inclusion of $p(\boldsymbol{x}^t|\mathcal{G})$ in Eq. (1) is essential because under the domain generalization setting whereby we cannot access input data $\boldsymbol{x}^t$ during training. Constructing $p(\mathcal{G} \mid \boldsymbol{x}^t, \mathcal{D})$ is generally infeasible due to the absence of complete data necessary for Bayesian causal discovery, particularly the values of $\boldsymbol{Z}, U$, and Y corresponding to the input $\boldsymbol{x}^t$. Additionally, estimating a posterior distribution for each $\boldsymbol{x}^t$ from the target domain is computationally prohibitive.

3.3.1 The Invariant Prediction Mechanism

To achieve robust domain generalization prediction performance, we aim to select a domain-invariant, transportable prediction mechanism as the $p(y|\boldsymbol{x}^t, \mathcal{G})$ in Eq. (1). According to our SCM in Figure 1, we choose to construct the prediction mechanism with the Causal Markov Blanket (CMB) variables $\boldsymbol{Z}_{cmb}$. The CMB set consists of parent variables $\boldsymbol{Z}_p$, child variables $\boldsymbol{Z}_c$, and spouse variables $\boldsymbol{Z}_s$.

It is the minimal and sufficient set of latent variables that conditioned on, the domain variable U is guaranteed to be d-separated from Y. Thus, the predictor $p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}})$[2] are free from the influence of domains. With this insight, we can obtain our predictor via Eq. (2):

$$p(y|\boldsymbol{x}^t, \mathcal{G}) = \int_{\boldsymbol{z}} \sum_{u} p(y|\boldsymbol{x}^t, \boldsymbol{z}, u, \mathcal{G}) p(\boldsymbol{z}, u|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z} = \int_{\boldsymbol{z}_{cmb}^{\mathcal{G}}} p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}}) p(\boldsymbol{z}_{cmb}^{\mathcal{G}}|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z}_{cmb}^{\mathcal{G}} \tag{2}$$

Please refer to the Appendix A.2 for detailed derivations. Eq. (5) indicates that we should employ the CMB set of latent variables and the domain-invariant predictor $p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}})$ to infer label Y. To identify the CMB latent variables from the inference input sample $\boldsymbol{x}^t$, we employ the advanced theoretical results and approach in the field of iVAE to reveal the domain-invariant mapping function from input $\boldsymbol{x}$ to latent variables $\boldsymbol{z}$. We will further elaborate on this in Section 4.

3.3.2 Sample Density Estimation in Graphs

This section aims to estimate $p(\boldsymbol{x}^t|\mathcal{G})$ in Eq. (1), where $\boldsymbol{x}^t$ represents any testing sample. Directly obtaining $p(\boldsymbol{x} \mid \mathcal{G})$ from the graph is challenging due to the unavailability of the U value for $\boldsymbol{x}^t$ in the target domain. Additionally, the causal mechanisms or conditional distributions in the target domain are also unknown. Thus, we propose to quantify $p(\boldsymbol{x}^t|\mathcal{G})$ through the epistemic uncertainty for the prediction $p(y|\boldsymbol{x}^t, \mathcal{G})$, which we denoted as single-graph prediction uncertainty $\mathcal{U}_e(\boldsymbol{x}|\mathcal{G})$. Epistemic uncertainty, also known as model uncertainty, emerges from incomplete knowledge about the most appropriate model to represent a process from insufficient data. This type of uncertainty is distinguishable because it can be reduced as additional relevant information or data becomes available, unlike aleatoric uncertainty, which is inherently random and irreducible. A fundamental characteristic of epistemic uncertainty is that it is inversely proportional to data density, which is well-explored in various literature [14, 23, 24]. Intuitively, since the classification model is trained based on the causal graph $\mathcal{G}$, it should know $\boldsymbol{x}^t$ well if $\boldsymbol{x}^t$ fits the graph well. Consequently, the epistemic uncertainty of the classification prediction on $\boldsymbol{x}^t$ should be small if $p(\boldsymbol{x}^t|\mathcal{G})$ is large.

There is no universally adopted function that best demonstrates the relationship between $\mathcal{U}_e(\boldsymbol{x}|\mathcal{G})$ and $p(\boldsymbol{x}|\mathcal{G})$. This function varies according to models and uncertainty quantification methods. Here, we adopt a generic function to quantify $p(\boldsymbol{x}^t|\mathcal{G})$ using $\mathcal{U}_e(\boldsymbol{x}|\mathcal{G})$, as outlined in Eq. (3). α is a hyperparameter to be tuned. It is worth noting that scaling uncertainty to a positive weight is well explored in various applications, where applying an exponential function with some hyperparameters is a common method.

$$p(\boldsymbol{x}^t|\mathcal{G}) \propto e^{-\alpha \mathcal{U}_e(\boldsymbol{x}|\mathcal{G})} \tag{3}$$

3.4 Uncertainty Quantification

As we indicated in Section 3.3, one of the advantages of adopting the Bayesian approach is to quantify the uncertainty and leverage it to real-world tasks, as we have illustrated in Section 3.3.2. With further exploration, we discover that we can quantify uncertainty in three different levels: causal graph uncertainty $\mathcal{U}(\mathcal{G})$, graph-data uncertainty $\mathcal{U}_e(\boldsymbol{x}|\mathcal{G})$, and data uncertainty $\mathcal{U}_e(\boldsymbol{x}|\mathcal{D})$.

Causal Graph Uncertainty $\mathcal{U}(\mathcal{G})$. The Bayesian prediction performance heavily depends on the causal graph's quality. Thus, we first introduce causal graph uncertainty $\mathcal{U}(\mathcal{G})$. This uncertainty directly arises from $p(\mathcal{G}|\mathcal{D})$ and can be calculated by the entropy or the variance of $p(\mathcal{G}|\mathcal{D})$. Estimating $\mathcal{U}(\mathcal{G})$ is crucial as it indicates when our method might outperform existing single-network methods. Due to the complexity of the real-world applications, we assume that the SCM for generating the data is a nonlinear additive noise model. According to the identifiability results in [20], given a distribution over all the variables $p_{\mathcal{D}}$, only one causal graph can be identified. However, only when we have access to indefinite data, can we identify the one true causal graph from training data $\mathcal{D}$. In practice, with sufficient data, the graph uncertainty $\mathcal{U}(\mathcal{G})$ may be small, meaning $p(\mathcal{G}|\mathcal{D})$ will be concentrated around one graph with high confidence. In such cases, existing deterministic methods might also perform well. However, with limited data, the graph uncertainty $\mathcal{U}(\mathcal{G})$ is likely to be large, necessitating the use of Bayesian inference across all possible graphs. In summary, $\mathcal{U}(\mathcal{G})$ serves as an indicator of when to employ our method.

Single-graph Prediction Uncertainty $\mathcal{U}(\boldsymbol{x}|\mathcal{G})$. We then introduce how to calculate the predictive uncertainty for $p(y|\boldsymbol{x}^t, \mathcal{G})$. We first decompose $p(y|\boldsymbol{x}^t, \mathcal{G})$ to show different sources of uncertainty as

[2]The CMB set is determined by causal graphs $\mathcal{G}$. We denote the CMB set in $\boldsymbol{z}$ according to graph $\mathcal{G}$ as $\boldsymbol{z}_{cmb}^{\mathcal{G}}$.

follows:

$$p(y|\boldsymbol{x}^t, \mathcal{G}) = \int \underbrace{p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}})}_{\text{aleatoric}} \underbrace{p(\boldsymbol{z}_{cmb}^{\mathcal{G}}|\boldsymbol{x}^t, \mathcal{G})}_{\text{epistemic}} dz_{cmb}^{\mathcal{G}}. \tag{4}$$

The classification model $p(y|\boldsymbol{x}^t, \mathcal{G})$ can be treated as an epistemic neural network [45] where the distribution of $z_{cmb}^{\mathcal{G}}$ provides the stochasticity of the model. Following [45], we leverage the entropy-based uncertainty quantification as shown in Eq. (5).

$$\begin{aligned} \underbrace{\mathcal{I}\left[y, \boldsymbol{z}_{cmb}^{\mathcal{G}}|\boldsymbol{x}, \mathcal{G}\right]}_{\text{Epistemic Uncertainty } \mathcal{U}_e(\boldsymbol{x}|\mathcal{G})} &= \underbrace{\mathcal{H}\left[p(y|\boldsymbol{x}, \mathcal{G})\right]}_{\text{Total Uncertainty } \mathcal{U}_t(\boldsymbol{x}|\mathcal{G})} - \underbrace{\mathbb{E}_{p\left(\boldsymbol{z}_{cmb}^{\mathcal{G}}|\boldsymbol{x}, \mathcal{G}\right)}\left[\mathcal{H}[p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}})]\right]}_{\text{Aleatoric Uncertainty } \mathcal{U}_a(\boldsymbol{x}|\mathcal{G})} \\ &\approx \mathcal{H}\left[\frac{1}{S}\sum_{s=1}^{S} p(y|(\boldsymbol{z}_{cmb}^{\mathcal{G}})^s)\right] - \frac{1}{S}\sum_{s=1}^{S} \mathcal{H}[p(y|(\boldsymbol{z}_{cmb}^{\mathcal{G}})^s)] \quad (\boldsymbol{z}_{cmb}^{\mathcal{G}})^s \sim p(\boldsymbol{z}_{cmb}^{\mathcal{G}}|\boldsymbol{x}, \mathcal{G}) \end{aligned} \tag{5}$$

Symbols $\mathcal{I}, \mathcal{H}, \mathbb{E}$ represent the mutual information, entropy, and expectation, respectively. Since the expectation term in Eq. (5) is often analytically intractable, we approximate it using the sample average, where S is the number of samples for CMB variables. We provide a detailed derivation in Appendix A.3. We use $\mathcal{U}_e(\boldsymbol{x}|\mathcal{G})$ for estimating density $p(\boldsymbol{x}|\mathcal{G})$, which intrinsically serves as the weights for different causal graphs in the Bayesian prediction. We expect the Bayesian inference with the weights can further boost the domain generalization prediction performance.

Bayesian Inference Uncertainty $\mathcal{U}(\boldsymbol{x}|\mathcal{D})$. With the Bayesian prediction model, not only can we obtain a joint prediction to improve prediction performance, but we can also gauge the reliability of these predictions by quantifying $\mathcal{U}(\boldsymbol{x}|\mathcal{D})$. $\mathcal{U}(\boldsymbol{x}|\mathcal{D})$ is a different type of prediction uncertainty for Bayesian inference of $p(y|\boldsymbol{x}, \mathcal{D})$. Similar to Eq. (4), we can decompose $p(y|\boldsymbol{x}, \mathcal{D})$ based on Eq. (1) where $p(y|\boldsymbol{x}, \mathcal{G})$ is the source of aleatoric uncertainty and $p(\mathcal{G}|\boldsymbol{x}, \mathcal{D})$ is the source of epistemic uncertainty. By leveraging the entropy-based uncertainty quantification, we have

$$\begin{aligned} &\underbrace{\mathcal{I}\left[y, \mathcal{G}|\boldsymbol{x}, \mathcal{D}\right]}_{\text{Epistemic Uncertainty } \mathcal{U}_e(\boldsymbol{x}|\mathcal{D})} = \underbrace{\mathcal{H}\left[p(y|\boldsymbol{x}, \mathcal{D})\right]}_{\text{Total Uncertainty } \mathcal{U}_t(\boldsymbol{x}|\mathcal{D})} - \underbrace{\mathbb{E}_{p(\boldsymbol{\mathcal{G}}|\boldsymbol{x}, \mathcal{D})}\left[\mathcal{H}[p(y|\boldsymbol{x}, \mathcal{G})]\right]}_{\text{Aleatoric Uncertainty } \mathcal{U}_a(\boldsymbol{x}|\mathcal{D})} \\ &= \mathcal{H}\left[\frac{\sum_{n=1}^{N} p(y|\boldsymbol{x}, \mathcal{G}^n) p(\boldsymbol{x}|\mathcal{G}^n)}{\sum_{n=1}^{N} p(\boldsymbol{x}|\mathcal{G}^n)}\right] - \left[\frac{\sum_{n=1}^{N} \mathcal{H}[p(y|\boldsymbol{x}, \mathcal{G}^n)] p(\boldsymbol{x}|\mathcal{G}^n)}{\sum_{n=1}^{N} p(\boldsymbol{x}|\mathcal{G}^n)}\right] \quad \mathcal{G}^n \sim p(\mathcal{G}|\mathcal{D}) \end{aligned} \tag{6}$$

where $p(y|\boldsymbol{x}, \mathcal{G}^n)$ can be calculated by Eq. (4) and approximated as shown in Eq. (5). We provide the derivation in Appendix A.4. $\mathcal{U}(\boldsymbol{x}|\mathcal{D})$ can be used for estimating the reliability of our weighted Bayesian predictions via Eq. (1).

4 Proposed Algorithm: UCD-Bayes

Guided by the theoretical results in Section 3, we propose a novel approach for performing domain generalization prediction, denoted as **U**ncertainty-guided **C**ausal **D**iscovery for **Bayes**ian Inference procedure (UCD-Bayes). In the training procedure, we perform Bayesian causal discovery to obtain samples of causal graphs and construct the invariant prediction mechanism using CMB features for each causal graph. We quantify the uncertainties during inference and then perform weighted Bayesian inference to predict the label.

4.1 The Training Procedure

The key technique in our proposed algorithm is sampling causal graphs from the posterior distribution $p(\mathcal{G}|\mathcal{D})$. However, constructing $p(\mathcal{G}|\mathcal{D})$ is challenging due to partial observations for the variables in $\mathcal{G}$. Specifically, the random variables in $\mathcal{G}$ include $V = \{U, Y, \boldsymbol{Z} = [Z_1, \cdots, Z_N], \boldsymbol{X}\}$, and we can only access observations of variables $\boldsymbol{X}, Y, U$ for training domain data. In some datasets, U is even unobservable. To address this challenge, we first estimate the values of latent variables using an existing identifiable VAE framework, obtaining complete observations of all variables in V, denoted as $\mathcal{D}^V = \{u(m), \boldsymbol{z}(m) = [z_1(m), z_2(m), \cdots, z_N(m)], \boldsymbol{x}(m), y(m)\}_{m=1}^{M}$. We then transform the learning problem into a standard Bayesian causal discovery problem, allowing us to estimate the posterior distribution $p(\mathcal{G}|\mathcal{D}^V)$ using existing algorithms. We train the invariant prediction mechanisms using the CMB features subject to each causal graph.

iVAE. Several recent works [25, 26, 38] have been devoted to developing identifiable latent variable learning frameworks. We employ the NF-iVAE [38] framework due to the consistency in the data generation assumptions. NF-iVAE constrains the learning of latent variables with a general form of prior $p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{Z}|Y,U)$ that follows the general exponential distribution. We train the framework by minimizing the objective in Eq. (7). $q_{\boldsymbol{\psi}}(\boldsymbol{Z}|\boldsymbol{X}), p_{\boldsymbol{\theta}}(\boldsymbol{X}|\boldsymbol{Z})$ are the encoder and decoder distributions that are parameterized by $\boldsymbol{\psi}$ and $\boldsymbol{\theta}$ respectively. Please refer to Appendix A.5 for detailed derivations for the $\mathcal{L}_{\text{ELBO}}$ and training procedure.

$$\mathcal{L}_{\text{iVAE}} = \underbrace{-\mathbb{E}_{q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) + \log p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{z}|y,u) - \log q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x})\right]}_{\mathcal{L}_{\text{ELBO}}} + \underbrace{\mathbb{E}_{q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\|\nabla_{\boldsymbol{z}} q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x}) - \nabla_{\boldsymbol{z}} p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{z}|y,u)\|^2\right]}_{\mathcal{L}_{\text{SM}}} \tag{7}$$

According to **Theorems 1** in [38], the estimated latent variables are component-wise identifiable subject to a few assumptions. We summarize these assumptions in Appendix B.2 and justify how their theoretical results can be applied to our framework with slight adjustments of certain assumptions. In particular, we replace the encoder distribution $q(\boldsymbol{Z}|\boldsymbol{X},Y,U)$ with $q(\boldsymbol{Z}|\boldsymbol{X})$ to accommodate our inference procedure. It is important to obtain latent variables with component-wise identifiability. It guarantees that for a set of true latent variables $\boldsymbol{z} = [z_1, z_2, \cdots, z_N]$, we can obtain a set of estimated latent variables $\hat{\boldsymbol{z}} = [\hat{z}_1, \hat{z}_2, \cdots, \hat{z}_N]$ whereby $\hat{z}_i = h_i(z_j)$. $h_1, h_2, \cdots, h_N$ are separable and invertible transformations. With such a level of identifiability, we can directly conclude that the causal graph over the random variables set with true latent variables $(\boldsymbol{x}, y, \boldsymbol{z}, u)$ is the same as the causal graph over the set with estimated latent variables $(\boldsymbol{x}, y, \hat{\boldsymbol{z}}, u)$. This insight justifies our proposed approach in performing Bayesian causal discovery on the variable set $(\boldsymbol{x}, y, \hat{\boldsymbol{z}}, u)$ when the observations of $\boldsymbol{z}$ are unavailable. However, we inherit the limitation from the NF-iVAE and other iVAE frameworks. There is a gap between theory and practice due to the violation of assumptions that guarantee the identifiability of the latent variables. Therefore, we investigate the identifiability of the obtained latent variables on real data. An empirical study in Figure 3 indicates that our estimated $\hat{\boldsymbol{Z}}$ can achieve a decent degree of identifiability even if certain assumptions are violated.

Bayesian Causal Discovery. The Bayesian causal discovery aims to estimate the posterior distribution of causal graph $\mathcal{G}$ given observations $\mathcal{D}^V$, i.e., $p(\mathcal{G}|\mathcal{D}^V)$. We obtain the values of the estimated latent variables using the mean of $q_{\boldsymbol{\psi}}(\boldsymbol{Z}|\boldsymbol{X})$. When $\boldsymbol{x}$ is high dimensional and $\boldsymbol{z}$ is low dimensional, which is common in real-world applications, $q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x})$ will be highly concentrated. We can still estimate $\boldsymbol{z}$ with high accuracy. After considering the efficiency and accuracy of the state-of-the-art methods, we employ the advanced DAG-GFlowNet [13] method to construct posterior distribution and sample different causal graphs. Since we have certain assumptions regarding the underlying causal graphs, we reject the graphs that violate these assumptions in practice and admit those that do not. In this procedure, we aim to obtain a set of L valid causal graph $\boldsymbol{\mathcal{G}} = \{\mathcal{G}^1, \mathcal{G}^2, \cdots, \mathcal{G}^L\}, \mathcal{G}^l \sim p(\mathcal{G}|\mathcal{D}^V)$.

Invariant Prediction Mechanism Learning. For each $\mathcal{G}^l \in \boldsymbol{\mathcal{G}}$, we can easily identify the CMB variables by searching for the parents, children, and spouses features of Y. We denote the CMB variables subject to $\mathcal{G}^l$ as $\boldsymbol{Z}_{cmb}^{\mathcal{G}^l}$, and then train the predictor $p_{\boldsymbol{\phi}^l}(Y|\boldsymbol{Z}_{cmb}^{\mathcal{G}^l})$ by minimizing the loss function in Eq. (8)

$$\hat{\boldsymbol{\phi}}^l = \arg\min_{\boldsymbol{\phi}^l} \mathbb{E}_{\boldsymbol{x},y\sim\mathcal{D}} \mathcal{L}_{pred}\Big(y, p_{\boldsymbol{\phi}^l}(y|\boldsymbol{z}_{cmb}^{\mathcal{G}^l})\Big) \quad \boldsymbol{z}_{cmb}^{\mathcal{G}^l} \sim q_{\hat{\boldsymbol{\psi}}}(\boldsymbol{z}|\boldsymbol{x}) \tag{8}$$

whereby the $\mathcal{L}_{pred}$ is the cross-entropy loss.$\boldsymbol{\phi} = (\boldsymbol{\phi}^1, \boldsymbol{\phi}^2, \cdots, \boldsymbol{\phi}^L)$ are the parameters of the predictors. We can obtain L predictors $\{p_{\hat{\boldsymbol{\phi}}^l}(Y|\boldsymbol{Z}_{cmb}^{\mathcal{G}^l})\}_{l=1}^L$ after training procedures.

Complexity Analysis. In the learning procedure, the Bayesian causal discovery step is computationally demanding. The scalability of UCD-Bayes depends on the number of variables the Bayesian causal discovery method can handle. We also want to emphasize that we can leverage more efficient Bayesian causal discovery methods. For the inference procedure, if we define the number of computational operations for the non-Bayesian inference approach as M, then for our Bayesian inference approach, it is $L(1+S)M$, where L is the number of causal graphs sampled, and S is the number of Z samples used for calculating the weights.

Necessity for Each Step. The main component contributes to improved domain generalization is the Bayesian inference. To perform our proposed Bayesian inference, we need to obtain the posterior distribution of the causal graph $\mathcal{G}$ over the variables of interest in the SCM. If this information is provided or pre-learned, the iVAE and Bayesian causal discovery approaches are unnecessary; we can directly sample the graph and estimate uncertainties for Bayesian inference. Our proposed algorithm aims to provide a feasible and reliable approach for obtaining $\mathcal{G}$ when it is unavailable.

4.2 The Inference Procedure

We obtain the learned causal graph set $\boldsymbol{\mathcal{G}} = \{\mathcal{G}^l\}_{l=1}^L$ and predictors $\{p_{\hat{\boldsymbol{\phi}}^l}(y|\boldsymbol{z}_{cmb}^{\mathcal{G}^l})\}_{l=1}^L$ from training procedure. For an input $\boldsymbol{x}^t$ from test domain, we first compute the graph-data uncertainty $\mathcal{U}_e(\boldsymbol{x}^t|\mathcal{G}^l)$ for each causal graph $\mathcal{G}^l$ from sampled causal graph set $\boldsymbol{\mathcal{G}}$ via Eq. (5). To obtain samples of $\boldsymbol{z}_{cmb}^{\mathcal{G}^l}$, we approximate $p(\boldsymbol{z}_{cmb}^{\mathcal{G}^l}|\boldsymbol{x}^t, \mathcal{G}^l)$ using $q_{\hat{\boldsymbol{\psi}}}(\boldsymbol{z}|\boldsymbol{x}^t)$ and known causal graph $\mathcal{G}^l$, i.e., we first obtain the values for all latent variables and select the CMB features according to $\mathcal{G}^l$. With the obtained uncertainties $\{\mathcal{U}_e(\boldsymbol{x}^t|\mathcal{G}^l)\}$, we calculate the the densities $\{p(\boldsymbol{x}^t|\mathcal{G}^l)\}_{l=1}^L$ using Eq. (3). We then substitute normalized $\{p(\boldsymbol{x}^t|\mathcal{G}^l)\}_{l=1}^L$ and $\{p_{\hat{\boldsymbol{\phi}}^l}(y|\boldsymbol{z}_{cmb}^{\mathcal{G}^l})\}_{l=1}^L$ into Eq. (1) to obtain our final predictions.

Effectiveness Analysis. The proposed method leverages Bayesian inference for Bayesian model averaging (BMA) to make joint predictions based on multiple graphs. BMA is known to have several theoretical benefits. As shown by Proposition 3.3 of [56], BMA reduces error compared to using a single model. We can extend this theorem to our setting with minor modifications. PAC-Bayesian theory [15] provides a framework for deriving explicit generalization bounds for models with distributions over their parameters. We aim to extend the theorems in [15] for our framework to provide OOD generalization bounds. Intuitively, the Bayesian approach performs well in scenarios where the data is insufficient or noisy. In such cases, methods that aim to identify a single set of invariant causal features for OOD prediction (such as [38]) may fail to select features accurately. Our Bayesian approach offers a more robust selection of features. On benchmark datasets, particularly CMNIST, where the data is sufficient, methods that perform point estimation of invariant causal features also perform well. It motivates us to develop causal graph uncertainty, which is the variance of $p(\mathcal{G}|\mathcal{D})$. This term indicates when our method is more likely to outperform non-Bayesian methods.

5 Experiments

We aim to investigate the effectiveness of our proposed algorithm UCD-Bayes. First, we show that we can obtain latent variables with a decent level of identifiability. Then we present the Bayesian inference results on multiple benchmark datasets and show how uncertainty can be employed for further improving the OOD prediction.

Dataset. We conduct experiments on four benchmark datasets, Colored-MNIST (CMNIST), PACS, VLCS, and OfficeHome. **CMNIST** [42] contains digit images from 10 different categories.[3] In the training domain, the data is generated such that digits are associated with different background or foreground colors. However, in the testing domain, the digits' colors are independent. **PACS** [30] contains images from four domains: Photo (P), Art painting (A), Cartoon (C), and Sketch (S), with each domain comprising images in 7 categories. **VLCS** [52] has images of 5 categories from four domains: PASCAL VOC 2007 (P), LabelMe (L), Caltech (C), and Sun (S). **OfficeHome** [53] includes images from four domains: Artistic (A), Clipart (C), Product (P), and Real World (R), with 65 object categories related to office and home settings. We use the standard leave-one-domain-out protocol, testing on images from one domain and training on the others.

Implementation Details. For the CMNIST dataset, we adopt the architectural configurations from one of our baselines, as described by [38], utilizing multilayer perceptrons (MLP) as both encoder and decoder components. For PACS, VLCS, and OfficeHome datasets, we leverage a pre-trained ResNet-50 on ImageNet as the encoder backbone, complemented by a decoder of comparable complexity. In all cases, the classifier consists of a two-layer MLP. All experiments are conducted on a Ti2080 GPU.

5.1 Verifying identifiability of latent variables

Our iVAE framework has a gap between theory and practice, as some assumptions may be violated in practice. Therefore, we first investigate the identifiability level of estimated $\boldsymbol{Z}$

[3] We adopt the most challenging setting of the colored MNIST dataset as one of our baselines [42]. This setting creates a significant difference between the distributions of training domain images and testing domain images, making correlation-based models capture spurious correlations between color and digits.

on CMNIST whereby only two training domains are available. Following the standard protocol in [25], we compute the average mean correlation coefficient (MCC) between samples of latent variables recovered by different models trained with different random initialization. Higher MCC scores indicate stronger identifiability. We compare the identifiability results for unidentifiable VAE [36], and original NF-iVAE method [38]. Our iVAE has a different encoder distribution compared to NF-iVAE. We set the number of latent variables as $|\boldsymbol{Z}| = 10$. Results in Table 3 show that our iVAE achieves similar latent variable identifiability as the NF-iVAE, outperforming the unidentifiable VAE. To reuse the learned encoder distribution for computing the uncertainties and perform Bayesian Inference, we replace $q(\boldsymbol{z} \mid \boldsymbol{x}, y, u)$ in original NF-iVAE with $q(\boldsymbol{z} \mid \boldsymbol{x})$. This change can still deliver latent variables with a reasonable degree of identifiability. However, the slight drop in identifiability is expected since we provide less information as input to the encoder.

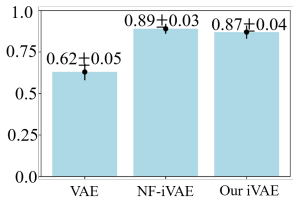

Figure 3: Average MCC on CMNIST

5.2 Domain Generalization Prediction

We validate our proposed algorithm UCD-Bayes on domain generalization tasks. We compare three types of approaches. First, we compare the state-of-the-art classifier as the ERM approach for each dataset. We then compare with causal DG approaches, including IRM[5], GenInt[41], MatchDG[40], SagNet[43], Causalrep [59], CIRL[39], CaSN[61], iCaRL[38], and Bayesian DG approaches, including PTG [49] and BiteBayes[60]. For a comprehensive comparison, especially on real datasets, we also compare to other DG methods.

Table 1: Comparison with SOTA methods on CMNIST.

Algorithms	Prediction Acc (%)	
	In-distribution	Out-of-distribution
ERM	**85.2**$\pm$0.5	10.8$\pm$0.2
Robust MIN MAC	84.3$\pm$0.4	10.9$\pm$0.5
F-IRM GAME	63.4$\pm$1.1	60.0$\pm$2.7
V-IRM GAME	64.0$\pm$1.0	49.2$\pm$3.4
RSC	76.3$\pm$0.3	20.5$\pm$1.0
IRM	59.3$\pm$4.4	62.8$\pm$9.6
Causalrep	70.1$\pm$1.5	68.6$\pm$5.5
iCaRL	70.6$\pm$0.8	68.8$\pm$1.5
iVAE + GES	84.9$\pm$1.2	12.4$\pm$0.7
iVAE + NOTEARS	84.1$\pm$1.0	15.5$\pm$0.5
iVAE + DAG-GFlowNet + BI	70.4$\pm$0.2	55.8$\pm$0.2
UCD-Bayes (iVAE + DAG-GFlowNet + BI + UQ)	72.8$\pm$0.2	**69.5**$\pm$0.1

Evaluation on Bayesian inference and single-graph prediction uncertainty. We show the empirical performance of our UCD-Bayes on the CMNIST dataset in Table 1. We compare several state-of-the-art causal DG approaches. In particular, we also show the prediction performance using the optimal causal graph learned from training domain data using causal discovery methods GES [9] and NOTEARS [64]. They only achieve marginal improvement in OOD prediction compared to ERM. We then show the performance using the Bayesian causal discovery method DAG-GFlowNet without using single-graph prediction uncertainty. According to Table 1, we can infer that the single-graph uncertainty $\mathcal{U}_e(\boldsymbol{x}|\mathcal{G})$ can further improve prediction performance for both in-distribution and OOD since it tells the Bayesian framework which causal graph is more suitable for the inferred samples. However, the UCD-Bayes relies on the performance of the iVAE and Bayesian causal discovery. A worse level of identifiability of learned latent variables and inaccurate Bayesian causal discovery will compromise the OOD prediction performance. We choose $|\boldsymbol{Z}| = 10$. Increasing the dimension of $\boldsymbol{Z}$ will increase the difficulty of Bayesian causal discovery, resulting in possible worse performance. Ablation study using $|\boldsymbol{Z}| = 15$ results in an OOD performance of 47.5%.

Evaluation on causal graph uncertainty For causal graph uncertainty $\mathcal{U}(\mathcal{G})$, we quantify it using the differences between the CMB selection results, i.e., if $Z_i \in \boldsymbol{Z}_{cmb}$, then we remark it as 1, otherwise 0. We obtained a $|\boldsymbol{Z}|$ vector for each sampled graph and calculated the average distance as $\mathcal{U}(\mathcal{G})$. We set the number of sampled graphs to $L = 5, 10, 20$, with more sampled graphs, the average distances decrease, resulting in decreasing improvements between the Bayesian model and the single model. In practice, this term can also be used to select hyperparameters.

Table 2: Evaluation on $\mathcal{U}(\mathcal{G})$

L	$\mathcal{U}(\mathcal{G})$	Δ OOD Acc
5	2.6	54.0
10	2.3	52.3
20	2.0	50.4

Evaluation on Bayesian inference uncertainty. We evaluate the Bayesian inference uncertainty $\mathcal{U}(\boldsymbol{x}|\mathcal{D})$ by using it to reduce the unconfident predictions in test domain data. We reduce 5% and 10% predictions with the highest Bayesian inference uncertainty, and the OOD prediction performance improves.

Table 3: Evaluation on $\mathcal{U}_e(\boldsymbol{x}|\mathcal{D})$

Reduction ratio	0%	5%	10%
OOD Prediction Acc	69.5	71.6	73.5

Evaluation on real-world datasets. To evaluate our UCD-Bayes more comprehensively, we applied it to more challenging, real-world image datasets. Our UCD-Bayes achieved optimal performance on the PACS and OfficeHome datasets, and state-of-the-art performance on the VLCS dataset. For comparisons with additional baselines, please refer to Tables 5 and 6 in the Appendix. As emphasized, the performance of our method partially relies on effective latent variable learning and Bayesian causal discovery. Therefore, for a specific dataset, the appropriate iVAE and Bayesian causal discovery methods should be selected using our causal graph uncertainty measure $\mathcal{U}(\mathcal{G})$.

Table 4: Empirical results on VLCS, PACS, and OfficeHome datasets in terms of OOD prediction accuracy (%). We report the average performance over all domains.

Algorithms	VLCS	PACS	OfficeHome
ERM	77.2	83.5	66.5
GroupDRO	77.9	84.4	66.0
RSC	77.5	85.2	65.5
Mixup	77.7	84.6	68.4
FACT	-	88.2	66.6
IRM	78.5	85.5	64.3
SagNet	77.5	86.3	68.1
iCaRL	81.8	88.7	-
CaSN	79.1	87.3	68.1
CIRL	-	**90.1**	67.1
BiteBayes	79.1	85.5	66.4
PTG	76.1	83.7	61.6
UCD-Bayes	81.5	**89.0**	**69.5**

6 Conclusion

In this work, we propose a novel Bayesian framework, UCD-Bayes, for DG prediction tasks. Empirical studies indicate that our Bayesian framework achieves state-of-the-art OOD prediction performance across multiple datasets. Specifically, we quantify the uncertainty of the causal graph at three levels. Our empirical results demonstrate that these uncertainty measurements not only enhance the generalization of Bayesian inference but also provide insights into the effectiveness of Bayesian inference on specific datasets and the confidence in our final predictions. **Limitations.** However, due to optimization difficulties, we adopt a phase-by-phase learning framework to recover the underlying causal graph from the data. Errors in the earlier stages can affect the accuracy of the final outputs. Additionally, there is a gap between theory and practice, as some assumptions may be violated in real-world applications. **Broader Impacts.** Our model provides a causal framework for modeling the data generation process, enhancing the explainability and trustworthiness of modern deep learning models.

References

[1] Alessandro Achille and Stefano Soatto. Emergence of invariance and disentanglement in deep representations. *The Journal of Machine Learning Research*, 19(1):1947–1980, 2018.

[2] Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. Invariance principle meets information bottleneck for out-of-distribution generalization. In *Advances in Neural Inf. Proc. Systems*, 2021.

[3] Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization game. In *International Conference on Machine Learning*, 2020.

[4] Kartik Ahuja, Jun Wang, Amit Dhurandhar, Karthikeyan Shanmugam, and Kush R. Varshney. Empirical or invariant risk minimization? a sample complexity perspective. In *International Conference on Learning Representations*, 2021.

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

[6] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[7] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in β-vae. *arXiv preprint arXiv:1804.03599*, 2018.

[8] Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

[9] David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of machine learning research*, 3(Nov):507–554, 2002.

[10] Peng Cui and Susan Athey. Stable learning establishes some common ground between causal inference and machine learning. *Nature Machine Intelligence*, 4(2):110–115, 2022.

[11] Peng Cui, Zheyan Shen, Sheng Li, Liuyi Yao, Yaliang Li, Zhixuan Chu, and Jing Gao. Causal inference meets machine learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3527–3528, 2020.

[12] Chris Cundy, Aditya Grover, and Stefano Ermon. Bcd nets: Scalable variational approaches for bayesian causal discovery. *Advances in Neural Information Processing Systems*, 34:7095–7110, 2021.

[13] Tristan Deleu, António Góis, Chris Emezue, Mansi Rankawat, Simon Lacoste-Julien, Stefan Bauer, and Yoshua Bengio. Bayesian structure learning with generative flow networks. In *Uncertainty in Artificial Intelligence*, pages 518–528. PMLR, 2022.

[14] Stefan Depeweg, Jose-Miguel Hernandez-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *International conference on machine learning*, pages 1184–1193. PMLR, 2018.

[15] Pascal Germain, Francis Bach, Alexandre Lacoste, and Simon Lacoste-Julien. Pac-bayesian theory meets bayesian inference. *Advances in Neural Information Processing Systems*, 29, 2016.

[16] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

[17] David Heckerman, Dan Geiger, and David M Chickering. Learning bayesian networks: The combination of knowledge and statistical data. *Machine learning*, 20:197–243, 1995.

[18] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 41–50, 2019.

[19] Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019.

[20] Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. *Advances in neural information processing systems*, 21, 2008.

[21] Dominik Janzing. Causal regularization. *Advances in Neural Information Processing Systems*, 32, 2019.

[22] Yibo Jiang and Victor Veitch. Invariant and transportable representations for anti-causal domain shifts. *arXiv preprint arXiv:2207.01603*, 2022.

[23] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

[24] Salman Khan, Munawar Hayat, Syed Waqas Zamir, Jianbing Shen, and Ling Shao. Striking the right balance with uncertainty. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 103–112, 2019.

[25] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR, 2020.

[26] Ilyes Khemakhem, Ricardo Monti, Diederik Kingma, and Aapo Hyvarinen. Ice-beem: Identifiable conditional energy-based deep models based on nonlinear ica. *Advances in Neural Information Processing Systems*, 33:12768–12778, 2020.

[27] Lingjing Kong, Shaoan Xie, Weiran Yao, Yujia Zheng, Guangyi Chen, Petar Stojanov, Victor Akinwande, and Kun Zhang. Partial disentanglement for domain adaptation. In *International Conference on Machine Learning*, pages 11455–11472. PMLR, 2022.

[28] Masanori Koyama and Shoichiro Yamaguchi. When is invariance useful in an out-of-distribution generalization problem? *arXiv preprint arXiv:2008.01883*, 2020.

[29] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[30] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017.

[31] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and Ling-Yu Duan. Uncertainty modeling for out-of-distribution generalization. *arXiv preprint arXiv:2202.03958*, 2022.

[32] Bing Liu, Dong Wang, Xu Yang, Yong Zhou, Rui Yao, Zhiwen Shao, and Jiaqi Zhao. Show, deconfound and tell: Image captioning with causal inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18041–18050, 2022.

[33] Quande Liu, Cheng Chen, Jing Qin, Qi Dou, and Pheng-Ann Heng. Feddg: Federated domain generalization on medical image segmentation via episodic learning in continuous frequency space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1013–1023, 2021.

[34] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pages 4114–4124. PMLR, 2019.

[35] Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *International Conference on Machine Learning*, pages 6348–6359. PMLR, 2020.

[36] Romain Lopez, Jeffrey Regier, Michael I Jordan, and Nir Yosef. Information constraints on auto-encoding variational bayes. *Advances in neural information processing systems*, 31, 2018.

[37] Lars Lorch, Jonas Rothfuss, Bernhard Schölkopf, and Andreas Krause. Dibs: Differentiable bayesian structure learning. *Advances in Neural Information Processing Systems*, 34:24111–24123, 2021.

[38] Chaochao Lu, Yuhuai Wu, José Miguel Hernández-Lobato, and Bernhard Schölkopf. Invariant causal representation learning for out-of-distribution generalization. In *International Conference on Learning Representations*, 2021.

[39] Fangrui Lv, Jian Liang, Shuang Li, Bin Zang, Chi Harold Liu, Ziteng Wang, and Di Liu. Causality inspired representation learning for domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8046–8056, 2022.

[40] Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. In *International Conference on Machine Learning*, pages 7313–7324. PMLR, 2021.

[41] Chengzhi Mao, Augustine Cha, Amogh Gupta, Hao Wang, Junfeng Yang, and Carl Vondrick. Generative interventions for causal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3947–3956, 2021.

[42] Chengzhi Mao, Kevin Xia, James Wang, Hao Wang, Junfeng Yang, Elias Bareinboim, and Carl Vondrick. Causal transportability for visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7521–7531, 2022.

[43] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8690–8699, 2021.

[44] Teppo Niinimäki, Pekka Parviainen, Mikko Koivisto, et al. Structure discovery in bayesian networks by sampling partial orders. 2016.

[45] Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Epistemic neural networks. *Advances in Neural Information Processing Systems*, 36, 2024.

[46] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):947–1012, 2016.

[47] Fengchun Qiao and Xi Peng. Uncertainty-guided model generalization to unseen domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6790–6800, 2021.

[48] Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. In *International Conference on Learning Representations*, 2021.

[49] Shiyu Shen, Bin Pan, Tianyang Shi, Tao Li, and Zhenwei Shi. Bayesian domain invariant learning via posterior generalization of parameter distributions. *arXiv preprint arXiv:2310.16277*, 2023.

[50] Xinwei Shen, Furui Liu, Hanze Dong, Qing Lian, Zhitang Chen, and Tong Zhang. Weakly supervised disentangled generative causal representation learning. *The Journal of Machine Learning Research*, 23(1):10994–11048, 2022.

[51] Yang Shu, Zhangjie Cao, Chenyu Wang, Jianmin Wang, and Mingsheng Long. Open domain generalization with domain-augmented meta-learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9624–9633, 2021.

[52] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

[53] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017.

[54] Jussi Viinikka, Antti Hyttinen, Johan Pensar, and Mikko Koivisto. Towards scalable bayesian learning of causal dags. *Advances in Neural Information Processing Systems*, 33:6584–6594, 2020.

[55] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *Advances in neural information processing systems*, 31, 2018.

[56] Hanjing Wang and Qiang Ji. Diversity-enhanced probabilistic ensemble for uncertainty estimation. In *Uncertainty in Artificial Intelligence*, pages 2214–2225. PMLR, 2023.

[57] Haotao Wang, Chaowei Xiao, Jean Kossaifi, Zhiding Yu, Anima Anandkumar, and Zhangyang Wang. Augmax: Adversarial composition of random augmentations for robust training. *Advances in neural information processing systems*, 34:237–250, 2021.

[58] Tan Wang, Jianqiang Huang, Hanwang Zhang, and Qianru Sun. Visual commonsense r-cnn. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10760–10770, 2020.

[59] Yixin Wang and Michael I Jordan. Desiderata for representation learning: A causal perspective. *arXiv preprint arXiv:2109.03795*, 2021.

[60] Zehao Xiao, Jiayi Shen, Xiantong Zhen, Ling Shao, and Cees Snoek. A bit more bayesian: Domain-invariant learning with uncertainty. In *International Conference on Machine Learning*, pages 11351–11361. PMLR, 2021.

[61] Mengyue Yang, Yonggang Zhang, Zhen Fang, Yali Du, Furui Liu, Jean-Francois Ton, Jianhong Wang, and Jun Wang. Invariant learning via probability of sufficient and necessary causes. *Advances in Neural Information Processing Systems*, 36, 2024.

[62] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019.

[63] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[64] Xun Zheng, Chen Dan, Bryon Aragam, Pradeep Ravikumar, and Eric Xing. Learning sparse nonparametric dags. In *International Conference on Artificial Intelligence and Statistics*, pages 3414–3425. PMLR, 2020.

A Detailed Derivations

A.1 Derivations for Eq. (1)

$$
\begin{aligned}
p(y|\boldsymbol{x}^t, \mathcal{D}) =& \int_{\mathcal{G}} p(y|\boldsymbol{x}^t, \mathcal{G}, \mathcal{D}) p(\mathcal{G}|\boldsymbol{x}^t, \mathcal{D}) \, \mathrm{d}\mathcal{G} \\
=& \int_{\mathcal{G}} p(y|\boldsymbol{x}^t, \mathcal{G}) p(\mathcal{G}|\boldsymbol{x}^t, \mathcal{D}) \, \mathrm{d}\mathcal{G} \quad \text{We assume } y \perp\!\!\!\perp \mathcal{D}|\boldsymbol{x}^t, \mathcal{G} \\
=& \int_{\mathcal{G}} p(y|\boldsymbol{x}^t, \mathcal{G}) \frac{p(\boldsymbol{x}^t|\mathcal{G}, \mathcal{D}) p(\mathcal{G}|\mathcal{D})}{p(\boldsymbol{x}^t|\mathcal{D})} \, \mathrm{d}\mathcal{G} \\
=& \int_{\mathcal{G}} p(y|\boldsymbol{x}^t, \mathcal{G}) \frac{p(\boldsymbol{x}^t|\mathcal{G}) p(\mathcal{G}|\mathcal{D})}{p(\boldsymbol{x}^t)} \, \mathrm{d}\mathcal{G} \quad \text{We assume } \boldsymbol{x}^t \perp\!\!\!\perp \mathcal{D} \text{ and } \boldsymbol{x}^t \perp\!\!\!\perp \mathcal{D}|\mathcal{G} \\
=& \frac{1}{p(\boldsymbol{x}^t)} \int_{\mathcal{G}} p(y|\boldsymbol{x}^t, \mathcal{G}) p(\boldsymbol{x}^t|\mathcal{G}) p(\mathcal{G}|\mathcal{D}) \, \mathrm{d}\mathcal{G} \\
\propto& \int_{\mathcal{G}} p(y|\boldsymbol{x}^t, \mathcal{G}) p(\boldsymbol{x}^t|\mathcal{G}) p(\mathcal{G}|\mathcal{D}) \, \mathrm{d}\mathcal{G} \\
=& \mathbb{E}_{\mathcal{G} \sim p(\mathcal{G}|\mathcal{D})} \Big[p(y|\boldsymbol{x}^t, \mathcal{G}) p(\boldsymbol{x}^t|\mathcal{G}) \Big]
\end{aligned}
\tag{9}
$$

A.2 Derivation for Eq. (2)

$$
\begin{aligned}
p(y|\boldsymbol{x}^t, \mathcal{G}) =& \int_{\boldsymbol{z}} \sum_u p(y|\boldsymbol{x}^t, \boldsymbol{z}, u, \mathcal{G}) p(\boldsymbol{z}, u|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z} \quad \text{introduce other variables} \\
=& \int_{\boldsymbol{z}} \sum_u p(y|\boldsymbol{x}^t, \boldsymbol{z}_o^{\mathcal{G}}, \boldsymbol{z}_{cmb}^{\mathcal{G}}, u, \mathcal{G}) p(\boldsymbol{z}, u|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z} \\
=& \int_{\boldsymbol{z}} \sum_u p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}}) p(\boldsymbol{z}_{cmb}^{\mathcal{G}}, \boldsymbol{z}_o^{\mathcal{G}}, u|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z} \quad (Y \perp\!\!\!\perp \boldsymbol{Z}_o^{\mathcal{G}}, U|\boldsymbol{Z}_{cmb}^{\mathcal{G}})_{\mathcal{G}} \\
=& \int_{\boldsymbol{z}} p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}}) \sum_u p(\boldsymbol{z}_{cmb}^{\mathcal{G}}, \boldsymbol{z}_o^{\mathcal{G}}, u|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z} \\
=& \int_{\boldsymbol{z}} p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}}) p(\boldsymbol{z}_{cmb}^{\mathcal{G}}, \boldsymbol{z}_o^{\mathcal{G}}|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z} \\
=& \int_{\boldsymbol{z}_{cmb}^{\mathcal{G}}} p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}}) \int_{\boldsymbol{z}_o^{\mathcal{G}}} p(\boldsymbol{z}_{cmb}^{\mathcal{G}}, \boldsymbol{z}_o^{\mathcal{G}}|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z}_o^{\mathcal{G}} \, \mathrm{d}\boldsymbol{z}_{cmb}^{\mathcal{G}} \\
=& \int_{\boldsymbol{z}_{cmb}^{\mathcal{G}}} p(y|\boldsymbol{z}_{cmb}^{\mathcal{G}}) p(\boldsymbol{z}_{cmb}^{\mathcal{G}}|\boldsymbol{x}^t, \mathcal{G}) \, \mathrm{d}\boldsymbol{z}_{cmb}^{\mathcal{G}}
\end{aligned}
\tag{10}
$$

A.3 Derivation of Eq. (5)

We first compute the total uncertainty term $\mathcal{U}_t(\mathcal{G})$, as shown in Eq. (11)

$$
\begin{aligned}
\mathcal{U}_t(\mathcal{G}) =& \mathcal{H}\left[p(y|\boldsymbol{x}, \mathcal{G})\right] \\
=& \mathcal{H}\left[\mathbb{E}_{p(\boldsymbol{z}_{cmb}|\boldsymbol{x}, \mathcal{G})}\left[p(y|\boldsymbol{z}_{cmb})\right]\right] \quad \text{According to Eq. (10)} \\
=& \mathcal{H}\left[\frac{1}{S}\sum_{s=1}^{S} p(y|\boldsymbol{z}_{cmb}^s)\right] \quad \boldsymbol{z}_{cmb}^s \sim p(\boldsymbol{z}_{cmb}|\boldsymbol{x}, \mathcal{G})
\end{aligned}
\tag{11}
$$

Then we compute the aleatoric uncertainty $\mathcal{U}_a(\mathcal{G})$, as outlined in Eq. (12)

$$
\begin{aligned}
\mathcal{U}_a(\mathcal{G}) =& \mathbb{E}_{p(\boldsymbol{z}_{cmb}|\boldsymbol{x}, \mathcal{G})}\left[\mathcal{H}[p(y|\boldsymbol{z}_{cmb})]\right] \\
=& \frac{1}{S}\sum_{s=1}^{S} \mathcal{H}[p(y|\boldsymbol{z}_{cmb}^s)] \quad \boldsymbol{z}_{cmb}^s \sim p(\boldsymbol{z}_{cmb}|\boldsymbol{x}, \mathcal{G})
\end{aligned}
\tag{12}
$$

Substituting Eq. (11) and Eq. (12) into Eq. (5), we have

$$\mathcal{U}_e(\mathcal{G}) = \mathcal{U}_t(\mathcal{G}) - \mathcal{U}_a(\mathcal{G}) = \mathcal{H}\left[\frac{1}{S}\sum_{s=1}^{S} p(y|\boldsymbol{z}_{cmb}^s)\right] - \frac{1}{S}\sum_{s=1}^{S}\mathcal{H}[p(y|\boldsymbol{z}_{cmb}^s)] \quad \boldsymbol{z}_{cmb}^s \sim p(\boldsymbol{z}_{cmb}|\boldsymbol{x},\mathcal{G}) \tag{13}$$

A.4 Derivation of Eq. (6)

$$\begin{aligned}
&\underbrace{\mathcal{H}\left[p(y|\boldsymbol{x},\mathcal{D})\right]}_{\text{Total Uncertainty } \mathcal{U}_t(\boldsymbol{x}|\mathcal{D})} = \mathcal{H}\left[\int_{\mathcal{G}} p(y|\boldsymbol{x}^t,\mathcal{G},\mathcal{D})p(\mathcal{G}|\boldsymbol{x}^t,\mathcal{D})\,\mathrm{d}\mathcal{G}\right] = \mathcal{H}\left[\frac{\mathbb{E}_{\mathcal{G}\sim p(\mathcal{G}|\mathcal{D})}\left[p(y|\boldsymbol{x}^t,\mathcal{G})p(\boldsymbol{x}^t|\mathcal{G})\right]}{\mathbb{E}_{\mathcal{G}\sim p(\mathcal{G}|\mathcal{D})}\left[p(\boldsymbol{x}^t|\mathcal{G})\right]}\right] \\
&\approx \mathcal{H}\left[\frac{\sum_{n=1}^{N} p(y|\boldsymbol{x},\mathcal{G}^n)p(\boldsymbol{x}|\mathcal{G}^n)}{\sum_{n=1}^{N} p(\boldsymbol{x}|\mathcal{G}^n)}\right] \quad \mathcal{G}^n \sim p(\mathcal{G}|\mathcal{D}) \\
&\underbrace{\mathbb{E}_{p(\boldsymbol{\mathcal{G}}|\boldsymbol{x},\mathcal{D})}\left[\mathcal{H}[p(y|\boldsymbol{x},\mathcal{G})]\right]}_{\text{Aleatoric Uncertainty } \mathcal{U}_a(\boldsymbol{x}|\mathcal{D})} = \mathbb{E}_{p(\boldsymbol{\mathcal{G}}|\mathcal{D})}\left[\frac{p(\boldsymbol{x}|\mathcal{G})\mathcal{H}[p(y|\boldsymbol{x},\mathcal{G})]}{\mathbb{E}_{\mathcal{G}\sim p(\mathcal{G}|\mathcal{D})}\left[p(\boldsymbol{x}^t|\mathcal{G})\right]}\right] = \left[\frac{\sum_{n=1}^{N}\mathcal{H}[p(y|\boldsymbol{x},\mathcal{G}^n)]p(\boldsymbol{x}|\mathcal{G}^n)}{\sum_{n=1}^{N} p(\boldsymbol{x}|\mathcal{G}^n)}\right] \quad \mathcal{G}^n \sim p(\mathcal{G}|\mathcal{D})
\end{aligned} \tag{14}$$

A.5 Derivation of Eq. (7)

We start with the joint distribution of observed variables $\boldsymbol{X}, Y, U$, i.e., $p(\boldsymbol{X}, Y, U)$.

$$\begin{aligned}
& -\log p(\boldsymbol{x}, y, u) \\
=& -\log p(\boldsymbol{x}|y,u)p(y,u) \quad y,u \text{ are observed variables. } p(y,u) \text{ is known.} \\
=& -\log p(y,u) - \log p(\boldsymbol{x}|y,u) \\
=& -\log p(\boldsymbol{x}|y,u) + c \quad -\log p(y,u) \text{ is constant.} \\
=& -\log \int_{\boldsymbol{z}} p(\boldsymbol{x},\boldsymbol{z}|y,u)\,\mathrm{d}\boldsymbol{z} + c \\
=& -\log \int_{\boldsymbol{z}} p(\boldsymbol{x}|\boldsymbol{z},y,u)p(\boldsymbol{z}|y,u)\,\mathrm{d}\boldsymbol{z} + c \\
=& -\log \int_{\boldsymbol{z}} p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}|y,u)\,\mathrm{d}\boldsymbol{z} + c \quad \boldsymbol{X} \perp\!\!\!\perp Y,U|\boldsymbol{Z} \\
=& -\log \int_{\boldsymbol{z}} \frac{p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}|y,u)}{q(\boldsymbol{z}|\boldsymbol{x})}q(\boldsymbol{z}|\boldsymbol{x})\,\mathrm{d}\boldsymbol{z} + c \\
=& -\log \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\frac{p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}|y,u)}{q(\boldsymbol{z}|\boldsymbol{x})}\right] + c \\
\leq& -\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}|y,u)}{q(\boldsymbol{z}|\boldsymbol{x})}\right] + c \\
=& -\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\log p(\boldsymbol{x}|\boldsymbol{z}) + \log p(\boldsymbol{z}|y,u) - \log q(\boldsymbol{z}|\boldsymbol{x})\right] + c
\end{aligned} \tag{15}$$

According to Eq. (15), $-\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\log p(\boldsymbol{x}|\boldsymbol{z}) + \log p(\boldsymbol{z}|y,u) - \log q(\boldsymbol{z}|\boldsymbol{x})\right] + c$ is upper bound of the negative log joint likelihood over observed variables $\boldsymbol{X}, Y, U$. We ignore the constant term in training since there is no parameter to optimize. Its expected value over data observations from the training distribution $p_{\mathcal{D}}$ is defined as ELBO loss $\mathcal{L}_{\text{ELBO}}$ in Eq. (7).

Training Details for Our iVAE: As described in Eq. (7), the training objective comprises an ELBO loss $\mathcal{L}_{\text{ELBO}}$ and a score matching loss $\mathcal{L}_{\text{SM}}$. The ELBO loss $\mathcal{L}_{\text{ELBO}}$ optimizes over encoder and decoder parameters $(\boldsymbol{\theta}, \boldsymbol{\psi})$, while the score matching loss $\mathcal{L}_{\text{SM}}$ minimizes over prior parameters (T, λ). The parameters $(\boldsymbol{T}, \boldsymbol{\lambda})$ are constants in $\mathcal{L}_{\text{ELBO}}$, and $\boldsymbol{\theta}, \boldsymbol{\psi}$ are constants in $\mathcal{L}_{\text{SM}}$. The score matching loss $\mathcal{L}_{\text{SM}}$ minimizes over prior parameters $(\boldsymbol{T}, \boldsymbol{\lambda})$. The parameters $\boldsymbol{T}, \boldsymbol{\lambda}$ are constants in $\mathcal{L}_{\text{ELBO}}$, and $\boldsymbol{\theta}, \boldsymbol{\psi}$ are constants in $\mathcal{L}_{\text{SM}}$.

$$\mathcal{L}_{\text{iVAE}}(\boldsymbol{\theta}, \boldsymbol{\psi}, \boldsymbol{T}, \boldsymbol{\lambda}) \coloneqq \mathcal{L}_{\text{ELBO}}(\boldsymbol{\theta}, \boldsymbol{\psi}, \hat{\boldsymbol{T}}, \hat{\boldsymbol{\lambda}}) + \mathcal{L}_{\text{SM}}(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\psi}}, \boldsymbol{T}, \boldsymbol{\lambda})$$

B Theoretical Analysis

B.1 General Form of SCM Assumptions

The general form of SCM we proposed in Figure 1 should satisfy the assumptions in **Assumption 1**.

Assumption 1. *(a) U is the root node and does not have direct links with Y or $\boldsymbol{X}$. (b) Z_i is generated by either Y or U for any i; (c) $\boldsymbol{Z}_p$, $\boldsymbol{Z}_c$, and $\boldsymbol{Z}_s$ collectively form the Causal Markov Blanket (CMB) set of target Y. The CMB set of Y does not contain U. Y does not have a direct link to $\boldsymbol{X}$. (d) $\boldsymbol{X}$ is the child of Z_i for any i. $\boldsymbol{X}$ is the leaf node in the graph. (e) The causal graph over $\{\boldsymbol{X}, Y, \boldsymbol{Z}, U\}$ is a DAG.*

B.2 Latent Variable Learning

We adopt the NF-iVAE framework [38] and tailored it to for our purpose. NF-iVAE train the VAE with a prior distribution on $\boldsymbol{Z}$ that is consistent with our **Assumption 1(b)** and belongs to a general exponential family, i.e.,

$$p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{Z}|Y,U) = \frac{\mathcal{Q}(\boldsymbol{Z})}{\mathcal{C}(Y,U)} \exp[\boldsymbol{T}(\boldsymbol{Z})^T \boldsymbol{\lambda}(Y,U)]$$

where $\mathcal{Q}$ is the base measure. $\mathcal{C}$ is the normalizing constant. $\boldsymbol{\lambda}$ is the arbitrary function. $\boldsymbol{T}$ is the sufficient statistics.[4]

Compatibility between $p(\boldsymbol{Z}|Y,U)$ and our SCM. As can be seen from the joint distribution $p(\boldsymbol{x}, y, u) \propto \int_{\boldsymbol{z}} p(\boldsymbol{x}|\boldsymbol{z}) p(\boldsymbol{z}|y,u)\, \mathrm{d}\boldsymbol{z}$, the prior $p(\boldsymbol{z}|y,u)$ is consistent with our SCM. In fact, based on our SCM, $p(\boldsymbol{z}|y,u)$ can be further decomposed into the product of conditional probabilities, which would result in further sparsity in $\boldsymbol{\lambda}$. This property makes our learnt model differ from existed work based on different SCMs [25, 26, 38]. However, the causal graph is generally unknown *a priori*, and hence we can not pre-define the sparsity of $\boldsymbol{\lambda}$. Therefore, in our learning algorithm we treat $p(\boldsymbol{z}|y,u)$ as a generic form of prior that satisfies **Assumption 1(b)**. Without a fully specified causal graph, we use this generic prior $p(\boldsymbol{z}|y,u)$ to constrain the learning of VAE to obtain the $\boldsymbol{Z}$ without knowing their causal identities. We summarize the assumptions and identifiability results in **Theorem 1**.

Theorem 1. *Assume the data is sampled from a generative model described by*

$$p_{\boldsymbol{\xi}=(\boldsymbol{\theta},\boldsymbol{T},\boldsymbol{\lambda})}(\boldsymbol{X},\boldsymbol{Z}|Y,U) = p_{\boldsymbol{\theta}}(\boldsymbol{X}|\boldsymbol{Z}) p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{Z}|Y,U),\ p_{\boldsymbol{\theta}}(\boldsymbol{X}|\boldsymbol{Z}) = p_{\boldsymbol{\epsilon}}(\boldsymbol{X} - g_{\boldsymbol{\theta}}(\boldsymbol{Z}))$$

Assume the following holds: (i) Denote the characteristic function of $p_{\boldsymbol{\epsilon}}$ as $\varphi_{\boldsymbol{\epsilon}}$, $\{\boldsymbol{X}|\varphi_{\boldsymbol{\epsilon}}(\boldsymbol{X}) = 0\}$ has measure zero. (ii) g has second-order cross derivatives and is injective. (iii) The sufficient statistics $\boldsymbol{T}(\boldsymbol{Z}) = [T_1(Z_1)^T, \cdots, T_N(Z_N)]^T$ have all second-order own derivatives, and all the $T_i(Z_i)$ have dimension larger or equal to 2. (iv) There exist $k+1$ distinct points $(Y^0,U^0),(Y^1,U^1),\cdots,(Y^k,U^k)$ such that the matrix $L = \Big(\boldsymbol{\lambda}(Y^1,U^1) - \boldsymbol{\lambda}(Y^0,U^0), \cdots, \boldsymbol{\lambda}(Y^k,U^k) - \boldsymbol{\lambda}(Y^0,U^0)\Big)$ of size $k \times k$ is invertible, where k is the dimension of $\boldsymbol{T}$. Then, the following holds: $\boldsymbol{\xi}$ is identifiable up to a permutation and component-wise transformation.

Theorem 1 is the same to **Theorem 1** in NF-iVAE. Please refer to [38] for detailed proof. Our contribution does not lie in proposing new assumptions for proving the component-wise identifiability of the latent variables. We merely aim to show that we can train the VAE with the same general prior distribution $p(\boldsymbol{Z}|Y,U)$ since it satisfies the data generation process of our proposed SCM.

As indicated in Section 4, we use a different encoder distribution $q(\boldsymbol{Z}|\boldsymbol{X})$, instead of $q(\boldsymbol{Z}|\boldsymbol{X},Y,U)$ in NF-iVAE. Therefore, we have a different theorem for obtaining the true parameters $\boldsymbol{\xi}^*$ using our proposed learning framework.

Theorem 2. *Assume the following assumptions hold: (i) The family of distributions $q_{\boldsymbol{\psi}}(\boldsymbol{Z}|\boldsymbol{X})$ contains $p_{\boldsymbol{\xi}}(\boldsymbol{Z}|\boldsymbol{X},Y,U)$, and $q_{\boldsymbol{\psi}}(\boldsymbol{Z}|\boldsymbol{X}) > 0$ everywhere. (ii) The NF-iVAE learning framework, which minimizes $\mathcal{L}_{iVAE}(\boldsymbol{\psi},\boldsymbol{\xi})$ in Eq.* (16) *with respect to both $\boldsymbol{\xi}$ and $\boldsymbol{\psi}$, can learn the true parameters $\boldsymbol{\xi}^*$ up to a permutation and simple transformation of the latent variable $\boldsymbol{Z}$ in the limit of infinite data.*

[4] Arbitrary function $\boldsymbol{\lambda}$ and sufficient statistics $\boldsymbol{T}$ are modeled by neural networks with ReLU activation due to their universal approximation ability.

Proof. We recall from the loss function in Phase I is as follows:

$$\mathcal{L}_{\text{iVAE}}(\boldsymbol{\theta}, \boldsymbol{\psi}, \boldsymbol{T}, \boldsymbol{\lambda}) \coloneqq \mathcal{L}_{\text{ELBO}}(\boldsymbol{\theta}, \boldsymbol{\psi}, \hat{\boldsymbol{T}}, \hat{\boldsymbol{\lambda}}) + \mathcal{L}_{\text{SM}}(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\psi}}, \boldsymbol{T}, \boldsymbol{\lambda}) \tag{16}$$

$$\mathcal{L}_{\text{ELBO}} \coloneqq -\mathbb{E}_{p_{\mathcal{D}}}\left[\mathbb{E}_{q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) + \log p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{z}|y,u) - \log q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x})]\right] \tag{17}$$

$$\mathcal{L}_{\text{SM}} \coloneqq \mathbb{E}_{p_{\mathcal{D}}}\left[\mathbb{E}_{q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x})}[\|\nabla_{\boldsymbol{z}} \log q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x}) - \nabla_{\boldsymbol{z}} \log p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{z}|y,u)\|^2]\right] \tag{18}$$

If the family of $q_{\boldsymbol{\psi}}(\boldsymbol{Z}|\boldsymbol{X})$ is flexible enough to contain $p_{\boldsymbol{\xi}}(\boldsymbol{Z}|\boldsymbol{X}, Y, U)$, then by optimizing the $\mathcal{L}_{\text{iVAE}}$ over its parameter $\boldsymbol{\xi}$, the score matching term $\mathcal{L}_{\text{SM}}$ is minimized and eventually reach zero. If we assume that the model is not degenerate and that $q_{\boldsymbol{\psi}} > 0$ everywhere, then we have

$$\begin{aligned} \mathcal{L}_{\text{SM}} = 0 &\implies \nabla_{\boldsymbol{z}} \log q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x}) = \nabla_{\boldsymbol{z}} \log p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{z}|y,u) \\ &\implies \log q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x}) = \log p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{z}|y,u) + c \end{aligned} \tag{19}$$

for some constant c. c is zero because both $q_{\boldsymbol{\psi}}(\boldsymbol{z}|\boldsymbol{x})$ and $p_{\boldsymbol{T},\boldsymbol{\lambda}}(\boldsymbol{z}|y,u)$ are pdf's. Therefore, the $\mathcal{L}_{\text{iVAE}}$ will be equal to the log-likelihood. Under such circumstances, the estimation in Eq. (16) inherits all the properties of maximum likelihood estimation (MLE). Since our identifiability is guaranteed up to a permutation and componentwise transformation, the consistency of MLE indicates that we converge to the true parameters $\boldsymbol{\xi}^*$ up to a permutation and component-wise transformation in the limit of infinite data. □

Automatically, we will have Theorem 3 that proves the identifiability of learned $\boldsymbol{Z}^*$.

Theorem 3. *Assume that Theorem 1 and Theorem 2 hold, then in the limit of infinite data, the true latent variables $\boldsymbol{Z}^*$ are identifiable up to a permutation and componentwise transformation.*

Proof. **Theorem 1** and **Theorem 2** guarantee that in the limit of infinite data, iVAE can obtain the true parameters $\boldsymbol{\xi}^* \coloneqq (\boldsymbol{\theta}^*, \boldsymbol{T}^*, \boldsymbol{\lambda}^*)$ up to a permutation and componentwise transformation of the latent variables. We denote the parameters obtained from NF-iVAE as $\hat{\boldsymbol{\xi}} \coloneqq (\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{T}}, \hat{\boldsymbol{\lambda}})$, i.e., $(\hat{\boldsymbol{\phi}}, \hat{\boldsymbol{T}}, \hat{\boldsymbol{\lambda}})$ and $(\boldsymbol{\phi}^*, \boldsymbol{T}^*, \boldsymbol{\lambda}^*)$ are identifiable up to a permutation and component-wise transformation. If there were no noise, we have $\hat{\boldsymbol{Z}} = g^{-1}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{X})$ that are equal to $\boldsymbol{Z}^* = g^{-1}_{\boldsymbol{\theta}^*}(\boldsymbol{X})$ up to a permutation and componentwise transformation. If with noise, we can obtain the posterior distribution of the latent variables up to an analogous indeterminacy. □

C Detailedd Empirical Performance

C.1 Detailed Empirical Results for PACS, VLCS, and OfficeHome

Table 5: Empirical results on VLCS and PACS datasets in terms of OOD prediction accuracy (%).

Algorithms	VLCS					PACS				
	C	**L**	**S**	**V**	**Avg**	**A**	**C**	**P**	**S**	**Avg**
ERM	98.0±0.4	62.6±0.9	70.8 ±1.9	77.5 ±1.9	77.2	84.8±1.3	76.4±1.1	96.7±0.6	76.1±1.0	83.5
GroupDRO	98.1± 0.3	66.4 ± 0.9	71.0 ± 0.3	76.1 ± 1.4	77.9	83.5±0.9	79.1±0.6	96.7±0.3	78.3±2.0	84.4
MLDG	98.5± 0.3	61.7± 1.2	73.6± 1.8	75.0± 0.8	77.2	85.5±1.4	80.1±1.7	97.4±0.3	76.6±1.1	84.9
CORAL	96.9± 0.9	65.7± 1.2	73.3± 0.7	78.7± 0.8	78.7	88.3±0.2	80.0±0.7	97.5±0.3	78.8±1.3	86.2
MMD	98.3± 0.1	65.6± 0.7	69.7± 1.0	75.7± 0.9	77.3	86.1±1.4	79.4±0.9	96.6±0.2	76.5±0.7	84.6
RSC	97.5± 0.6	63.1± 1.2	73.0± 1.3	76.2± 0.5	77.5	85.4±0.8	79.7±1.8	97.6±0.3	78.2±1.2	85.2
Mixup	98.4± 0.3	63.4± 0.7	72.9± 0.8	76.1± 1.2	77.7	86.1±0.7	78.9±0.8	97.6±0.1	75.8±1.8	84.6
DANN	98.5± 1.3	64.9± 1.3	72.6± 1.4	78.7± 1.7	78.2	86.4±0.8	77.4±0.8	97.3±0.4	73.5±2.3	83.6
CDANN	97.6± 0.6	65.2± 0.8	73.4± 1.4	76.9± 0.5	78.3	84.6±1.8	75.5±0.9	96.8±0.3	73.5±0.6	82.6
MTL	97.6± 0.6	60.6± 1.3	71.0± 1.2	77.2± 0.7	76.6	87.5±0.8	77.1±0.7	96.4±0.8	77.3±1.8	84.6
ARM	97.2± 0.5	62.7± 1.	70.6± 0.6	75.8± 0.9	76.6	86.8±0.6	76.8±0.7	97.4±0.3	79.3±1.2	85.1
IRM	**98.6**±0.1	66.0 ±0.9	72.3 ±0.6	77.3 ±0.9	78.5	84.7±0.4	80.0±0.6	97.2±0.3	79.3±1.0	85.5
SagNet	97.3± 0.4	61.6± 0.8	73.4± 1.9	77.6± 0.4	77.5	87.4±1.0	80.7±0.6	97.1±0.1	80.0±0.4	86.3
iCaRL	-	-	-	-	**81.8**	-	-	-	-	88.7
CaSN	98.1±0.3	67.5±0.8	72.9±0.7	78.3±0.9	79.1	88.5±0.6	83.2±1.0	97.2±0.3	80.4±0.5	87.3
BiteBayes	97.3± 0.2	67.2± 0.1	73.0± 0.2	78.8± 0.1	79.1	83.9±0.7	81.6±81.6	96.0±0.2	80.3±0.9	85.5
PTG	-	-	-	-	76.1	-	-	-	-	83.7
UCD-Bayes	98.6±0.2	**69.2** ±0.1	**76.5**±0.4	**81.7**±0.1	81.5	**89.2**±0.7	**85.6**±1.2	**97.6**±0.5	**83.6**±0.6	**89.0**

We provided the detailed empirical results for each domain in Table 5 and 6. For algorithms with read-to-use implementations, we run the algorithms for 5 trials and report the mean and std in the

Table 6: Empirical results on OfficeHome datasets in terms of OOD prediction accuracy (%).

Algorithms	OfficeHome				
	A	**C**	**P**	**R**	**Avg**
ERM	61.3 ± 0.7	52.4 ± 0.3	75.8 ± 0.1	76.6 ± 0.3	66.5
GroupDRO	60.4 ± 0.7	52.7 ± 1.0	75.0 ± 0.7	76.0 ± 0.7	66.0
MLDG	61.5 ± 0.9	53.2 ± 0.6	75.0 ± 1.2	77.5 ± 0.4	66.8
CORAL	65.3 ± 0.4	54.4 ± 0.5	76.5 ± 0.1	78.4 ± 0.5	68.7
MMD	60.4 ± 0.2	53.3 ± 0.3	74.3 ± 0.1	77.4 ± 0.6	66.3
RSC	60.7 ± 1.4	51.4 ± 0.3	74.8 ± 1.1	75.1 ± 1.3	65.5
Mixup	62.4 ± 0.8	54.8 ± 0.6	77.3 ± 0.3	**79.2 ± 0.2**	68.4
DANN	59.9 ± 1.3	53.0 ± 0.3	73.6 ± 0.7	76.9 ± 0.5	65.9
CDANN	61.5 ± 1.4	50.4 ± 2.4	74.4 ± 0.9	76.6 ± 0.8	65.8
MTL	61.5 ± 0.7	52.4 ± 0.6	74.9 ± 0.4	76.8 ± 0.4	66.4
ARM	58.9 ± 0.8	51.0 ± 0.5	74.1 ± 0.1	75.2 ± 0.3	64.8
IRM	58.9 ± 2.3	52.2 ± 1.6	72.1 ± 2.9	74.0 ± 2.5	64.3
SagNet	**63.4± 0.2**	54.8± 0.4	75.8± 0.4	78.3± 0.3	68.1
CaSN	63.5 ±0.2	54.5±0.2	**77.8**±0.3	76.5±0.3	68.1
BiteBayes	61.8 ±0.4	53.3±0.4	74.3±0.4	76.3±0.2	66.4
PTG	-	-	-	-	61.6
UCD-Bayes	63.1 ±0.1	**56.9**±0.2	**78.8**±0.2	79.1±0.1	**69.5**

tables. If the reported results for an algorithm are better then we report the results in their original papers. For algorithms without implementations, such as iCaRL and PTG, we use the reported results.

D Limitations.

Our UCD-Bayes method requires knowledge of the domain variables during the iVAE and Bayesian causal discovery phases and is not applicable when these domain variables are unknown. The iVAE step estimates the latent variable Z given the input X, target Y, and domain variable U. The outcome of the iVAE step directly affects the quality of the data ($\mathcal{D} = (X, Y, U, Z)$), particularly the latent variables Z, for Bayesian causal discovery. The accuracy of the Bayesian causal discovery process, in turn, influences the selection of CMB features for prediction. Moreover, we want to emphasize that with poor quality observational data, our method gains more benefits through Bayesian model averaging for OOD predictions, as a single graph cannot capture the correct graph distribution.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: [NA]

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: [NA]

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: [NA]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: [NA]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: [NA]

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: [NA]

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.