Semi-Supervised Learning under General Causal Models

Archer Moore, Heejung Shim, Jingge Zhu, Mingming Gong

***Abstract*—Semi-supervised learning (SSL) aims to train a machine learning model using both labelled and unlabelled data. While the unlabelled data have been used in various ways to improve the prediction accuracy, the reason why unlabelled data could help is not fully understood. One interesting and promising direction is to understand SSL from a causal perspective. In light of the independent causal mechanism principle, the unlabelled data can be helpful when the label causes the features but not vice versa. However, the causal relations between the features and labels can be complex in real world applications. In this paper, we propose a SSL framework that works with general causal models in which the variables have flexible causal relations. More specifically, we explore the causal graph structures and design corresponding causal generative models which can be learned with the help of unlabelled data. The learned causal generative model can generate synthetic labelled data for training a more accurate predictive model. We verify the effectiveness of our proposed method by empirical studies on both simulated and real data.**

***Index Terms*—semi-supervised learning, causal graph, generative model**

I. Introduction

The goal of classification is to identify a meaningful attribute from contextual information. This encompasses a myriad of useful research applications, including the identification of pedestrians for self-driving cars [1] or misinformation in social media networks [2]. While modern technology engenders large datasets, unsolved ML classification problems require human identification of ground-truth labels. This may be slow and costly. Such issues motivate Semi-Supervised Learning (SSL), where we aim to use small number of ground-truth labelled examples, as well as a larger set of unlabelled examples, to achieve the ultimate objective of class prediction. One useful application of SSL is to develop medical diagnostic models which currently necessitate finite expert knowledge [3], [4]. In a different context, one recent work employs SSL methods for the detection of fake online review data [5].

Given some label Y, with observable feature variables X, the goal of semi-supervised classification is to estimate $P(Y|X)$ from both labelled and unlabelled data [6]. A broad range of approaches for semi-supervised learning exist [7], [8], encompassing discriminative and generative methods. Discriminative approaches use only the unlabelled empirical sample $P(X)$ with regularisation heuristics which link $P(X)$ to $P(Y|X)$, and avoid modelling $P(X)$ directly [6]. Such models are more parsimonious and require fewer assumptions. In contrast, generative approaches model the distribution $P(X)$ or $P(X|Y)$ with a generator G, using synthetic samples drawn from G to improve the discriminator. The rationale follows from a rewriting of the classification probability according to Bayes rule: $P(Y|X) = P(X,Y)/P(X)$. It is worth mentioning that there are a large number of methods specifically designed for image data: for example, by using consistency regularisation under different augmentations of the input images [8].

Despite the success of SSL, the reason why unlabelled data could help is still not fully understood. An interesting direction of SSL research is to understand SSL from a causal perspective. In the seminal work [9], the authors consider two learning scenarios under two possible causal graphs between labels and features, including causal learning $X \to Y$ and anticausal learning $Y \to X$, and conjecture that SSL methods can only be successful in the anticausal scenario with empirical illustrations. This conjecture is proven from an information-theoretical perspective under a class of parametric models in a recent work [10]. [11] considers the scenario where part of the features are causes of the label and the other part is the effect of label and propose two more general SSL approaches based on generative modeling and self-training, respectively. While outcomes along this line are encouraging, the existing methods are limited to relatively simple causal graphs and cannot handle real situations well.

In this paper, we propose a semi-supervised learning framework under general causal models. More specifically, we consider the possible causal graphs one can obtain from real-word applications and divide them into different categories. Under each category, we develop semi-supervised approaches based on causal generative models, i.e., generative models following causal structures, to effectively leverage labelled and unlabelled data for training predictive models. To the best of our knowledge, this is the first compreshesive causal SSL solution that is exhaustive in this respect. In comparative experiments, our method demonstrates superior performance over all models on synthetic and real-world data.

II. Related Works

There is a vast literature on semi-supervised learning. Broadly speaking, we can divide existing SSL methods into three categories: discrminative methods, generative methods, and hybrid methods.

Archer Moore, Heejung Shim, and Mingming Gong are with the School of Mathematics and Statistics, Faculty of Science, The University of Melbourne, Melbourne, Australia (e-mail: a.moore9@student.unimelb.edu.au; hee.shim@unimelb.edu.au; mingming.gong@unimelb.edu.au).

Jingge Zhu is with the Department of Electrical and Electronic Engineering, Faculty of Engineering and Information Technology, The University of Melbourne (email: jingge.zhu@unimelb.edu.au).

Discriminative methods train a predictive model by minimizing the risk on labelled data and adding regularisations built from unlabelled data. Entropy minimisation [12] optimises the decision boundary to lie within a low-density region, and is grounded in the assumption that features in the same cluster should belong to the same class. A more elaborate recent contribution uses an entropy-based clustering approach in the latent space to improve the quality of representations [13]. Pseudo-labelling (also known as label propagation) [14] uses the softmax label estimates of unlabelled data as a regularisation technique. At each training epoch, softmax predictions are compared with hard labels. The optimisation objective encourages the classifier to make predictions that are closer to the hard labels. Meta Pseudo-Labelling (MPL) [15] is an extension which uses a Student-Teacher architecture. Student-Teacher architectures consist of a Teacher discriminator which learns the decision function $P(Y|X)$, and a Student model which learns to emulate the decision predictions of the Teacher. Some further methods are Co-Training [16], which trains on complementary aspects of the data, and TriNET [17], which forms consensus among three pseudo-labelling models. In general, these methods suffer from confirmation-bias, as models are incentivised to produce consistent predictions over successive training epochs, and minimise uncertainty in ambiguous cases [18].

Consistency Regularisation methods assume that predictions on unlabeled data should be consistent over different discriminative models or a model on different augmentations of input X. VAT [19] extends the adversarial framework of GAN. For a perturbation γ in the most vulnerable direction within a small radius around a data point $x \in X$, the model encourages identical predictions for $P(Y|x)$ and $P(Y|x+\gamma)$. Given that the ground-truth labels are never observed for unlabelled data, the loss is denoted as 'virtual'. Further approaches abound: VAdD [20] perturbs network weights and changes network structure to increase consistency of predictions that are vulnerable to adversarial attack. Stochastic Weight Averaging [21] considers various schemes to average weights at differing points during training. UWA is a more exhaustive approach to text and image augmentation [22]. More complicated methods such as the Π model [23] enforce consistency regularisation across differing augmentations of identical data, dropout in neural network layers, or random max pooling. There exist an extensive suite of approaches which enforce consistencies across multiple networks: Dual Student [24], Temporal Ensembling [25] and Mean Teacher [26].

Generative methods typically model the marginal distribution $P(X)$ by a generative model and make use of it to improve the prediction performance. Among the generative methods, GAN architectures [27] have shown considerable success in SSL. The adversarial framework employs a discriminator $D : \mathcal{X} \to d, 0 \leq d \leq 1$, which is a judge of how accurately G models $P(X)$. SGAN/Improved GAN is the most direct implementation for SSL, using D as both a judge of G, and an auxiliary classifier over k classes [28]. Fake data is mapped to some extra hypothetical class $k+1$. Triple GAN is an alteration to SGAN which separates the discriminator into a real/fake discriminator and an auxiliary classifier for the class [29]. There are many SSL models employing simliar frameworks: GoodBadGAN [30], Localised GAN [31], BiGAN [32], Triangle GAN [33], Structured GAN [34], and others [7], [8]. These extensions exploit specific assumptions of the data generating process, such as semantic segmentation, local feature maps, latent feature maps, or domain shift: these may be employed where appropriate for specific datasets.

A different type of generative model implements the Variational Autoencoder (VAE) architecture for use in SSL [35], [36]. As per regular VAE, the observational distribution is a function of a set of low-dimensional, independent, normally-distributed latent factors Z. VAE methods exploit an identical parameterisation of the latent space for both labelled and unlabelled data. In their original paper, the authors explored a variety of architectures: the most effective model, which we refer to as SSVAE (commonly referred to as M1+M2 in the literature), combines two VAE networks M1 and M2. M1 learns a representation of latent variables Z_1 of the marginal distribution $P(X)$ only. M2 then learns a representation of some latent Z_2 which depends on the both the label Y and Z_1. The marginal probability $P(Z_2|Z_1)$ is used to infer $P(Y|X)$ for unlabelled X. ReVAE [37] and ADGM [38] extend SSVAE by introducing auxiliary variables. Infinite VAE [39] combines multiple individual VAE models. Disentangled VAE [40] and SVAE [41] enforce stricter assumptions on the data structure. In comparison to SSVAE (M1+M2) and ADGM, SeGMA [42] is a recent autoencoder approach which demonstrates superior consistency under style mixing and interpolation between classes.

The existing generative models either directly model $P(X)$ or $P(Y|X)$, which implicit assume that the underlying causal generative process follows $Y \to X$. While these methods work well for image data, they do not work well if the underlying causal model is not $Y \to X$. Therefore, in the current work we wish to explore the effect of a causal structure on the performance of these models, and consider learning under general causal generative models.

Hybrid Methods combine elements of methods previously discussed. A central feature of most of these approaches is Mixup [43], which interpolates synthetic samples from labelled pairs indexed by a convex combination of both features and labels. ICT [44] implements Mixup consistency regularisation in a student-teacher framework. Recently, ICL-SSL [45] explores an interpolation scheme which preserves semantic consistency. Mixup has also been combined with meta-learning for considerable SSL performance [46]. Mixmatch [47], ReMixMatch [48] and Fixmatch [49] comprise a similar vein of work combining elements of data augmentation, consistency regularisation and/or pseudo-labelling. These particular methods are applicable to image data, and are not used for comparison in our current work. Broadly, these models find success in implementing various aspects of consistency regularisation to maximise discriminative confidence, while also incorporating assumptions about the data generating process.

-

III. Problem setup and notation

Let two random variables X and Y denote the features and the labels, respectively. In semi-supervised learning, we have labelled examples $\{(x_1, y_1), \ldots, (x_l, y_l)\} \in \mathcal{X} \times \mathcal{Y}$ drawn from an unknown joint distribution $P(X, Y)$ and unlabelled examples $\{x_{l+1}, \ldots, x_{l+u}\} \in \mathcal{X}$ drawn from the marginal distribution $P(X)$. The labelled examples are usually much less than unlabelled examples, i.e., $l << u$. The goal of semi-supervised learning is to learn a predictive function f from both labelled and unlabelled data, with the expectation that f performs better than the function learned from only labelled data. In this paper, we consider the classification problem with $\mathcal{Y} = \{1, \ldots, K\}$ and $\mathcal{X} = \mathbb{R}^d$.

IV. Preliminaries

A. Causal Models

Directed Acyclic Graph (DAG). We require a framework for articulating causal relationships between all variables in $V = \{X, Y\}$. Following Pearl's framework [50], we use the Directed Acyclic Graph (DAG) to represent causal interactions. A DAG is a directed graph (DiGraph) which is acyclic. A DiGraph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ is a pair over vertices $\mathcal{V}$ and edges $\mathcal{E}$, where each edge $e \in \mathcal{E}$ is an ordered tuple (v_i, v_k) over two distinct vertices $v_i, v_k \in \mathcal{V}$. The direction is visually depicted using an arrowhead. A path is an ordered collection of edges which share common source/target vertices. If the DiGraph contains edges $e_x = (v_i, v_k), e_{x'} = (v_k, v_m)$, we say that there is a path from v_i to v_m. A DiGraph is acyclic if there exist no paths which start and end at the same vertex. If there is an edge from v_i to v_k, we say that v_i is a parent of v_k. We denote all parents of v_k as $\mathbf{Pa}_{v_k}$. If v_k has no parents, then $\mathbf{Pa}_{v_k} = \{\emptyset\}$ and v_k is called a root node. Considering the correspondence between vertices $v_i \in \mathcal{V}$ and variables in $V = \{X, Y\}$, we use v_i to interchangeably refer to a variable in V, or its corresponding vertex in $\mathcal{G}$.

The DAG can be used as a representation of causal mechanisms between all variables $v_i \in \{X, Y\}$, and the causal mechanism of v_i is represented by the expression $P(v_i|\mathbf{Pa}_{v_i})$. Using a causal framework, we seek an understanding of how the observational distribution $P(X, Y)$ may arise from a hierarchy of causal mechanisms. In addition to assuming a DAG representation, we require further assumptions on the data generating process: these are encompassed by Causal Bayesian Networks and Structural Causal Models.

Causal Bayesian Networks. For some DAG $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, let each vertex $v_i \in \mathcal{V}$ correspond to a particular variable $v_i \in V = \{X, Y\}$ and let P be a joint probability distribution over V. The pair $\langle \mathcal{G}, P \rangle$ is a causal Bayesian network if the Causal Markov condition and Modularity condition hold. The **Causal Markov condition** implies that the joint distribution $P(V)$ can be factorised into the product of each $v_i \in V$ conditional on its parents $\mathbf{Pa}_{v_i}$:

$$P(V) = \prod_{i=1}^{d} P(v_i|\mathbf{Pa}_{v_i}). \tag{1}$$

The DAG entails that if $\mathbf{Pa}_{v_i}$ is a cause of v_i, then an intervention on $\mathbf{Pa}_{v_i}$ should affect the value of v_i. In contrast, an intervention on v_i does not imply a change in $\mathbf{Pa}_{v_i}$. This is the *Modularity principle* [50]: the causal mechanism of unintervened variables is invariant under intervention. While Causal Bayesian Networks are an elegant representation of a joint probability, they may suffer from identifiabiliity guarantees as, in general, it can be difficult to confirm or deny a unique causal DAG $\mathcal{G}$ for some P. In this project, we avoid this limitation by assuming a prior knowledge of the DAG.

Structural Causal Model (SCM) is an alternative way to formalise a model of the causal mechanism for each $v_i \in V$. The SCM consists of a tuple $\langle S, P \rangle$ where $S = (S_1, \ldots, S_d)$:

$$S_i := f_i(\mathbf{Pa}_{X_i}, N_i) \qquad i = 1, ..., d, \tag{2}$$

where each S_i corresponds to each $v_i \in \{V\}$. N_i is an approximation of all possible external causes of v_i, including errors in measurement. Writing $N = \{N_1, \ldots, N_d\}$, all N_i are mutually independent:

$$P(N) = \prod_{i=1}^{d} P(N_i). \tag{3}$$

In the SCM framework, f_i describes a deterministic relationship between v_i and its parents. This is grounded in the assumption that the causal mechanism is an interaction of physical phenomena, and is therefore encompassed by some set of deterministic set of physical laws [50]. f_i is a function of the random variable N_i, implying that S_i is also a random variable. However, f_i itself does not change, as the laws of physics are not mutable. This is known as the *Independent Causal Mechanisms* principle. The ensemble of causal mechanisms described by each S_i is a model of $P(v_i|\mathbf{Pa}_{v_i})$, and the hierarchical collection of all S_i describes the generative process of how i.i.d. samples of $P(X, Y)$ arise.

B. Markov Blanket

While previous discussion illustrated the relevance of causality for modelling $P(X, Y)$, the causal DAG can also inform which features are relevant for estimating $P(Y|X)$. The classification task can be simplified by identifying and using features in the Markov Blanket of Y, written as $X_{\mathbf{MB}}$. The Markov Blanket of Y is the set of features $X_{\mathbf{MB}} \in X$ s.t. $Y \perp\!\!\!\perp X \setminus X_{\mathbf{MB}} | X_{\mathbf{MB}}$. Intuitively, this means that given information contained in the features in $X_{\mathbf{MB}}$, Y is conditionally independent of all other features. Considering our classification objective is to estimate $P(Y|X)$, the following equivalence should therefore hold:

$$P(Y|X) = P(Y|X_{\mathbf{MB}}), \tag{4}$$

where $X_{\mathbf{MB}}$ contains features X which are either parents X_C, spouses X_S or children X_E of Y. The associated causal relationships are depicted in Table I.

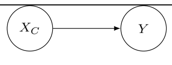
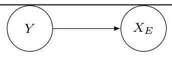
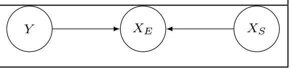
Terminology	Notation	Example
Direct cause / parent of Y	X_C	
Direct effect / child of Y	X_E	
Spouse of Y	X_S	

TABLE I: Parents, children and spouses in $X_{\textbf{MB}}$

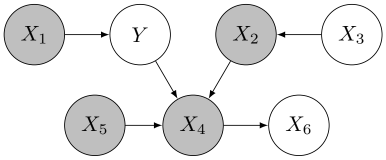


Fig. 1: Markov Blanket $X_{\textbf{MB}} = \{X_1, X_5, X_4, X_2\}$

In Figure 1, we illustrate an example where features in $X_{\textbf{MB}}$ are shaded grey, and $X_{\textbf{MB}} = \{X_1, X_5, X_4, X_2\}$, since X_1 is a direct cause (parent) of Y, X_4 is a direct effect (child) of Y and X_2, X_5 are both spouses of Y. In contrast, X_3 and X_6 are neither parents, spouses or children of Y, and hence are not in $X_{\textbf{MB}}$. For a classification task, observations of X_1, X_2, X_4, X_5 are sufficient for estimating $P(Y|X)$: we gain no extra information from X_3, X_6 when all features $X \in X_{\textbf{MB}}$ are observed.

V. Causal Semi-Supervised Learning using $X_{\textbf{MB}}$

Our goal is to exploit the causal structure underlying $P(X, Y)$ for semi-supervised learning. This task is simplified considering Equation 4, as the Markov Blanket contains all useful information for estimating $P(Y|X)$. In the current section, we analyse the plausibility of SSL over any Markov Blanket structure. Such results extend naturally to any DAG, providing a unified framework for causal SSL. A key conjecture of previous works is that for data $P(X, Y)$, unlabelled samples could facilitate SSL if the causal structure is $Y \rightarrow X$, but not if $X \rightarrow Y$ [9], [51], [52], with the relationship proven analytically for a class of parametric models [10]. Since such results account for Markov Blankets containing either X_C or X_E only, we expand the analysis to consider how more features could be used. To gain some intuition for different types of Markov Blanket structures, consider the DAG in Figure 2, where $X_S, X_C, X_E \in X_{\textbf{MB}}$.

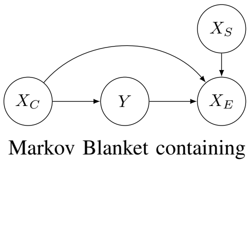
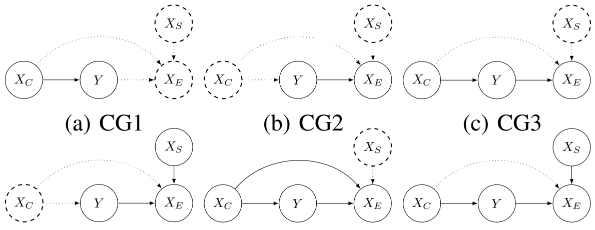


Fig. 2: CG6: Markov Blanket containing X_S, X_C, X_E



(a) CG1 (b) CG2 (c) CG3

(d) CG4 (e) CG5 (f) CG7

Fig. 3: Markov Blanket Subgraphs of CG6

There are 6 subgraphs of Figure 2 which are also a Markov Blanket over Y. Each variation is depicted in Figure 3, with dotted elements excluded. If we group all variations based on whether $X_C, X_E, X_S \in X_{\textbf{MB}}$, we see that there are five possible cases. Using notation $X^1_{\textbf{MB}}, \ldots, X^5_{\textbf{MB}}$ for each case, we summarise these in Table II. For example, $X^1_{\textbf{MB}}$ is a Markov Blanket which contains direct causes X_C of Y, no spouses X_S and no children X_E. Clearly then, CG1 corresponds to $X^1_{\textbf{MB}}$. In contrast, $X^5_{\textbf{MB}}$ is a Markov Blanket containing parents X_C, children X_E and spouses X_S. This topology applies to CG6, CG7, and Figure 1.

$X_{\textbf{MB}}$	$X_C \in X_{\textbf{MB}}$	$X_E \in X_{\textbf{MB}}$	$X_S \in X_{\textbf{MB}}$	**DAG**
$X^1_{\textbf{MB}}$	yes	no	no	CG1
$X^2_{\textbf{MB}}$	no	yes	no	CG2
$X^3_{\textbf{MB}}$	no	yes	yes	CG4
$X^4_{\textbf{MB}}$	yes	yes	no	CG3, CG5
$X^5_{\textbf{MB}}$	yes	yes	yes	CG6, CG7

TABLE II: Possible $X_{\textbf{MB}}$ topologies

Although in general the structure of a given Markov Blanket may be very complicated, any Markov Blanket will correspond to one of the cases as given in Table II. Therefore, if we are able to elucidate the utility of unlabelled data in each case, we provide a unified framework for whether unlabelled data could help over any causal structure. The proposed grouping allows us to make arguments based on a causal factorisation over the DAG. As such, each $X^1_{\textbf{MB}}, \ldots, X^5_{\textbf{MB}}$ should inherit the same theoretical guarantees. We now discuss each case, starting from $X^1_{\textbf{MB}}$.

$X^1_{\textbf{MB}}$. Recalling that we seek to use the unlabelled data distribution, $P(X_C)$, to improve our estimate of $P(Y|X_C)$. $X^1_{\textbf{MB}}$ corresponds to CG1 in Table 3, which has been studied in previous works [9], [10]. The causal factorisation of the the joint distribution is $P(X_C, Y) = P(Y|X_C)P(X_C)$. According to the independent causal mechanism, $P(Y|X_C)$ and $P(X_C)$ do not contain information about each other, i.e., algorithmically independent [53]. As a result, a better estimation of $P(X_C)$ from unlabeled data could not benefit the estimation of $P(Y|X_C)$.

$X^2_{\textbf{MB}}$. This scenario is depicted in CG2 of Table 3, which is a widely studied scenario [9], [35] and it is often applied to image data [35], [54]. The causal factorisation of the data generating process is: $P(X_E, Y) = P(X_E|Y)P(Y)$.

$P(X_E|Y)$ and $P(Y)$ are independent causal mechanisms, i.e., they do not contain information about each other. In contrast, in the anticausal direction, $P(X_E)$ and $P(Y|X_E)$ contain information about each other. This is because $P(X_E)$ is associated with both $P(X_E|Y)$ and $P(Y)$, which induces the labelling function $P(Y|X_E)$.

$X^3_{\mathbf{MB}}$. In this Markov Blanket shown in CG4 of Table 3 , the task is to improve $P(Y|X_E, X_S)$ using the unlabelled data in either X_E, X_S. This case has some familiar structure to $Y \rightarrow X_E$, but with an additional casual variable of X_S. One can compute $P(Y|X_E, X_S)$ as follows:

$$\begin{aligned}P(Y|X_E, X_S) &= \frac{P(Y, X_E, X_S)}{P(X_E, X_S)} \\ &= \frac{P(X_E|Y, X_S)P(Y)P(X_S)}{\int_Y P(X_E|Y, X_S)P(Y)P(X_S)dY} \\ &= \frac{P(X_E|Y, X_S)P(Y)}{\int_Y P(X_E|Y, X_S)P(Y)dY}.\end{aligned}$$

Obviously, the final predictive distribution $P(Y|X_E, X_S)$ and the marginal distribution $P(X_S)$ contain no information about each other. Thus, the unlabeled data of X_S alone is not beneficial for learning the predictive model. By contrast, the conditional distribution $P(X_E|X_S) = \int_Y P(X_E|Y, X_S)P(Y)dY$ contains information about $P(X_E|Y, X_S)$ and $P(Y)$, which induce the predictive distribution $P(Y|X_E, X_S)$. Therefore, semi-supervised learning would be possible by using unlabeled data to better estimate $P(X_E|X_S)$, which could improve the estimation of $P(Y|X_E, X_S)$.

$X^4_{\mathbf{MB}}$. Two topologies are depicted as CG3 and CG5 in Table 3. CG5 is analysed in [11], which argues that the clusters $P(X_E|X_C)$ should correspond to different parameterisations based on Y. Here we generalize the results to make use of the whole data generating process. More specifically, one can write the predictive distribution as

$$P(Y|X_E, X_C) = \frac{P(X_E|Y, X_C)P(Y|X_C)}{\int_Y P(X_E|Y, X_C)P(Y|X_C)dY}.$$

It can be seen that $P(X_E|X_C) = \int_Y P(X_E|Y, X_C)P(Y|X_C)dY$ contains information about $P(X_E|Y, X_C)$ and $P(Y|X_C)$, which induce the predictive distribution $P(Y|X_E, X_C)$. In our method section, we will present our algorithms that learn $P(X_E|Y, X_C)$ and $P(Y|X_C)$ with the help of unlabeled data.

As a special case of CG5, CG3 removes the confounding path from X_C to X_E. CG3 still benefits from a better estimation of $P(X_E|X_C)$ using unlabeled data, but one can also utilize unlabeled data to estimate $P(X_E)$ alone so as to improve the learning of $P(X_E|Y)$ as in CG2, due to the absence of confounding.

$X^5_{\mathbf{MB}}$. In this Markov Blanket, we aim to predict Y using X_C, X_S and X_E: hence the objective is to estimate $P(Y|X_E, X_S, X_C)$. We note that all features X_C, X_E, X_S must be used together. If X_E is parameterised by X_S, Y, X_C, as depicted in CG6 of Figure 2, we can write the predictive distribution as:

$$\begin{aligned}P(Y|X_E, X_S, X_C) &= \frac{P(Y, X_E, X_S, X_C)}{P(X_E, X_S, X_C)} \\ &= \frac{P(X_E|Y, X_S, X_C)P(Y|X_C)}{\int_Y P(X_E|Y, X_S, X_C)P(Y|X_C)dY}.\end{aligned} \tag{5}$$

Here, the conditional distribution $P(X_E|X_S, X_C) = \int_Y P(X_E|Y, X_S, X_C)P(Y|X_C)dY$ contains information about $P(Y|X_C)$ and $P(X_E|Y, X_S, X_C)$, which induce the predictive distribution $P(Y|X_E, X_S, X_C)$. Thus, we can use unlabelled data to better estimate $P(X_E|X_S, X_C)$, which could lead to a more accurate estimate of $P(X_E|Y, X_S, X_C)$ and $P(Y|X_C)$. As a result, the predictive distribution $P(Y|X_E, X_S, X_C)$ induced from $P(X_E|Y, X_S, X_C)$ and $P(Y|X_C)$ can be better estimated.

CG7 in Figure 3 could be considered similarly to CG4, since $P(X_E|X_S) = \int_Y P(X_E|Y, X_S)P(Y)dY$. These terms could be used to induce the predictive distribution $P(Y|X_S, X_E)$ without confounding from X_C. Alternatively, we can also use X_C by considering the conditional probability $P(X_E|X_S, X_C) = \int_Y P(X_E|Y, X_S)P(Y|X_C)dY$. Since it contains info about $P(X_E|Y, X_S)$ and $P(Y|X_C)$, this could be used to induce the predictive disribution $P(Y|X_S, X_E, X_C)$. We should expect CG7 to benefit from an improved estimate of $P(X_E|X_S, X_C)$ in the latter case.

VI. Method for Causal SSL

Method Overview. While in theory causal relationships may improve a semi-supervised model, it is not obvious how one is supposed to exploit causal information in a practical modelling context. Our goal is to create a method which could be used to benefit from unlabelled data over any causal structure. Recalling that we assume $P(X, Y) = \prod_{i=1}^{d} P(v_i|\mathbf{Pa}_{v_i})$, we produce a generative model $G := \hat{P}(X, Y)$ by creating a structural model for each $P(v_i|\mathbf{Pa}_{v_i})$ separately. We then synthesise extra data $(x', y') \sim G$, which augments the original labelled sample $(x, y) \in D_l$, and the augmented sample is used to train a classifier in a fully-supervised regime. The motivation for a generative approach is to encode the ICM assumption, and thus the flow of causal information, into the generated data. Our method modifies existing nonparametric approaches to SCM modelling [55] in order to estimate G under missing labels.

Modelling each $P(v_i|\mathbf{Pa}_{v_i})$. By fully factorising $P(X, Y) = \prod P(v_i|\mathbf{Pa}_{v_i})$ according to the Causal Markov condition in Equation 1, we identify the factors for structural modelling. In CG4, for example, we can see that there are three factors:

$$P(X, Y) = \underbrace{P(X_E|Y, X_S)}_{\text{Factor 1}}\underbrace{P(Y)}_{\text{Factor 2}}\underbrace{P(X_S)}_{\text{Factor 3}}.$$

It is impractical to use an identical estimation procedure for each $P(v_i|\mathbf{Pa}_{v_i})$, since some factors may need to be estimated in the presence of missing label data, and our choice of loss function depends on whether $v_i \in X$ or $v_i \in Y$. In CG4, $P(X_E|Y, X_S)$ must be estimated in the presence of missing labels, but $P(X_S)$ does not suffer from the same issue. Similarly, our model of $P(Y)$ maps to a label, while $P(X_S)$ maps to a feature. Each factor requires a different approach. To determine the appropriate estimation procedure, we identify some structure in $P(v_i|\mathbf{Pa}_{v_i})$ according to whether v_i is a feature, and whether $\mathbf{Pa}_{v_i}$ contains any features, any labels, or is empty. There are two sets of approaches we will describe: disjoint, and joint.

Modelling Approach I: Disjoint. Under our disjoint approach, we focus on identifying structure in each $P(v_i|\mathbf{Pa}_{v_i})$ separately. Our set of rules characterises any single $P(v_i|\mathbf{Pa}_{v_i})$ as belonging to five separate scenarios, **A**,**B**,**C**,**D**,**E**, listed in Table III.

Scenario	**Description**	v_i	$P(v_i\|\mathbf{Pa}_{v_i})$
A	Y is a root node, $\mathbf{Pa}_Y = \{\varnothing\}$	Y	P(Y)
B	Y is not a root node, $\mathbf{Pa}_Y \neq \{\varnothing\}$	Y	$P(Y\|X_K,\dots)$
C	X_I is a root node, $\mathbf{Pa}_{X_I} = \{\varnothing\}$	X_I	$P(X_I)$
D	$X_K \in \mathbf{Pa}_{X_I}$ and $Y \notin \mathbf{Pa}_{X_I}$	X_I	$P(X_I\|X_K,\dots)$
E	$Y \in \mathbf{Pa}_{X_I}$	X_I	$P(X_I\|Y,\dots)$

TABLE III: Disjoint modelling approach for $P(v_i|\mathbf{Pa}_{v_i})$

To use this table, we could either match the form of $P(v_i|\mathbf{Pa}_{v_i})$ to an appropriate expression in the fourth column, or use the **Description** column. For example, if $P(v_i|\mathbf{Pa}_{v_i}) = P(Y)$, then this form corresponds to scenario **A**. In contrast, $P(v_i|\mathbf{Pa}_{v_i}) = P(Y|X_C)$ corresponds to scenario **B**. We use notation $P(v_1|v_2,\dots)$ to indicate that $\mathbf{Pa}_{v_1}$ must contain v_2, and possibly extra variables. $P(X_1|Y)$, $P(X_1|Y,X_2)$ and $P(X_1|Y,X_2,X_3)$ all conform to the pattern $P(X_I|Y,\dots)$, and therefore they are all identified as scenario **E**: even though the conditional information is different in each case, all factors depend on Y. In contrast, $P(X_1)$ and $P(X_1|X_2)$ cannot conform to this pattern, and are identified as scenario **C** and **D** respectively. This structural identification process is used for Markov Blankets CG1-CG7 given earlier. For CG1, the causal factorisation $P(X,Y) = P(Y|X_C)P(X_C)$ contains two causal modules $P(Y|X_C)$ and $P(X_C)$. From Table III, it is clear that $P(X_C)$ is scenario **C** and $P(Y|X_C)$ is scenario **B**. For CG1-CG7, all factorised modules and scenarios are listed in Figure 4.

Graph	Module	Scenario
CG1	$P(X_C)$	C
	$P(Y\|X_C)$	B
CG2	$P(Y)$	A
	$P(X_E\|Y)$	E
CG3	$P(X_C)$	C
	$P(Y\|X_C)$	B
	$P(X_E\|Y)$	E
CG4	$P(Y)$	A
	$P(X_S)$	C
	$P(X_E\|Y,X_S)$	E

Graph	Module	Scenario
CG5	$P(X_C)$	C
	$P(Y\|X_C)$	B
	$P(X_E\|Y,X_C)$	E
CG6	$P(X_C)$	C
	$P(X_S)$	C
	$P(Y\|X_C)$	B
	$P(X_E\|Y,X_C)$	E
CG7	$P(X_C)$	C
	$P(X_S)$	C
	$P(Y\|X_C)$	B
	$P(X_E\|Y,X_C,X_S)$	E

Fig. 4: Disjoint approach: factors and scenarios for CG1-CG7

Modelling Approach II: Joint. The scenarios given in Table III may be used to characterise any $P(v_i|\mathbf{Pa}_{v_i})$, and hence determine a method for structural modelling of each factor separately. In contrast, in the joint approach, we identify some shared structure between factors, and these factors are modelled together. Specifically, if **B** and **E** both occur in the same factorised $P(X,Y)$, our procedure is modified. While the reasons for this will be explained later, for now we are merely focused on identifying the structures. In this special case, the product $P(X_E|Y,\dots)P(Y|X_C,\dots)$ must be present. This scenario is referred to as **F**, and entails a different method which replaces the methods of **B** and **E**. In Table IV, we give an updated set of rules: note that scenarios **A**, **C**, **D** are unchanged. For CG7, the structures identified by our disjoint approach are given in Equation 6:

$$P(X,Y) = \underbrace{P(X_E|Y,X_S)}_{\text{E}}\underbrace{P(Y|X_C)}_{\text{B}}\underbrace{P(X_S)}_{\text{C}}\underbrace{P(X_C)}_{\text{C}}. \tag{6}$$

In contrast, the scenarios identified by the joint approach are depicted in Equation 7. This illustrates that the joint identification scheme is trivially different from the disjoint identification scheme trivial in practice, as it amounts to merely recategorising scenarios **B** and **E** as scenario **F**.

$$P(X,Y) = \underbrace{P(X_E|Y,X_S)P(Y|X_C)}_{\text{F}}\underbrace{P(X_S)}_{\text{C}}\underbrace{P(X_C)}_{\text{C}}. \tag{7}$$

Scenario	**Description**	$P(v_i\|\mathbf{Pa}_{v_i})$
A	Y is a root node, $\mathbf{Pa}_Y = \{\varnothing\}$	P(Y)
C	X_I is a root node, $\mathbf{Pa}_{X_I} = \{\varnothing\}$	$P(X_I)$
D	$X_K \in \mathbf{Pa}_{X_I}$ and $Y \notin \mathbf{Pa}_{X_I}$	$P(X_I\|X_K,\dots)$
F, **(Special Case)**	$Y \in \mathbf{Pa}_{X_I}$ and $\mathbf{Pa}_Y \neq \{\varnothing\}$	$P(X_I\|Y,\dots)P(Y\|\dots)$

TABLE IV: Joint modelling approach for $P(v_i|\mathbf{Pa}_{v_i})$

We show how this changes earlier classifications for CG1-CG7 in Table 4. CG1, CG2 and CG4 are unchanged by this new rule, since they do not contain the special case. Structures for CG3, CG5, CG6 and CG7 are reidentified in Table V.

Graph	Module	Scenario
CG3	$P(X_C)$	C
	$P(X_E\|Y)P(Y\|X_C)$	F
CG5	$P(X_C)$	C
	$P(X_E\|Y,X_C)P(Y\|X_C)$	F
CG6	$P(X_C)$	C
	$P(X_S)$	C
	$P(X_E\|Y,X_C,X_S)P(Y\|X_C)$	F
CG7	$P(X_C)$	C
	$P(X_S)$	C
	$P(X_E\|Y,X_S)P(Y\|X_C)$	F

TABLE V: Joint approach: factors and scenarios for CG3, CG5, CG6, CG7

Structural Modelling of features in scenario C, D, E, F.

For $v_i \in X$, $P(v_i|\mathbf{Pa}_{v_i})$ may be modelled as SCM $f_{\theta_{v_i}}$, and $f_{\theta_{v_i}}$ is some universal function approximator, such as a feed forward neural network, with parameters θ_{v_i}. Where $f_{\theta_{v_i}} := \hat{p}(v_i)$, and $p(v_i)$ is the empirical sample of v_i, we use the Maximum Mean Discrepancy (MMD) [56] to match the estimate $\hat{p}(v_i) \overset{i.i.d.}{\sim} p(v_i)$. For some feature map $\Phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a Reproducing Kernel Hilbert Space (RKHS), MMD computes the mean distance between feature embeddings of distributions P and Q:

$$\mathbf{MMD}(P \parallel Q) = \|\mathbb{E}_{X \sim P}[\Phi(X)] - \mathbb{E}_{Y \sim Q}[\Phi(Y)]\|_{\mathcal{H}}. \quad (8)$$

If $\mathbf{MMD}(P \parallel Q) = 0$, the distributions P, Q are identical. A kernel k is a function that computes the dot product of X, X' in feature space: $k(x, x') = \Phi(x)^T \Phi(x')$. For empirical samples $x \in X, x' \in X'$, we can derive an estimate $\hat{\mathbf{MMD}}(X \parallel X')$ by computing $k(x, x')$ for an RKHS kernel such as the RBF kernel k_{σ_x}. For bandwidth $\sigma_x \in \mathbb{R}^+$,

$$k_{\sigma_x}(x, x') = \exp\left[\frac{-\|x - x'\|^2}{2\sigma_x}\right]. \quad (9)$$

We now move to a detailed explanation of the estimation procedure in scenarios **A**-**F**. We employ notations given in Table VI.

Notation	Meaning
$P(X_A)$	Distribution of random variable X_A
$P(X_1) \overset{\mathrm{d}}{=} P(X_2)$	Distributions of random variables X_1, X_2 match
$\mathbf{MMD}(P \parallel Q)$	MMD between distributions P, Q
$p(x_a)$	Empirical samples x_a from distribution $P(X_A)$
$x_a \in D_u$	Empirical samples $x_a \in X_A$ in unlabelled sample D_u
$\hat{\mathbf{BCE}}(p \parallel q)$	Sample Binary Cross Entropy [6] between realisations from P, Q
$\hat{\mathbf{MMD}}(p \parallel q)$	Sample MMD between realisations from P, Q

TABLE VI: Notations used in estimation procedures, scenario **A**-**F**

Scenario A: $P(v_i|\mathbf{Pa}_{v_i}) = P(Y)$
The factor $P(Y)$ corresponds to a root node in the DAG, and we consider $P(Y)$ as a random variable, rather than SCM. For $Y \in \{1, 2, \ldots, K\}$, where y_i counts the number of occurrences of value y_i in n observations, and π_i is the probability that y_i occurs in a single observation, Y is modelled using the multinomial distribution with PMF:

$$P(Y|\pi, n) = n! \prod_{i=1}^{K} \frac{\pi_i^{y_i}}{y_i!} \quad (10)$$

The labelled sample, consisting of n_l observations, may be used to estimate $\hat{\pi}_i = \frac{y_i}{n_l}$ via maximum likelihood. We can then draw samples from a categorical distribution with parameters $\pi = (\pi_1, \pi_2, \ldots, \pi_K)$. Intuitively, for two classes this reduces to a Bernoulli random variable.

Scenario B: $P(v_i|\mathbf{Pa}_{v_i}) = P(Y|X_K, \ldots)$
For brevity, we write $X_C = \{X : X \in \mathbf{Pa}_Y\}$, so that $P(Y|X_K, \ldots) = P(Y|X_C)$. Denote $f_{\theta_Y} := \hat{P}(Y|X_C)$. We fit Y against X_C from the labelled pairs. As such, this is a fully-supervised learning problem. If there are two classes, we use the Binary Cross Entropy loss over the empirical sample, employing the following objective for update of θ_Y:

$$\theta_Y = \underset{\theta_Y}{\arg\min} \underset{(x_c, y) \in \{D_l\}}{\hat{\mathbf{BCE}}} \Big[y \parallel f_{\theta_Y}(x_c)\Big].$$

Scenario C: $P(v_i|\mathbf{Pa}_{v_i}) = P(X_I)$
Denoting our model $f_{\theta_{X_i}} := \hat{P}(X_I)$, our theoretical intuition is that if the MMD distance between the true distribution $P(X_I)$ and our model $\hat{P}(X_I)$ is zero, then $P(X_I) \overset{\mathrm{d}}{=} \hat{P}(X_I)$. Therefore, our goal is to find the model parameters θ_{X_i} of $f_{\theta_{X_i}}$ which minimise the sample MMD between the empirical distribution of X_I, and samples estimated from $f_{\theta_{X_i}}$. Following the SCM framework, $f_{\theta_{X_i}}$ is a function of the independent noise term N_{X_i} only. Since the causal mechanism has no dependence on a latent Y, all samples are observed in labelled and unlabelled data. This provides the following objective:

$$\theta_{X_i} = \underset{\theta_{X_i}}{\arg\min} \underset{x_i \in \{D_l, D_u\}}{\hat{\mathbf{MMD}}} \Big[p(x_i) \parallel f_{\theta_{X_i}}(N_{X_i})\Big].$$

Scenario D: $P(v_i|\mathbf{Pa}_{v_i}) = P(X_I|X_K, \ldots)$
Using shorthand notation so that $X_C = \{X : X \in \mathbf{Pa}_{X_I}\}$, $P(X_I|X_K, \ldots) = P(X_I|X_C)$. Similar to Scenario C, where the feature X_I is a root node, we do not have to deal with missing labels, $f_{\theta_{X_i}} := \hat{P}(X_I|X_C)$ can be estimated from paired observations (x_i, x_c) in unlabelled and labelled data. $f_{\theta_{X_i}}$ is a function of the noise term N_{X_i} and parent features $X_C \in \mathbf{Pa}_{X_i}$. Since a perfect model of the causal mechanism $f_{\theta_{X_i}} := \hat{P}(X_I|X_C)$ equates to the following,

$$\mathbf{MMD}\Big[P(X_I|X_C)P(X_C) \parallel \hat{P}(X_I|X_C)P(X_C)\Big] = 0,$$

our goal is to find model parameters θ_{X_i} which minimise the sample MMD between the empirical joint distribution $P(X_i, X_c)$ and the causal factorisation $\hat{P}(X_I|X_C)P(X_C)$:

$$\theta_{X_i} = \underset{\theta_{X_i}}{\arg\min} \underset{(x_i, x_c) \in \{D_l, D_u\}}{\hat{\mathbf{MMD}}} \Big[p(x_i, x_c) \parallel f_{\theta_{X_I}}(N_{X_I}, x_c)p(x_c)\Big].$$

Scenario E: $P(v_i|\mathbf{Pa}_{v_i}) = P(X_I|Y, \ldots)$
Since $f_{\theta_{X_i}} := \hat{P}(X_I|Y, \ldots)$ is a function of root node Y, we seek to benefit from unlabelled and labelled samples. First, we discuss the case where there are no features $X_S \in \mathbf{Pa}_{X_i}$, so that $P(v_i|\mathbf{Pa}_{v_i}) = P(X_I|Y)$. The optimisation procedure iterates over minibatches from labelled and unlabelled samples separately. For labelled sample, we use $f_{\theta_{X_i}}$ to estimate $P(X_I|Y)$. If the estimate $\hat{P}(X_I|Y)$ is optimal, then $\mathbf{MMD}\Big[P(X_I|Y)P(Y) \parallel \hat{P}(X_I|Y)P(Y)\Big] = 0$, which suggests the following training objective for labelled data:

$$\theta_{X_i} = \underset{\theta_{X_i}}{\arg\min} \underset{(y, x_i) \in \{D_l\}}{\hat{\mathbf{MMD}}} \Big[p(x_i, y) \parallel f_{\theta_{X_i}}(y, N_{X_i})p(y)\Big].$$

As $f_{\theta_{X_i}}$ must be provided some Y during training, we must modify this for the unlabelled batch: unlabelled sample is used to estimate the marginal $P(X_I)$ from $f_{\theta_{X_i}}$, rather than $P(X_I|Y)$. We create a sample (X_I, Y^*), where $X_I \in D_u$, and Y^* is a bootstrapped sample randomly drawn from labels $Y \in D_l$. As Y^* are randomly drawn with replacement, this conforms to the probabilistic measure which renders Y^* and X_I independent. Comparing the bootstrapped and original sample, we expect the following to hold:

$$\sum_{Y^* \in D_l} P(X_I|Y^*)P(Y^*) \overset{\mathrm{d}}{=} \sum_{Y \in D_l} P(X_I|Y)P(Y).$$

If $\hat{P}(X_I)$ is optimal, we have $\mathbf{MMD}\Big[P(X_I) \parallel \hat{P}(X_I)\Big] = 0, X_I \in D_u$. Hence, we use the following objective function to estimate the marginal $P(X_I)$ from unlabelled data:

$$\theta_{X_i} = \underset{\theta_{X_i}}{\arg\min} \underset{x_i \in D_u, y^* \in D_l}{\hat{\mathbf{MMD}}} \Big[p(x_i) \parallel f_{\theta_{X_i}}(N_{X_i}, y^*)p(y^*)\Big].$$

If we have some $X_S \in \mathbf{Pa}_{X_I}$, then we need to pair each observation with an appropriate sample from X_S. We modify the the labelled objective as follows:

$$\begin{aligned}\theta_{X_i} =& \underset{\theta_{X_i}}{\arg\min} \underset{(y, x_i, x_s) \in \{D_l\}}{\hat{\mathbf{MMD}}} \\ & \Big[p(x_i, y, x_s) \parallel f_{\theta_{X_i}}(N_{X_i}, y, x_s)p(y)p(x_s)\Big].\end{aligned}$$

And we modify the unlabelled objective as follows:

$$\begin{aligned}\theta_{X_i} =& \underset{\theta_{X_i}}{\arg\min} \underset{(x_i, x_s) \in D_u, y^* \in D_l}{\hat{\mathbf{MMD}}} \\ & \Big[p(x_i, x_s) \parallel f_{\theta_{X_i}}(N_{X_i}, y^*, x_s)p(y^*)p(x_s)\Big].\end{aligned}$$

Scenario F: $P(X_I|Y, \ldots)P(Y|\ldots)$
We illustrate this method on the causal structure given in Figure 3, for CG3 $X_C \to Y \to X_I$. For more complicated structures, ie if X_I is a function of any X_S, X_C, we modify the SCM for X_I, $f_{\theta_{X_i}}$, and use appropriate samples from unlabelled and labelled data. The key contribution of this section is to show that if Y is not a root node, and hence the DAG contains some $X_C \to Y$ as well as $Y \to X_I$, it may be advantageous to estimate structural model parameters for X_I, Y together, because Y mediates the relationship between X_C and X_I: this is the motivation for employing Scenario **F** in the joint approach, instead of using scenario **B** and **E** in the disjoint approach. Beginning with the causal factorisation $P(X_C, Y, X_I) = P(X_I|Y)P(Y|X_I)P(X_C)$, denote structural models $g_{\theta_Y}(X_C) := \hat{P}(Y|X_C)$, and $f_{\theta_{X_i}}(N_{X_i}, Y) := \hat{P}(X_I|Y)$. Under the disjoint approach, g_{θ_Y} is modelled as a regression/classification function via scenario **B**, using only labelled samples $\{X_C, Y\} \in D_l$, and $f_{\theta_{X_i}}$ is modelled as scenario **E**. Combining these structural models, we derive an expression for $\hat{P}(X_I|X_C)$:

$$\hat{P}(X_I|X_C) = \int_Y \hat{P}(X_I|Y)\hat{P}(Y|X_C).$$

However, given DAG $X_C \to Y \to X_i$, $\hat{P}(X_I|X_C)$ should depend on Y, and this dependence is not well captured if we optimise g_{θ_Y} and $f_{\theta_{X_i}}$ separately. In this instance, we expect the composite causal mechanism $f \circ g$ to be modelled for data in the labelled dataset only. In scenario **F**, we wish to capture information about Y from the unlabelled (x_i, x_c) pairs. In this case, we use the Gumbel-Softmax [57] trick to train models for f, g together. In our disjoint method, we used bootstrapped y^* to match unlabelled $x_i \in D_u$. In contrast, we now instead sample from the estimate $\hat{P}(y|x_c), x_c \in D_u$ to exploit information in unlabelled (x_i, x_c). Using Gumbel-Softmax, hard-labelled estimates $\hat{y} \sim \hat{P}(Y|X_C)$ may be drawn by sampling $G^i \sim -\log(-\log(\text{Uniform}(0, 1)))$, adding to π_y, which are the normalised outputs from $g_{\theta_Y} := \hat{P}(Y|X_C)$, and finally taking argmax: $\hat{y} = \arg\max\{\theta_y + G^i\}$. By using this estimate, we link the causal mechanism $f \circ g$ for paired unlabelled observations $(X_C, X_I) \in D_u$:

$$\begin{aligned}P(X_I, X_C) =& \int\limits_Y P(X_I|Y)P(Y|X_C)P(X_C)dY, \\ =& P(X_I|X_C)P(X_C), \{X_I, X_C\} \in D_u.\end{aligned}$$

Considering that $\mathbf{MMD}(p \parallel q) = 0$ iff $p \overset{\mathrm{d}}{=} q$, and according to the causal factorisation each factor shares no information, then the disjoint terms must match, and we expect to improve $\hat{P}(Y|X_C)$: $\mathbf{MMD}\Big[P(X_I|Y)P(Y|X_C)||\hat{P}(X_I|Y)\hat{P}(Y|X_C)\Big] = 0$.

Sampling data D_G from our model $G := \hat{P}(X, Y)$. Once we have optimised the structural models of G, we perform ancestor sampling to generate novel data $D_G = (x', y')$ from G. This procedure is straightforward: we generate samples of root node variables from structural models $f_{\theta_v}(N_v)$, and these samples are then used as input to estimate all subsequent variables. We refer to the generated data as D_G. The number of examples drawn from G is the same as the number of unlabelled data, D_u, as recorded in Table VII and Table IX.

Training the classifier. For the classifier $\mathcal{C} : \mathcal{X} \to \mathcal{Y}$, we first train on labelled pairs $(x, y) \in D_l$. We can think of this as 'pre-training'. Then, we augment the sample with D_G and use $D' = D_G \cup D_l$ to perform further training.

VII. Experiments

To develop benchmarks for the improvement from an SSL method, we train a classifier on labelled data only. We call this model the Partial Supervised Classifier, or **P-SUP**. We train a separate classifier using a modified dataset where all of the labels from the unlabelled data are given to the model. We call this model the Fully Supervised Classifier, or **F-SUP**. We expect F-SUP to indicate an upper bound on the performance achievable by any SSL method. For each dataset, we take $n = 100$ examples, and report the average difference in classification accuracy relative to the P-SUP model.

Neural Network Implementation Details. We employ two neural network architectures for our experiments, both using ReLU activation. The first architecture is a 3-layer MLP with hidden layer size 100. This architecture is used in all benchmark models, and in the classifier network for our method. The second architecture is identical except the hidden layer size is 50. This architecture is used for all SCM models. During training, we implement early stopping regulariser if classification accuracy over D_v, the validation partition, does not increase over 10 epochs. Recall that the MMD loss requires a kernel satisfying the RKHS property, in equation 8. K is a mixture of five RBF kernels, as given in Equation 9. To derive all kernels, set σ_x to the median pairwise distance between all points $x \in X_i$. Then the mixture, with each component indexed by $n = [1, \dots, 5]$, and $\sigma_{x_n} = 2^{n-3}\sigma_x$, is given by:

$$K(x, x') = \sum_{n=1}^{5} k_{\sigma_{x_n}}(x, x').$$

Our causal SSL method. We demonstrate two implementations of our method, corresponding to the joint / disjoint modelling approaches. The joint method, which uses the the Gumbel-Softmax to jointly optimise modules under identification of scenario **F**, is denoted **GCGAN-SSL**. The disjoint method, which identifies scenarios **B**, **E** instead of **F**, is denoted **CGAN-SSL**.

Benchmark SSL models for comparison. We report the performance of benchmark SSL models as tabulated in Figure VIII: SSL-GAN [28], Triple-GAN [29], SSL-VAE [41], VAT [19], Entropy Minimisation (ENT-MIN) [12] and Label Propagation / Pseudo-Labelling (L-PROP) [14]. In SSL-VAE, we use a latent embedding of dimension 5. Data analysis was undertaken using The University of Melbourne's Research Computing Services, supported by the Petascale Campus Initiative.

A. Synthetic data

We first demonstrate the utility of our approach on seven different synthetic datasets, each of which corresponds to the causal graphs CG1-CG7, as depicted in Figure 2 and Figure 3. Each dataset conforms to a different Markov Blanket over Y, allowing us to test the ideas proposed in the current work.

How we generate synthetic data. For each DAG, our goal is to generate synthetic data with a nonlinear decision boundary to be used for a semi-supervised binary classification task. Each dataset instance consists of n=2080 cases, which are split into labelled, unlabelled, validation and test partitions. The generative processes for CG1-CG7 are explained sequentially, starting with CG1. **CG1**. For dataset instance i, $X_C \sim N(0, \Sigma_{s_i})$, $\Sigma_{s_i} = \begin{bmatrix} s_i & 0 \\ 0 & s_i^{-1} \end{bmatrix}$, and each s_i is a single sample drawn from $\text{Uniform}(1, 2)$, ie $1 \leq s_i \leq 2$. We use quadratic feature map $\Phi_i(X_C) : \mathbb{R}^2 \rightarrow \mathbb{R}$ to set a decision boundary $P(Y|X)$, where the elements of $\Phi_i(X_C)$ are randomly assigned for each i. Write $X_C = \begin{bmatrix} X_{C_1} \\ X_{C_2} \end{bmatrix}$. Φ_i is determined by 6 scalars a, b, c, d, e, f: $\Phi_i(X_C) = a_i X_{C_1}^2 + b_i X_{C_2}^2 + c_i X_{C_1} + d_i X_{C_2} + e_i X_{C_1} X_{C_2} + f_i$. Set $f_i = 0, \forall i$, and each a_i, b_i, c_i, d_i, e_i is a separate single observation from distribution $u \sim \text{Uniform}(0, 1)$. Now, we use the feature map $\Phi_i(X_C)$ to parameterise a Bernoulli distribution, assigning the label Y from X_C. $\sigma(x_c, \mu_{\Phi_i})$ is a Sigmoid function centered at μ_{Φ_i}: $\sigma(x_c, \mu_{\Phi_i}) = \frac{1}{1+e^{-(\Phi_i(x_c) - \mu_{\Phi_i})}}$. μ_{Φ_i} is the sample mean embedding, ie $\mu_{\Phi_i} = \frac{1}{2080} \sum\limits_{x_c \in X_C} \Phi_i(x_c)$. Then write $p_{\Phi_i}(x_c) = \sigma(x_c, \mu_{\Phi_i})$, and each $y \in Y$ is a single sample from $\text{Bernoulli}(p_{\Phi_i}(x_c))$, corresponding to each $x_c \in X_C$. For our synthetic data, we ideally want to keep $P(Y = 0) \approx P(Y = 1) \approx 0.5$. For each random parameterisation, we synthesise 2080 cases and keep the dataset if $0.45 \leq P(Y) \leq 0.55$. If this is not the case, we discard the dataset and randomly draw new parameters. This procedure is repeated until we have n=100 instances for CG1. **CG2**. Set $Y \sim \text{Bernoulli}(0.5)$, and for each dataset instance i, w_i is a single sample drawn from $\text{Uniform}(4, 6)$, and Σ_{w_i} is defined analogously to Σ_{s_i} for CG1. The causal mechanism f_{X_E} is a nonlinear transform of $N_{X_E} \sim N([0, 0], \mathbb{I}_2)$, resulting in a nonlinear decision boundary between classes [58]. Write $X_E = \begin{bmatrix} X_{E_1} \\ X_{E_2} \end{bmatrix}$. Define a template causal mechanism $f^{\mathbf{T}}_{X_E}(N_{X_E}, Y, A)$, to be used in CG2-CG7, in Equation 11, where $A \in \mathbb{R}$ is an offset in the second dimension:

$$f^{\mathbf{T}}_{X_E}(N_{X_E}, Y, A) = \begin{cases} \Sigma_{w_i} N_{X_E} + \begin{bmatrix} 0 \\ 4\cos\frac{N_{X_{E_2}}}{2} \end{bmatrix}, & Y = 0 \\ \\ \Sigma_{w_i} N_{X_E} + \begin{bmatrix} 0 \\ 4\cos\frac{N_{X_{E_2}}}{2} + A \end{bmatrix}, & Y = 1 \end{cases} \tag{11}$$

Write $f^{\mathbf{CG}X}_{X_E}$ as the causal mechanism of f_{X_E} for dataset **CG**X. Set $f^{\mathbf{CG2}}_{X_E}(N_{X_E}, Y) := f^{\mathbf{T}}_{X_E}(N_{X_E}, Y, \frac{w_i}{2})$. The decision boundary is visibly curved, as depicted in Figure 5. **CG3**. X_C and Y are generated as per CG1, and X_E is generated as per CG2. **CG4**. For dataset instance i, $X_S \sim N([0, 0], \Sigma_{t_i})$, each t_i is a single sample drawn from $\text{Uniform}(1, 2)$, with Σ_{t_i} generated analogously as for CG1. Set $Y \sim \text{Bernoulli}(0.5)$. We set the causal mechanism $f^{\mathbf{CG4}}_{X_E}(N_{X_E}, Y, X_S) := f^{\mathbf{T}}_{X_E}(N_{X_E}, Y, \frac{t_i}{2} + \frac{w_i}{2}) + X_S$. **CG5**. X_C and Y are generated as per CG1. For X_E, define the causal mechanism $f^{\mathbf{CG5}}_{X_E}(N_{X_E}, Y, X_C) := f^{\mathbf{T}}_{X_E}(N_{X_E}, Y, \frac{s_i}{2} + \frac{w_i}{2}) + X_C$. **CG6**. X_C, Y are generated as per CG1, and X_S is generated as per CG4, and set $f^{\mathbf{CG6}}_{X_E}(N_{X_E}, Y, X_C, X_S) := f^{\mathbf{T}}_{X_E}(N_{X_E}, Y, \frac{s_i}{2} + \frac{t_i}{2} + \frac{w_i}{2}) + X_C + X_S$. **CG7**. X_C, Y are generated as per CG1, and X_S is generated as per CG4. For X_E, set $f^{\mathbf{CG7}}_{X_E}(N_{X_E}, Y, X_S) := f^{\mathbf{T}}_{X_E}(N_{X_E}, Y, \frac{t_i}{2} + \frac{w_i}{2})$.

Partition splits for synthetic data. The synthetic datas is created with n=2080 cases, randomly split into partitions given in Table VII. The test partition D_t is used to evaluate accuracy once at end of training. In our experiments, we observe similar performance between D_u and D_t, but we only report accuracy over D_u for brevity. D_G denotes the number of cases that we create via our causal generative models **CGAN-SSL / GCGAN-SSL**.

Partition	Labelled	Unlabelled	Validation	Test	Generated
Notation	D_l	D_u	D_v	D_t	D_G
Size	40	1000	40	1000	1000

TABLE VII: Partition splits used for CG1-CG7

SSL performance on synthetic data. We tabulate classification accuracy over D_u in Table VIII, with corresponding box plots in Figure 6. CG1 illustrates the anticausal SSL conjecture in benchmark methods [9]. In contrast, our method demonstrates some improvement over baseline, although the

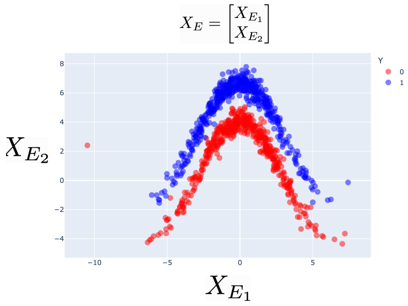


Fig. 5: Curved decision boundary between classes for CG2

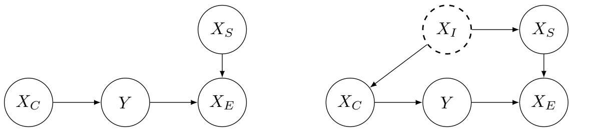
variability is also higher, as depicted in Figure 6 (a). The results for CG2 suggest that in general this decision boundary is difficult to learn. By keeping a similar causal mechanism f_{X_E} across CG2-CG7, we illustrate the potential of extra features to improve the model. CG3, CG5 and CG7 illuminate the strongest support for causality in SSL. By parameterising a composite causal mechanism over Y for unlabelled data, the joint Gumbel-Softmax method GCGAN-SSL seems able to exploit information in unlabelled data more effectively than the disjoint method CGAN-SSL. CG5 illustrates that a disjoint causal approach via CGAN-SSL may actually worsen model performance in this instance. To a lesser extent, this notion is also reflected in CG6: GCGAN-SSL performs better than CGAN-SSL. CG4 shows a weaker but still compelling case for a causal approach to SSL: although SSL-GAN is superior, CGAN-SSL is the next-best performing model.

B. *Real-world data*

Datasets. We use two real-world datasets: in the Breast Cancer Wisconsin dataset [59], the goal is to classify tumour status as either malignant or benign. Using Tetrad software [60], we search for the Markov Blanket using FGES-MB [61] with Degenerate Gaussian BIC score [62]. From the Sachs cell-flow Cytometry dataset [63], we identify the Markov Blanket of binarised variable RAF. All datasets are originally given with no missing labels: hence, we create unlabelled partitions for SSL, using partition splits given in Table IX. Partition allocation is randomised over $n = 100$ trials, partition sizes are kept constant, and $P(Y = 0) \approx 0.5$.

Source Data	D_l	D_u	D_v	D_G
Breast Cancer Wisconsin	10	424	10	424
Sachs	10	7446	10	7446

TABLE IX: Partition splits for real data experiments: see Table VII for partition notation



(a) No confounders

(b) Latent confounder X_I

Fig. 7: X_I is a latent confounder of X_C, X_S

SCM approach for real-world data. When creating the hierarchical SCM model for real-world data, we assume that the full DAG is not known as root nodes cannot be identified. The true data-generating causal model may contain a latent confounding feature X_I. This is depicted in Figure 7. In this instance, we elect to model all parent features X_C and spouse features X_S as a single joint random variable. Hence, the generative model encompasses the possible correlation between any X_C, X_S.

SSL performance on real-world data. Both of our approaches, disjoint CGAN-SSL and joint GCGAN-SSL, outperform benchmark generative approaches on both datasets and improve on the baseline. Given that benchmark generative methods implicitly model the causal structure as $Y \to X$, it is plausible that they exhibit confounding by modelling spurious correlations for features that are actually causes of Y. In comparison to synthetic results, our joint modelling approach GCGAN-SSL, which aims to exploit information in unlabelled X_C, X_E pairs, performs worse than the disjoint approach CGAN-SSL. If this is due to unobserved confounding variables, it is unclear how robust this approach may be in such cases.

Does this depend on the causal structure of the data? The authors state that "the joint Gumbel-Softmax method GCGAN-SSL seems able to exploit information in unlabelled data more effectively ...". Why does this not holld for the real data sets?

GUmbel Softmax can work if we have causal structure $X_C \to Y \to X_E$, and we assume that $P(X_E|X_C)$ could benefit from $\hat{P}(Y|X_C)$. If performance degrades, this suggests that our esitmate is incorrect, and should not be used in estimatino procedure.

Disjoint = pessimistic.

Se how it works with more dat.

VIII. Conclusions

Our primary goal in this work is to provide a unified procedure for semi-supervised classification over causal graphical models. While our results confirm previous insights that the relationship $Y \to X_E$ is crucial for SSL, we show that a structural model f_{X_E} which encapsulates explicit causal relations between Y, X_E and extra features X_C, X_S can improve the classifier by exploiting information in the paired observations in the unlabelled dataset. Our results illustrate the crucial importance of causality in SSL, which we hope may serve as a basis for more detailed analysis in future works.

References

[1] J. Janai, F. Güney, A. Behl, and A. Geiger, "Computer vision for autonomous vehicles: Problems, datasets and state of the art," 2017. [Online]. Available: https://arxiv.org/abs/1704.05519

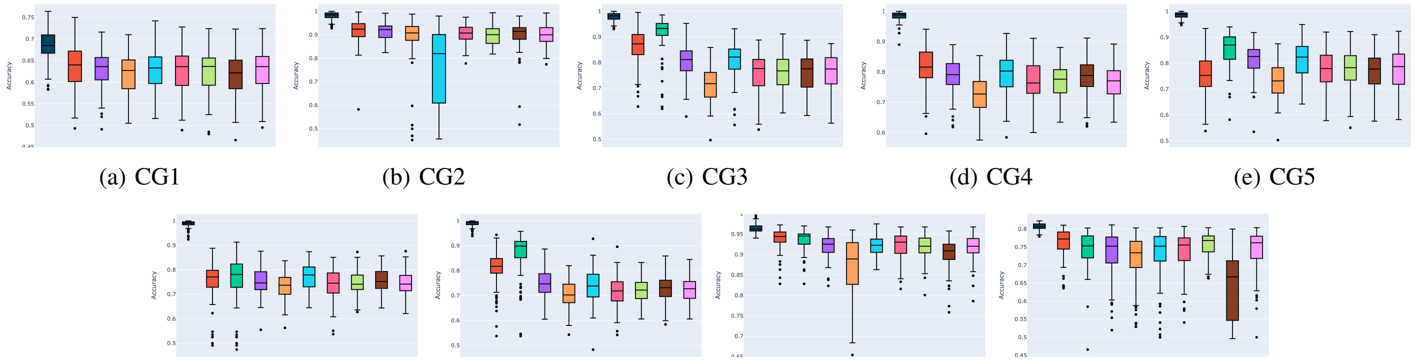
[2] B. Rath, A. Salecha, and J. Srivastava, *Detecting Fake News Spreaders in Social Networks Using Inductive Representation Learning*. IEEE Press, 2020, p. 182–189. [Online]. Available: https://doi.org/10.1109/ASONAM49781.2020.9381466

[3] L. Gu, X. Zhang, S. You, S. Zhao, Z. Liu, and T. Harada, "Semi-supervised learning in medical images through graph-embedded random forest," *Frontiers in Neuroinformatics*, vol. 14, p. 49, 2020. [Online]. Available: https://www.frontiersin.org/article/10.3389/fninf.2020.601829

[4] S. Roychowdhury, "Deep-reap: Deep representations and partial label learning for multi-pathology classification," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2021, pp. 3557–3560.

	KEY	CG1	CG2	CG3	CG4	CG5	CG6	CG7	BCANCER	RAF
F-SUP	⬛	5.644 ± 4.253	8.090 ± 4.371	21.804 ± 6.904	21.873 ± 5.366	20.842 ± 7.497	24.025 ± 4.625	26.628 ± 5.105	4.495 ± 3.108	6.429 ± 5.538
CGAN-SSL	🟥	**0.910 ± 3.079**	**1.681 ± 5.558**	10.158 ± 8.177	2.326 ± 8.005	-2.187 ± 8.314	0.870 ± 8.365	8.569 ± 7.651	**1.898 ± 3.100**	**1.969 ± 6.581**
GCGAN-SSL	🟩	-	-	**15.418 ± 8.685**	-	**7.686 ± 9.035**	2.668 ± 9.886	**14.238 ± 9.456**	1.699 ± 3.043	0.274 ± 6.240
SSL-GAN	🟦	-0.096 ± 4.074	-14.860 ± 16.499	4.317 ± 6.887	**3.002 ± 5.981**	3.436 ± 7.193	**2.761 ± 5.146**	1.535 ± 5.891	0.154 ± 2.925	-1.995 ± 9.176
TRIPLE-GAN	🟧	-0.646 ± 4.047	-1.666 ± 9.958	-4.626 ± 7.432	-4.152 ± 6.669	-4.491 ± 7.747	-1.083 ± 5.234	-1.664 ± 5.828	-5.433 ± 8.600	-2.412 ± 6.549
SSL-VAE	🟥	0.039 ± 2.739	0.554 ± 4.256	-0.452 ± 5.821	0.141 ± 6.033	0.075 ± 5.920	-0.078 ± 5.395	-0.642 ± 5.521	0.289 ± 2.678	-0.279 ± 5.543
VAT	🟪	0.388 ± 2.852	1.286 ± 4.459	4.278 ± 5.608	2.232 ± 5.221	3.330 ± 6.414	1.011 ± 3.638	2.745 ± 5.167	0.100 ± 1.364	-0.832 ± 6.521
ENT-MIN	🟩	-0.065 ± 2.188	-0.154 ± 2.775	0.040 ± 3.818	0.328 ± 3.920	0.133 ± 4.240	-0.021 ± 3.921	-0.019 ± 3.751	0.050 ± 0.979	1.368 ± 4.603
L-PROP	🟫	-0.778 ± 2.898	-0.080 ± 6.707	-0.260 ± 5.672	1.512 ± 4.909	-0.927 ± 6.005	1.216 ± 4.212	1.063 ± 4.401	-1.694 ± 3.206	-10.147 ± 10.727
P-SUP	🟪	62.606 ± 5.013	89.936 ± 4.276	76.082 ± 6.984	76.534 ± 5.576	77.672 ± 7.674	74.564 ± 4.580	72.206 ± 5.131	91.915 ± 2.907	74.229 ± 5.469
n		100	100	100	100	100	100	100	100	100

TABLE VIII: Prediction accuracy for synthetic and real data over D_U, optimal performance over each dataset in **bold**



(a) CG1 (b) CG2 (c) CG3 (d) CG4 (e) CG5

(f) CG6 (g) CG7 (h) BCANCER (i) RAF

Fig. 6: Box plots: Prediction accuracy for synthetic and real data over D_U. Colour and model correspondence is given in column 'KEY' of Table VIII

[5] Z. Shunxiang, Z. Aoqiang, Z. Guangli, W. Zhongliang, and L. KuanChing, "Building fake review detection model based on sentiment intensity and pu learning," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2023.

[6] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[7] X. Yang, Z. Song, I. King, and Z. Xu, "A survey on deep semi-supervised learning," 2021.

[8] Y. Ouali, C. Hudelot, and M. Tami, "An overview of deep semi-supervised learning," *CoRR*, vol. abs/2006.05278, 2020. [Online]. Available: https://arxiv.org/abs/2006.05278

[9] B. Schölkopf, D. Janzing, J. Peters, E. Sgouritsa, K. Zhang, and J. Mooij, "On causal and anticausal learning," in *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ser. ICML'12. Madison, WI, USA: Omnipress, 2012, p. 459–466.

[10] X. Wu, M. Gong, J. H. Manton, U. Aickelin, and J. Zhu, "On causality in domain adaptation and semi-supervised learning: an information-theoretic analysis," 2022. [Online]. Available: https://arxiv.org/abs/2205.04641

[11] J. von Kügelgen, A. Mey, M. Loog, and B. Schölkopf, "Semi-supervised learning, causality, and the conditional cluster assumption," in *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, ser. Proceedings of Machine Learning Research, J. Peters and D. Sontag, Eds., vol. 124. PMLR, 08 2020, pp. 1–10. [Online]. Available: https://proceedings.mlr.press/v124/kugelgen20a.html

[12] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Proceedings of the 17th International Conference on Neural Information Processing Systems*, ser. NIPS'04. Cambridge, MA, USA: MIT Press, 2004, p. 529–536.

[13] S. Wu and W.-S. Zheng, "Semisupervised feature learning by deep entropy-sparsity subspace clustering," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 2, pp. 774–788, 2022.

[14] D.-H. Lee, "Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks," *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*, 07 2013.

[15] H. Pham, Z. Dai, Q. Xie, and Q. V. Le, "Meta pseudo labels," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 11 552–11 563.

[16] S. Qiao, W. Shen, Z. Zhang, B. Wang, and A. Yuille, *Deep Co-Training for Semi-Supervised Image Recognition: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XV*, 09 2018, pp. 142–159.

[17] D.-D. Chen, W. Wang, W. Gao, and Z.-H. Zhou, "Tri-net for semi-supervised deep learning," 07 2018, pp. 2014–2020.

[18] E. Arazo, D. Ortego, P. Albert, N. E. O'Connor, and K. McGuinness, "Pseudo-labeling and confirmation bias in deep semi-supervised learning," 2019. [Online]. Available: https://arxiv.org/abs/1908.02983

[19] T. Miyato, S. ichi Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," 2018.

[20] S. Park, J. Park, S.-J. Shin, and I.-C. Moon, "Adversarial dropout for supervised and semi-supervised learning," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[21] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, and A. G. Wilson, "Averaging weights leads to wider optima and better generalization," 2018, cite arxiv:1803.05407Comment: Appears at the Conference on Uncertainty in Artificial Intelligence (UAI), 2018. [Online]. Available: http://arxiv.org/abs/1803.05407

[22] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le, "Unsupervised data augmentation for consistency training," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 6256–6268. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/44feb0096faa8326192570788b38c1d1-Paper.pdf

[23] M. Sajjadi, M. Javanmardi, and T. Tasdizen, "Regularization with

stochastic transformations and perturbations for deep semi-supervised learning," *CoRR*, vol. abs/1606.04586, 2016. [Online]. Available: http://arxiv.org/abs/1606.04586

[24] Z. Ke, D. Wang, Q. Yan, J. S. J. Ren, and R. W. H. Lau, "Dual student: Breaking the limits of the teacher in semi-supervised learning," *CoRR*, vol. abs/1909.01804, 2019. [Online]. Available: http://arxiv.org/abs/1909.01804

[25] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," *CoRR*, vol. abs/1610.02242, 2016. [Online]. Available: http://arxiv.org/abs/1610.02242

[26] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/68053af2923e00204c3ca7c6a3150cf7-Paper.pdf

[27] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014.

[28] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/file/8a3363abe792db2d8761d6403605aeb7-Paper.pdf

[29] C. Li, K. Xu, J. Zhu, and B. Zhang, "Triple generative adversarial nets," 2017.

[30] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. Salakhutdinov, "Good semi-supervised learning that requires a bad gan," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6513–6523.

[31] G. Qi, L. Zhang, H. Hu, M. Edraki, J. Wang, and X. Hua, "Global versus localized generative adversarial nets," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 1517–1525. [Online]. Available: http://openaccess.thecvf.com/content_cvpr_2018/html/Qi_Global_Versus_Localized_CVPR_2018_paper.html

[32] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. [Online]. Available: https://openreview.net/forum?id=BJtNZAFgg

[33] Z. Gan, L. Chen, W. Wang, Y. Pu, Y. Zhang, H. Liu, C. Li, and L. Carin, "Triangle generative adversarial networks," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 5247–5256. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/bbeb0c1b1fd44e392c7ce2fdbd137e87-Abstract.html

[34] Z. Deng, H. Zhang, X. Liang, L. Yang, S. Xu, J. Zhu, and E. P. Xing, "Structured generative adversarial networks," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 3899–3909. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/c3535febaff29fcb7c0d20cbe94391c7-Abstract.html

[35] D. P. Kingma, D. J. Rezende, S. Mohamed, and M. Welling, "Semi-supervised learning with deep generative models," 2014.

[36] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2014.

[37] T. Joy, S. M. Schmon, P. H. S. Torr, N. Siddharth, and T. Rainforth, "Rethinking semi-supervised learning in vaes," *CoRR*, vol. abs/2006.10102, 2020. [Online]. Available: https://arxiv.org/abs/2006.10102

[38] L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther, "Auxiliary deep generative models," 2016. [Online]. Available: https://arxiv.org/abs/1602.05473

[39] E. Abbasnejad, A. R. Dick, and A. van den Hengel, "Infinite variational autoencoder for semi-supervised learning," *CoRR*, vol. abs/1611.07800, 2016. [Online]. Available: http://arxiv.org/abs/1611.07800

[40] Y. Li, Q. Pan, S. Wang, H. Peng, T. Yang, and E. Cambria, "Disentangled variational auto-encoder for semi-supervised learning," *CoRR*, vol. abs/1709.05047, 2017. [Online]. Available: http://arxiv.org/abs/1709.05047

[41] W. Xu and H. Sun, "Semi-supervised variational autoencoders for sequence classification," *CoRR*, vol. abs/1603.02514, 2016. [Online]. Available: http://arxiv.org/abs/1603.02514

[42] M. Śmieja, M. Wołczyk, J. Tabor, and B. C. Geiger, "Segma: Semi-supervised gaussian mixture autoencoder," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 9, pp. 3930–3941, 2021.

[43] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," 2017. [Online]. Available: https://arxiv.org/abs/1710.09412

[44] V. Verma, A. Lamb, J. Kannala, Y. Bengio, and D. Lopez-Paz, "Interpolation consistency training for semi-supervised learning," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 7 2019, pp. 3635–3641. [Online]. Available: https://doi.org/10.24963/ijcai.2019/504

[45] X. Yang, X. Hu, S. Zhou, X. Liu, and E. Zhu, "Interpolation-based contrastive learning for few-label semi-supervised learning," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2022.

[46] Z. Mai, G. Hu, D. Chen, F. Shen, and H. T. Shen, "Metamixup: Learning adaptive interpolation policy of mixup with metalearning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 7, pp. 3050–3064, 2022.

[47] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/1cd138d0499a68f4bb72bee04bbec2d7-Paper.pdf

[48] D. Berthelot, N. Carlini, E. D. Cubuk, A. Kurakin, K. Sohn, H. Zhang, and C. Raffel, "Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring," *arXiv preprint arXiv:1911.09785*, 2019.

[49] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 596–608. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/06964dce9addb1c5cb5d6e3d9838f733-Paper.pdf

[50] J. Pearl, *Probabilistic reasoning in intelligent systems*. Oxford, England: Morgan Kaufmann, May 1997.

[51] D. Janzing, J. Mooij, K. Zhang, J. Lemeire, J. Zscheischler, P. Daniušis, B. Steudel, and B. Schölkopf, "Information-geometric approach to inferring causal directions," *Artificial Intelligence*, vol. 182-183, pp. 1–31, 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0004370212000045

[52] J. Peters, D. Janzing, and B. Schlkopf, *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press, 2017.

[53] D. Janzing and B. Schölkopf, "Causal inference using the algorithmic markov condition," *IEEE Transactions on Information Theory*, vol. 56, no. 10, pp. 5168–5194, 2010.

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

[55] C.-L. Li, W.-C. Chang, Y. Cheng, Y. Yang, and B. Poczos, "Mmd gan: Towards deeper understanding of moment matching network," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/dfd7468ac613286cdbb40872c8ef3b06-Paper.pdf

[56] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, no. 25, pp. 723–773, 2012. [Online]. Available: http://jmlr.org/papers/v13/gretton12a.html

[57] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *International Conference on Learning Representations*, 2017. [Online]. Available: https://openreview.net/forum?id=rkE3y85ee

[58] R. Duda, P. Hart, and D. Stork, *Pattern classification*. New York: Wiley, 2001.

[59] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic)

[60] J. D. Ramsey, K. Zhang, M. Glymour, R. Romero, B. Huang, I. Ebert-Uphoff, and C. Glymour, "Tetrad—a toolbox for causal discovery," 2018. [Online]. Available: https://www.ccd.pitt.edu/tools/

[61] J. Ramsey, M. Glymour, R. Sanchez-Romero, and C. Glymour, "A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images," *International Journal of Data Science and Analytics*, vol. 3, no. 2, pp. 121–129, Mar 2017. [Online]. Available: https://doi.org/10.1007/s41060-016-0032-z

[62] B. Andrews, J. Ramsey, and G. F. Cooper, "Learning high-dimensional directed acyclic graphs with mixed data-types," in *Proceedings of Machine Learning Research*, ser. Proceedings of Machine Learning Research, vol. 104. PMLR, 05 Aug 2019, pp. 4–21. [Online]. Available: https://proceedings.mlr.press/v104/andrews19a.html

[63] K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan, "Causal protein-signaling networks derived from multiparameter single-cell data," *Science*, vol. 308, no. 5721, pp. 523–529, 2005. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1105809